Platform Documentation

ProVision

Application Version 7.0.0 / 7.0.1

Covering:

- Installation Guide
- Getting Started
- User Guide
- Admin Guide
- Developer Tools
- Help & Support

For additional information, please visit https://docs.6connect.com or contact 6connect at support@6connect.com

1. ProVision Installation Guide . . . 3
1.1 Hosted Instances Guide . . . 4
1.2 Local / VM Installation Guide . . . 5
1.2.1 System Requirements . . . 6
1.2.2 Backup and Redundancy: Local / VM . . . 8
1.2.3 ProVision Local Installation For CentOS 6 . . . 9
1.2.4 ProVision Local Installation For CentOS 7 . . . 15
1.2.5 ProVision Local Installation For Debian 7/8 . . . 21
1.2.6 ProVision Local Installation For Ubuntu . . . 25
1.2.7 6connect Local Software Upgrades . . . 31
1.2.8 Local Installations - Peering Setup . . . 33
1.3 ESXi VM Image Setup . . . 35
2. ProVision Getting Started . . . 36
2.1 First Steps . . . 37
2.2 Resource Concepts . . . 42
2.3 Workflow Concepts . . . 46
3. ProVision User Guide . . . 57
3.1 The Dashboard . . . 58
3.1.1 Customizing the Dashboard . . . 61
3.1.2 Working with Dashboard Widgets . . . 65
3.2 Working with Resources . . . 77
3.2.1 Working with Entries . . . 80
3.2.2 Customizing Sections . . . 89
3.2.3 Customizing Fields . . . 92
3.2.4 Gadgets . . . 94
3.2.4.1 XML Specifications . . . 114
3.2.5 Contact Manager . . . 115
3.3 DNS Tab . . . 127
3.3.1 Working with DNS Groups . . . 131
3.3.2 Working with DNS Zones - Common Tasks . . . 148
3.3.3 Editing DNS Zones . . . 159
3.4 DHCP Tab . . . 166
3.4.1 Working with DHCP Groups . . . 168
3.4.2 Working with DHCP Pools . . . 182
3.4.3 Copy of DHCP Tab . . . 196
3.5 IPAM Tab . . . 210
3.5.1 Working with IP Blocks . . . 216
3.5.2 Working with IP Rules . . . 231
3.6 Peering v2 . . . 237
3.6.1 Peering - Common Tasks . . . 238
3.6.2 Peering Exchanges . . . 243
3.6.3 Peering Routers . . . 246
3.6.4 Peering Sessions . . . 248
3.6.5 Managing Peers . . . 251
3.7 Reporting . . . 254
3.8 User Preferences . . . 257
4. ProVision Admin Guide . . . 258
4.1 Admin Preferences . . . 259
4.2 Authentication Options . . . 270
4.2.1 LDAP Authentication . . . 271
4.2.2 LDAP Authentication on Windows Server . . . 275
4.2.3 RADIUS Authentication . . . 279
4.3 IPAM Administration . . . 282
4.3.1 IPAM Parameters . . . 284
4.3.2 IPAM Rules . . . 291
4.3.3 Holding Tank Management . . . 296
4.3.4 LIR Management and Use . . . 298
4.3.4.1 ARIN LIR Setup and Use . . . 302
4.3.4.2 RIPE LIR Setup and Use . . . 307
4.4 VLAN Administration . . . 311
4.4.1 Working with the VLAN Manager . . . 314
4.5 DNS Administration . . . 333
4.5.1 Working with DNS Servers . . . 339
4.5.2 Configuring ISC BIND Support . . . 350
4.5.3 Configuring PowerDNS Support . . . 354
4.5.4 Configuring Secure64 Support . . . 359
4.5.5 Configuring Split Horizon/Views . . . 365
4.5.6 Configuring DNSSEC . . . 370
4.6 DHCP Administration . . . 374
4.6.1 Working with DHCP Servers . . . 377
4.7 Importing Your Data . . . 388
4.7.1 Resource Import from CSV . . . 393
4.7.2 Resource Import Tool . . . 400
4.7.2.1 Resource Importer Walkthrough - Step 1 . . . 402
4.7.2.2 Resource Importer Walkthrough - Step 2 . . . 404

4.7.2.3 Resource Importer Walkthrough - Step 3 . . . 406
4.7.2.4 Resource Importer Walkthrough - Step 4 . . . 407
4.7.2.5 Resource Importer Walkthrough - Step 5 . . . 411
4.7.2.6 Resource Importer Walkthrough - Step 6 . . . 412
4.7.3 Import Sessions . . . 413
4.7.4 IP Import from CSV . . . 415
4.7.5 Import Aggregate Blocks . . . 423
4.7.6 Import DNS Zones . . . 426
4.7.6.1 BIND DNS Zone Upload / Import . . . 428
4.7.6.2 PowerDNS Zone Import . . . 434
4.7.6.3 InfoBlox Zone Import . . . 436
4.7.6.4 NS One Zone Import . . . 439
4.7.6.5 Dyn DNS Zone Import . . . 442
4.7.6.6 DNSMadeEasy Zone Import . . . 445
4.7.6.7 IPPLAN Zone Import . . . 448
4.8 Users & Permissions . . . 450
4.8.1 Global Permissions . . . 452
4.8.2 Resource Permissions . . . 454
4.8.3 Users and Groups . . . 457
4.8.4 Verifying Permissions . . . 463
4.9 API Tab . . . 466
4.9.1 Reverse API 1 . . . 467
4.10 Scheduler . . . 471
4.11 Log . . . 477
4.12 RIR Integration: SWIP/RPSL . . . 480
5. ProVision Developer Tools . . . 484
5.1 API v1 . . . 485
5.1.1 1 - Overview . . . 486
5.1.2 2 - Making API Requests . . . 487
5.1.3 3 - SDK . . . 489
5.1.4 API Module - Admin and Audit . . . 491
5.1.4.1 Authentication Testing . . . 492
5.1.4.2 Backup . . . 494
5.1.4.3 Log Management . . . 495
5.1.4.4 Scheduler 1 . . . 498
5.1.4.5 Zone Templates . . . 503
5.1.5 API Module - DHCP . . . 506
5.1.5.1 API Module - DHCPv2 . . . 511
5.1.6 API Module - DNS . . . 528
5.1.7 API Module - IPAM . . . 564
5.1.8 API Module - LIR . . . 596
5.1.9 API Module - Peering . . . 606
5.1.10 API Module - Resource . . . 617
5.1.11 API Module - VLAN . . . 630
5.1.12 How Do I... . . . 644
5.1.13 Reverse API . . . 655
5.1.13.1 Reverse API - Detail . . . 657
5.2 CLI (Alpha) . . . 662
5.3 Toolkit . . . 663
5.3.1 IPPlan Importer . . . 667
5.4 Resource Concepts 1 . . . 670
5.4.1 Resource Classes Doc . . . 671
5.4.2 Database Layout 1 . . . 673
5.4.3 Asset System . . . 675
5.5 Portable Gadgets . . . 677
5.5.1 Getting Started With Portable Gadgets . . . 686
5.5.2 Portable Gadget Customization . . . 689
6. Help & Support . . . 691
6.1 Tutorials . . . 692
6.1.1 Common Tasks . . . 693
6.1.2 UI Tours . . . 694
6.2 FAQ . . . 695
6.3 Additional Resources . . . 698
6.3.1 Import Templates . . . 699
6.3.2 List of Abbreviations . . . 700
6.4 Feedback and Feature Requests . . . 701

ProVision Installation Guide

Installing ProVision

You have 6connect ProVision and now it's time to set it up! 6connect offers both cloud hosted instances and local installations of ProVision. Follow the links below for specific instructions on each instance type.

For setup assistance or additional information, you can contact our support team at support@6connect.com.

Table of Contents

- Hosted Instances Guide
- Local / VM Installation Guide
- ESXi VM Image Setup

Hosted Instances Guide

Hosted Instances Guide

With a cloud hosted instance of ProVision, all you need is one of the following web browsers with an internet connection and login credentials!

Once you have confirmed that you have a supported browser and valid login, you can proceed to ProVision Getting Started, the ProVision User Guide, or the ProVision Admin Guide to learn more about ProVision.

- Hosted Instances Guide
  - 6connect Cloud Hosted Instance: Browser Requirements
  - Backup and Redundancy

6connect Cloud Hosted Instance: Browser Requirements

6connect makes every effort to maintain broad compatibility across browser vendors and versions.
Web Browsers Supported:

- Firefox, Safari, Chrome, Microsoft Edge (current versions).

Backup and Redundancy

Backup Schedule

Both local and hosted instances are provided with a default Scheduler task to perform a backup every 24 hours, with a 1 month retention policy.

Restoration
Is a phone call or email away. We can spin up a new instance with your preferred data set.

Local / VM Installation Guide

Installing ProVision

Local and VM installs of ProVision have specific requirements and configuration settings. Please follow the links below for detailed instructions on how to set up your local installation of ProVision.

For setup assistance or additional information, you can contact our support team at support@6connect.com.

Previous versions of this installation documentation contained a dedicated page, "6connect Local Software Installation" for the ProVision-only section of the installation process. This information is now available under each OS-specific installation page as the last section ("Install 6connect ProVision Software:").

Table of Contents

- System Requirements
- Backup and Redundancy: Local / VM
- ProVision Local Installation For CentOS 6
- ProVision Local Installation For CentOS 7
- ProVision Local Installation For Debian 7/8
- ProVision Local Installation For Ubuntu
- 6connect Local Software Upgrades
- Local Installations - Peering Setup

System Requirements

ProVision System Requirements

- ProVision System Requirements
  - 6connect Locally Hosted Instance
    - Hardware Requirements:
    - Software Requirements:
    - Port Requirements:

6connect Locally Hosted Instance

Initial application installation is included with the purchase of a license from 6connect. If modifications need to be made, we recommend contacting 6connect prior to any changes to ensure there is no negative impact to production systems or product functionality.

Hardware Requirements:

The optimum resource mix will be based on page views/refreshes. A larger concurrent user base with constant editing may benefit from additional RAM.
The minimum recommended hardware is:

	Minimum	Recommended
Processor	Dual-core Xeon class processor or equivalent	Quad-core Xeon
RAM	2GB RAM	4GB RAM
Storage	20 GB	100GB

*Virtual instances are also acceptable. We have confirmed functionality with Citrix Xen Essentials, VMware, KVM, etc.

VMware Statement of Support

6connect, Inc. will support customers who run 6connect ProVision products on supported Operating Systems, irrespective of whether they are running in VMware environments or not. 6connect, Inc. supports Operating Systems, not specific hardware configurations. Accordingly, VMware operates as a hardware abstraction layer.

VMware supports a set of certified Operating Systems and Hardware, and the customer and VMware will be responsible for any interactions or issues that arise at the Hardware or Operating System layer as a result of their use of VMware.

6connect, Inc. will not require clients to recreate and troubleshoot every issue in a non-VMware environment; however, 6connect, Inc. does reserve the right to request our customers to diagnose certain issues in a native certified Operating System environment, operating without the virtual environment. 6connect, Inc. will only make this request when there is reason to believe that the virtual environment is a contributing factor to the issue.

Any time spent on investigation of problems that may, in the sole opinion of 6connect, Inc. be related to VMware, will be handled in the following fashion:

1) 6connect, Inc. will provide standard support to all 6connect ProVision products.

2) If a problem is encountered while 6connect ProVision is/are running in a VMware environment, the client may be required to recreate the problem on a non-VMware server unit, at which time 6connect, Inc. will provide regular support.

3) The client can authorize 6connect, Inc. to investigate the VMware related items at normal time and materials rates. If such investigation shows that the problem is VMware related, the client may contract 6connect, Inc. to provide a software change to resolve the issue if such a resolution is possible.

4) Regardless of the problem type or source, if the problem is determined to be a non VMware related issue - time spent on investigation and resolution will be covered as part of regular maintenance, and support will be provided as usual.

Software Requirements:

Base Software Needed:

	Required	Link
Operating System	Linux/BSD/OSX	

Apache	Apache 2.4	http://httpd.apache.org/
PHP	PHP 7.x	http://php.net/downloads.php
MySQL	MySQL 5.7	http://www.mysql.com/downloads/

MySQL Triggers

6connect does not support custom MySQL triggers at this time - please email support@6connect.com if you have any questions.

Port Requirements:

Open outbound ports 443 and port 80

- cloud.6connect.com is used for license check
- checkip.dyndns.org validates the IP address of the machine to communicate with the licensing server

The ProVision DNS Module also supports zone checks on attached DNS Server(s) as an external user. Port 53 will need to be open and usable for ProVision to query the servers. If not, zone error checking and verification will not work as intended.

Depending on the Connectors in use, there may be other port requirements needed. Port 22 (SSH) is used extensively for control of downstream devices.

Backup and Redundancy: Local / VM

Backup and Redundancy

Local/VM Instance

Backup Schedule
Both local and hosted instances are provided with a default Scheduler task to perform a backup every 24 hours, with a 1 month retention policy.

Restoration
Is a phone or email away. We can spin up a new instance with your preferred data set, or send you a link to download your database. Optionally, we can even help you set it up and import your data to your new instance or assist with redundant configuration options depending on your RPO /RTO guidelines.

Backup your Data

For local customers, you should be backing up the following items:

mysqldump

And system folders off the 6connect root:

/scans

/zones

/keys

/archive

/data

ProVision Local Installation For CentOS 6

CentOS 6

- CentOS 6
- Before You Begin
- Install Requirements
    - 1) Upgrade your current packages
    - 2) Install Required Packages
        - PHP
        - MySQL
        - DNS and Additional Utilities
        - DNSSEC-Tools
    - 3) Configuring the requirements:
        - SSL
        - Apache
        - MySQL
    - 4) Optional configurations:
        - Configure SELinux
        - Configure IPTables
        - Radius (Optional)
        - SSH
    - 4) Install 6connect ProVision Software:

Before You Begin

Ensure that System Requirements have been met prior to proceeding with the CentOS Configuration Guide.

Install Requirements

1) Upgrade your current packages

Upgrade your current packages.

```
yum update
```

2) Install Required Packages

PHP

Installations of ProVision 7.x and newer require PHP 7.0 (and related extensions). CentOS 6 comes with PHP 5.3 by default. You can either add a repository which provides PHP 7.0 or install PHP manually.

The Webtatic and Remi repos both have versions of PHP which are newer than those in the official repos. For this example, we'll be using Webtatic

Add the repository:

```
rpm -Uvh https://mirror.webtatic.com/yum/el6/latest.rpm
```

Update:

```
yum update
```

Install:

Click here to expand...

PHP 7.0 / Apache2 / extensions

- httpd
- php70w
- php70w-opcache
- php70w-mysqlnd

- php70w-pdo
- php70w-ldap
- php70w-pecl-memcache
- php70w-bcmath
- php70w-devel
- php70w-pear
- php70w-cli

Development tools for pecl / additional system packages:

- curl
- openssl
- memcached
- mod_ssl

```
yum install httpd php70w php70w-opcache php70w-mysqlnd php70w-pdo php70w-ldap php70w-pecl-memcache
php70w-bcmath php70w-devel php70w-pear php70w-cli curl openssl memcached mod_ssl
```

MySQL

Install MySQL 5.7 to use a local database.
MySQL is included with most CentOS installs, check for it with:

```
yum list installed | grep mysql
```

The default MySqL version included with most CentOS installs will need to be upgraded to the latest version:

```
rpm -Uvh https://mirror.webtatic.com/yum/el6/latest.rpm
```

If you have an existing installation, you can replace it with:

```
yum install mysql.`uname -i` yum-plugin-replace
yum replace mysql --replace-with mysql57w
```

For a fresh install:

```
    yum install mysql57w mysql57w-server
```

Then, re-start and configure.

```
    service mysqld start      chkconfig mysqld on
```

If a new install, set the MySQL root password:

```
     /usr/bin/mysqladmin -u root password 'new-password'
```

For an existing install / upgrade, you will need to upgrade the existing tables after the restart.

```
mysql_upgrade -u root -p
```

This will issue a password prompt for the user. If you don't have a root user password, remove the "-p".

DNS and Additional Utilities

5. Install the DNS and other remaining utilities:
Click here to expand...

- curl
- openssl
- nmap
- bind-utils
- bind

- expect

```
yum install curl openssl nmap bind-utils bind expect
```

DNSSEC-Tools

Install and Compile DNSSEC-Tools

```
yum groupinstall 'Development Tools'
yum install openssl-devel perl-devel perl-CPAN
cd /usr/src
wget https://www.dnssec-tools.org/download/dnssec-tools-2.1.tar.gz
tar -xzf dnssec-tools-2.1.tar.gz
./configure
make
make install
```

3) Configuring the requirements:

SSL

Self signed certificates in CentOS 6 by default have been already installed.

If you want to change it, follow the steps below:

Note: For production install, it is **HIGHLY RECOMMENDED** to use organization signed certs

Generate private key, CSR, and temporary key if one hasn't been provided.

```
openssl genrsa -out ca.key 1024    openssl req -new -key ca.key -out ca.csr
openssl x509 -req -days 365 -in ca.csr -signkey ca.key -out ca.crt
```

Copy the files to the correct locations

```
   cp ca.crt /etc/pki/tls/certs    cp ca.key /etc/pki/tls/private/ca.key    cp ca.csr /etc/pki/tls
/private/ca.csr
```

Make sure that you copy the files and do not move them if SELinux is enabled (which it is by default)

Edit the apache ssl config and put in the appropriate options:
(shown using the vi editor, though you may use the editor of your choice)

```
vi /etc/httpd/conf.d/ssl.conf
```

Find the lines that start with SSHCertificateFile and change them to be like:

```
SSLCertificateFile /etc/pki/tls/certs/ca.crt
SSLCertificateKeyFile /etc/pki/tls/private/ca.key
```

Then restart

```
/etc/init.d/httpd restart
```

Add 443 virtual hosts as needed in httpd.conf.

Apache

Allow overwrites in the apache vhosts

```
sed -i 's/AllowOverride None/AllowOverride All/g' /etc/httpd/conf/httpd.conf
```

Start Apache and make it to start on boot

```
    chkconfig httpd on     service httpd start
```

mod rewrite REQUIRED

Please note that mod_rewrite is required! If it is not enabled in Apache, key elements will not work as expected.

MySQL

Set the MySQL Configuration:

```
mysql -p -e "SET GLOBAL sql_mode='NO_ENGINE_SUBSTITUTION';SET SESSION
sql_mode='NO_ENGINE_SUBSTITUTION';"
```

then enter the MySQL root password when prompted.

4) Optional configurations:

Configure SELinux

READ THIS BEFORE YOU USE SELINUX

SELinux is a very powerful method of securing the CentOS environment, but it is not "turn key" and requires expertise to configure it correctly. If you do not know how to configure SELinux, please do not use it. A badly configured SELinux install will not work well and result in frustration. If you have any questions or concerns about this - please contact 6connect Support at support@6connect.com.
RE-IP WARNING

Please remember - if you change the IP address of the your server, then you will need to update SELinux functions accordingly

Most CentOS install have SELinux enabled by default. One of its protections is to not allow httpd daemon to make network connections, we need to disable this for license checks.

To view the SELinux configuration for http:

```
/usr/sbin/getsebool -a | grep httpd
```

To turn protection off for the httpd daemon for creating network connections:

```
   /usr/sbin/setsebool -P httpd_can_network_connect 1
```

Configure IPTables

IPTables is enabled by default on CentOS. Add a new rule to allow 443 from anywhere. Make sure that this rule is in the chain BEFORE any blanket reject rule:

If you are going to run iptables, click here

To list all current IPTable rules:

```
/etc/rc.d/init.d/iptables status
```

To add a rule for 443:

```
/sbin/iptables  RH-Firewall-1-INPUT -I 5 -m state --state NEW -m tcp -p tcp --dport 443 -j ACCEPT
```

Note

The -I 5 is what adds the rule to the 5th chain position. You might need to change this depending on existing rules. Look at what rules are there before running.

To save the new config:

```
/etc/rc.d/init.d/iptables save
```

OR (some versions of centOS have different iptables names, so the above won't work)

```
vi /etc/sysconfig/iptables
```

With the file open for editing, add:

```
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 443 -j ACCEPT
```

Once complete - restart the iptables service:

```
/etc/init.d/iptables restart
```

Customers can alter this post install to allow only their IP space, plus the 6connect management space.

Radius (Optional)

This section only needs to be followed if the customer will be using Radius for authentication.

If you are going to use radius authentication, click here.

Install radius module:

```
    pecl install radius     echo extension=radius.so > /etc/php.d/radius.ini
```

SSH

Install ssh module:

```
    yum install libssh2-devel     pecl install -f ssh2     echo extension=ssh2.so > /etc/php.d/ssh2.
ini
```

4) Install 6connect ProVision Software:

1. Remove the current contents in the ProVision web folder (currently the www root) and after extract the archive contents (where 7.x.x is the version number for the build) :

```
tar -xf productionBuild-7.x.x-php7.0.tar -C /var/www/html
```

2. Change the permissions to be the web user permissions

chown -R apache.apache /var/www/html

3. Go to http://<web root>/install/configTest.php. Follow the provided instructions, correcting any configuration errors if they occur. Once all steps are completed, you are ready to use your ProVision instance!

ProVision Local Installation For CentOS 7

CentOS 7

- CentOS 7
- Before You Begin
- Install Requirements
    - 1) Upgrade your current packages
    - 2) Install Required Packages
        - PHP
        - MySQL
        - DNS and Additional Utilities
        - DNSSEC-Tools
    - 3) Configuring the requirements:
        - SSL
        - Apache
        - MySQL
    - 4) Optional configurations:
        - Configure SELinux
        - Configure IPTables
        - Radius (Optional)
        - SSH
    - 5) Install 6connect ProVision Software:

Before You Begin

Ensure that System Requirements have been met prior to proceeding with the CentOS Configuration Guide.

Install Requirements

1) Upgrade your current packages

Upgrade your current packages.

```
yum update
```

2) Install Required Packages

PHP

Installations of ProVision 7.x and newer require PHP 7.0 (and related extensions). CentOS 6 comes with PHP 5.3 by default. You can either add a repository which provides PHP 7.0 or install PHP manually. The Webtatic and Remi repos both have versions of PHP which are newer than those in the official repos. For this example, we'll be using Webtatic

Add the repository:

```
rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
rpm -Uvh https://mirror.webtatic.com/yum/el7/webtatic-release.rpm
```

Install PHP and extensions:

Click here for the list of extensions...

PHP 7.0 / Apache2 / extensions

- httpd
- php70w
- php70w-opcache
- php70w-mysqlnd
- php70w-pdo
- php70w-ldap
- php70w-pecl-memcache
- php70w-bcmath
- php70w-devel
- php70w-pear
- php70w-cli

Development tools for pecl / additional system packages:

- curl
- openssl
- memcached
- mod_ssl

```
yum install httpd php70w php70w-opcache php70w-mysqlnd php70w-pdo php70w-ldap php70w-pecl-memcache
php70w-bcmath php70w-devel php70w-pear php70w-cli curl openssl memcached mod_ssl
```

MySQL

Install MySQL 5.7 / MariaDB to use a local database.
MySQL is included with most CentOS installs, check for it with:

```
yum list installed | grep mysql
```

If MySQL is not installed, click here...

If it is not installed:

```
rpm -Uvh http://repo.mysql.com/mysql-community-release-el7-5.noarch.rpm
yum install mysql-server
service mysqld start
chkconfig mysqld on
```

Set the MySQL root password:

```
/usr/bin/mysqladmin -u root password 'new-password'
```

To install MariaDB instead of MySQL, click here...

If you prefer to install MariaDB:

```
yum install mariadb-server mariadb
systemctl start mariadb
```

Set the root password, as currently it is not set, just hit ENTER on the current password:

```
mysql_secure_installation
```

Set so that it starts on boot:

```
systemctl enable mariadb.service
```

DNS and Additional Utilities

5. Install the DNS and other remaining utilities:

Click here for the list of utilities...

- curl
- openssl
- nmap
- bind-utils
- bind
- expect
- wget
- bzip2

```
yum install curl openssl nmap bind-utils bind expect wget bzip2
```

DNSSEC-Tools

Install and Compile DNSSEC-Tools

```
      yum groupinstall 'Development Tools'
yum install openssl-devel perl-devel perl-CPAN
cd /usr/src
wget https://www.dnssec-tools.org/download/dnssec-tools-2.2.tar.gz
tar -xzf dnssec-tools-2.2.tar.gz
./configure
make
make install
```

3) Configuring the requirements:

SSL

1. Self signed certificates in CentOS 7 by default have been already installed.

If you want to change it, follow the steps below:

Note: For production install, it is **HIGHLY RECOMMENDED** to use organization signed certs

Click here to expand...

Generate private key, CSR, and temporary key if one hasn't been provided.

```
openssl genrsa -out ca.key 1024
openssl req -new -key ca.key -out ca.csr
openssl x509 -req -days 365 -in ca.csr -signkey ca.key -out ca.crt
```

Copy the files to the correct locations

```
cp ca.crt /etc/pki/tls/certs
cp ca.key /etc/pki/tls/private/ca.key
cp ca.csr /etc/pki/tls/private/ca.csr
```

Make sure that you copy the files and do not move them if SELinux is enabled (which it is by default)

Edit the apache ssl config and put in the appropriate options:
(shown using the vi editor, though you may use the editor of your choice)

```
vi /etc/httpd/conf.d/ssl.conf
```

Find the lines that start with SSHCertificateFile and change them to be like:

```
SSLCertificateFile /etc/pki/tls/certs/ca.crt
SSLCertificateKeyFile /etc/pki/tls/private/ca.key
```

Then restart.

```
/etc/init.d/httpd restart
```

Add 443 virtual hosts as needed in httpd.conf.

Apache

Allow overwrites in the apache vhosts

```
sed -i 's/AllowOverride None/AllowOverride All/g' /etc/httpd/conf/httpd.conf
```

Start Apache and make it start on boot

```
    systemctl start httpd.service
    systemctl enable httpd.service
```

mod rewrite REQUIRED

Please note that mod_rewrite is required! If it is not enabled in Apache, key elements will not work as expected.

MySQL

Set the MySQL Configuration:

```
mysql -p -e "SET GLOBAL sql_mode='NO_ENGINE_SUBSTITUTION';SET SESSION
sql_mode='NO_ENGINE_SUBSTITUTION';"
```

Then enter the MySQL root password when prompted.

4) Optional configurations:

Configure SELinux

READ THIS BEFORE YOU USE SELINUX

SELinux is a very powerful method of securing the CentOS environment, but it is not "turn key" and requires expertise to configure it correctly. If you do not know how to configure SELinux, please do not use it. A badly configured SELinux install will not work well and result in frustration. If you have any questions or concerns about this - please contact 6connect Support at support@6connect.com.
RE-IP WARNING

Please remember - if you change the IP address of the your server, then you will need to update SELinux functions accordingly

Most CentOS install have SELinux enabled by default. One of its protections is to not allow httpd daemon to make network connections, we need to disable this for license checks.

To view the SELinux configuration for http:

```
/usr/sbin/getsebool -a | grep httpd
```

To turn protection off for the httpd daemon for creating network connections:

```
 /usr/sbin/setsebool -P httpd_can_network_connect 1
```

Configure IPTables

IPTables is enabled by default on CentOS. Add a new rule to allow 443 from anywhere. Make sure that this rule is in the chain BEFORE any blanket reject rule:

If you are going to run iptables, click here

To list all current IPTable rules:

```
iptables -L
```

To add a rule for 443:

```
/sbin/iptables  RH-Firewall-1-INPUT -I 5 -m state --state NEW -m tcp -p tcp --dport 443 -j ACCEPT
```

Note

The -I 5 is what adds the rule to the 5th chain position. You might need to change this depending on existing rules. Look at what rules are there before running.

To save the new config:

```
/etc/rc.d/init.d/iptables save
```

OR (some versions of centOS have different iptables names, so the above won't work)

```
vi /etc/sysconfig/iptables
```

With the file open for editing, add:

```
-A RH-Firewall-1-INPUT -m state --state NEW -m tcp -p tcp --dport 443 -j ACCEPT
```

Once complete - restart the iptables service:

```
/etc/init.d/iptables restart
```

Customers can alter this post install to allow only their IP space, plus the 6connect management space.

Radius (Optional)

This section only needs to be followed if the customer will be using Radius for authentication.

If you are going to use radius authentication, click here.

Install radius module:

```
    pecl install radius
    echo extension=radius.so > /etc/php.d/radius.ini
```

SSH

Install ssh module:

```
yum install libssh2-devel
pecl install -f ssh2
echo extension=ssh2.so > /etc/php.d/ssh2.ini
```

5) Install 6connect ProVision Software:

1. Remove the current contents in the ProVision web folder (currently the www root) and after extract the archive contents (where 7.x.x is the version number for the build) :

```
tar -xf productionBuild-7.x.x-php7.0.tar -C /var/www/html
```

2. Change the permissions to be the web user permissions

```
chown -R apache.apache /var/www/html
```

3. If enabled SELinux you must execute the following command:

```
chcon -R -t httpd_sys_rw_content_t /var/www/html

chcon -R -t httpd_sys_rw_content_t /tmp
```

You must also execute the same command for the 6c secure path created from **configSecureKeys.sh**

4. Go to http://<web root>/install/configTest.php. Follow the provided instructions, correcting any configuration errors if they occur. Once all steps are completed, you are ready to use your ProVision instance!

ProVision Local Installation For Debian 7/8

Debian 7/ Debian 8

- Debian 7/ Debian 8
- Before You Begin
- Install Requirements
    - 1) Upgrade your current packages
    - 2) Install Required Packages
        - PHP
        - MySQL
        - DNS and Additional Utilities
    - 3) Configuring the requirements:
        - SSL
        - Apache
        - MySQL
    - 4) Optional configurations:
        - Radius (Optional)
    - 5) Install 6connect ProVision Software:

Before You Begin

Ensure that System Requirements have been met prior to proceeding with the Installation Guide.

Install Requirements

1) Upgrade your current packages

Upgrade your current packages.

```
apt-get update
apt-get upgrade
```

2) Install Required Packages

PHP

This step is only applicable to Debian 7 users. If you are using Debian 8, please skip to the next step.

Installations of ProVision 7.x and newer require PHP 7.0 (and related extensions). ProVision 7.x installations using Debian versions prior to Stretch 9 require updating to PHP 7.0. Debian Wheezy comes with PHP 5.4 by default. You can either add a repository which provides PHP 7.0 or install PHP manually. In this example, we are going to use 3rd party repository that also upgrades the Apache to 2.4 as well as some other packages.

Edit sources.list:
(shown using the vi editor, though you may use the editor of your choice)

```
vi /etc/apt/sources.list
```

Add the following lines to sources.list:

```
deb http://packages.dotdeb.org wheezy-php70 all
deb-src http://packages.dotdeb.org wheezy-php70 all
```

Install the GPG key:

```
wget https://www.dotdeb.org/dotdeb.gpg
apt-key add dotdeb.gpg
```

And update the packages list:

```
apt-get update
```

Install Apache2/PHP7/MySQL and development tools for pecl:

Click here for the list of extensions...

PHP 7.0 / Apache2 / extensions

- httpd
- php70w
- php70w-opcache
- php70w-mysqlnd
- php70w-pdo
- php70w-ldap
- php70w-pecl-memcache
- php70w-bcmath
- php70w-devel
- php70w-pear
- php70w-cli

Development tools for pecl / additional system packages:

- curl
- openssl
- memcached
- mod_ssl

```
apt-get install apache2 libapache2-mod-php7 php7 php7-cgi php7-cli php7-gd php7-curl php7-ldap
php7-mysqlnd php7-ssh2 php-pear php7-dev curl openssl memcached php7-memcache
```

While installing you will be asked to set the MySQL root password.

MySQL

Install MySQL 5.7 to use a local database.

```
apt-get install mysql-server
```

You will be asked to enter a root password for the mysql server.

DNS and Additional Utilities

Install the DNS and other utilities:

Click here to expand...

- curl
- openssl
- memcached
- php7-memcache
- nmap
- dnsutils
- bind9utils
- dnssec-tools
- expect

```
apt-get install curl openssl memcached php7-memcache nmap dnsutils bind9utils dnssec-tools expect
```

3) Configuring the requirements:

SSL

Enable the necessary apache modules with the following command:

```
a2enmod php7 ssl rewrite
```

In order to create self signed certificates and to work properly, the hostname must be resolved properly.
Open the hosts file for editing:

```
vi /etc/hosts
```

Add a record for your hostname

```
yourhostname X.X.X.X
```

Create self signed certificate:

```
make-ssl-cert generate-default-snakeoil --force-overwrite
```

Apache

mod rewrite REQUIRED

Please note that mod_rewrite is required! If it is not enabled in Apache, key elements will not work as expected.
Enable the default vhost for apache

```
a2ensite default-ssl
```

Allow overwrites in the apache vhosts

```
sed -i 's/AllowOverride None/AllowOverride All/g' /etc/apache2/sites-available/default*
```

6. Reload apache

```
service apache2 reload
```

MySQL

MySQL Configuration:

```
mysql -p -e "SET GLOBAL sql_mode='NO_ENGINE_SUBSTITUTION';SET SESSION
sql_mode='NO_ENGINE_SUBSTITUTION';"
```

Then, enter the MySQL root password when prompted.

4) Optional configurations:

Radius (Optional)

This section only needs to be followed if the customer will be using Radius for authentication.

If you are going to use radius authentication, click here.

1. Install radius module:
Debian 7:

Debian 7:

```
pecl install radius
```

Debian 8:

Debian 8:

```
apt-get install php7-radius
```

Create module loading configuration:
(shown using the vi editor, though you may use the editor of your choice)

```
vi /etc/php5/mods-available/radius.ini
```

Add the following lines to radius.ini:

```
; configuration for php radius module
; priority=20
extension=radius.so
```

To enable the radius module, type the following command:

```
php7enmod radius
```

5) Install 6connect ProVision Software:

1. Remove the current contents in the ProVision web folder (currently the www root) and after extract the archive contents (where 7.x.x is the version number for the build):

```
tar -xf productionBuild-7.x.x-php7.0.tar -C /var/www/
```

2. Change the permissions to be the web user permissions

```
chown -R www-data.www-data /var/www
```

3. Go to http://<web root>/install/configTest.php. Follow the provided instructions, correcting any configuration errors if they occur. Once all steps are completed, you are ready to use your ProVision instance!

ProVision Local Installation For Ubuntu

Ubuntu 14.04 LTS

- Ubuntu 14.04 LTS
- Before You Begin
- Install Requirements
    - 1) Upgrade your current packages
    - 2) Install Required Packages
        - PHP
        - MySQL
        - DNS and Additional Utilities
    - 3) Configuring the requirements:
        - SSL
        - Apache
        - MySQL
    - 4) Optional configurations:
        - Radius (Optional)
        - SSH
    - 5) Install 6connect ProVision Software:

Before You Begin

Ensure that System Requirements have been met prior to proceeding with the Installation Guide.

Permissions

Before installing, verify that you have appropriate install permissions. Depending on your permissions and system setup, commands may need to be run as superuser by prepending "sudo" to the listed commands.

Install Requirements

1) Upgrade your current packages

Upgrade your current packages.

```
apt-get update
apt-get upgrade
```

2) Install Required Packages

PHP

Installations of ProVision 7.x and newer require PHP 7.0 (and related extensions). You can either add a repository which provides PHP 7.0 or install PHP manually. In this example, we use the default package sources for Ubuntu 14.04 to install Apache 2.4 and PHP 7.0. We also provide an option to install PHP 5.7 through a third-party repository.

If you are installing PHP 7.0 click here...

If you are installing ProVision 7.x, or after, make sure that your build archive has "php7.0" as part of its name. For example "productionBuild-7.0.0-php7.0.tar".

1. Add PHP 7.0 package sources to your system:

```
add-apt-repository ppa:ondrej/php
```

and confirm with ENTER to continue.

If you get an error here, you may need to install python-software-properties first and then repeat the repository add as follows:

```
apt-get update
apt-get install python-software-properties

add-apt-repository ppa:ondrej/php
```

2. Update packages:

```
apt-get update
```

3. Install PHP:

```
apt-get install php7.0
```

and confirm with “y” and ENTER.

You’ll now see “PHP 7.0.x” in the scrolling installation logs.

To verify the installed version of PHP, run:

```
php7 -v
```

Install Apache2/PHP 7.0/MySQL 5.7 and development tools for pecl:

Click here for the list of extensions...

PHP 7.0 / Apache2 / extensions

- httpd
- php70w
- php70w-opcache
- php70w-mysqlnd
- php70w-pdo
- php70w-ldap
- php70w-pecl-memcache
- php70w-bcmath
- php70w-devel
- php70w-pear
- php70w-cli

Development tools for pecl / additional system packages:

- curl
- openssl
- memcached
- mod_ssl

```
apt-get install apache2 libapache2-mod-php7 php7 php7-cgi php7-cli php7-gd php7-curl php7-ldap php7-
mysqlnd php7-ssh2 php-pear php7-dev curl openssl memcached php7-memcache
```

MySQL

ProVision 7.x and above require MySQL 5.7 and above.

To install MySQL 5.6.20+, click here...

Add new repository for MySQL 5.7 distribution.

```
apt-key adv --keyserver pgp.mit.edu --recv-keys 5072E1F5
echo "deb http://repo.mysql.com/apt/ubuntu/ trusty mysql-5.7" >> /etc/apt/sources.list.d/mysql.list
```

Update current packages.

```
apt-get update
```

Install MySQL 5.7 to use a local database.

```
apt-get install mysql-server-5.7
```

You will be asked to enter a root password for the mysql server.

MySQL is now installed. You can proceed to next section DNS And Additional Utilities

Install MySQL to use a local database.

```
apt-get install mysql-server
```

You will be asked to enter a root password for the mysql server.

DNS and Additional Utilities

Install the DNS and other utilities:

Click here to expand...

- curl
- openssl

- nmap
- dnsutils
- bind9utils
- dnssec-tools
- expect

```
apt-get install curl openssl nmap dnsutils bind9utils dnssec-tools expect
```

3) Configuring the requirements:

SSL

Enable the necessary apache modules with the following command:

```
a2enmod php7 ssl rewrite headers
```

In order to create self signed certificates and to work properly, the hostname must be resolved properly.
Open the hosts file for editing:

```
    vi /etc/hosts
```

Add a record for your hostname

```
    X.X.X.X yourhostname
```

Create self signed certificate:

```
make-ssl-cert generate-default-snakeoil --force-overwrite
```

Apache

mod rewrite REQUIRED

Please note that mod_rewrite is required! If it is not enabled in Apache, key elements will not work as expected.
Enable the default vhost for apache

```
a2ensite default-ssl
```

Allow overwrites in the apache vhosts

```
sed -i 's/AllowOverride None/AllowOverride All/g' /etc/apache2/apache2.conf
```

Update site configurations to use /var/www for docroot

```
sed -i 's/\/var\/www\/html/\/var\/www/' /etc/apache2/sites-available/000-default.conf
sed -i 's/\/var\/www\/html/\/var\/www/' /etc/apache2/sites-available/default-ssl.conf
```

Restart apache

```
service apache2 restart
```

MySQL

MySQL Configuration:

```
mysql -p -e "SET GLOBAL sql_mode='NO_ENGINE_SUBSTITUTION';SET SESSION
sql_mode='NO_ENGINE_SUBSTITUTION';"
```

Then, enter the MySQL root password when prompted.

4) Optional configurations:

Radius (Optional)

This section only needs to be followed if the customer will be using Radius for authentication.

If you are going to use radius authentication, click here.

Install radius module:

Debian 8:

```
apt-get install php7-radius
```

Create module loading configuration:
(shown using the vi editor, though you may use the editor of your choice)

```
vi /etc/php5/mods-available/radius.ini
```

Add the following lines to radius.ini:

```
; configuration for php radius module
; priority=20
extension=radius.so
```

To enable the radius module, type the following command:

```
php7enmod radius
```

SSH

Install ssh module:

```
apt-get install libssh2-1-dev
```

then run

```
pecl install -f ssh2
```

Create module loading configuration:
(shown using the vi editor, though you may use the editor of your choice)

```
vi /etc/php7/mods-available/ssh2.ini
```

Add the following lines to ssh2.ini (adding extension=ssh2.so to your ssh2.ini):

```
; configuration for php ssh2 module
; priority=20
extension=ssh2.so
```

To enable, type the following command:

```
php5enmod ssh2
```

Reload apache

```
service apache2 reload
```

5) Install 6connect ProVision Software:

The latest version of ProVision can be found at https://cloud.6connect.com/Download/Latest/ and downloaded using the credentials provided to you.

If you need credentials provided to you, or any other assistance, please contact our support team atsupport@6connect.com.

1. Remove the current contents in the ProVision web folder location (currently the www root) and after extract the archive contents (where 7.x.x is the version number for the build):

```
tar -xf productionBuild-7.x.x-php7.0.tar -C /var/www/
```

2. Change the permissions to be the web user permissions

```
chown -R www-data.www-data /var/www
```

3. Go to http://<web root>/install/configTest.php. Follow the provided instructions, correcting any configuration errors if they occur. Once all steps are completed, you are ready to use your ProVision instance!

6connect Local Software Upgrades

Local Software Upgrades

- Local Software Upgrades
  - Upgrade Methods for Local Installs of 6connect Provision
    - Old Method
    - Command Line
    - GUI

Upgrade Methods for Local Installs of 6connect Provision

Upgrades after 4.1.20 and up

You must be running at least 4.1.20 to follow any of the upgrade methods listed below. If you are not yet at this version, upgrade to 4.1.20 using the old upgrade process first, and then continue using the new upgrade process or contact support@6connect.com or any questions or to schedule an upgrade to the latest version.

There are now 3 different methods to run upgrades.

Old Method

(See Upgrades prior to 4.1.19 for detailed instructions)

Download the latest 6connect tar file from https://cloud.6connect.com/Download/Latest/

Extract in web root.

Run the upgrade scripts located in upgrade/scripts in order of version number via php <upgrade-script.php> -v

Command Line

In upgrade/scripts run 'php upgrade.php -h' to get the help and full usage of upgrade.php. This script will automatically get the latest tar file, create a backup, and run all the necessary upgrades between the current and latest version. The most common usage of upgrade will look like this 'php upgrade.php -v -b </path/to/store/backup>'

GUI

In the 6connect tool, navigate to Admin. If there is a new version available, an Upgrade button will be available. Click on the Upgrade Now button to go to the upgrade page. It will automatically download the latest version available, run all upgrade scripts, and create a log of the upgrade process.

Upgrades prior and up to 4.1.19

IMPORTANT NOTE FOR 4.1.15 - The configDir.sh script must be run as root after the 4.1.15 tar file is unpacked and before running upgrade-4.1.15.php.

1. Create a database backup.
mysqldump -u <user> -p<pass> <6connect database name> > /tmp/6connectDBBackup.<date>.sql

2. Create a directory backup. Even if you have offsite backup's with 6connect enabled, perform this step to ensure the most current data is saved.
tar -cvf 6connectFileBackup.<date>.tar /path/to/webroot

3. Move the tar file in 6connect web root.
tar -xof productionBuild-4.1.4.tar
This will place all the new files into your web root directory.

4. Run database upgrades, located in ./dev.
The simple rule of thumb is to run every database upgrade from the version after yours, to the version you want to get to. Here is the short cut list:
If upgrading from 4.1.0 or higher:
php upgrade-4.1.3.php -v
php upgrade-4.1.4.php -v
php upgrade-4.1.5.php -v
php upgrade-4.1.6.php -v
php upgrade-4.1.7.php -v
php upgrade-4.1.8.php -v

php upgrade-4.1.9.php -v
php upgrade-4.1.10.php -v
php upgrade-4.1.11.php -v
php upgrade-4.1.12.php -v
php upgrade-4.1.13.php -v
php upgrade-4.1.14.php -v
configDir.sh <web user> (after tar 4.1.15 tar file unpacked)
php upgrade-4.1.15.php -v
php upgrade-4.1.16.php -v
php upgrade-4.1.17.php -v
php upgrade-4.1.18.php -v
php upgrade-4.1.19.php -v

If upgrading from 3.9.3:
Contact 6connect Support - support@6connect.com

5. Check directory/file permissions for the following and make sure they read/write for the web user:
archive
keys
scans
zones
data/globals.php
images/custom

configDir.sh can be run to correct any permissions issues.

Check the imports directory for read/write permission in the configured php session dir.

6. http://<web root>/install/configTest.php. If there are any configuration errors listed, they must be corrected.

7. Login and use!

Local Installations - Peering Setup

Peering Setup - Local Installations:

ProVision versions 5.3.0 to 6.1.2

For ProVision instances 5.3.0 to 6.1.2...

ProVision uses a locally-hosted mirror of the PeeringDB database in order to perform non-edit Peering functions. There are a few steps to take in order to set up your locally hosted instance to coordinate with PeeringDB information.

As of PeeringDB 2.0, SQL dump files are no longer provided. If you are using ProVision 5.3.0 or higher, you must follow this new install process. If you are using a lower version of ProVision, then please follow the instructions in the previous version of this page.

1) Create a new database to store the PeeringDB data. This **must** be on the same server as the database which is used by ProVision.

2) Download, install, and use the PeeringDB Python Client to populate the database.

The PeeringDB documentation is available here: http://peeringdb.github.io/peeringdb-py/

3) Once this has been done, edit the ProVision global configuration file located here:

```
[ProVision Root]/data/globals.php
```

It must be updated with the following variables to inform ProVision of the location of this new install. The username and password fields correspond to the username and password of the MySQL account which has access to the database (Not the username and password to your PeeringDB account).

This can, but does not have to be, the same MySQL user which is used for the ProVision database. However, the ProVision MySQL user **must** have at least READ access to the PeeringDB database.

```
$peeringdb_host = 'localhost';          // Database host must be the same for provision and
peering
$peeringdb_username = 'username';       // username for the MySQL user
$peeringdb_password = 'password';       // password for the MySQL user
$peeringdb_db_name = 'peeringdb';       // name of the database in MySQL
```

4) Periodically sync with the PeeringDB server to get the latest updates. This can be done manually, or there are instructions in the PeeringDB documentation on how to automatically schedule syncs using cron (http://peeringdb.github.io/peeringdb-py/cli/#sync).

ProVision versions 7.x

For ProVision versions 7.x and later, ProVision directly interfaces with PeeringDB's API to update exchange and peering data, caching the data for a default time of 12 hours.

This requires a PeeringDB account, and for the account credentials to be set in ProVision. The credentials may either be hard coded into globals.php, or set into the database via the Admin/Peering GUI.

Additional Peering constants may be added into globals.php to change the PeeringDB URL between the main and beta site (some users may find the beta site to have faster response times), and to adjust the PeeringDB cache TTL.

For real-time updates, TTL may be set to 0. However, some users may experience severe lag with a TTL = 0; we recommend using a 10 to 15 minute or greater TTL if this occurs.

Peering Constants

In globals.php, the following constants can be defined to tweak the Peering internals:

Peering Constants Descriptions and Syntax...

PEERINGDB_USERNAME

```
define('PEERINGDB_USERNAME', 'username');
```

Default value: none

The username for the account used to connect to the PeeringDB API

Instead of saving the username and password in the database, the values can be hard coded into globals.php

PEERINGDB_PASSWORD

```
define('PEERINGDB_PASSWORD', 'mypass');
```

Default value: none

The password for the account used to connect to the PeeringDB API

PEERINGDB_URL

```
define('PEERINGDB_URL', 'https://peeringdb.com/api/');
```

Default value: https://peeringdb.com/api/

The URL of the PeeringDB API. Alternate value: https://beta.peeringdb.com/api/

PEERINGDB_CACHE_TTL

```
define('PEERINGDB_CACHE_TTL', 43200);
```

Default value: 43200 (12 hours)

How often (in seconds) to purge the cached PeeringDB API calls. If a customer wants real time access, this can be set to 0.

If experiencing major lag issues with real time access, it is recommended to increase to increase the cache TTL from 0 to 5, 10, or 15 minutes.

Additional Information:

For additional information on working with Peering, see the following documentation sections:

Peering v2

Import Sessions

ESXi VM Image Setup

Step 0

Ensure that you have the root user information for the VM and a license key - please contact 6connect Support (support@6connect.com) or your Reseller/VAR for more information.

Step 1

Download OVF and VMDK files from 6connect (or approved) download area. If you do not have this information, please contact 6connect Support (support@6connect.com) or your Reseller/VAR for more information.

Step 2

Option 1: Import OVF and VMDK to your ESXi VM environment (VMware ESXi 6.5 or later).

Option 2: If you do not have ESXi 6.5 or later, then you can create a new VM (CentOS, 64-bit, 4GB RAM) and use the VMDK disk image that you downloaded in Step 1.

Step 3

Power on the VM and login using the root user account information from Step 0

Step 4

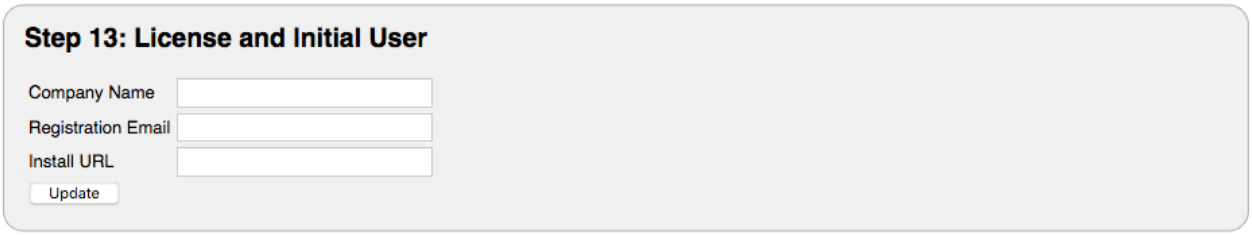
Customize the VM environment for the following items:

1. Change the root user password from its default
2. edit /etc/sysconfig/network-scripts/ifcfg-ens192 with your IP information
3. edit /etc/hostname
4. edit /etc/resolv.conf
5. Optional - add SSL certs (non self signed) per normal CentOS7 installation
6. Reboot the VM

Step 5a

Finish setup of ProVision for your organization:

1. Go to https://<hostname or IP address>/install/configTest.php in a web browser
2. Scroll to the bottom to complete Step 13 (see image below). The Registration Email is your License Email that would have been sent to you by 6connect or ProVision VAR. If your VM is in an "offline" or SCIF environment please go to Step 5b.



3. You should now be able to go to your Instance URL and login with the credentials generated for your user/registration email address.

Step 5b

For an "offline" license, you will be emailed a license file and path to place it. Without this file, you will not be able to login to ProVision. One the license file is applied, you should now be able to go to your Instance URL and login with the credentials generated for your user/registration email address.

Please note that the offline license file is version specific, so before you upgrade to a new version of ProVision, you will also need to make sure you update the license file after the upgrade.

ProVision Getting Started

Welcome to ProVision!

Our Getting Started documents provide an overview of concepts to orient you to working in ProVision. Below are some of the resources available. If you need setup assistance or additional information, you can contact our support team at support@6connect.com.

- Welcome to ProVision!
  - ProVision Getting Started
  - ProVision User Guide
  - ProVision Admin Guide
  - ProVision Developer Tools
  - Additional Resources

ProVision Getting Started

First Steps - Not sure where to begin? Here are some key items to consider when setting up your first ProVision instance, high level overviews of Resources, Permissions, and Importing data into ProVision.

Resource Concepts - The Resource Management System is a key component of ProVision. This system supports a variety of hierarchies and metadata - understanding how these pieces can be used is important prior to importing data or setting up the application.

Workflow Concepts - ProVision has two distinct interfaces depending on the user level and task. It is important to understand how these interfaces work together from the centralized data. This is important for user on-boarding and training of internal operations staff, developers or engineering teams.

UI Element Legend - ProVision has some UI elements that you should be familiar with for easy day to day operation.

ProVision User Guide

The user guide gives you an overview on the standard UI functions of ProVision and installed Modules.

ProVision Admin Guide

The Admin Guide provides an overview of administrative functions of the different functional areas of ProVision.

ProVision Developer Tools

The Developer Tools section has details on our API and related information - including code samples.

Additional Resources

You can also browse the Tutorials and FAQ, if you have any questions, please contact our support team at support@6connect.com.

First Steps

ProVision First Steps

- ProVision First Steps
  - Overview Video: First Steps - Part 1
  - Before you Begin
    - 1) What type of physical and non-physical components do you wish to track?
    - 2) What is your current data structure? What is your ideal data structure?
    - 3) Who needs access to what data?
  - Overview Video: First Steps - Part 2
  - Getting your Data into ProVision
    - 1) Gather and Prep your data
    - 2) Import or Manually Add Data
      - A) Resources
      - B) IP Aggregates and Blocks
      - C) Add DNS servers and zones
    - 3) Add Users and Groups
  - Working In Provision

Overview Video: First Steps - Part 1

This video gives a high level overview of the "Before you Begin" content on this page, as well as an introduction to ProVision's Resource System.

Note: Recorded in ProVision v5.1.x, newer versions may have interface updates.

This video may also be viewed at https://www.youtube.com/watch?v=apJRcQv3ZQ0.

Before you Begin

We recommend that new users work through the following questions with their internal team to plan their ProVision instance:

Need Help?

Remember that 6connect's engineers are here to help. If you have questions, or want to test out some ideas, our team has worked with a variety of data sets and can help get you started on the right path. You can reach us any time at support@6connect.com.

1) What type of physical and non-physical components do you wish to track?

Impacts: What Sections and Resources are created

An important first step is determining what items you currently are, or will be, tracking, and what relationship they have with each other. ProVision's flexible Resource system allows you create and customize detailed entries for any type of item you may want to track: Customers, Contacts, Data Centers, Routers, VMs, and more. These types of resources are labeled as "Sections" in ProVision. Once a section is made, individual items (resources) may be created as a part of that Section. Each Section may have different Gadgets selected, which then provide additional functionality on a resource entry page.

See: Working with Resources, Customizing Sections, Gadgets

2) What is your current data structure? What is your ideal data structure?

Impacts: Resource Hierarchy, Assignment Behavior

The resource hierarchy structure in ProVision allows for "child" resources to be created under a parent resource (for example: servers as children under a datacenter resource entry, or subsidiaries under a parent company, who then share IP aggregates). The structure decided upon will influence how resources are set up in ProVision, as well as the behaviors of functions while working with items such as IP blocks or DNS zones.

See: Working with Resources, Resource Concepts

3) Who needs access to what data?

Impacts: User and Group Permissions

In ProVision, standard user permissions are set by resource and functional area (IPAM, Resource, Peering, etc). Global Admin permissions give access to additional functions such as configuration settings, importing, and Scheduler tasks. Determine which users will require administrative

access, and which will have access to only specific resources or functional areas. You will need to get more specific later, but having some high level groups to work with is a great start.

See: Users & Permissions

Overview Video: First Steps - Part 2

This video gives a high level overview of the "Getting your Data into ProVision" content on this page.

Note: Recorded in ProVision v5.1.x, newer versions may have interface updates.

This video may also be viewed at https://www.youtube.com/watch?v=2e0H1H4rTTs.

Getting your Data into ProVision

After determining your internal goals and processes, it's time to get your data into ProVision!

Start Small

When importing data into ProVision, data validation is a key step to ensure that everything is accurate. Upon importing your data, you may see some errors that result in a stop in the import process! It is recommended that you break up your imports to both keep them manageable and give you a chance to normalize your data prior to importing.

1) Gather and Prep your data

Determine where your data will be from:

Excel / Spreadsheets:

May be used to import: Resources, IP Aggregates/ Blocks, DNS BIND Zones

If you currently use Excel or other spreadsheet program for tracking, you will need to verify that your spreadsheets are "cleaned up" according to the information on the Importing Your Data page under "Preparing for Data Import". Make sure that you use UTF-8 encoding, remove extraneous blank rows, and compare your data to the data fields available in ProVision, shown under "Which Import Tool Should I Use?". Review sample files if desired to see example formats. If you plan to track custom types of Resources by creating Sections, you will need to create a Section with custom fields, and verify that your spreadsheet contains the same fields.

Once your verification / cleanup is complete, export your spreadsheet as a .csv file.

See: Importing Your Data, Working with Resources, Import DNS Zones

RIR

May be used to import: IP Aggregates

No advance preparation is needed for aggregates imported from RIR. ProVision's built-in importer will ask for your ORGID or an IP, and then populate an aggregate list from that information. Simply choose which aggregates you wish to import.

See: Import Aggregate Blocks

DNS Servers

May be used to import: DNS Zones

ProVision provides automated tools for importing DNS zones from the following server types: BIND, PowerDNS, InfoBlox, NS One, Dyn DNS, and DNSMadeEasy servers, as well as IPPlan MySQL Databases.

Before importing DNS zones, it is recommended that a DNS Group be created to hold the zones being imported, and that the DNS server be added into ProVision so that zone updates may be pushed.

For general DNS taks, see: DNS Administration, DNS Tab, Working with DNS Groups, Working with DNS Servers

For specific DNS zone import instructions for each DNS server type, see: Import DNS Zones

Manually Adding Data

ProVision allows manual adding of data at any time. We recommend verifying that the item has not already been added beforehand (to prevent duplicates), and keeping your desired data structure in mind.

2) Import or Manually Add Data

The order in which items are added will depend on what ProVision functional area (Resources, IPAM, DNS) you will be using, and what the current / desired data structure is. In general, the following order is recommended:

A) Resources

Importing or creating your resources first allows subsequent items to be associated with those resources.

Note: If you already have "Resource" data associated with your IP block data (ie, as fields in a spreadsheet with Resource Name and Resource ID), you may choose not to create those resources ahead of time. The IP Import from CSV tool will give you an option to create those Resources during the IP import process.

Import customers, physical devices, locations, and so forth through the Resource Import from CSV tool. If you wish for additional customization, you can create a custom Section, add the desired fields, and import resources under that Section through the Resource Import Tool.

Adding resources manually may be done at any time under the **Resources** Tab, by clicking the "Add Entry" button.

See: Working with Resources, Importing Your Data

B) IP Aggregates and Blocks

Import your IP Aggregates through the Import from RIR tool, or from a .csv file via IP Import from CSV.

The Import from CSV tool will create Top-Level Aggregates and place blocks under those aggregates based on the following method:

First, the importer will parse through the provided data, order all blocks from largest to smallest, then attempt to split the largest block out of an existing block matching the IP space and RIR. If that fails (no larger block exists), then that block is added in the system as its own Top-Level-Aggregate. Subsequent blocks will undergo the same process.

This method ensures that your list of IP blocks does not need to be organized in any particular order.

If your import includes a large number of small blocks (/30s, /32s), be sure you've included at least one large block which encompasses them (/24, /22). This will ensure that the smaller blocks are neatly organized under the larger block, rather than imported as their own Top-Level Aggregates.

See: Importing Your Data

C) Add DNS servers and zones

If using the DNS functionality in ProVision, you will need to add your servers prior to importing zones.

Add DNS Severs

Adding DNS Servers requires administrative access.

Servers are added under the DNS Tab of ProVision, under "DNS Servers". Click on "Add Server" and fill out the server information.

See: Working with DNS Servers

Information for specific server types and options is available under the following sections:

Configuring ISC BIND Support

Configuring PowerDNS Support

Configuring Secure64 Support

Configuring Split Horizon/Views

Configuring DNSSEC

Import DNS Zones

After creating the applicable servers in ProVision, you may import or manually add DNS zones.

ProVision offers multiple DNS zone import options, available under the Data Import tab in the Admin section. For more information on importing DNS zones, see Importing your Data and Import DNS Zones.

BIND Zone Import

- Imports using the named.conf configuration file tied to the zones you are uploading, a .zip or .tar file of the zones themselves, and an optional .csv file mapping zones to customers.

PowerDNS Zone Import

- Option is available after configuring a PowerDNS server with a MySQL backend. Connects to the selected server and imports all zones.

InfoBlox Zone Import

- Imports DNS zones using a provided Host, Username, and Password. The InfoBlox import pulls all zones on the InfoBlox LOCAL grid and adds them to a designated Group. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

NS One Zone Import

- Imports DNS zones using a NS One API Key. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

Dyn DNS Zone Import

- Imports DNS zones using a Dyn DNS Customer Name, Username, and Password. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

DNSMadeEasy Zone Import

- Imports DNS zones using a DNSMadeEasy API Key and API Secret. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

IPPlan Zone Import

- Imports DNS zones using IPPlan MySQL database options. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

Manually adding zones may be done at any time from the **DNS** Tab. See Working with DNS Zones - Common Tasks and Editing DNS Zones.

3) Add Users and Groups

In ProVision, the permissions structure is handled by assigning users to groups, then setting specific resource-level C/R/U/D permissions for that group. It is designed to give you as much flexibility as you need to accommodate most use cases. When mapping out the permissions structure for your organization, keep in mind who you want to access to application:

- Internal Users and Roles (Admins, Read Only, etc.)
- Partners related to multiple specific Resources/Accounts
- Customers/Departments with limited view to only their respective Resources/Accounts

See: Users & Permissions, Users and Groups

Working In Provision

With the basic data now imported, and users set up, you (and your team) are ready to work in ProVision!

Refer to our User Guide and Admin Guide for details on standard user and admin level areas of ProVision. Or, follow the links below for additional details grouped by specific task areas:

Concepts:

Resource Concepts

Workflow Concepts

Resources:

Working with Resources

Customizing Sections

Customizing Fields

Gadgets

IPAM:

IPAM Tab

Working with IP Blocks

Working with IP Rules

IPAM Administration

DNS:

DNS Tab

Working with DNS Groups

Working with DNS Zones - Common Tasks

Editing DNS Zones

DNS Administration

Working with DNS Servers

DHCP:

DHCP Tab

Working with DHCP Groups

Working with DNS Zones - Common Tasks

Copy of DHCP Tab

DHCP Administration

Peering:

Peering v2

Peering Exchanges

Peering Routers

Peering Sessions

Import Sessions

Resource Concepts

The Resource System

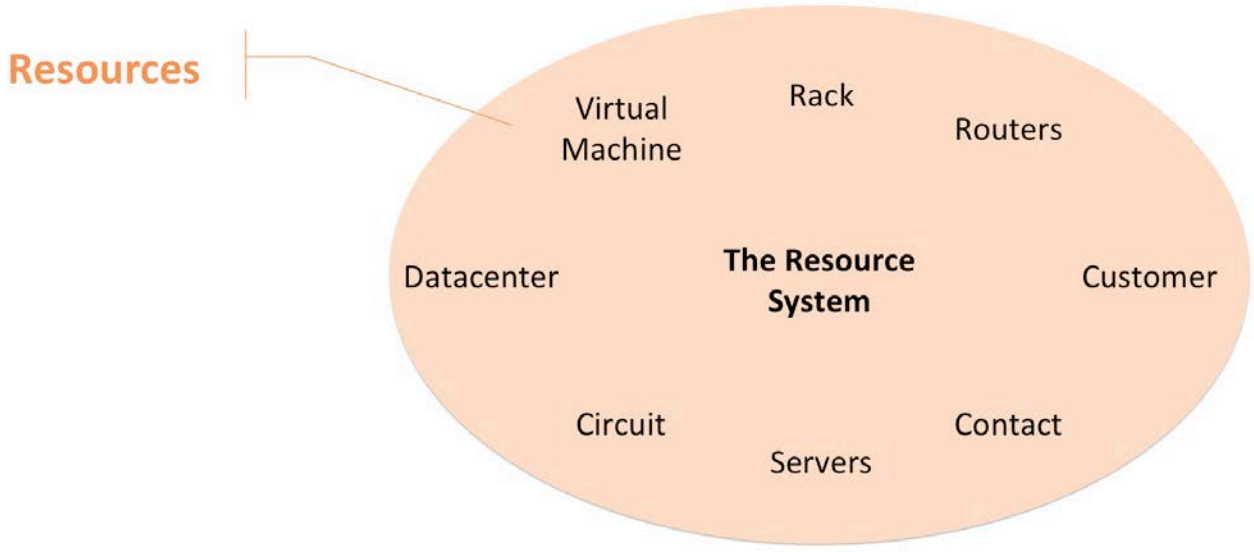
- The Resource System
  - Resource System Components
    - Resources
    - Section
    - Entry
    - Category
  - Resource Hierarchy
    - Additional Information:

Resource System Components

Resources

In ProVision, a “Resource” is simply an umbrella term for the components that you are tracking. Resources may be:

- Physical assets such as servers and routers
- Virtual assets such as DNS Zones or VMs
- Individual people or companies (customers)
- Places holding your assets, such as Datacenters

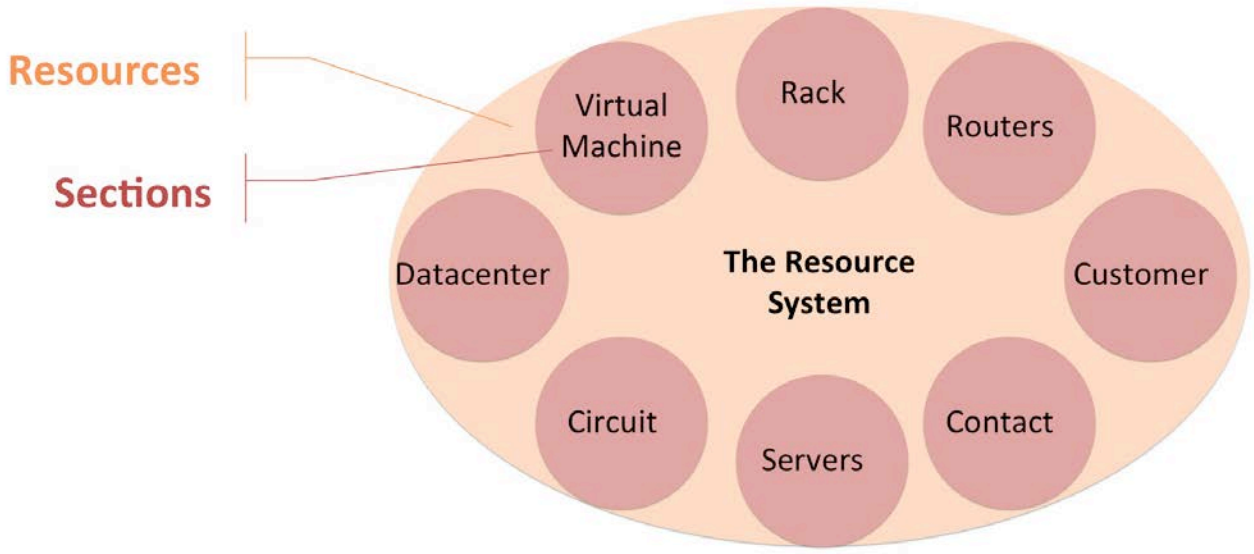


Section

The generic type of a resource is called a Section. General terms such as Rack, Routers, Customer, Servers, and Datacenters would be considered Sections in ProVision.

You might have an individual server named "Test Server", but its generic type of resource is that of a “Server” – so we would associate “Test Server” with the Section “Server”.

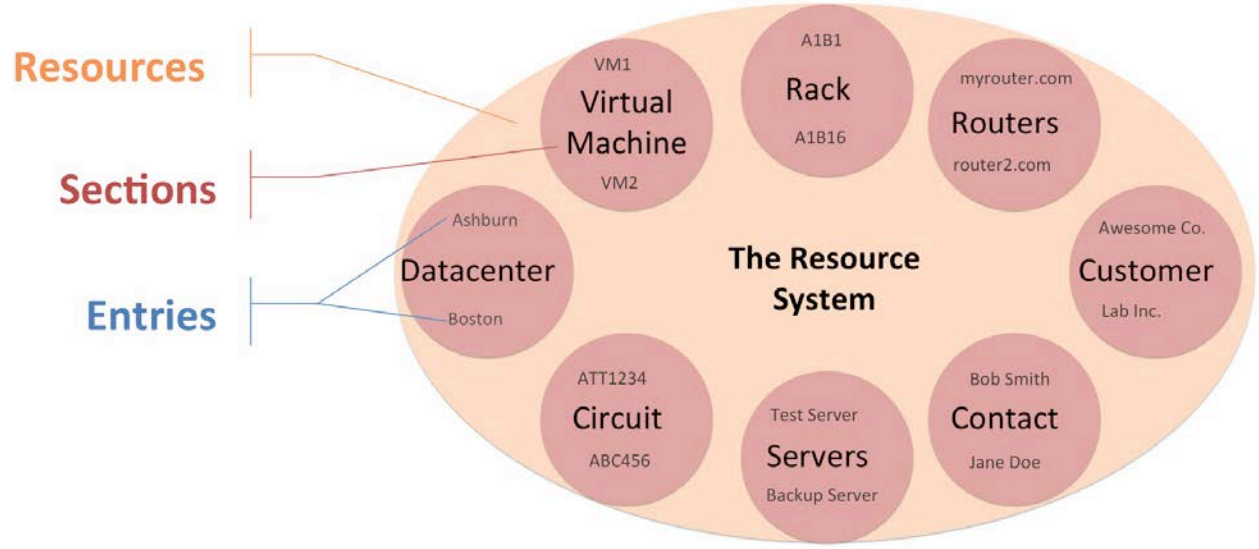
Think of Sections as templates for your resources. When a Section is created, you can associate specific fields and gadgets with that Section. In the case of our "Server" section, we would want to associate fields that are common server properties, such as Make, Model, Operating System, and Domain. Then, anytime we looked up a specific server in ProVision, we can see and update the Make, Model, OS, and Domain information.



Entry

Think of an Entry as the individual item you are tracking – it has a specific name, it might have an IP Address, a physical address, or additional descriptive data that is associated with it.

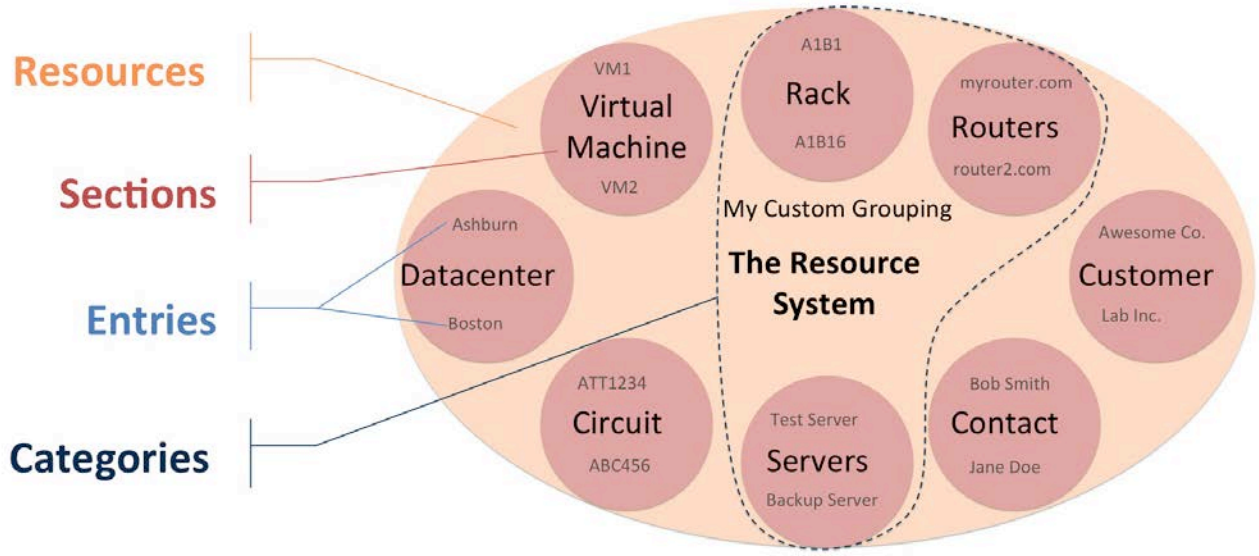
Our "Test Server" resource is an Entry under the "Server" Section.



Here, we can see that under the “Datacenter” section, we have two entries – one is the Ashburn Datacenter, the other is the Boston Datacenter. Under “Servers”, we have entries for “Test Server” and “Backup Server”.

Category

Another part of ProVision’s resource system is “Categories”. Categories allow you to group resources together under a name of your choosing to further organize your resources. A category that is often used is “Customer”, grouping customer resources together

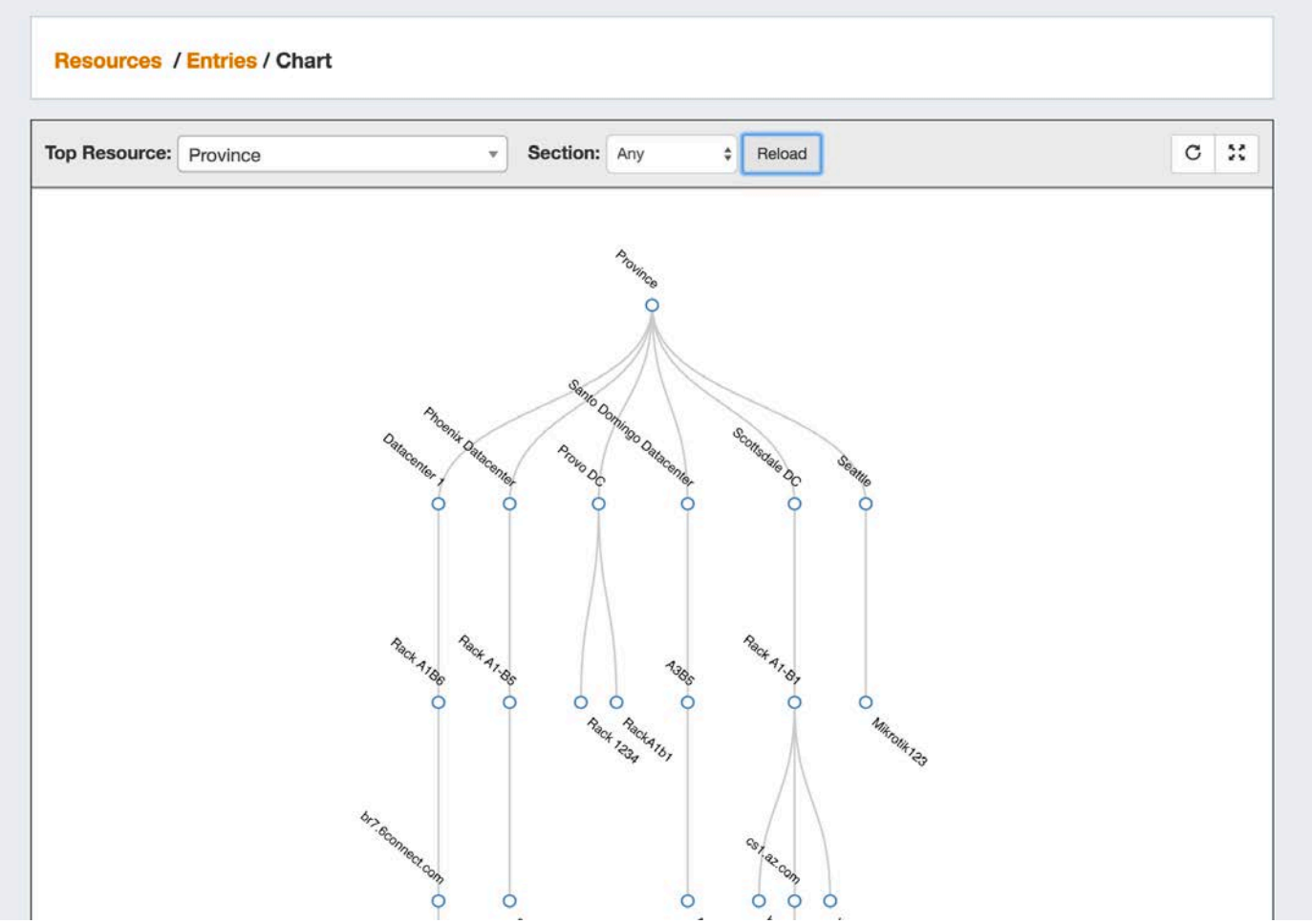


In this example, we are grouping Racks, Servers, and Routers under the “My Custom Grouping” Category.

Creating categories is entirely optional, so you can decide internally if the additional organization is needed in your instance.

Resource Hierarchy

In ProVision, "Child" Resources can be created underneath a "Parent" resource. This functionality allows for hierarchies to be created to match the desired organizational structure. An example of this would be to create racks and servers as children under a datacenter entry. Chart View, available from the **Resources** Tab, lets you see this structure graphically.



The structure decided upon will influence how resources are set up in ProVision, as well as behaviors of functions while working with items such as IP Blocks or DNS zones.

Additional Information:

Working with Resources

Working with Entries

Customizing Sections

Customizing Fields

Gadgets

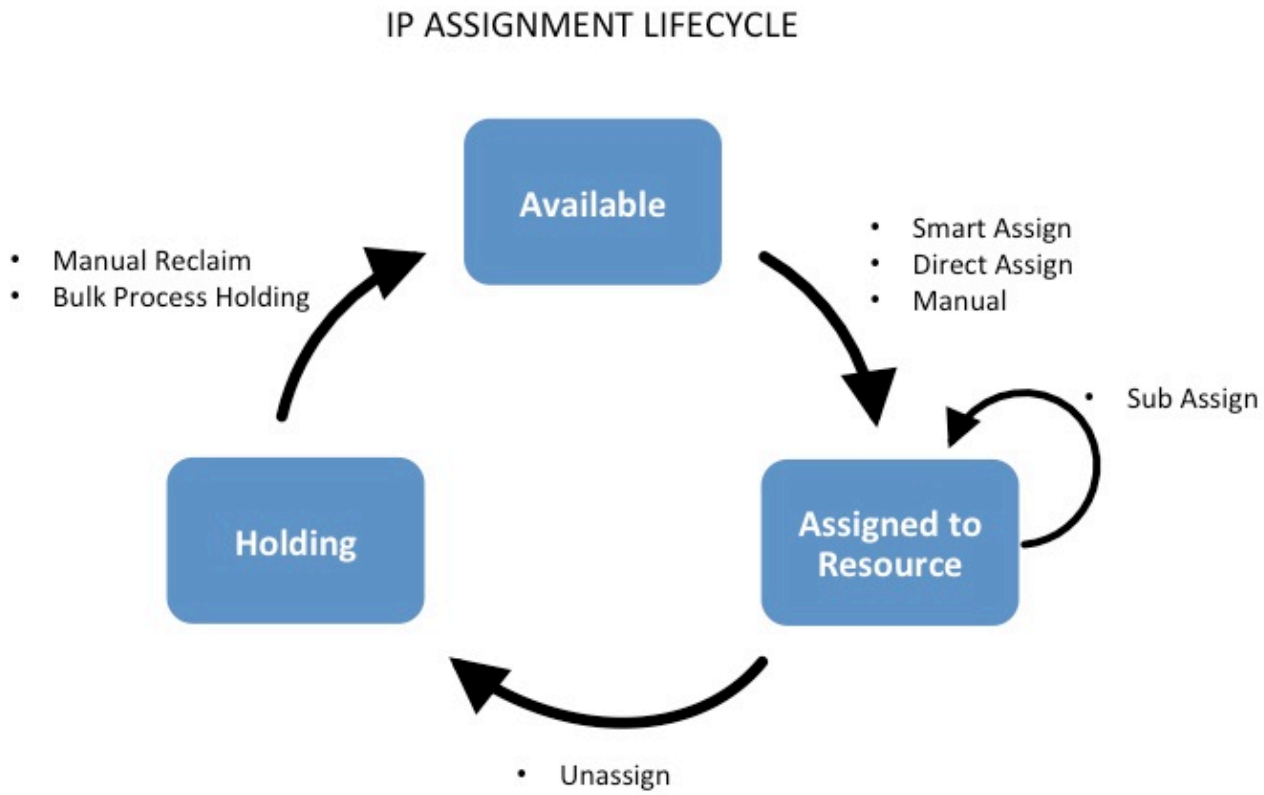
Workflow Concepts

Workflow Concepts

- Workflow Concepts
  - IP Assignment Lifecycle
  - IP Management
  - Peering
  - VLAN Manager
    - Standard flow (without VLAN tags):
    - Optional flow (with VLAN tags):
  - DHCP
  - DNSv3 Workflow

IP Assignment Lifecycle

In ProVision, the IP assignment lifecycle starts with an available block which is free to be assigned to any IPAM-enabled resource holder. There are multiple methods that may be used to assign a block to a resource holder: Smart Assign, Direct Assign, or Manual Assign (Smart Browse). Once an IP block is assigned, blocks can be further subassigned via the same methods if desired. When an assigned block is un-assigned it proceeds into the Holding Tank: a special resource where blocks are held until either a set time has elapsed or until they are manually reclaimed to 'available' status.



For more information on performing tasks in this IP Assignment Lifecycle, see the following documentation sections:

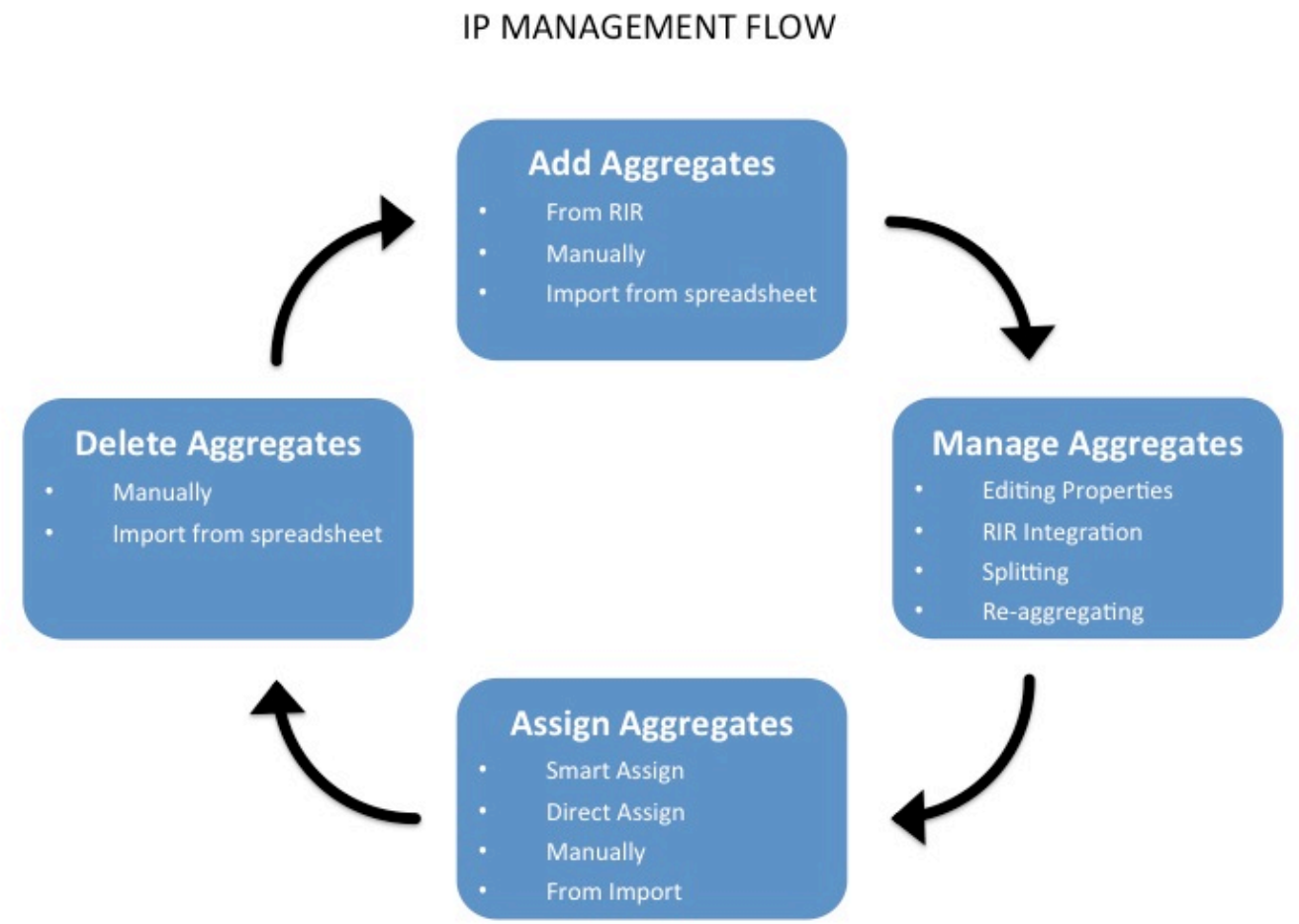
Working with IP Blocks

IPAM Administration

IP Management

IP Management is comprised of four basic functions: adding aggregates into ProVision, managing those aggregate blocks, assigning them to a resource, and deleting the aggregates.

ProVision provides multiple ways for you to achieve each step, depending on your needs. For example, if your organization currently uses spreadsheet data to track aggregates, ProVision provides tools that can import your existing spreadsheets for bulk updates, saving you time. Need to just quickly assign a single IP? Direct Assign will allow you to do so with just a few clicks.



For more information on performing tasks in this IP Management Flow, see the following documentation sections:

Working with IP Blocks

IPAM Administration

Importing Your Data

Import Aggregate Blocks

Peering

ProVision versions 7.0.0 and later:

In ProVision, Peering starts with designating Routers. Routers may be added through the Resource system under a router Section, but the simpler approach is to add a new router via the "Add Router" button under the Peering Tab Routers sub-tab. Through this dialog, Peer Groups may be added at the same time.

After the router(s) and Peer Groups have been created in ProVision, Sessions need to be added. Sessions may be added manually through the "Add Session" dialog in the Peering v2 tab, or Imported from a router (requires ProVision Admin permissions).

Once a session has been created, it can be configured and managed through the Sessions list and Session details page, or selected and configured for a specific Peer via the "Configure Multiple Sessions" option.

ProVision versions 6.2.1 and earlier:

Click here to expand...

PEERING WORKFLOW





For more information on performing tasks under Peering, see the following documentation sections:

Peering:

Peering

Peering Exchanges

Peering Routers

Peering Sessions

Import Peering Sessions

VLAN Manager

The VLAN Manager allows Admin users to add domains and VLANs to their ProVision instance, and associate them with IP Blocks. Optionally, VLAN tags may be associate with ranges or individual VLANs under a domain, to help limit VLAN search results when enabling VLANs.

Standard flow (without VLAN tags):

The workflow starts with creating a domain in the VLAN Admin section of ProVision. During creation, domains may be selected as "standard" or "extended" domains, declaring the size of the VLAN pool from which VLANs are available to be enabled. Next, VLANs must be enabled and added to the domain. This is done under the IPAM Tab -> VLAN section of ProVision. Add VLANs to the domains by clicking "Add", searching for the desired range of VLANs to enable from the standard / extended pool, and selecting the desired VLANs to enable. Once enabled, VLANs may be edited or have IP blocks associated with that VLAN. Editing VLANs and Direct / Smart Browse / Search IPs functions for adding blocks to VLANs are available from the Domain/ VLAN list under under the IPAM Tab -> VLAN section of ProVision by expanding the desired domain and clicking on the VLAN link. IP blocks may also be edited individually through the IPAM gadget, IPAM Manage, and VLAN Manage areas to add domain and VLAN information to the block.

Optional flow (with VLAN tags):

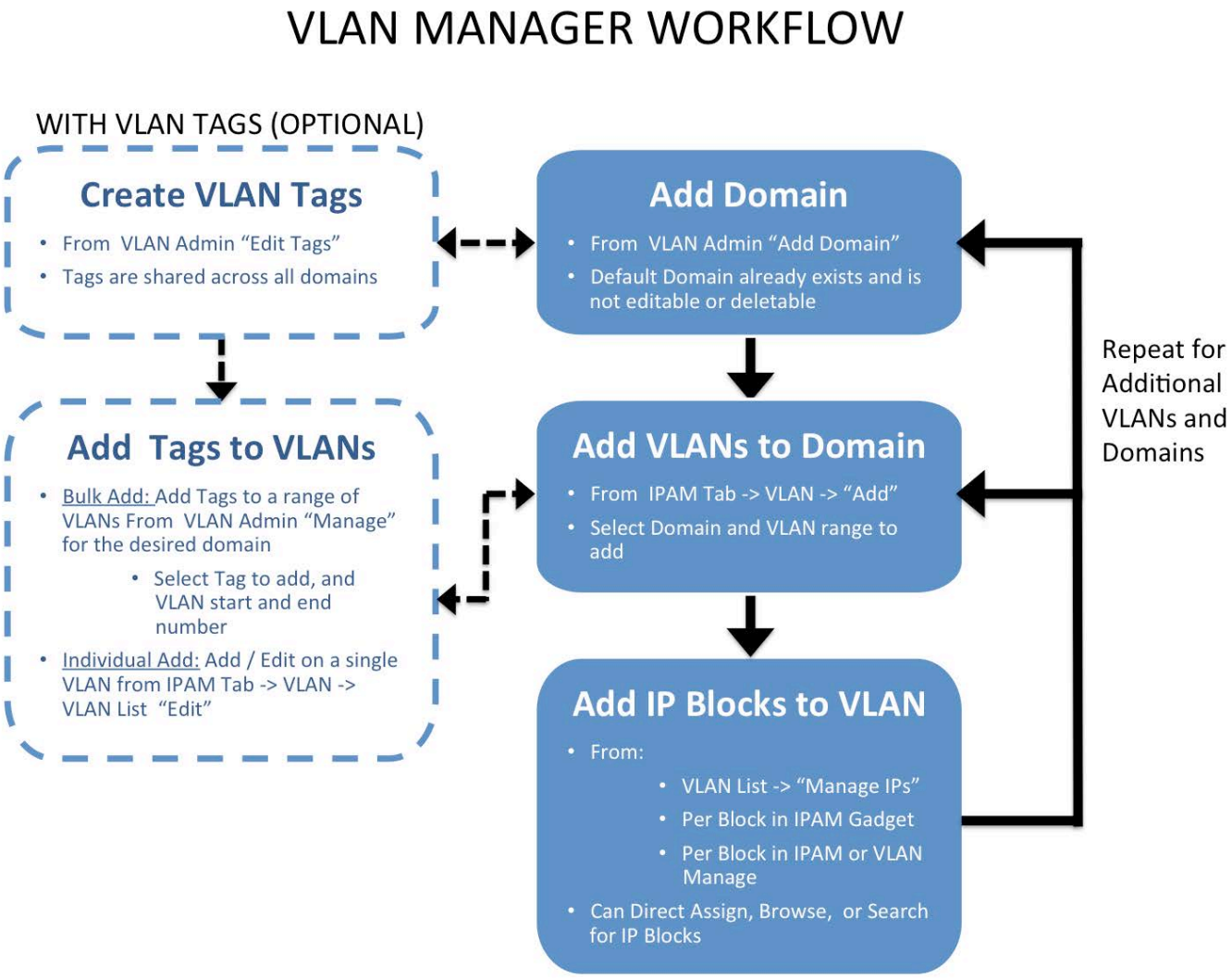
The workflow starts with creating a domain in the VLAN Admin section of ProVision. During creation, domains may be selected as "standard" or "extended" domains, declaring the size of the VLAN pool from which VLANs are available to be enabled. VLAN tags may be added (from the VLAN Admin Tab -> Edit Tags submenu) before or after domain creation. Tags created in this area are available to all domains and VLANs.

If the tags created need to be added to a large range of VLANs, the next step would be to add tags to the desired range(s) of VLANs from the VLAN Admin -> Domain "Manage" button. Domain Manage allows you to select an existing tag, type in the numeric VLAN start and end point, and assign that tag to that entire range of VLANs under the current domain.

Once a bulk range of tags is associated with VLANs, you can use the tag as a search criteria when adding VLANs to a domain. This is done under the IPAM Tab -> VLAN section of ProVision. Add VLANs to the domains by clicking "Add", using the tag name as a search field for the desired range of VLANs, and then selecting the desired VLANs to enable.

If tags are only desired for individual VLANs, they may be added when editing a VLAN after the VLAN has already been added/enabled for the domain. Editing VLANs may be accessed from the IPAM Tab -> VLAN section, then expanding the desired domain and clicking on the VLAN link, selecting "Edit".

Next, VLANs must be enabled and added to the domain. This is done under the IPAM Tab -> VLAN section of ProVision. Add VLANs to the domains by clicking "Add", searching for the desired range of VLANs to enable from the standard / extended pool, and selecting the desired VLANs to enable. Once enabled, VLANs may be edited or have IP blocks associated with that VLAN. Editing VLANs and Direct / Smart Browse / Search IPs functions for adding blocks to VLANs are available from the Domain/ VLAN list under under the IPAM Tab -> VLAN section of ProVision. IP blocks may also be edited individually through the IPAM gadget, IPAM Manage, and VLAN Manage areas to add domain and VLAN information to the block.



DHCP

For ProVision versions 6.2.1 and earlier:

Click here to expand...

The DHCP Workflow in ProVision begins with denoting the DHCP IP Aggregate(s) from the IPAM Tab -> "Add Aggregate". When creating DHCP aggregates, ensure that the block is designated as 1918 space, and has the check boxes next to "DHCP Aggregate" and "Allow Sub-assignment" selected. This will ensure the block is automatically added to the DHCP Available Resource, and is usable when defining DHCP Pools.

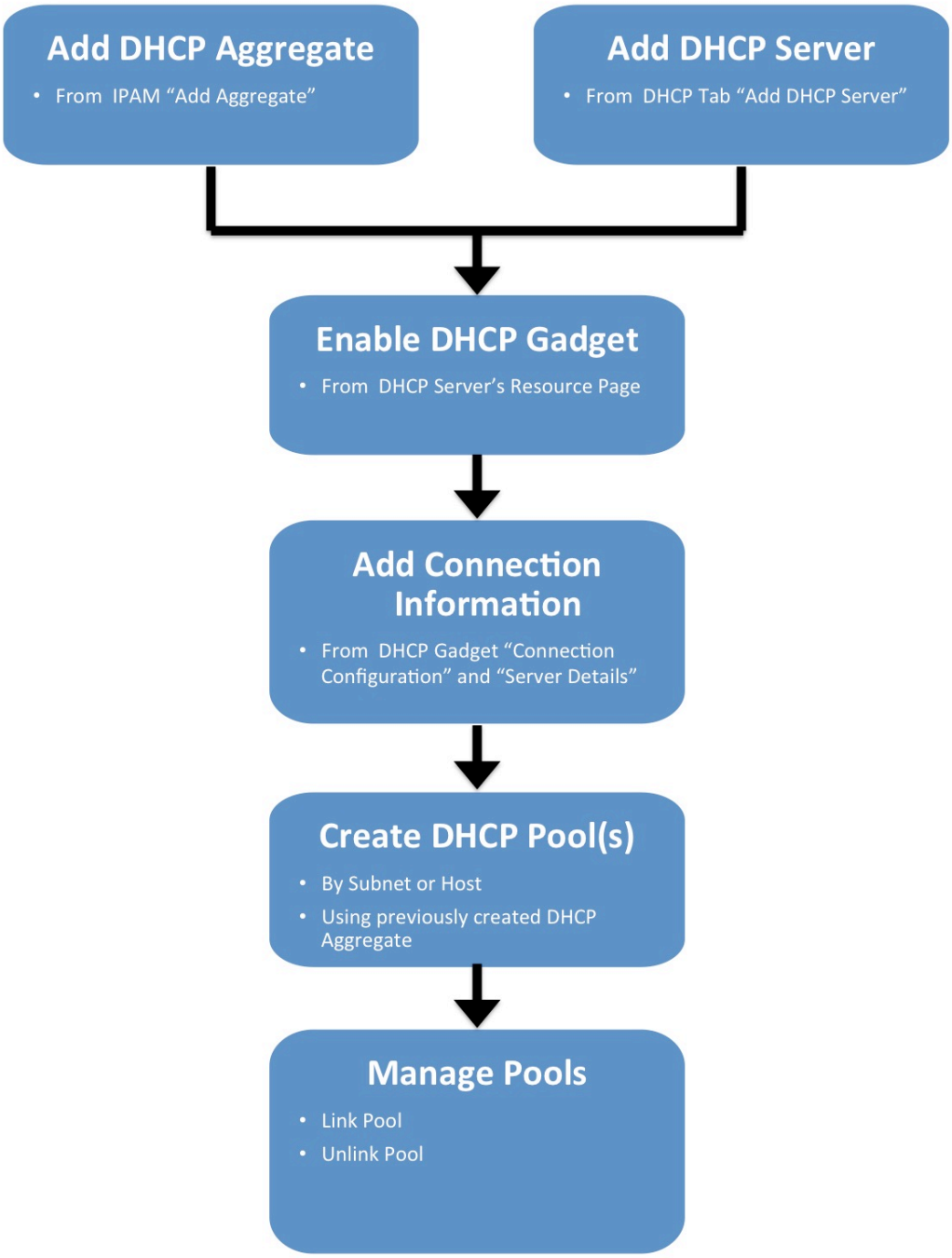
Separately, ensure that the DHCP server is added into ProVision. New DHCP servers may be added from the DHCP Tab "Add DHCP Server" button. Type the server name, then choose the resource to which the DHCP server belongs. This creates a hierarchical relationship, with the server as a child resource under the selected parent.

Once the server has been created in ProVision, click on the server name in the DHCP server list to go to the server's Entry Page. If the server was created through the DHCP Tab, the DHCP Management Gadget will already be enabled for the DHCP server. If created elsewhere, the DHCP Gadget may need to be enabled by simply selecting the "On" radio toggle next to "DHCP Services", and clicking the "Update" button. Next, add in the connection configuration and server details for the DHCP server in the DHCP Gadget.

Now that the DHCP server is set up and DHCP aggregates created, DHCP Pools may be created and managed. Pools may be created by subnet or host. Creating by subnet allows for a block to either be directly assigned manually, or for ProVision to SmartAssign a block based on the provided criteria. Creating by host requires providing a host and MAC address, and then likewise may have an IP directly assigned or

Smart Assigned. DHCP Pools, once created, are managed by linking or unlinking to the DHCP server. IP assignments and pool details may be changed at any time by clicking on the pool name, editing the details as desired, then clicking "Save".

DHCP WORKFLOW



For more information on working with DHCP in ProVision, see the DHCP Tab documentation.

DNSv3 Workflow

DNSv3 revolves around Groups. Zones are gathered under Groups, servers attached to Groups, and pushes may be done on a per Group level. Thus, the first workflow step in DNSv3 is to set up one or more DNS Groups. A "Default Group" is automatically provided in ProVision, but other Groups may be desired to organize zones and default values.

To create a new DNS Group, click the "Add Group" button from the **DNS Groups** tab. Enter the desired default values for the Group, and save. If only using the Default Group, ensure the default parameter values are set as needed. For more information, see Working with DNS Groups.

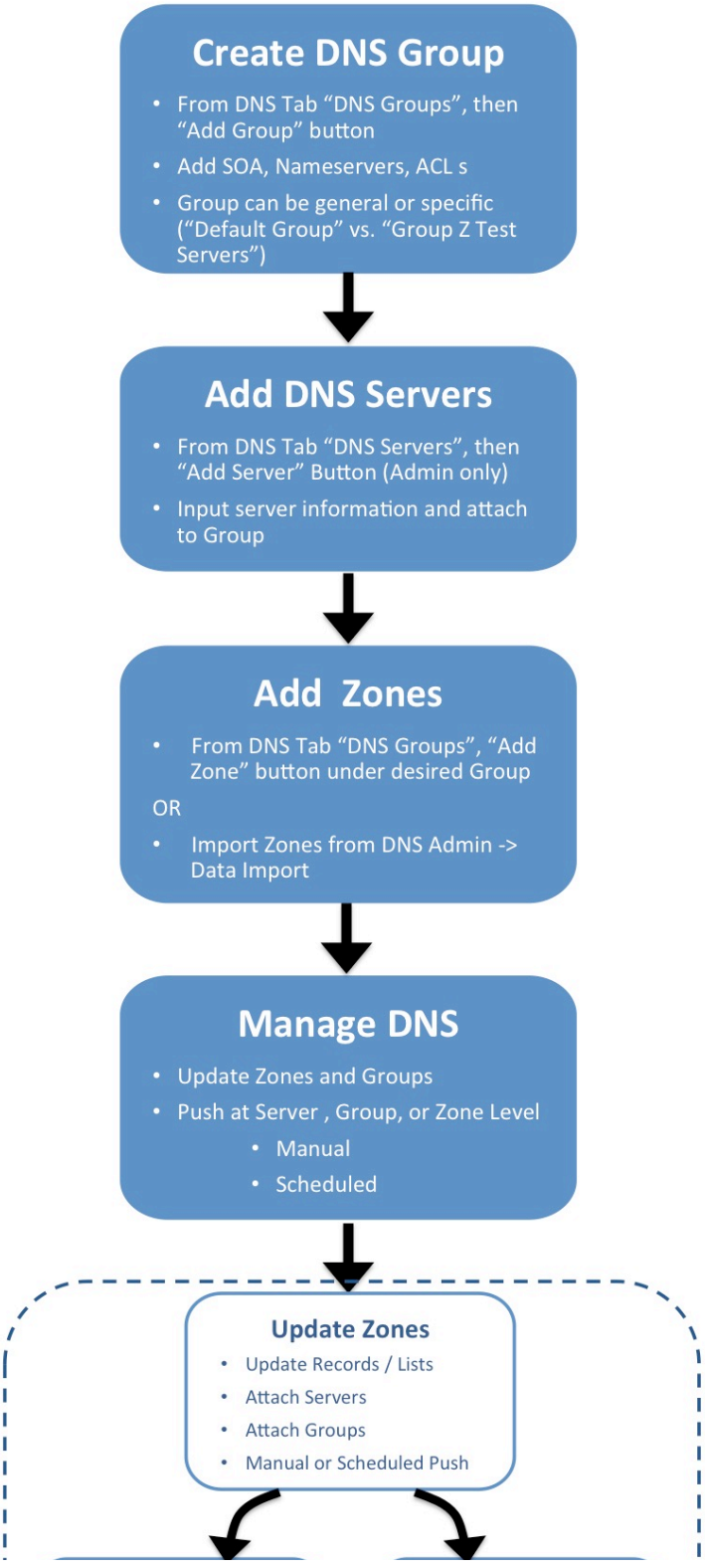
After Groups have been set up, DNS servers should be added or settings verified. Admin users may add DNS servers from the **DNS Servers** tab "Add Server" button. Input the server information and save. Existing servers may be reviewed and edited by clicking on the server name in the DNS Server List. Once a server is created in ProVision, it may be attached to any DNS Group under the Group's "Attached Servers" module. Attaching a server to a Group will allow for zones in that group to be pushed to the attached server(s). See Working with DNS Servers.

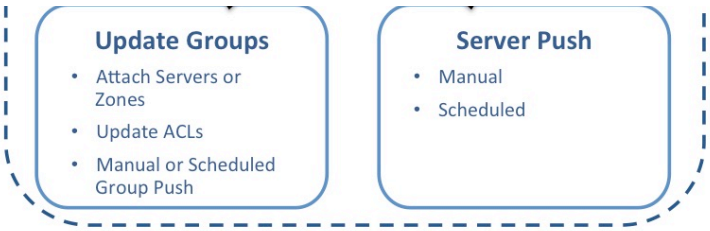
Next, add zones to your groups. Zones may be manually added under each group by clicking the "Add Zone" button, or it may be imported via DNS Importers into a selected Group. Add the zone and record information, and save. See Working with DNS Zones - Common Tasks for additional information.

Zones may be only exist once per Group, but may be duplicated under multiple Groups. Zones may also be moved from Group to Group as needed.

At this point, all major components of the ProVision DNS system have been added - from here management tasks take over. Zones may be updated and moved to or from Groups; Groups may be edited with different default values or servers, and pushes maybe be performed for an individual zone, a full Group, or for an entire server. Pushes may be manual or scheduled for a future time through the Scheduler.

DNS WORKFLOW





For more information on DNSv3 tasks, see the following sections:

DNS Tab

Working with DNS Groups

Working with DNS Zones - Common Tasks

DNS Administration

Working with DNS Servers

Import DNS Zones

ProVision User Guide

User Guide

The ProVision User Guide provides information on features accessible in the standard user tabs within ProVision. For more detailed information on features accessible with Admin permissions, see the ProVision Admin Guide.

Table of Contents

- The Dashboard
- Working with Resources
- DNS Tab
- DHCP Tab
- IPAM Tab
- Peering v2
- Reporting
- User Preferences

The Dashboard

The Dashboard

The Dashboard is your first stop when logging into 6connect Provision, giving you a quick graphical status overview as well as convenient links for reference and support. The Dashboard is comprised of modular "Widgets" that show different types of information. These Widgets may be added, moved, edited, and customized to create personalized Dashboards.

There are two levels of Dashboards:

Default Dashboard: The Default Dashboard is a shared dashboard for all users of the ProVision instance. It may only be edited by admin level users, but it is viewable by all users.

Individual Users' Dashboards: Additional Dashboards may be created by individual users. These dashboards are tied to the user logged in, and are only viewable / editable by that user.

Dashboards and Permissions

Note: An individual's permission levels may limit what that user can view in both default and personal Dashboards. Although all users can view the Default Dashboard, they may be not be able to view all elements or data inside the dashboard.

- The Dashboard
  - ProVision Default Dashboard Overview:
    - Activity Chart (Viewable by Admins Only):
    - IP Charts:
    - Clock:
    - Status:
    - Resource Tree:
    - Contact Us:
    - ProVision Learning Links: (Markdown Widget)
    - 6connect Professional Services: (iFrame Widget)
    - 6connect Links (Links Widget):
    - 6connect RSS (RSS Feed Widget):
- Additional Information

ProVision Default Dashboard Overview:

Although each user may have access to or create wildly differing dashboards, below are descriptions of the Widgets available shown on the ProVision Default Dashboard (from left to right):

Activity Chart (Viewable by Admins Only):

Illustrates activity level for API, IPAM, DNS, Peering, and Resource areas of ProVision.

Select the desired ProVision area by clicking on the radio buttons. Then, hover over the chart, and use the mouse scroll wheel to zoom in or out of specific date / times. Clicking on a bubble will show activity detail, and resource / IP block links if applicable.

Only one Activity Chart Widget may exist on a dashboard.

IP Charts:

Illustrates the percentage of assigned, unassigned, allocated, or holding tank hosts for 1918 / IPv4 /

IPv6 space out of the total available hosts in ProVision viewable by the user. Hovering over any section of the ring will display additional data.

Only one IP Charts Widget may exist on a dashboard.

Clock:

Shows the current time and date for the selected time zone, based on format set by the user.

Multiple Clock Widgets may exist on a dashboard.

Status:

General status information on number of user / admin accounts, ProVision version number, and a 'Coming Soon' link to the future releases roadmap in the the documentation.

Only one Status Widget may exist on a dashboard.

Resource Tree:

An interactive horizontal or vertical treeview of a selected Resource and its children. Select a top-level resource from the Widget's Edit menu. Users may click and drag to move within the Widget, as well as zoom in and out using a scroll wheel. Clicking on filled dots will expand the tree to show the children of that resource. Double-clicking on a resource in the tree will give you the option to navigate to the resource's entry page.

Multiple Resource Tree Widgets may exist on a dashboard.

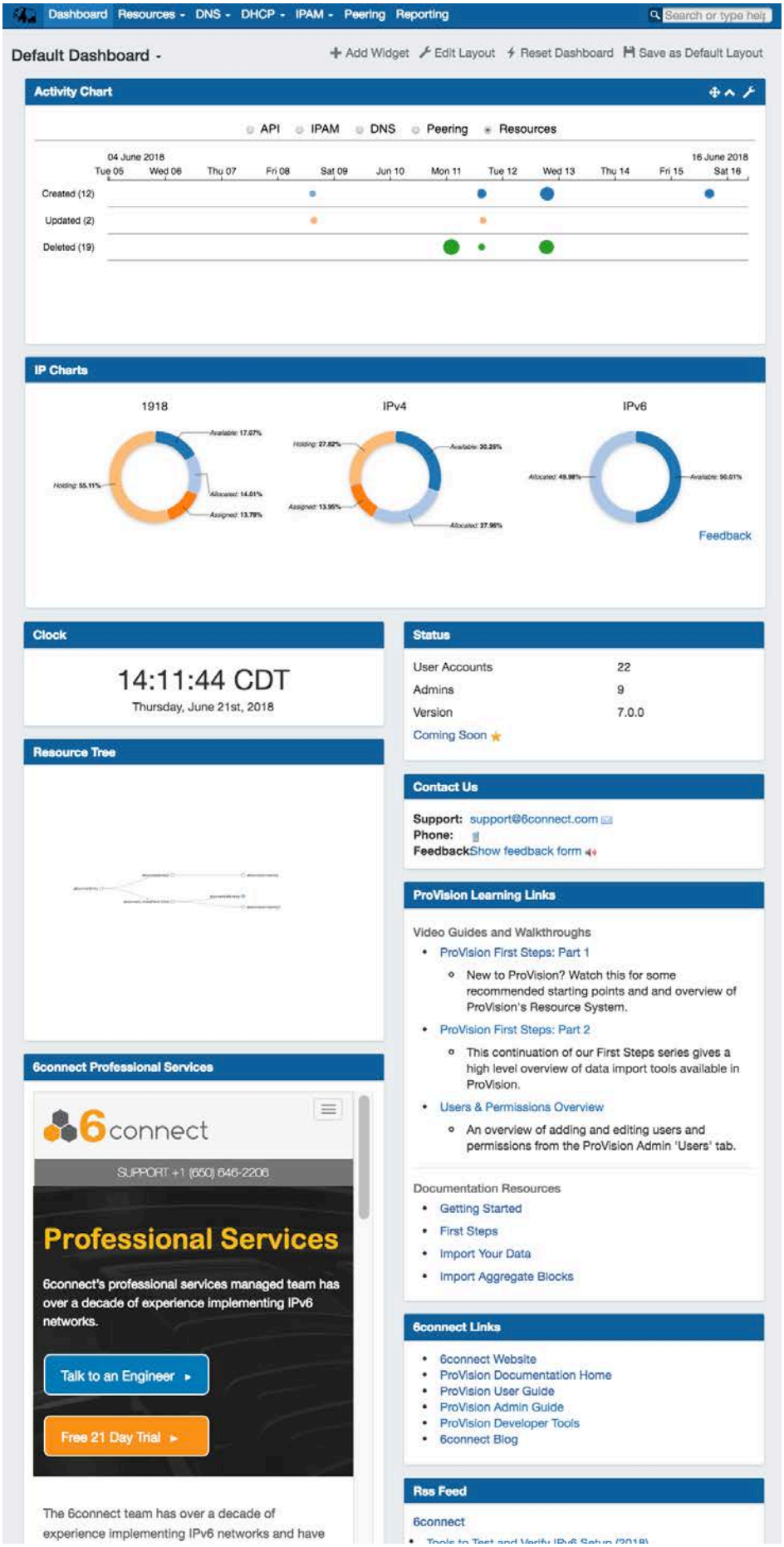
Contact Us:

Provides support email, phone, and feedback form information. Support phone number and email address are set from the ProVision Admin settings. The feedback form email address is editable within the Widget.

Only one Contact Us Widget may exist on a dashboard.

ProVision Learning Links: (Markdown Widget)

Pre-created Markdown Widget containing links to commonly referenced ProVision documentation sections, video guides, and walkthroughs.



The Markdown Widget is a blank slate where users may add links, text, or other content through use of the Markdown language.

Multiple Markdown Widgets may exist on a dashboard.

6connect Professional Services: (iFrame Widget)

Pre-created iFrame Widget accessing the 6connect Professional Services website.

The iFrame Widget sets a https URL to be viewed in an iFrame, to view stats or commonly referenced websites.

Multiple iFrame Widgets may exist on a dashboard.

6connect Links (Links Widget):

Pre-created Links Widget listing 6connect ProVision company and documentation links.

Link URL and display may be set from within the Widget.

Multiple Links Widgets may exist on a dashboard.

6connect RSS (RSS Feed Widget):

Pre-created RSS Feed Widget accessing the 6connect Blog RSS.

Shows the most recent five entries of a selected https RSS Feed. Feed links open in a new window when clicked.

Multiple RSS Feed Widgets may exist on a dashboard.



Additional Information

Continue on for details on dashboard customization, as well as how to add, move, update, and delete individual Widgets.

- Customizing the Dashboard
- Working with Dashboard Widgets

Customizing the Dashboard

Dashboard Customization

The Dashboard is your first stop when logging into 6connect Provision, giving you a quick graphical status overview as well as convenient links for reference and support. The Dashboard is comprised of modular "Widgets" that show different types of information. These Widgets may be added, moved, edited, and customized to create personalized Dashboards.

There are two levels of Dashboards:

Default Dashboard: The Default Dashboard is a shared dashboard for all users of the ProVision instance. It may only be edited by admin level users, but it is viewable by all users.

Individual Users' Dashboards: Additional personal dashboards may be created by individual users. These dashboards are tied to the user logged in, and are only viewable / editable by that user.

Dashboards and Permissions

Note: An individual's permission levels may limit what that user can view in both default and personal Dashboards. Although all users can view the Default Dashboard, they may be not be able to view all elements or data inside the dashboard.
Browser Compatibility

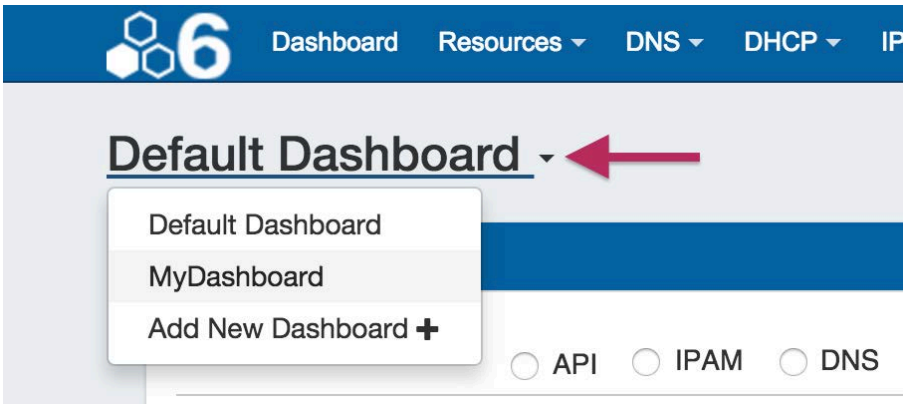
It is recommended to use the most recent versions of the Chrome or Firefox browsers when working with the dashboard. Due to incompatibilities with Internet Explorer, the dashboard is set to read-only mode and is not editable when accessed from IE at this time.

- Dashboard Customization
- Working with Dashboards
  - Viewing an Existing Dashboard
  - Adding a Dashboard
  - Editing a Dashboard
    - To Edit the Page Layout:
    - To Edit the Location of Individual Widgets:
  - Saving a Dashboard
  - Deleting a Dashboard
  - Reset the Default Dashboard
- Next Step: Widgets

Working with Dashboards

Viewing an Existing Dashboard

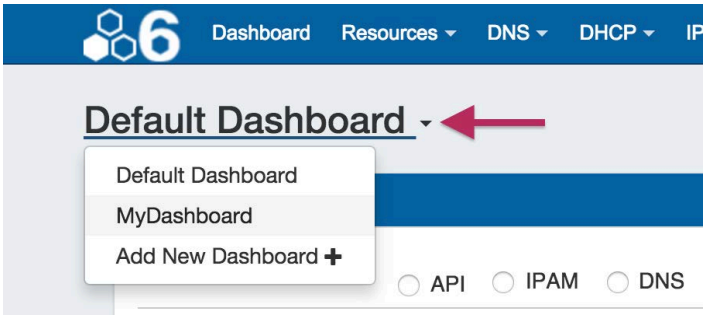
The Default Dashboard will be the primary dashboard that appears when clicking on the **Dashboard** Tab. To view another, already created dashboard, click on the dashboard name at the top left of the Dashboard page. This will open the list of existing dashboards. From there, click on the name of the dashboard you wish to view.



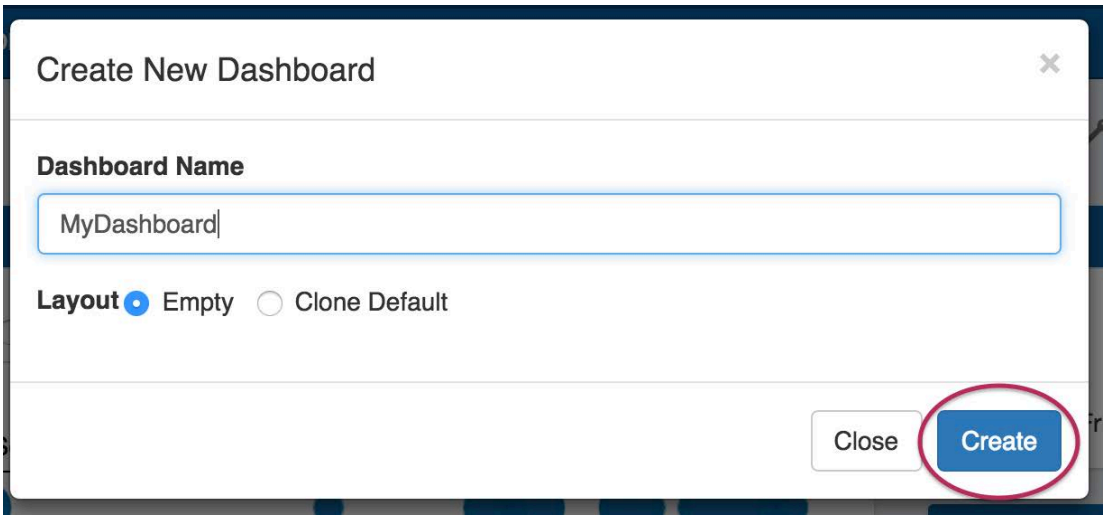
Adding a Dashboard

To add a new personal dashboard, click on the dashboard name at the top left of the Dashboard page. This will open the list of existing dashboards.

Click on "Add New Dashboard"



Fill out the desired dashboard name, and choose whether the dashboard will be empty or cloned from the default dashboard. Selecting "Empty" will provide a (mostly) blank canvas for selecting and placing Widgets from scratch; selecting "Clone" will duplicate the current state of the Default Dashboard, which the user can then edit as desired.

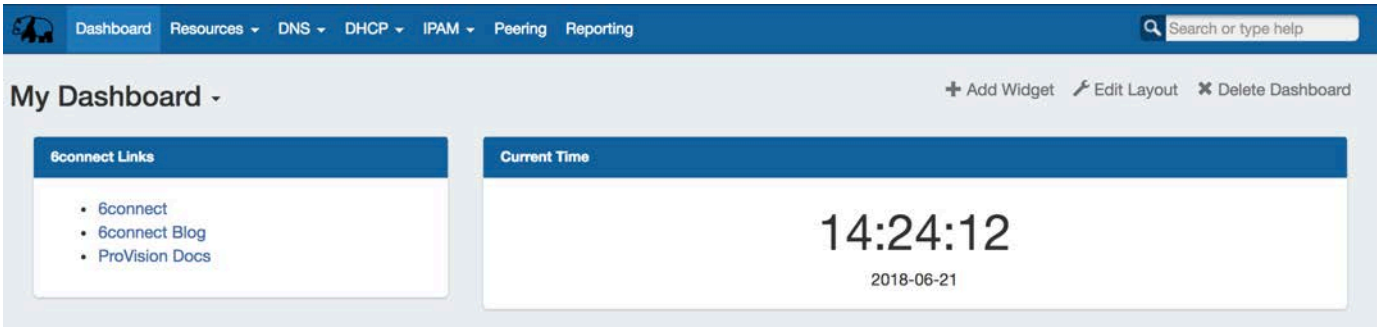


Note

Note: If "Empty" is selected, a default Widget will still show in the head of each page column, this is to assist in both visualizing the current page layout and to aid in placing new Widgets. These widgets may be edited or deleted as desired during the process of selecting layout and adding other Widgets.

For best performance of the dashboard, ensure that at least one Widget is placed in each column, or columns may be difficult to "detect" when moving Widgets to different page areas.

MyDashboard has now been created as an "Empty" dashboard, is accessible from the dropdown list of dashboards, and is ready to set up!

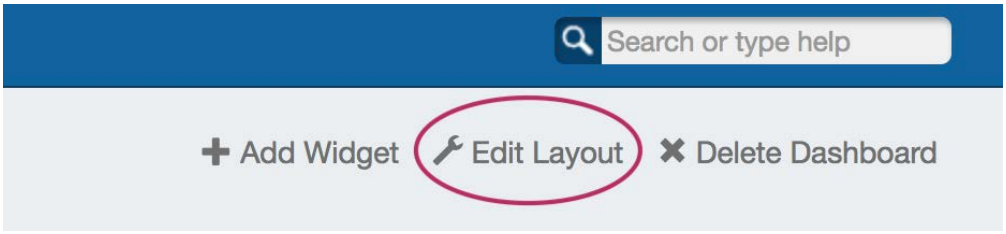


Editing a Dashboard

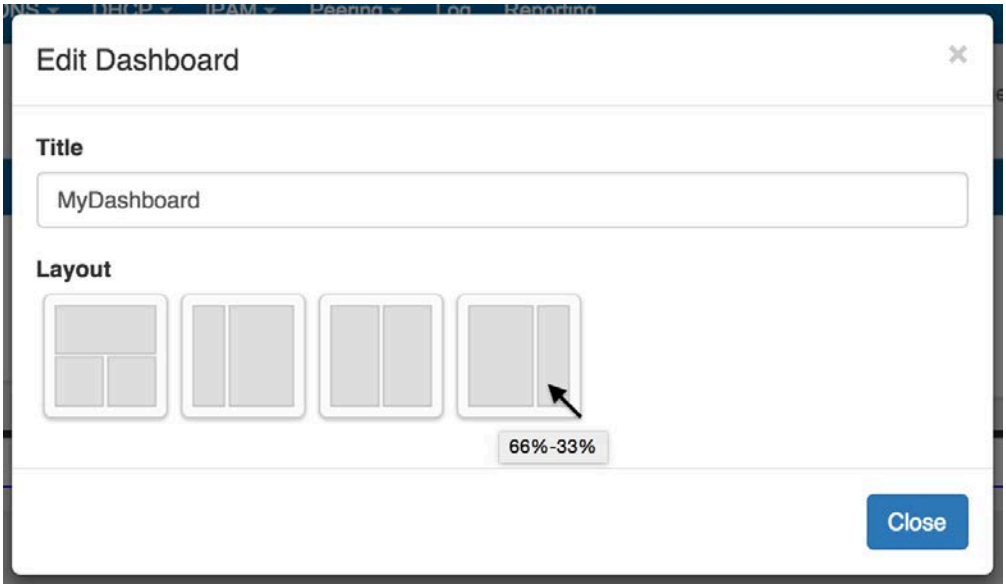
You can change the layout of your dashboard by selecting overall page layout options (for column size and location on the page), as well as move individual widgets around the page.

To Edit the Page Layout:

At the top of the page, click "Edit Layout".



This will bring up an option box where you can choose to rename your dashboard, and / or select a column organization structure.



Title: Type in how you want your dashboard title to display.

Layout: Click on a layout option to immediately apply that layout to your dashboard. Hovering over the option will show the column width percentages.

100% / (50% / 50%): Creates a page with a full width section at the top, and two equal sized columns below that.

33% / 66%: Creates a page with two columns, with the left column being one-third of the page width, and the right column as two-thirds of the page width.

50% / 50%: Creates a page with two equal sized columns, split down the middle.

66% / 33%: Creates a page with two columns, with the left column as two-thirds of the page width, and the right column as one-third of the page width.

To Edit the Location of Individual Widgets:

Individual Widgets may be moved around the page by clicking and dragging on the "Move" icon on the right side of the Widget header.

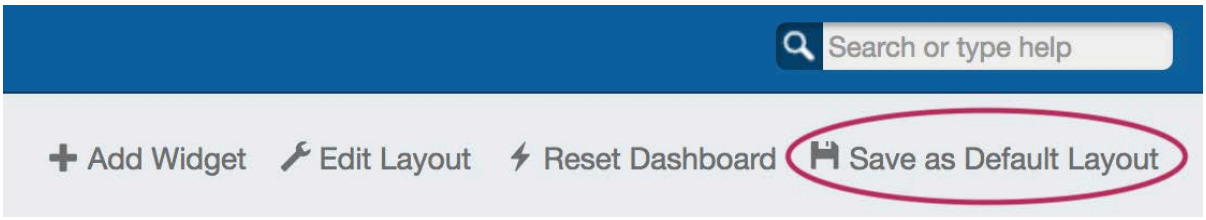
While mousing over the move icon, when the cursor changes to a hand icon, click and hold your left mouse button, then drag the widget up or down to the desired location. To move the Widget to another column, drag the widget sideways to the column, then up to briefly hover over a Widget already in the desired column (this "selects" the column for the Widget). Then, drop the Widget in the desired location in that column.

Saving a Dashboard

Personal dashboards are automatically saved after making layout or Widget edits (the exception is that Widget resizing must be manually saved - see Working with Dashboard Widgets).

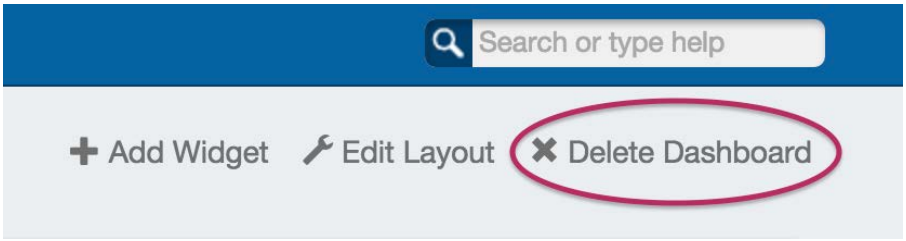
Default Dashboards must be manually saved after edits for those changes to apply to all dashboard users.

Admin users can save Default Dashboard changes by clicking "Save as Default Layout" at the top right section of the dashboard page, and the changes will be applied for all users.



Deleting a Dashboard

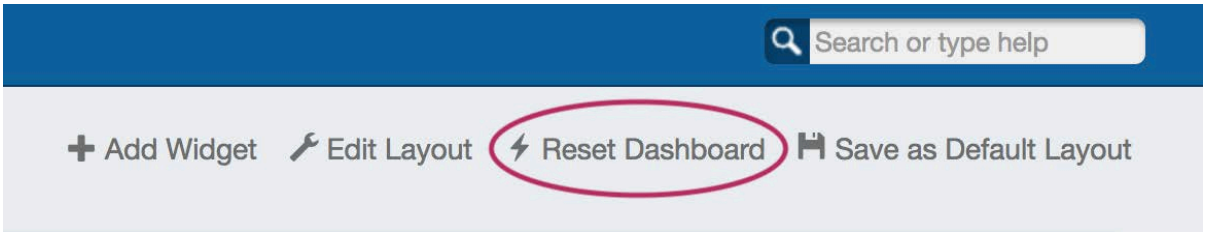
Personal dashboards may be deleted by clicking the "Delete Dashboard" button at the top right of the page. Default Dashboards may not be deleted.



Reset the Default Dashboard

The Default Dashboard may be reset to the ProVision default dashboard by clicking "Reset Dashboard".

This will remove any admin changes, and set the Default dashboard to its original state. This action may only be made by admin users.



Next Step: Widgets

Continue on to Working with Dashboard Widgets for information on each Widget, as well as how to add, move, update, and delete individual Widgets.

- Working with Dashboard Widgets

Working with Dashboard Widgets

Dashboard Widgets

The previous section, Customizing the Dashboard, gave a high level overview of the dashboard types, how to create new personal dashboards, edit the dashboard layout, and remove personal dashboards. This section will give an overview of how to work with individual Widgets, and options available in each Widget type.

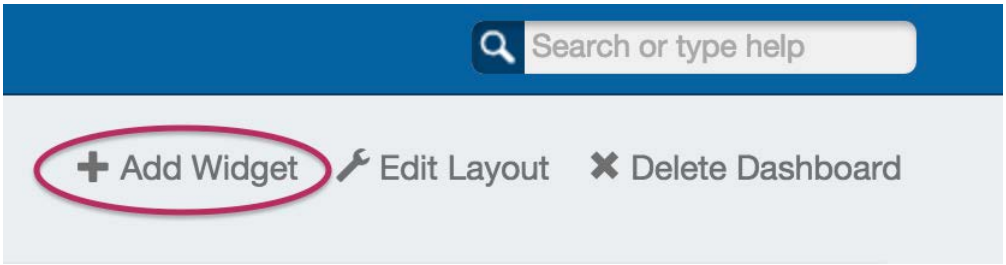
Before you begin working with individual Widgets, make sure that you have reviewed the information contained in Customizing the Dashboard and are familiar with the tasks outlined there.

- Dashboard Widgets
  - Working with Widgets
    - Adding Dashboard Widgets
    - Move a Widget:
    - Maximize / Pop-Out a Widget:
    - Edit a Widget (Widget Action Menu):
  - Available Widgets
    - Activity Chart (Admin only)
    - Resource Tree
    - IP Charts
    - RSS Feed
    - Status
    - Clock
    - Contact Us
    - Markdown
    - Links
    - iFrame

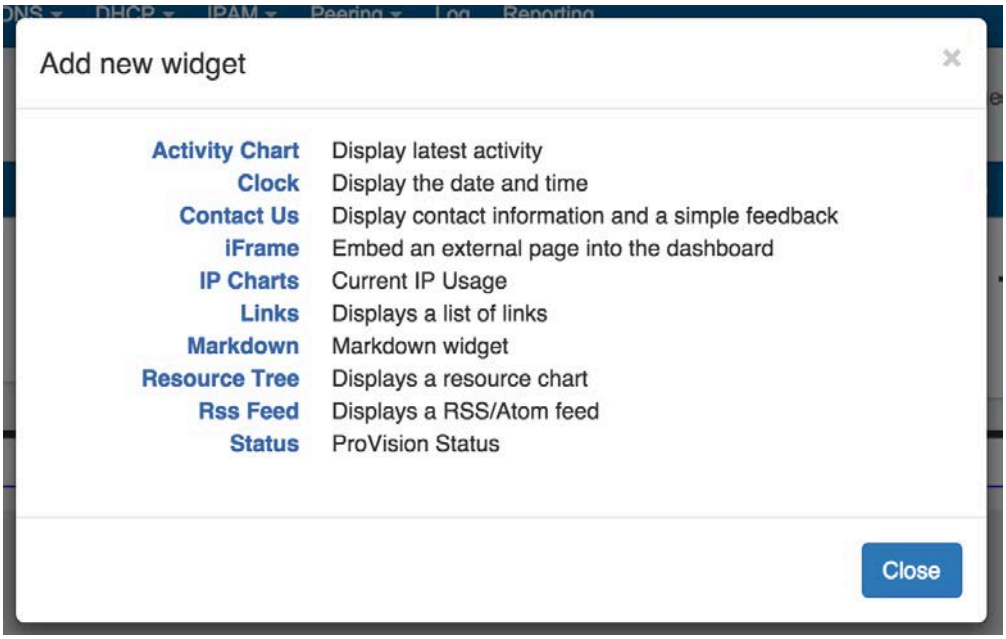
Working with Widgets

Adding Dashboard Widgets

To add a widget to personalize your dashboard, click on the "Add Widget" button at the top of the page.



Then, click on the name of the Widget you want to appear on your page. It will automatically be added into the leftmost column of your page.

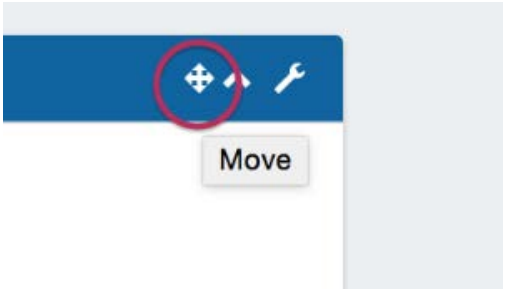


Some Widgets may only have one instance per dashboard, whereas other Widgets may be able to have multiple instances per dashboard. The Add New Widget dialog will only show you the widgets you have available to add.

For detailed information on each individual Widget, view the .Working with Dashboard Widgets v7.0.0#Available Widgets section on this page.

Move a Widget:

Individual Widgets may be moved around the page by clicking and dragging on the "Move" icon on the right side of the Widget header.



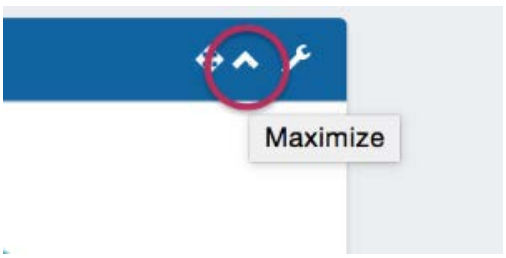
While mousing over the move icon (when the cursor changes to a hand icon) click and hold your left mouse button, then drag the widget up or down to the desired location.

To move the Widget to another column, drag the widget sideways to the column, then up to briefly hover over a Widget already in the desired column (this "selects" the column for the Widget). Then, drop the Widget in the desired location in that column.

Your browser does not support the HTML5 video element

Maximize / Pop-Out a Widget:

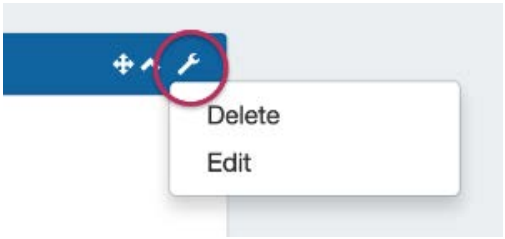
Widgets can temporarily be viewed at a larger size by hitting the "Maximize" icon on the Widget header. This will "pop out" the widget information to a larger page size for easier viewing.



While maximized, you may interact with the widget as normal. When done, hit the "Close" button.

Edit a Widget (Widget Action Menu):

To open the edit menu for a widget, click on the Action Menu (wrench icon) on the right side of the Widget header. A drop down menu will appear with some of the following options:



Edit: Opens an edit widget pop up, where widget-specific options may be changed. Options may include changing the title, links, applicable resources, and so forth depending on the particular Widget.

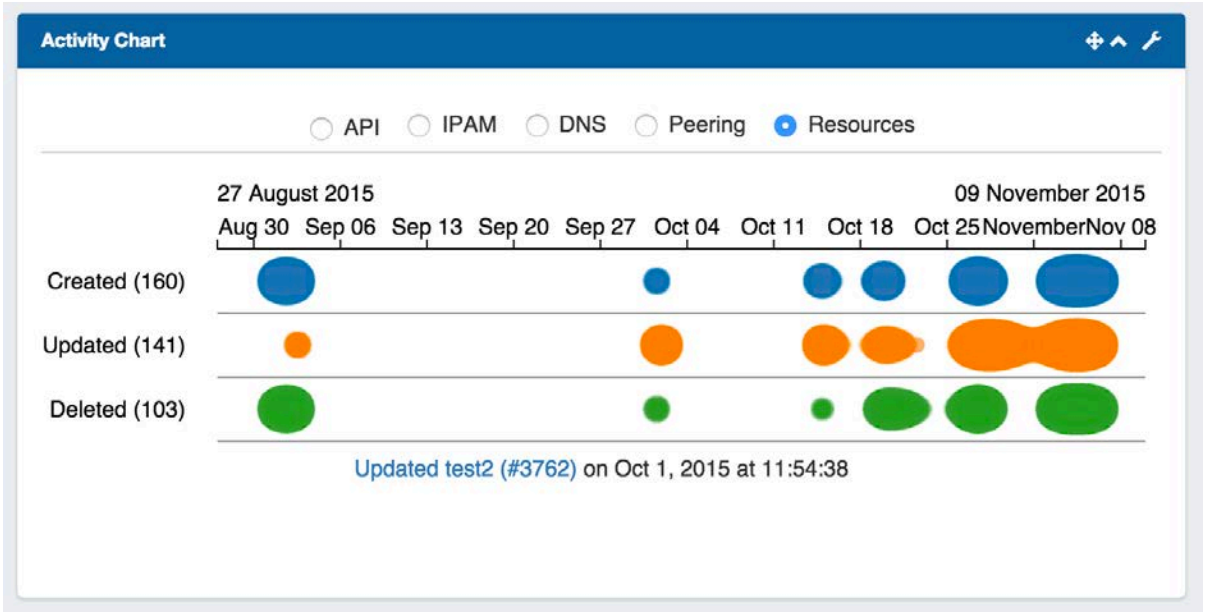
Resize: For Widgets able to be resized vertically, brings up a draggable black bar at the bottom of the widget. Drag the bar up or down to the desired widget height, then select the Action Menu again and click on "Save Resize" to save and complete the size edit.

Save Resize: Available only after the "Resize" action has been used. After resizing a Widget, selecting the Action Menu again and clicking on "Save Resize" will save the Widget size.

Delete: Deletes the widget.

Available Widgets

Activity Chart (Admin only)



Illustrates activity level for API, IPAM, DNS, Peering, and Resource areas of ProVision. The Activity Chart is only visible to Admin users. Only one Activity Chart Widget may exist on a dashboard.

Activity Chart Actions:

Radio Buttons: Select the desired ProVision activity area to view by clicking on the radio buttons (API, IPAM, DNS, Peering, or Resources).

Chart: While hovering over the chart, you can:

Use the Scroll Wheel to zoom in or out of the date/time detail level

Double click to zoom into the next date/time detail level

Click & Drag left or right to move to an earlier or later date/time at the current zoom level

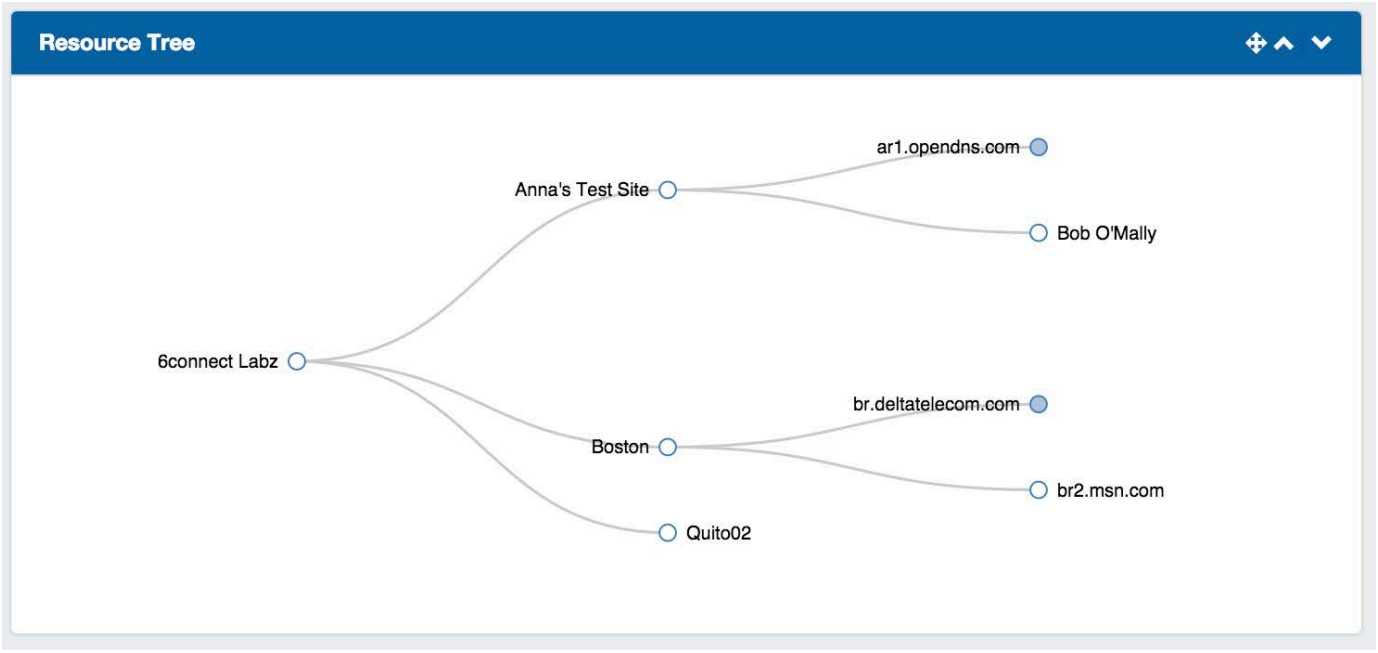
Single Click on a bubble will show the activity detail, and resource / IP block links if applicable.

Activity Chart Edit Options:



Title: Edits the title of the Widget shown on the header.

Resource Tree



Interactive graphical display of the resource structure for a parent resource.

Resource Tree Actions:

Users may:

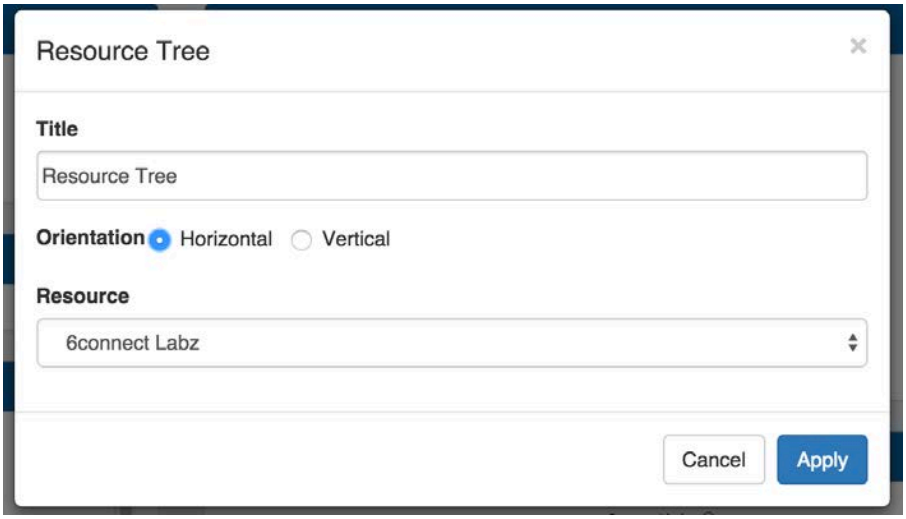
Click & Drag to move the chart within the Widget,

Use the Scroll Wheel to zoom in or out

Click on Filled dots to expand the tree to show the children of that resource.

Double click on a resource in the tree to give you the option to navigate to the resource's entry page.

Resource Tree Edit Options:



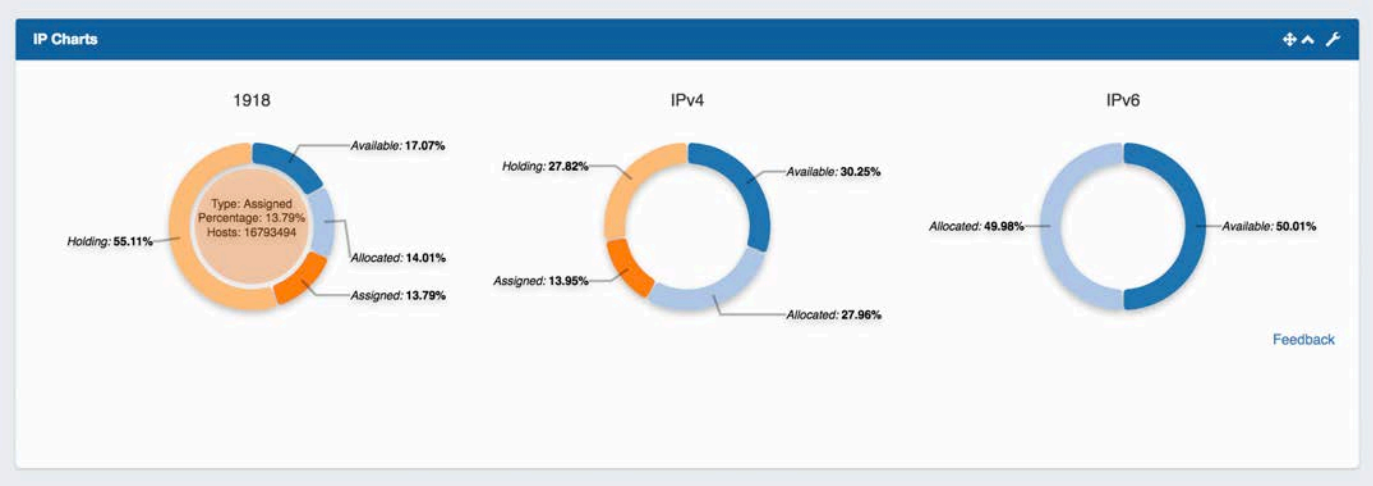
Title: Edits the title of the Widget shown on the header.

Orientation: Select "Horizontal" to view the tree in horizontal form, with the top level resource on the leftmost side of the widget, and children expanding to the right. "Vertical" will show the tree with the top level parent at the top of the widget, with children expanding downward.

Resource: A text search box to select the resource that will be the top level resource for the tree.

When done editing, hit "Apply" to save your changes, or "Cancel" to exit without saving.

IP Charts



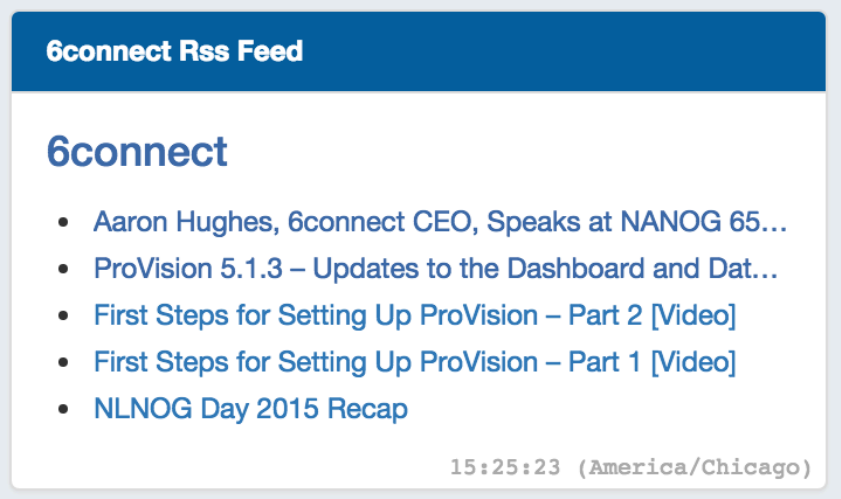
Illustrates the percentage of assigned, unassigned, allocated, and holding tank hosts for 1918 / IPv4 / IPv6 space out of the total available hosts in ProVision viewable by the user.

Hovering over a particular status will display detailed data in the center of the chart for that status type.

IP Charts Edit Options:

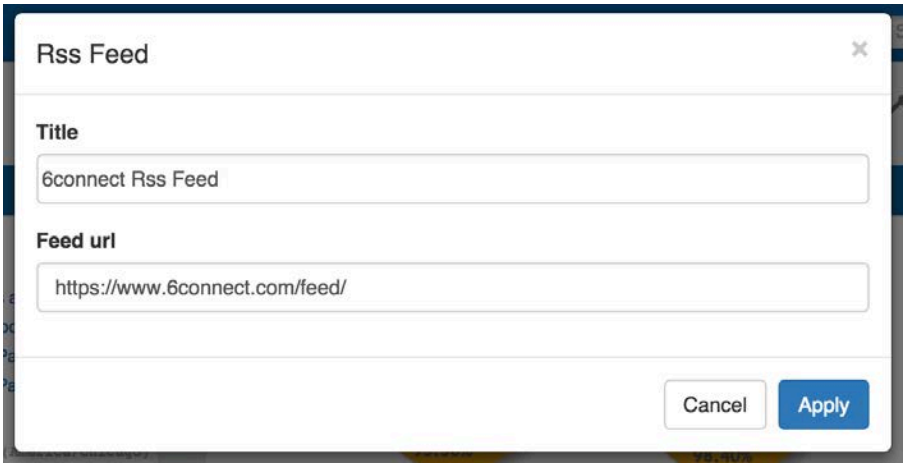
Title: Edits the title of the Widget shown on the header.

RSS Feed



Shows the most recent five entries of a selected RSS Feed. Feed links open in a new window when clicked.

RSS Feed Edit Options:



Title: Edits the title of the Widget shown on the header.

Feed URL: Edits the URL of the RSS Feed to show.

Status

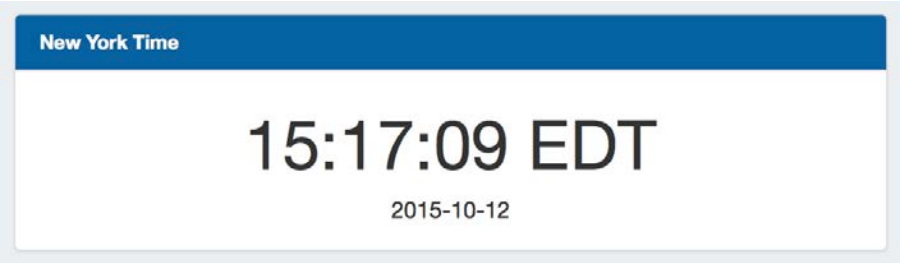


Shows 6connect ProVision status information, including number of user accounts, number of admin accounts, current version number, and a link to the "Coming Soon" section of the ProVision documentation.

Status Edit Options:

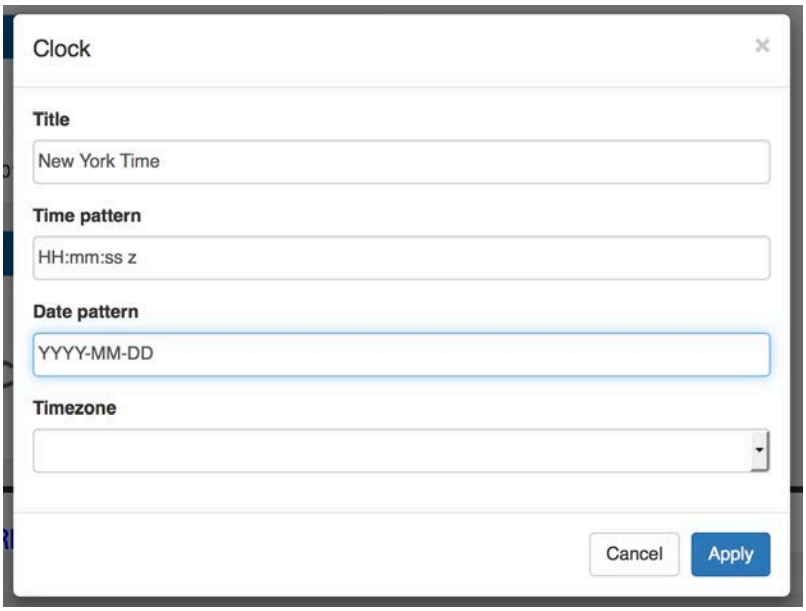
Title: Edits the title of the Widget shown on the header.

Clock



Shows the current time for the selected time zone.

Clock Edit Options:



Title: Edits the title of the Widget shown on the header.

Time Pattern: Sets the desired format for the current time. Example: "HH:mm:ss z" shows hours, then minutes, then seconds, then time zone.

Date Pattern: Sets the desired format for the current date. Example: "YYYY-MM-DD" shows 4 digit year, then month, then day.

Time Zone: The desired time zone to show the current time for.

Click here for detail on date / time patterns...

Time / Date Patterns

For applicable time / date patterns, see http://momentjs.com/docs/#/displaying/format/

Some common date patterns might be:

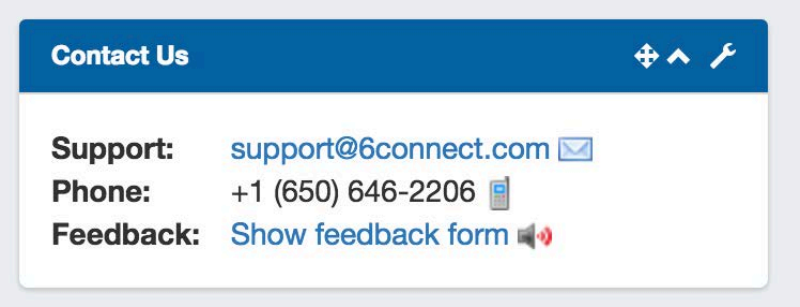
Display	Pattern
Wednesday, October 10th, 2015	dddd, MMMM Do, YYYY
10/21/2015	l

October 21, 2015	MMMM DD, YYYY
10-21-15	MM-DD-YY
2015-10-21	YYYY-MM-DD

Some common time patterns might be:

Display	Pattern
15:36:14 EDT	HH:mm:ss z
3:36:12 PM EDT	h:mm:ss A z
03:36:12 pm EDT	hh:mm:ss a z
15:36 -04:00	HH:mm Z

Contact Us



Provides support email, phone, and feedback form information. Only one Contact Us Widget may exist on a dashboard.

Contact Us Edit Options:

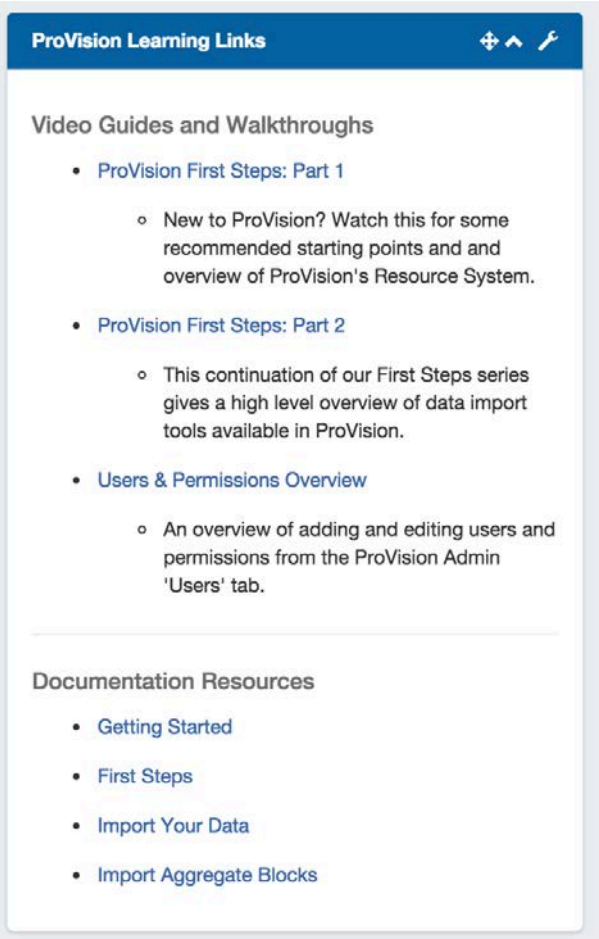


Title: Edits the title of the Widget shown on the header.

Feedback Email: Sets the desired email address for the feedback form.

Note: Support phone number and email address are set from the ProVision Admin settings.

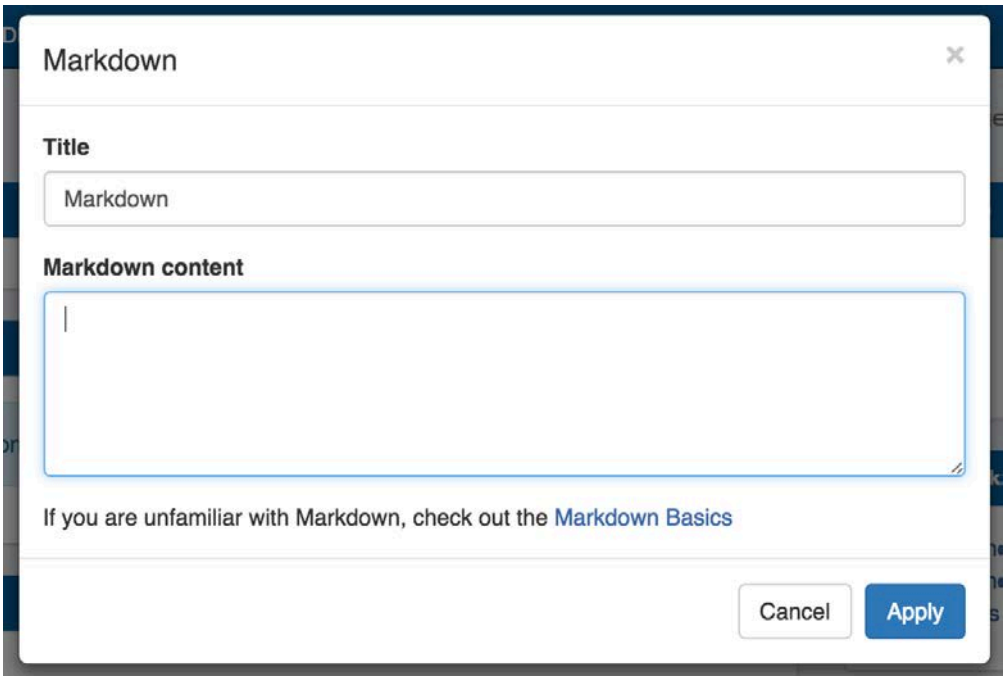
Markdown



The Markdown Widget is a blank slate where users may add links, text, or other content through use of the Markdown language. Multiple Markdown Widgets may exist on a dashboard.

If you are unfamiliar with Markdown, check out Markdown Basics

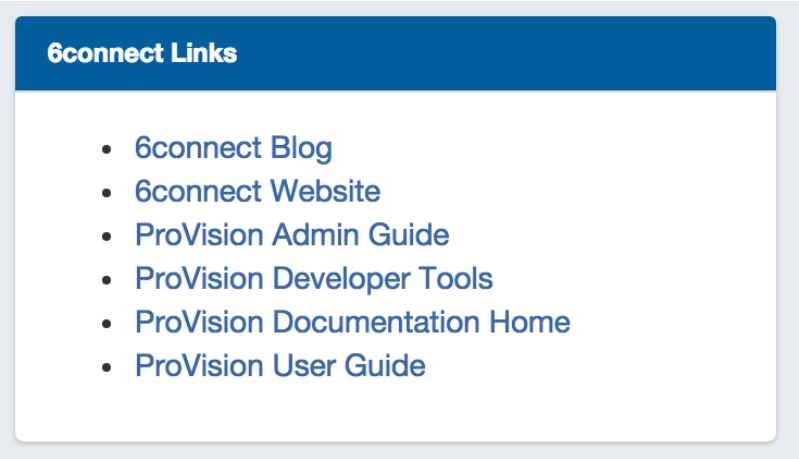
Markdown Edit Options:



Title: Edits the title of the Widget shown on the header.

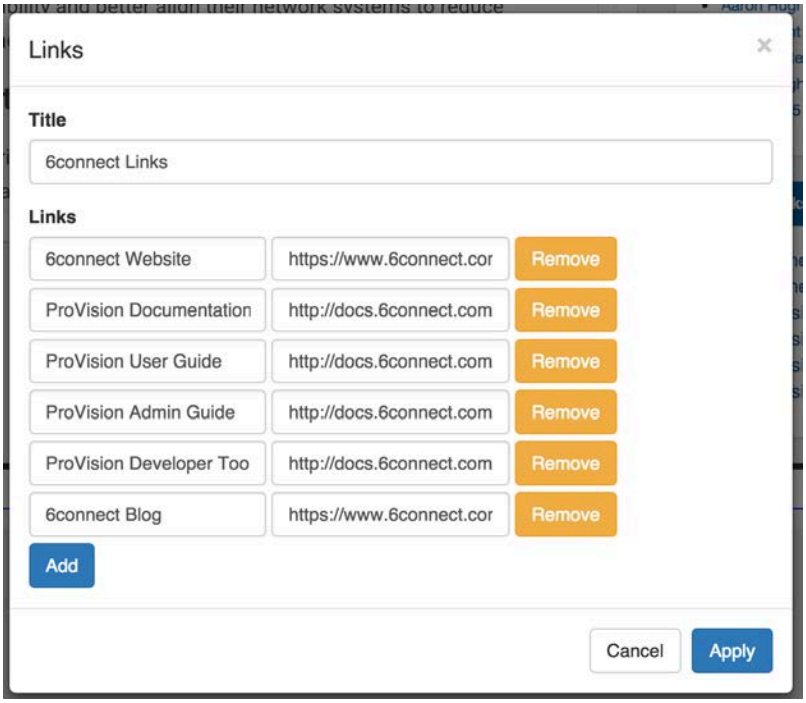
Markdown Content: Area to enter the markdown content.

Links



Provides a list of website links. Multiple Links Widgets may exist on a dashboard.

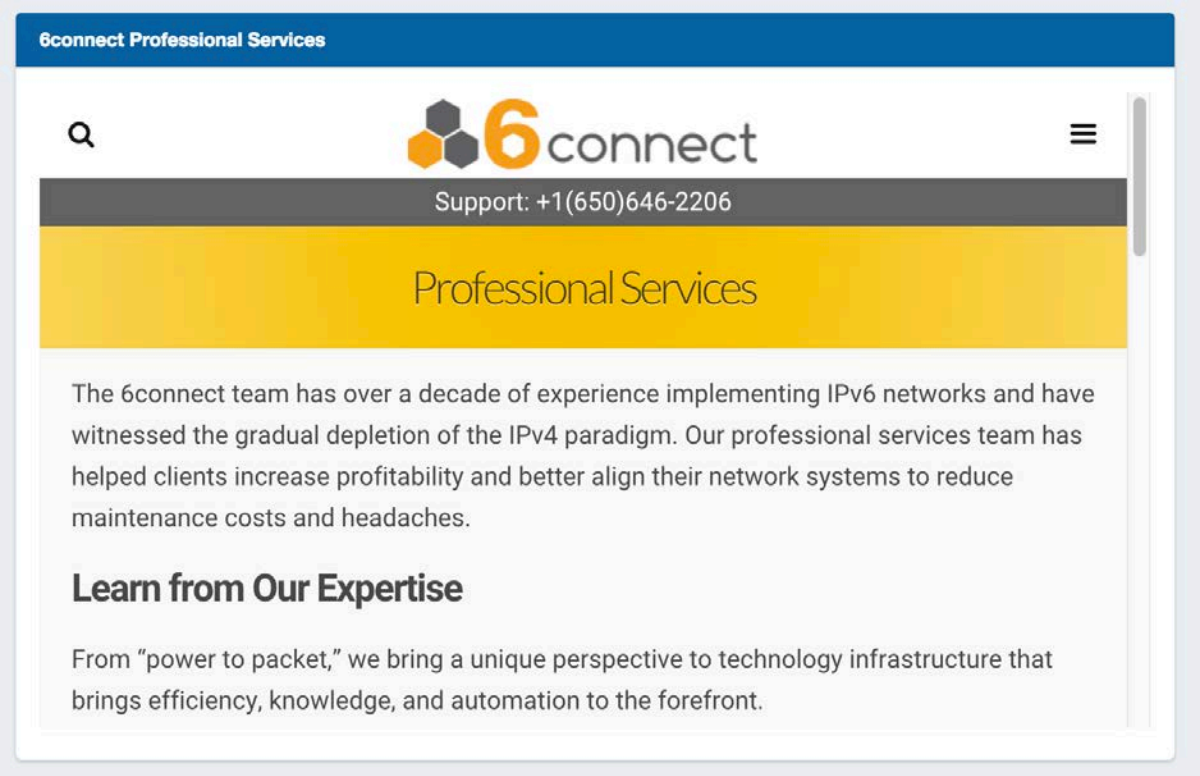
Link Edit Options:



Title: Edits the title of the Widget shown on the header.

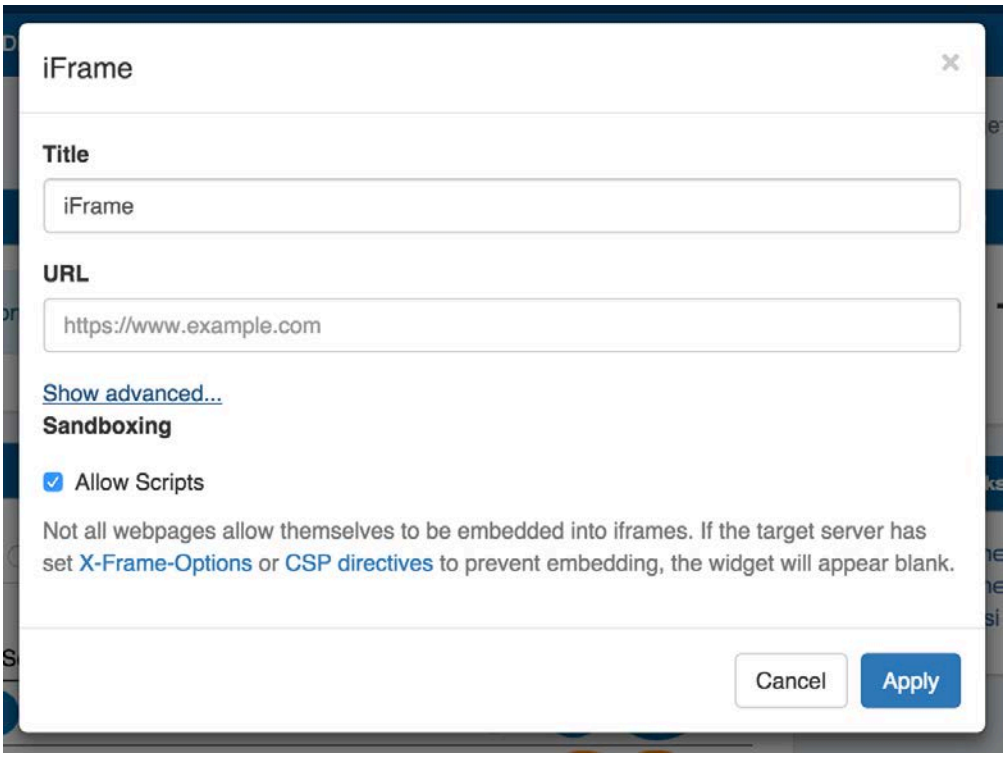
Links: Enter the desired link text, then the URL of the website to link. Click "Add" to add a new link, or "Remove" to remove an individual link. Hit "Apply" to save your changes.

iFrame



Set a https URL to be viewed in an iFrame. Useful to view stats or commonly referenced websites.

iFrame Edit Options:



Title: Edits the title of the Widget shown on the header.

URL: Sets the desired https address of the website to display in the frame

Advanced:

Sandboxing: Allow Scripts: Check to allow scripts.

Not all webpages allow themselves to be embedded into iframes. If the target server has set X-Frame-Options or CSP directives to prevent embedding, the widget will appear blank.

Working with Resources

Working with Resources

- Working with Resources
  - What is a Resource?
  - How to Work with Resources?
  - Additional Information

What is a Resource?

The "Resource" system is tied to the Permissions structure. What this means is that you get granular control on a resource level and can create groups around a single resource or even groups of resources. Since Resources can inherit permissions from others - it can be an easy way to categorize generic objects.

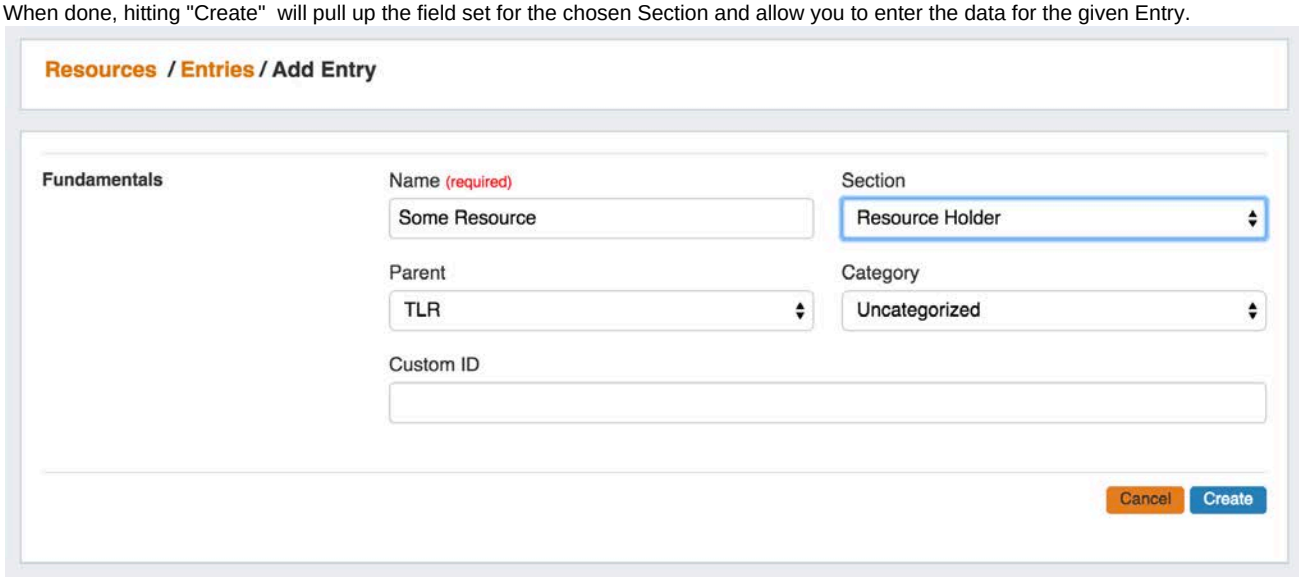
WARNING!

There are key Resources that are used by the System that should not be deleted. We have put in some safeguards in the UI, but the API can delete these resources if prompted. The resources that you should not remove are "Holding" and "Reverse". The "Available" Resource can be renamed - simply not deleted.

How to Work with Resources?

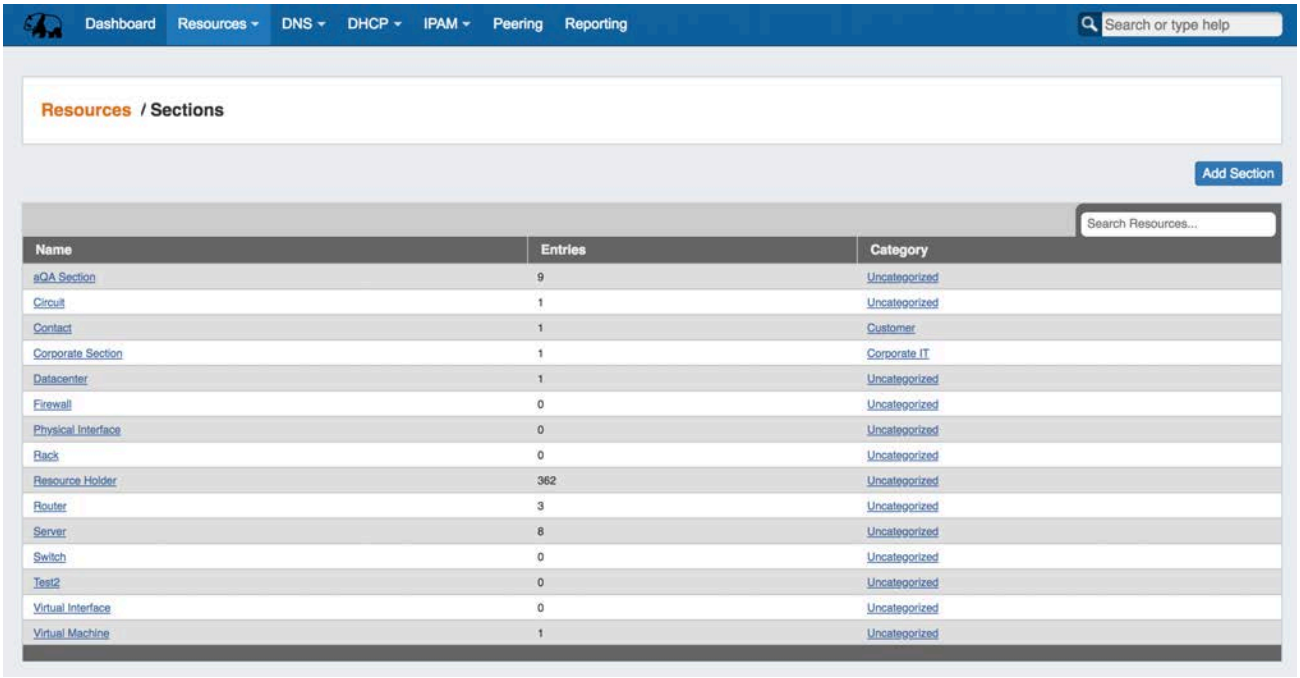
The Resource is an entity that users can assign Network Resources to (IP blocks, hosts, DNS zones, etc.). You can also create hierarchies between resources which allows you to leverage permissions to control who can view and interact with any given resource and its assigned elements. Please note that you can also have Resources that do NOT have anything assigned to them regarding Network Resources. The result of this flexible architecture is that you can work with Resources in three ways:

- **Resource Entries:**
  These are the actual Resource names. When you click the "Add Entry" button you can customize various elements of the entry and assign the Parent Resource, Section and Category from their respective dropdown menus. You may also add a Custom ID if desired. When done, hitting "Create" will pull up the field set for the chosen Section and allow you to enter the data for the given Entry.

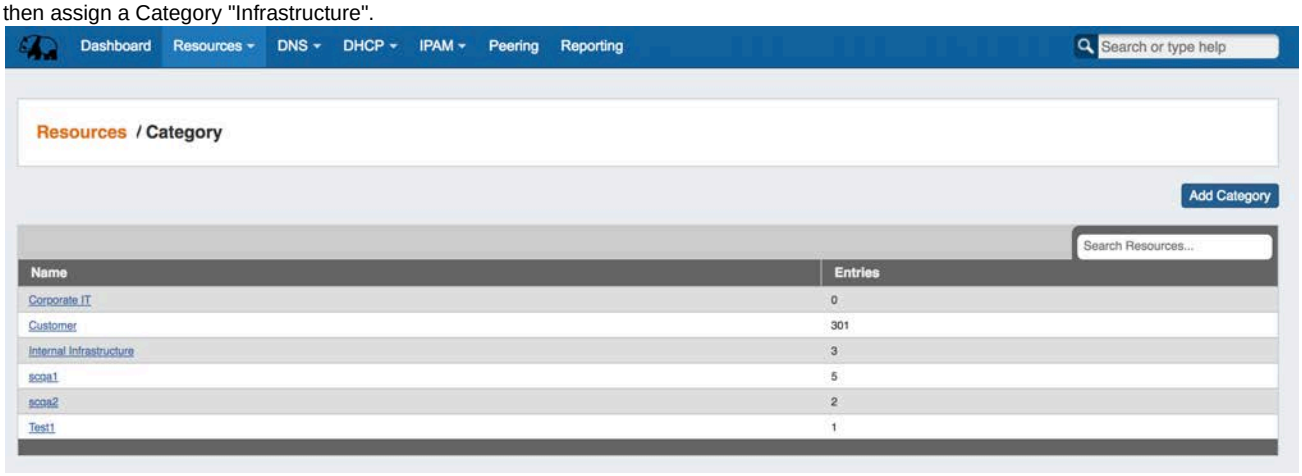


- **Resource Sections:**
  These can be anything from "customers" to "firewalls" to "cross-connects". Since you can customize the fields for these elements, and assign them to a Parent Section, you have flexibility in organizing the data. Check out Customizing Sections and Customizing Fields for

more details on how to fit these elements to your business.



- **Resource Categories:**
  Categories can be used to create some filtered views for given Resources and Sections. For example, you can create a Section called "Resource Holder" and then assign a Category "Customer". Then you can view a list of Resources that have been assigned to Category "Customer". In the same way, you could also assign a Section called "Router" under the Parent Resource "Corporate Datacenter" and then assign a Category "Infrastructure".



Want to customize Sections? Check out Customizing Sections and Customizing Fields for more details!

Some examples:

1) Service Provider

2) Managed Service Provider

3) Datacenter/Colocation Provider

4) Enterprise

Additional Information

- Working with Entries
- Customizing Sections
- Customizing Fields
- Gadgets

    - XML Specifications
- Contact Manager

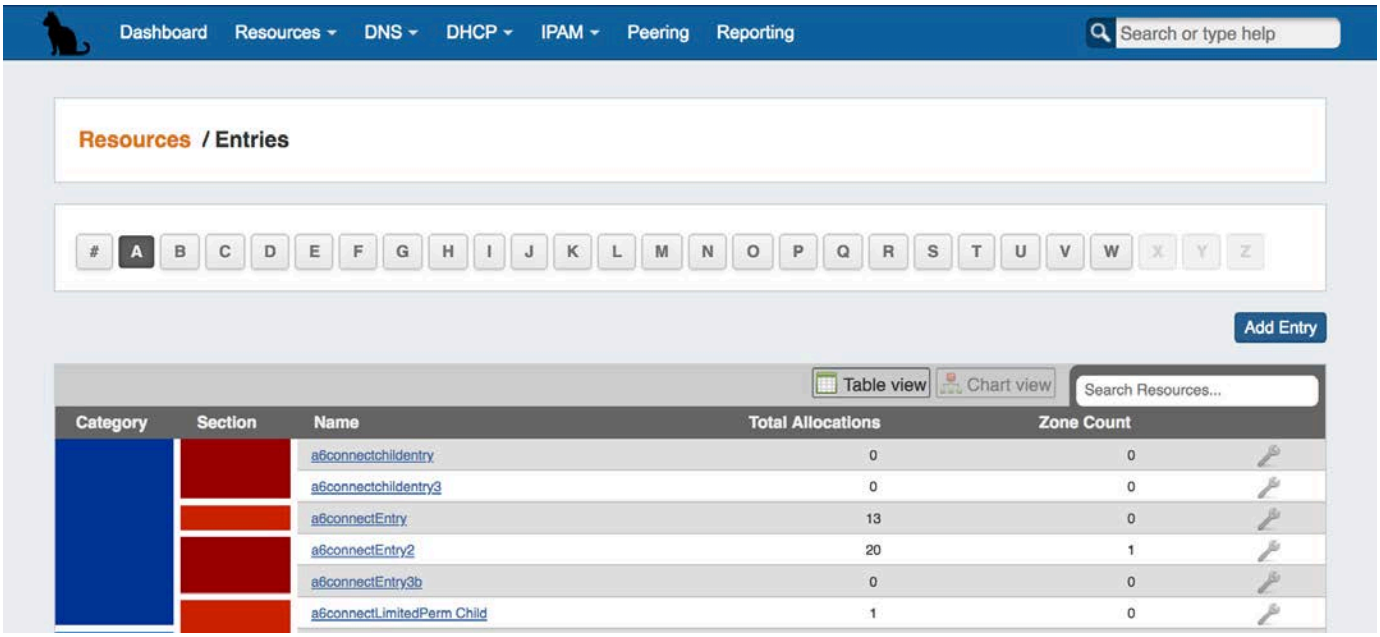
Working with Entries

Working with Entries

The list of Resource Entries is under the **Resource** Tab. To access it, you may either click on the Resource Tab, or select "Entries" from the Resource Tab dropdown menu.

- Working with Entries
  - Resource Tab / Entry List User Interface
    - Entry List Action Menu
  - Chart View
  - Create an Entry
  - Edit or Delete an Entry
  - Add Child Entries
  - Resource Clone
    - To Clone an Entry
  - Additional Information

Resource Tab / Entry List User Interface



1) Alphabetical Filter: Entries are organized by the first letter of the name. Click on a letter to see entries starting with that letter.

2) Add Entry Button: Click to add a new entry.

3) Pagination: Click to view additional pages of Entries under the selected letter.

4) Viewing Options: Select "Table View" or "Chart View" .

Table View (shown above), lists the Entries, Allocations, and Zone Count along with color-corded Category and Section information.

Chart View graphically shows the entries in filterable, hierarchy form.

5) Search Box: This text box allows the user to enter in criteria to filter the list of Entries.

6) Category: Color-coded Category indicator. The Category name will appear upon mouseover. Clicking on the Category box when the hand icon appears will redirect to a filtered list of entries of that Category.

7) Section: Color-coded Section indicator. The Section name will appear upon mouseover. Clicking on the Section box when the hand icon appears will redirect to a filtered list of entries of that Section.

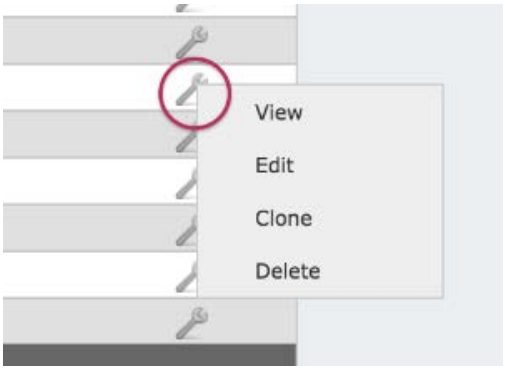
8) Name: A list of the Resource Entry names. Clicking on a name will take the user to that resource's individual entry page.

9) Allocations: The number of IP blocks assigned to that resource.

10) Zone Count: The number of zones assigned to that resource.

11) Action Menu: The Action Menu (wrench icon) gives a list of additional actions to perform on the zone

Entry List Action Menu



Clicking on the Action Menu in the Entry List View will bring up the following options:

View: Opens the resource's Entry page

Edit: Opens to the resource's Edit page

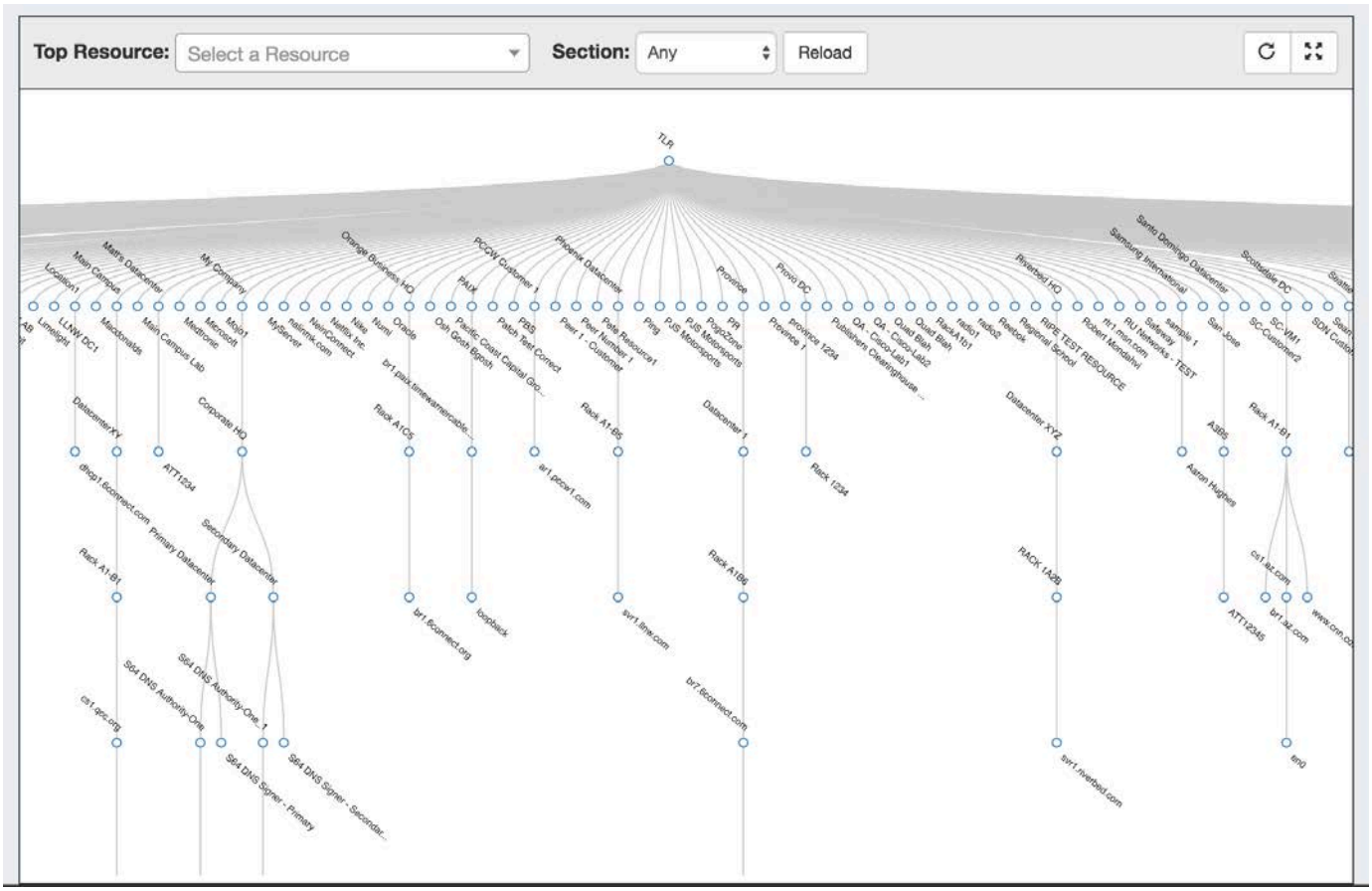
Clone: Opens the clone resource page for the selected resource

Delete: Deletes the resource, if appropriate permissions exist.

Chart View

Chart View illustrates the resources created in a node-based tree.

Select the Top Resource and/or the Section Type from the dropdown menus to limit the view, if desired, and then click "Reload". Users will only be able to view sections of the resource tree on which they have view permissions.



While viewing the main chart, you may:

- View in horizontal or vertical tree mode by clicking the "Switch Chart Orientation" button.
- View fullscreen by clicking the "View Fullscreen" button.
- Left click and drag within the tree view, or scroll wheel in/out to change the area viewed.
- Single or double-clicking on a node with children will collapse or expand the tree at that resource.
- Clicking and dragging a resource node to another resource will relocate the resource to be a child of the resource it is dropped on.

Hovering over a specific Resource or node, then right clicking will open a context menu with the following options:

Quick View: Shows summarized details about the resource and its children, and links to the Resource Entry page(s).

Edit: Opens to the resource's Edit page

Clone: Opens the clone resource page for the selected resource

Delete: Deletes the resource, if appropriate permissions exist.

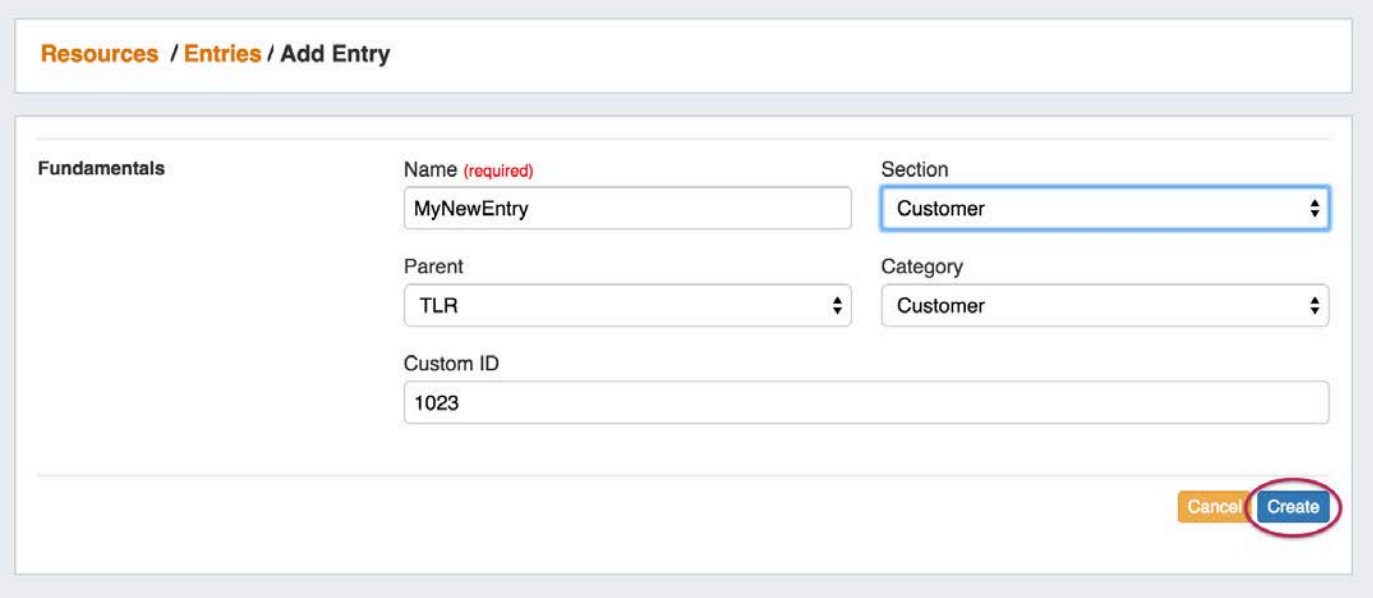
Create an Entry

1) To create a new Entry, click on the "Add Entry" button from the Resource Tab / Entry List Page.

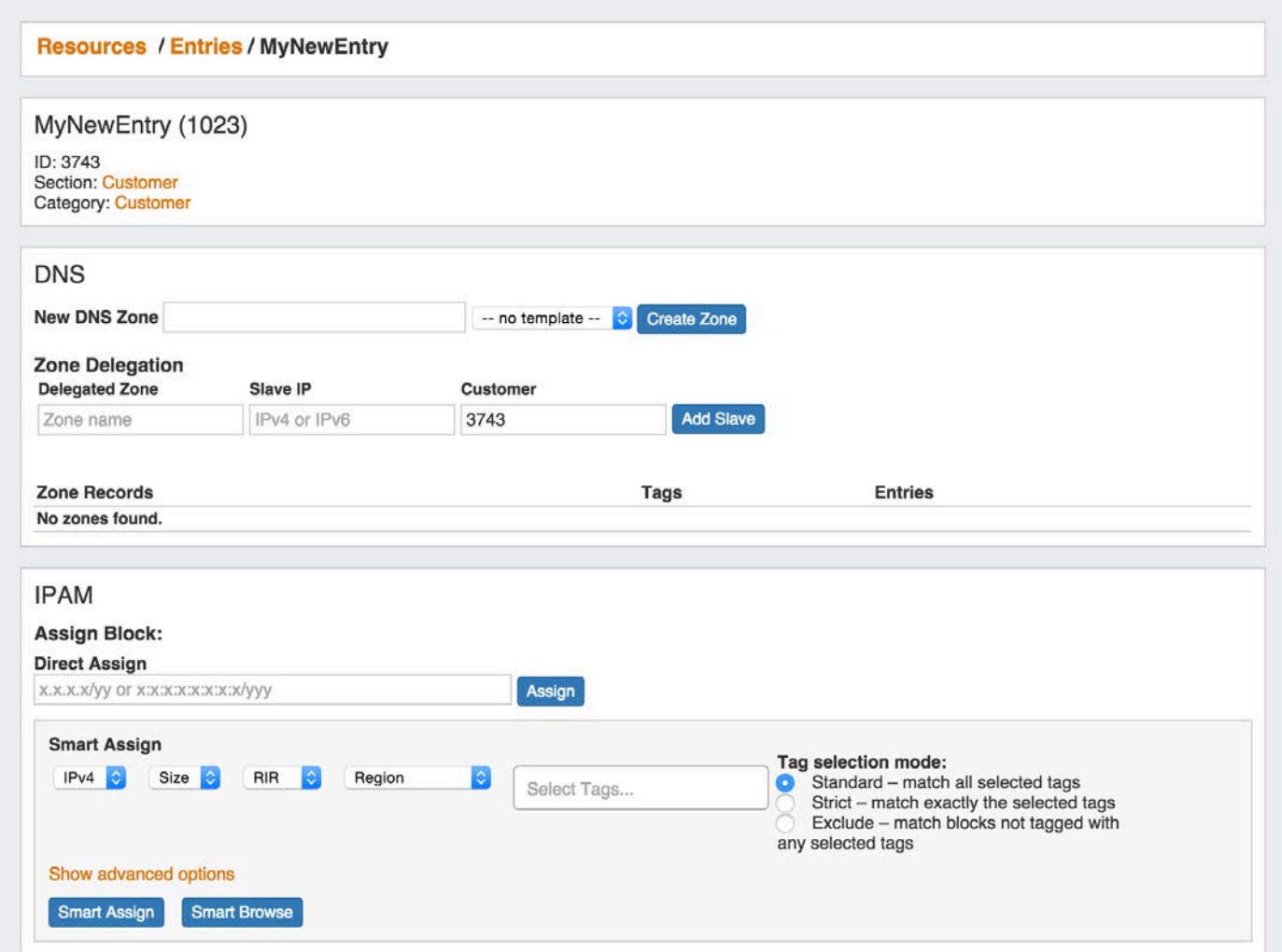


This will take you to the Add Entry page.

2) Fill in the Name of your new Resource Entry, select a Section (the chosen Section determines what gadgets will show on your new resource's Entry page), the Parent Resource, Category, and enter a Custom ID number if desired. When complete, click the "Create" button. If you wish to exit without saving your changes, click "Cancel".



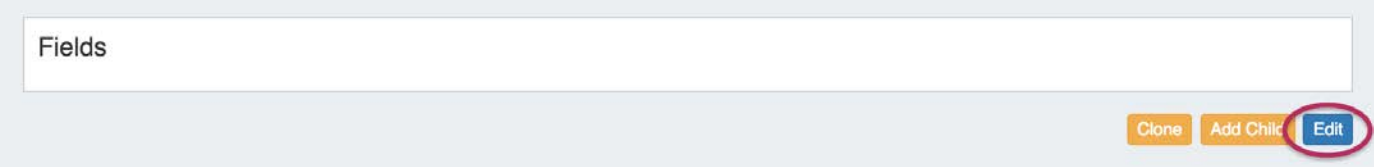
Once the Entry is created, the Resource Entry page appears for that Resource, with the gadgets applicable to the selected Section.



From here, you can work with your selected Gadgets, performing tasks such as assigning DNS zones, IP Blocks, assign Contacts, and more. See the Gadgets Page for detail on working with individual Gadgets.

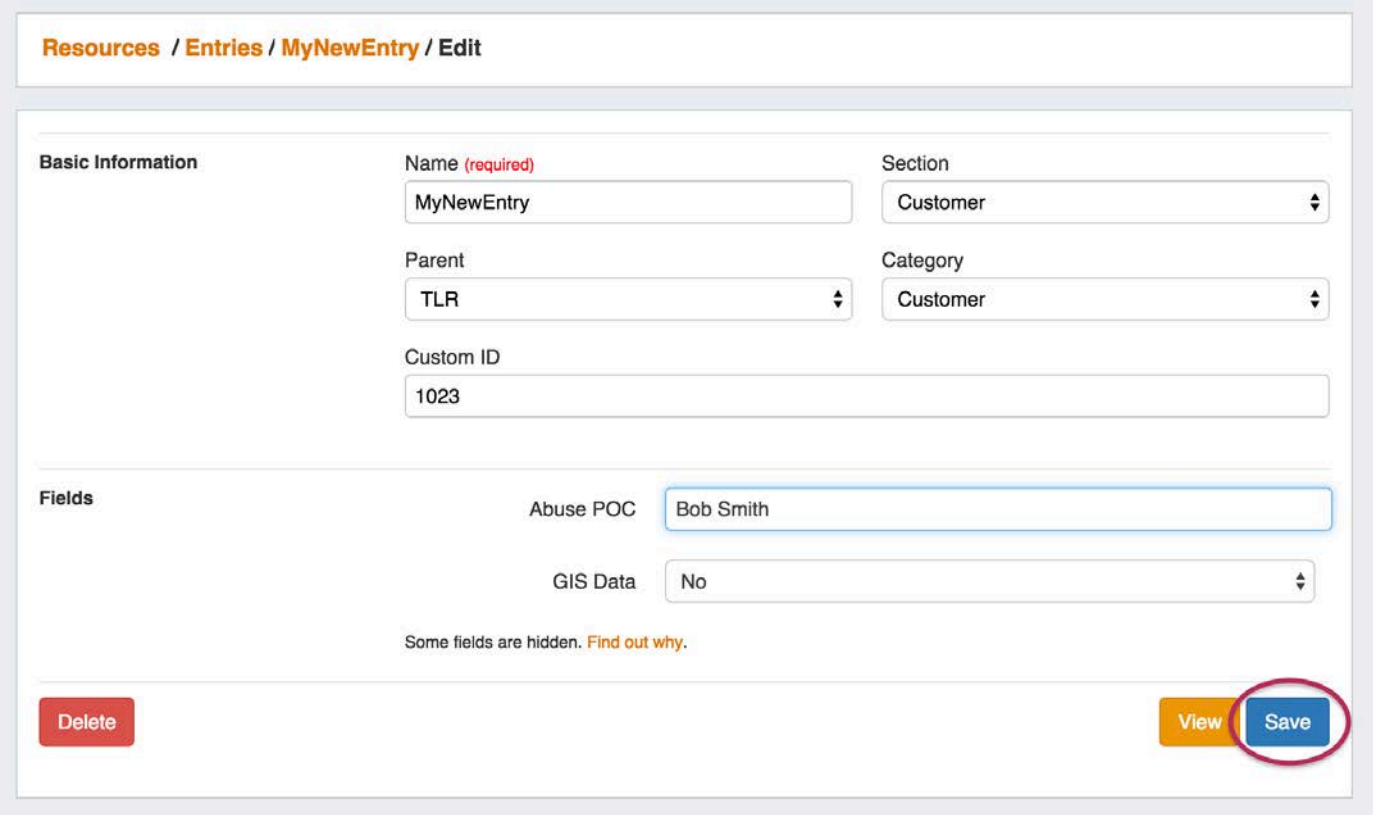
Edit or Delete an Entry

To edit or delete an Entry, go to the bottom of that resource's Entry Page, and select "Edit".



This will bring up the Edit Entry Page, where you can edit basic information such as Name, Section, Parent, Category, or Custom ID. If Fields are associated with the Section type, those may be edited as well.

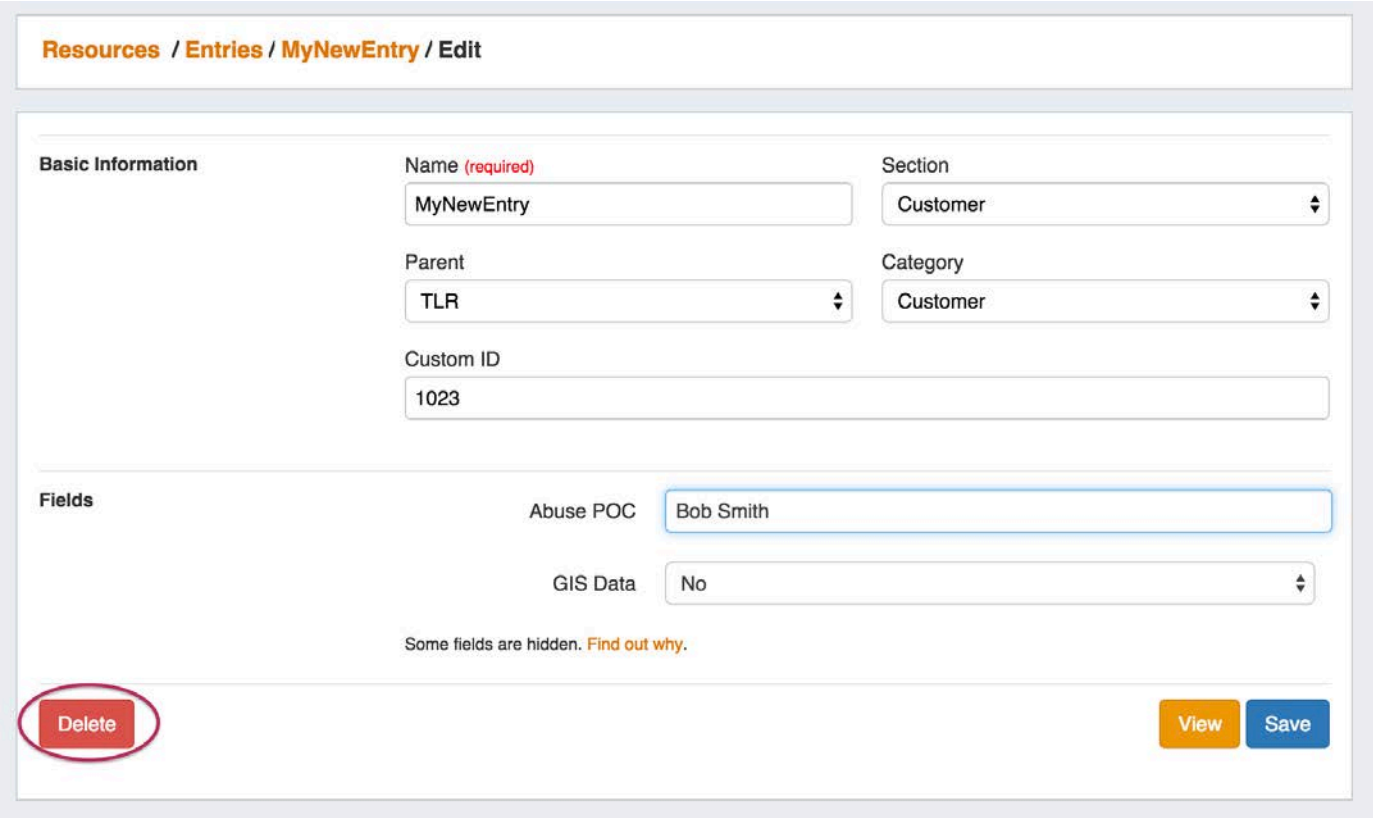
When done, hit the "Save" button. If information was added to Fields, that information will now show up on the Entry Page.



If you wish to delete the Entry, select the "Delete" button.

Note

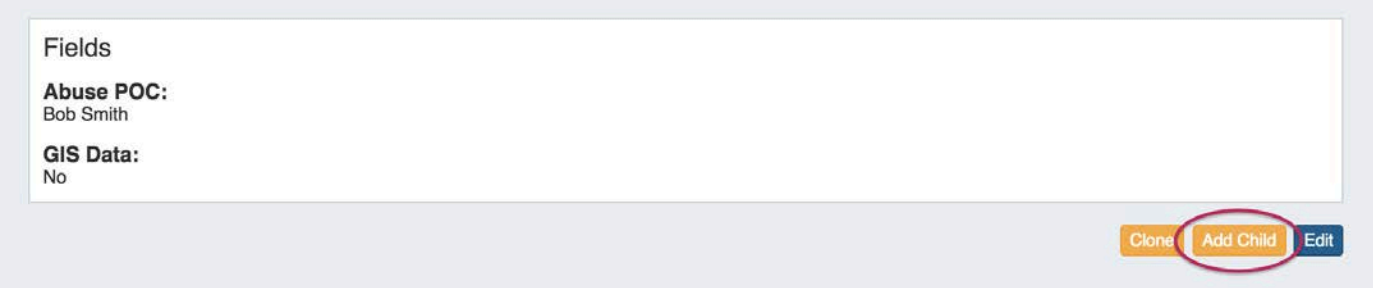
If an entry has children, the delete option will not be available. Child Entries must be deleted before a Parent Entry can be deleted.



Add Child Entries

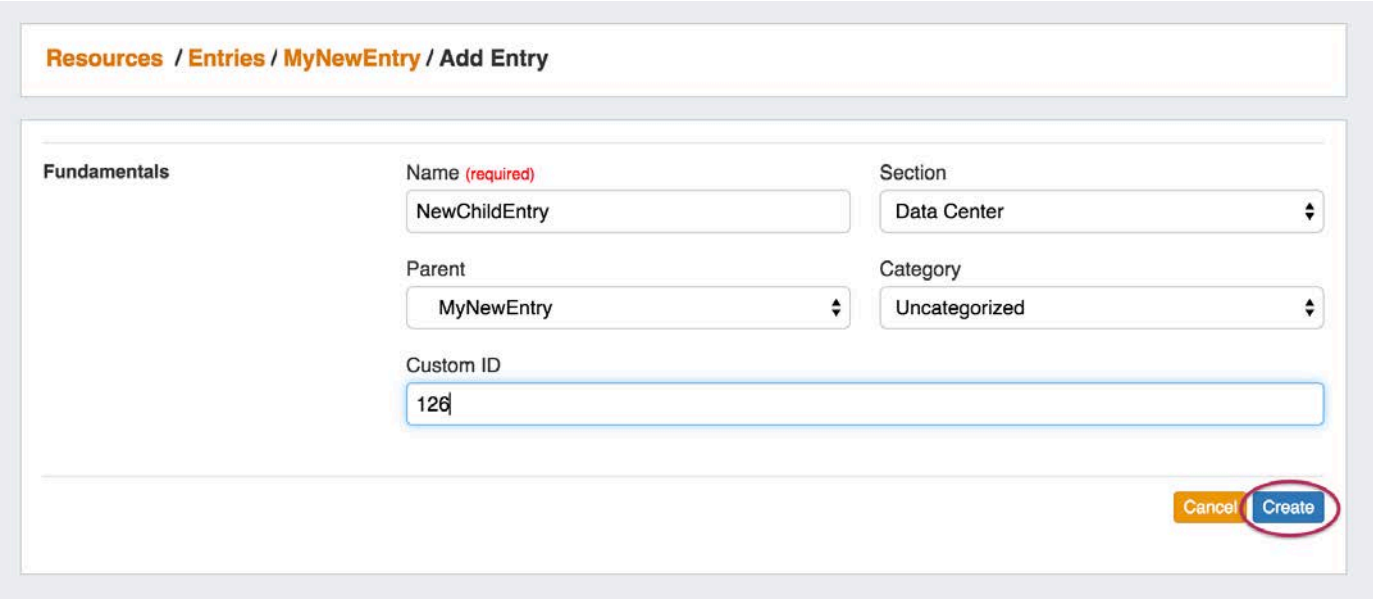
Adding a Child Entry creates a resource underneath the current resource in the hierarchy.

To add a Child Entry, at the bottom of the Resource Entry page, click "Add Child".



This brings up the Add Entry dialog. Looking at the top navigation breadcrumbs, you can see that we are adding an Entry underneath the resource "MyNewEntry".

Fill out the Name, Section, Category, and Custom ID (if desired), and hit the "Create" button. The Parent field is pre-selected for you.



Once created, the Child Entry will show in a list at the bottom of the Parent Entry page.



Resource Clone

Cloning an Entry duplicates the Entry and, if desired, any Child Entries existing under it.

This feature can be used to quickly and efficiently create multiple resources with same format, field information, or similar hierarchy structure. One example may be a datacenter with x number racks with y number of servers on each. One rack, with its servers as child entries could be created, and then cloned until the necessary quantity is reached. As text field information is copied during the clone, changes in individual server information would still need reviewed and updated. Think of cloning as creating a template from an existing Resource Entry.

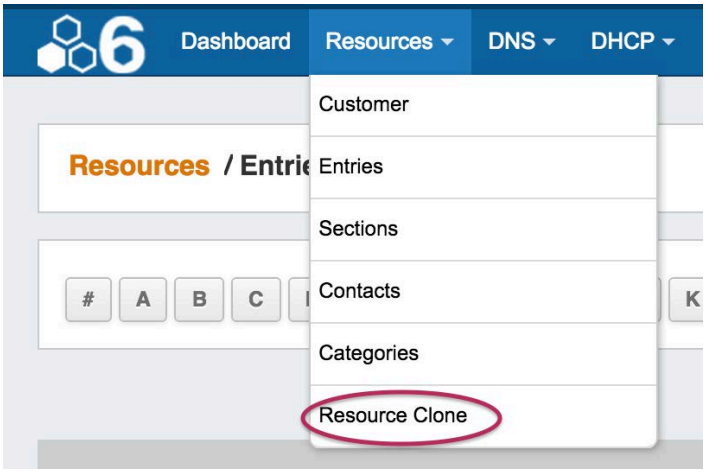
Information Cloned: Format of the original Resource (which gadgets are enabled, location, fields), contact / tech information, field information, and (if selected) Child Entries, and Child Entry sections, fields, and field informaiton - in general, items that are chosen or input at the entry creation, or input into text fields.

Information not Cloned: IP Blocks, DNS Zones, uploaded documents - in general, items that are *assigned* to the entry rather than input into fields.

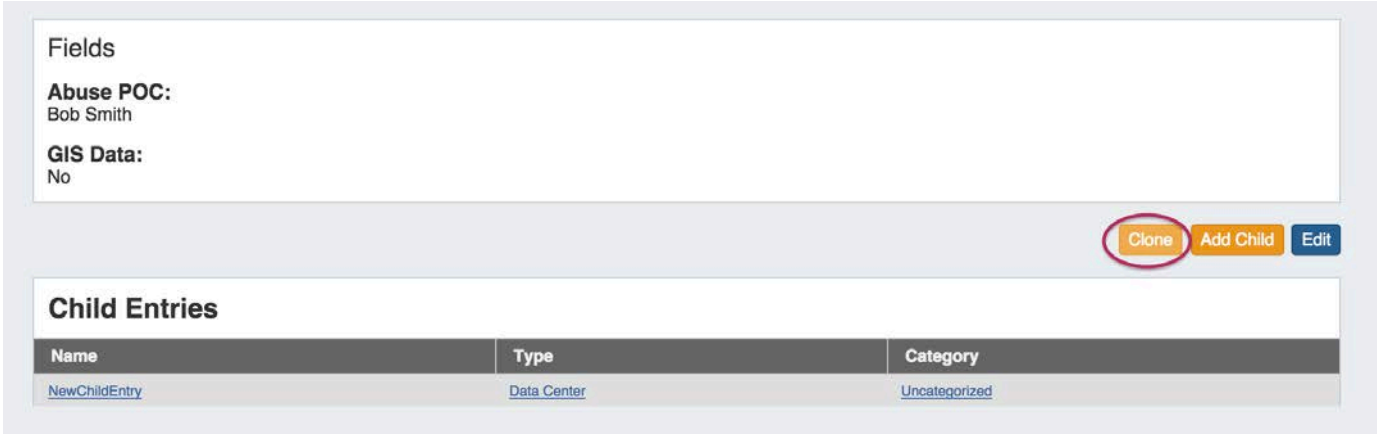
To Clone an Entry

The Resource Clone feature may be accessed from either the **Resources** Tab dropdown menu, or from a Resource Entry page.

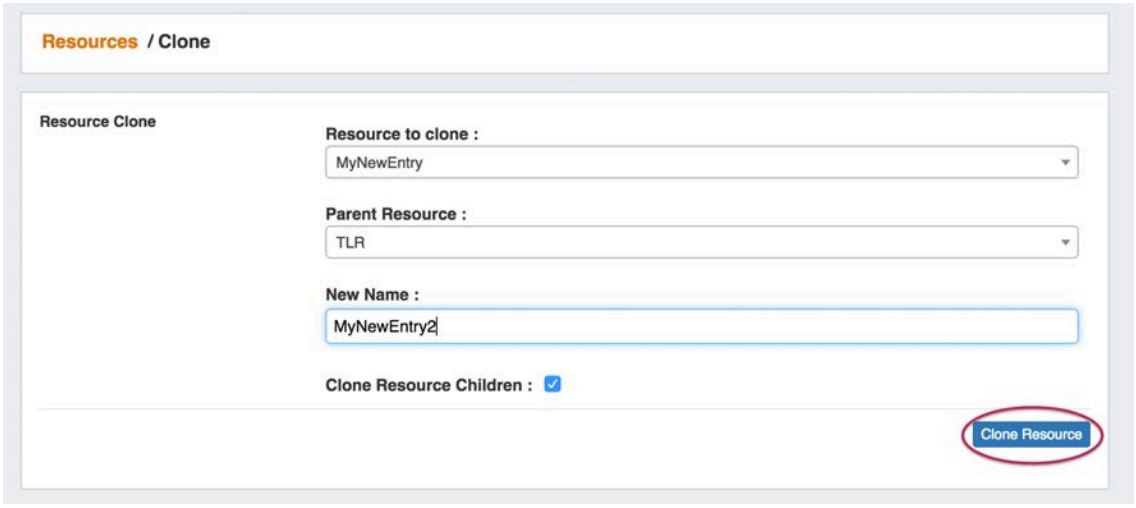
From the **Resources** Tab dropdown, select "Resource Clone". This will take you to the Resource Clone page.



You may also select "Clone" from the bottom of a Resource Entry page, in which case the Entry page you clicked "Clone" from will be automatically selected as the resource to clone.



Once on the Resource Clone Page, enter the Resource to clone, Parent Resource, and the New Name for the clone. If you wish for all children of that resource (and their children, if applicable) to be cloned, check the "Clone Resource Children" checkbox. When complete, click the "Clone Resource" button.



To repeatedly clone the resource, simply change the "New Name" field (if you want the subsequent clones to have a different name), and click "Clone Resource" again. You may repeatedly click "Clone Resource" and each click will produce a new clone.

When done, you may click on the link provided for the most recently created clone, or hit the back button from your browser to return to the parent resource.

Tip

Review the cloned resources using the chart view to see the hierarchy structure!

Additional Information

Continue on to the following pages for additional information on working with Working with Resources in Provision:

- Resource Concepts
- Customizing Sections
- Customizing Fields
- Gadgets
- Contact Manager

Customizing Sections

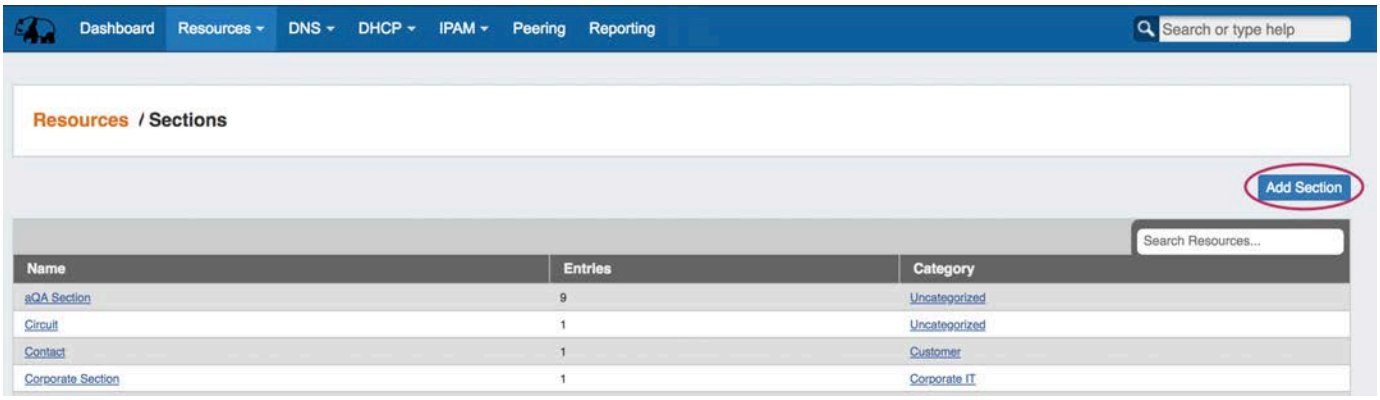
Customizing Sections

You can create as many Sections as you wish (Firewall, Server, VM, Virtual Interface, etc.) and customize the fields that you care about for each Section. For example, you may not need to track the console port for your virtual firewall, so you would simply not use that field for the "Virtual Firewall" Section. This way you can still track the console port for your physical firewalls like normal.

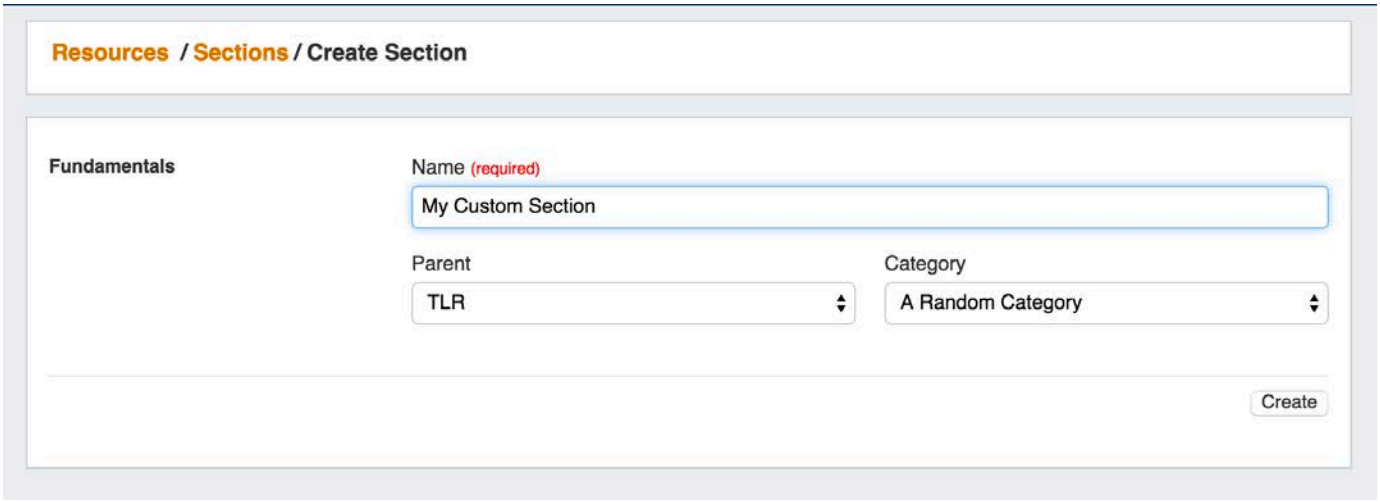
- Customizing Sections
  - Step 1: Create a New Section
  - Step 2: Add a Custom Field to a Section
  - Step 3: Edit Custom Field Data
  - Step 4: Add Gadgets to your Section
  - Additional Information

Step 1: Create a New Section

Click "Add Section" from the **Sections** sub-tab under the **Resources** Tab



Create a new Section by specifying a Name, Parent, and Category. Then hit the "Create" button.

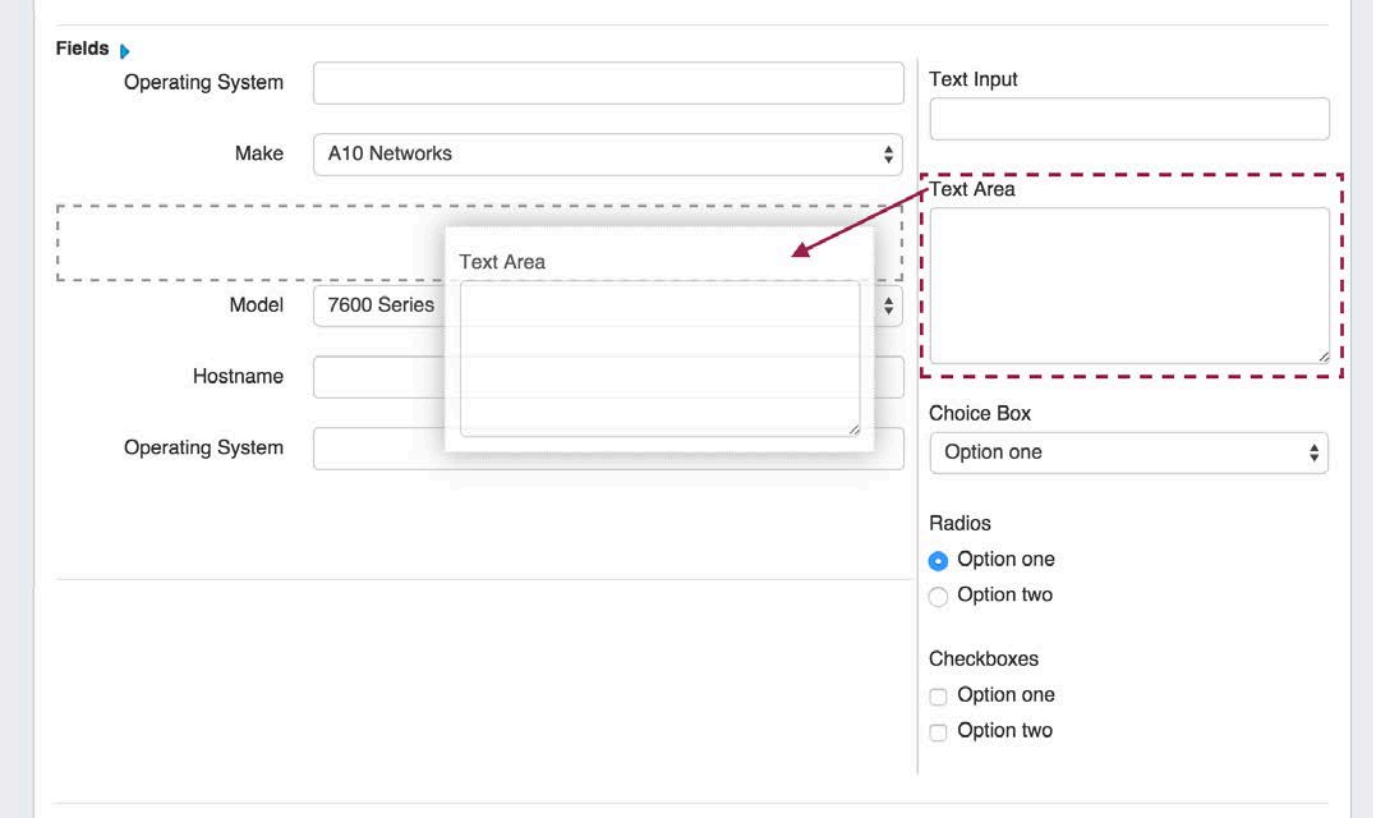


Step 2: Add a Custom Field to a Section

Manage existing fields and add custom fields for the selected Section by clicking **"**Edit Section"

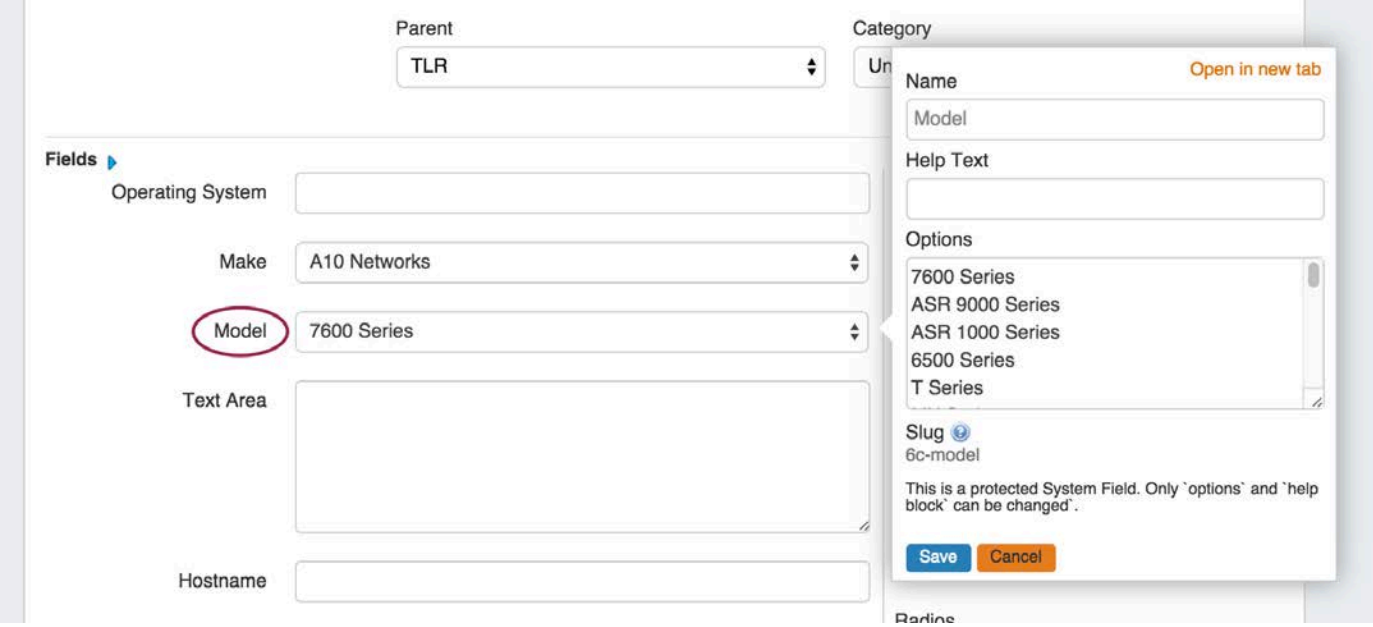


Add existing or Custom Fields for your Section. You can add new Custom Fields of different types (text input, text area, choice box, radios, checkbox) by dragging and dropping the fields as well as using any existing fields that are available. See the Customizing Fields page for more details.



Step 3: Edit Custom Field Data

Select the field name and you will get an editing window to modify the parameters of the field. Custom fields may be renamed and have other attributes updated, whereas protected system fields may have noted restrictions.



Step 4: Add Gadgets to your Section

You will notice on this customization screen that you also have an area for Gadgets. Gadgets are modules of additional functionality that can be added to the UI of a given Resource. Simply select the Gadget you want to show for that section, hit "Add", then organize by dragging into the order you wish them to appear on the page. Once added to the Section, Gadgets will be visible for all Resources of that Section.



For a detailed list of gadgets and descriptions, see the Gadgets page.

Additional Information

Continue on to the following pages for additional information on Working with Resources in Provision:

- Customizing Fields
- Gadgets
- Contact Manager

Customizing Fields

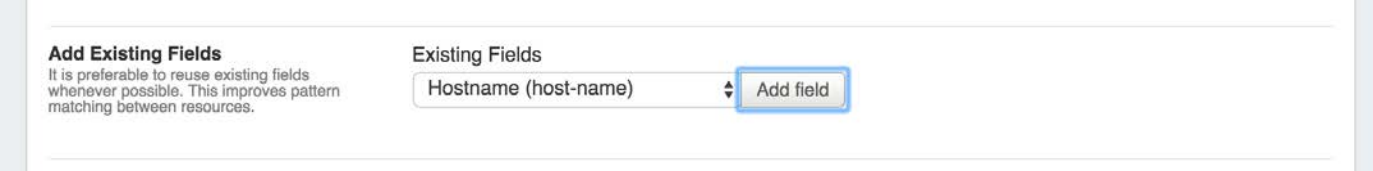
Working with Fields

Fields are individual pieces of data that you can associate with a particular Section. For example, you may want to have "Operating System", "Make", and "Model" fields for a server or device. Provision includes many default fields to choose from, but you may also create custom fields and edit fields to suit your needs.

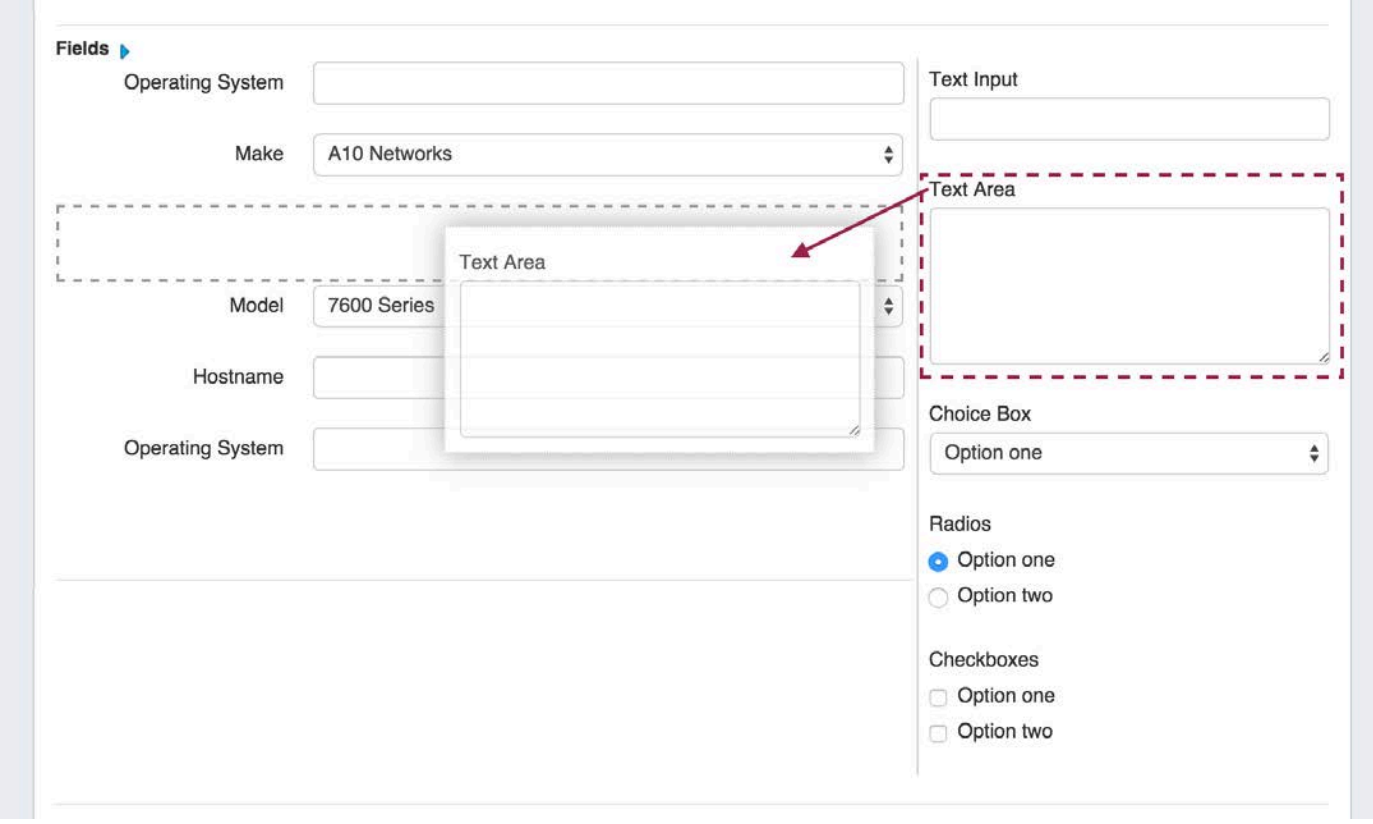
- Working with Fields
  - Creating Fields
  - Editing/Removing Fields
  - Additional Information

Creating Fields

To add an existing field to a Section, select the field name from the dropdown menu and click on the "Add Field" button.

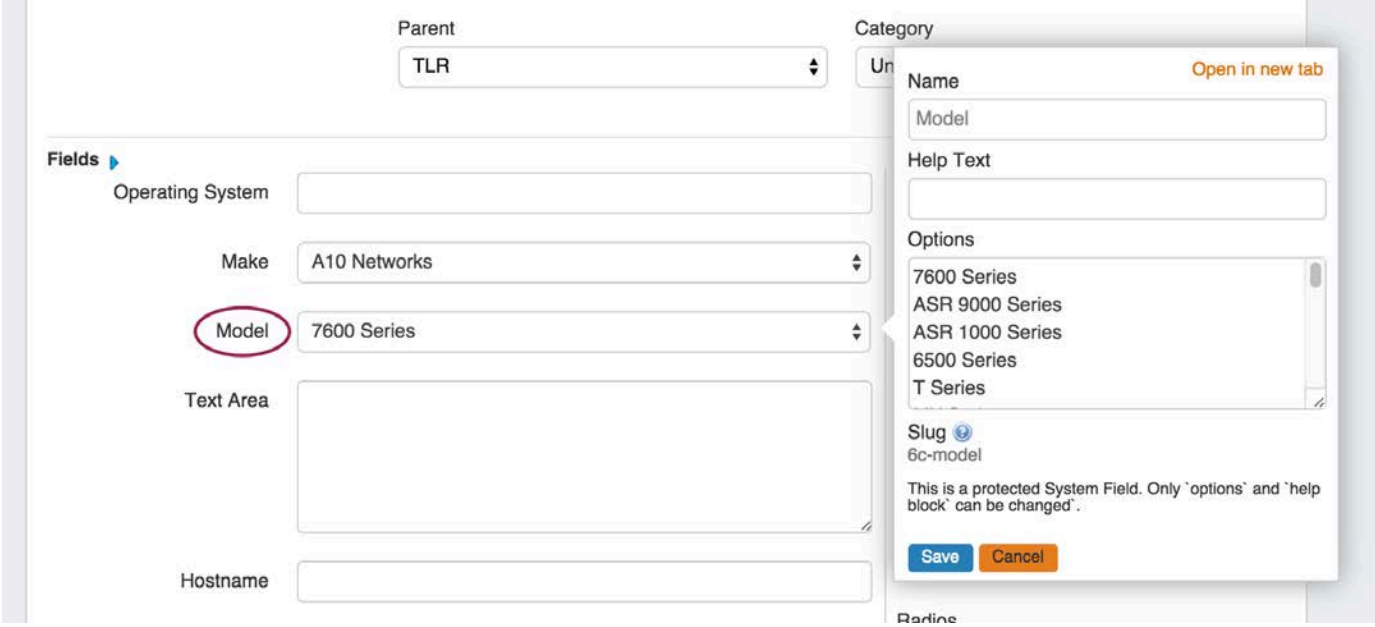


To add a new custom field to a Section, simply click on the custom field type name (Text Input, Text Area, Choice Box, etc), then drag the field over to the field list and release in the desired location. Edit the field name and options as described in Editing / Removing Fields.



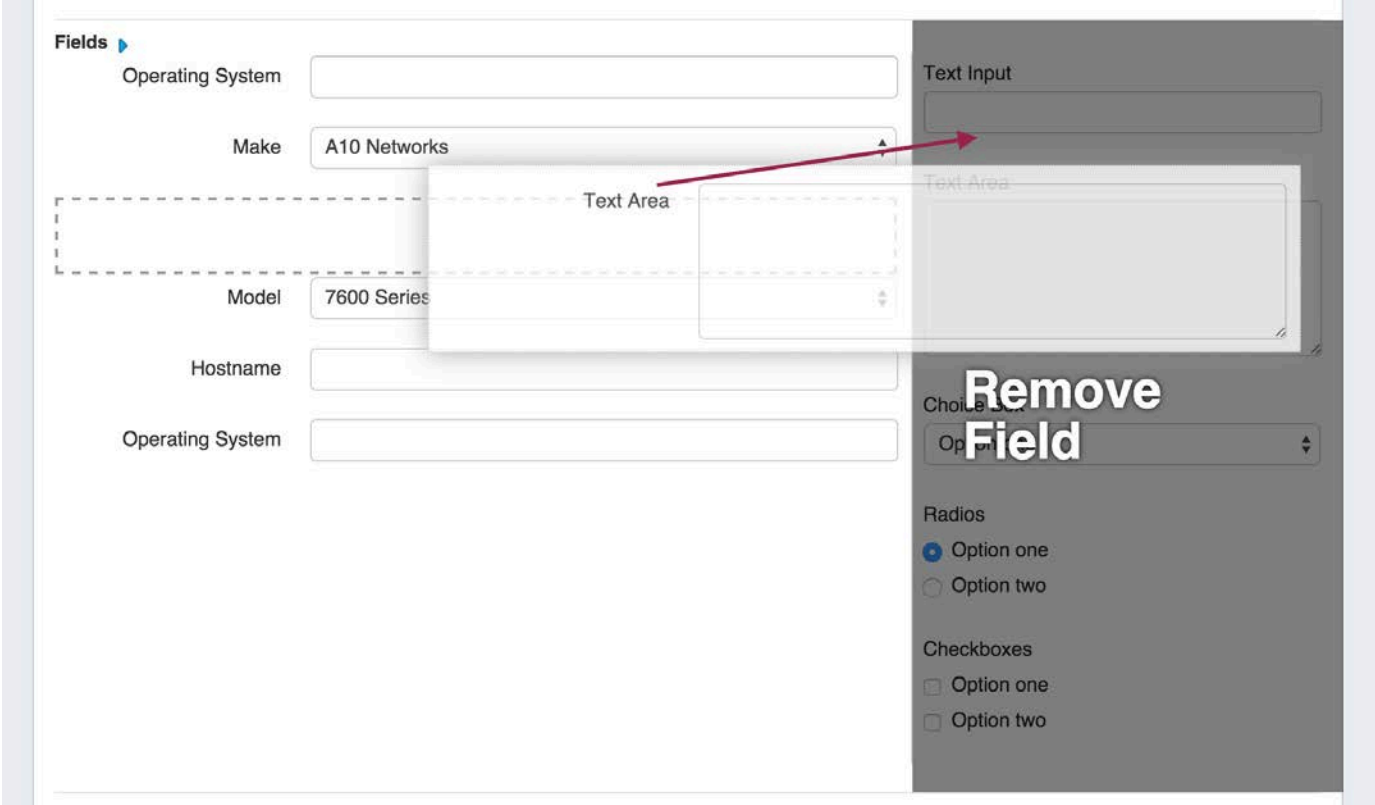
Editing/Removing Fields

Once fields are added to a Section, you can click on the field name to make additional changes to the fields. Custom fields may be renamed and have other attributes updated, whereas protected System Fields may have noted restrictions.



To rearrange the field list order, click and hold on the field name, then drag and drop into the preferred order.

To remove a field, click and hold on the field name, then simply drag and drop the field to the right side of the screen to where the "Remove Field" prompt is visible.



Additional Information

Continue on to the following pages for additional information on Working with Resources in Provision:

- Gadgets
- Contact Manager

Gadgets

Gadgets

- Gadgets
  - What are Gadgets?
  - Available Gadgets
    - Contact Info
    - Contacts
    - DHCP Server Management (Legacy)
    - DHCP Customer Configuration
    - DNS
    - DNS Autogenerator
    - Document Storage
    - IPAM (Legacy)
    - IPAMv2
    - Notes
    - Peer Groups
    - Peering Sessions
    - Peering VRFs
    - Resource Linkage Gadget
    - Resource View
    - Reverse API Console
    - Tech Info
  - Creating your own Gadgets
  - Additional Information

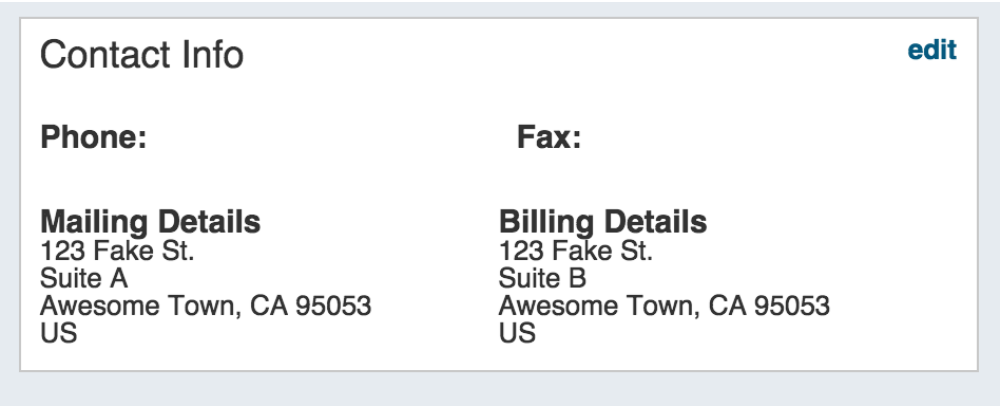
What are Gadgets?

Our gadget system is similar to the Atlassian Gadget system (and Google Gadgets). When creating or editing a Section, gadgets can be added in a way similar to how you would add or remove a field (see Customizing Sections). Gadgets are best described as self contained webapps; widgets but with more power. Gadgets can have their own fields, HTML templates, and even accompanying scripts and stylesheets. They can interface with the API to display simple information such as the Type of the Resource, or they can perform much more complex functions as demonstrated with the IPAM gadget in the following section.

Available Gadgets

Contact Info

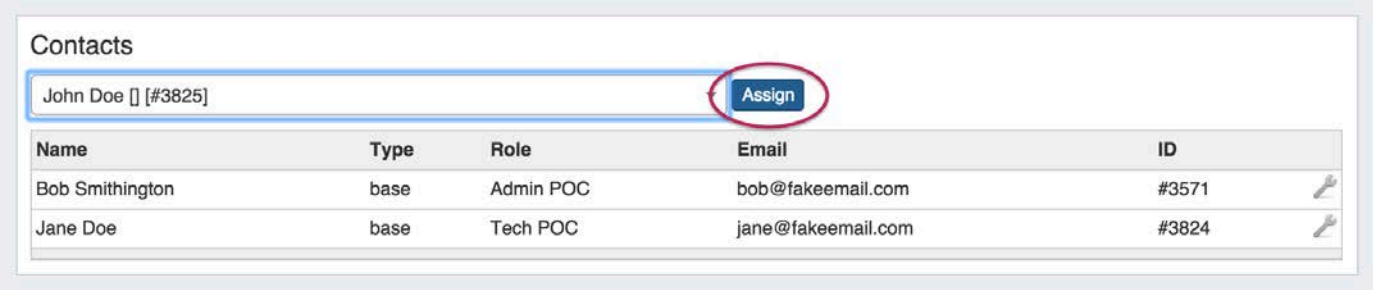
This Gadget is used on the Resource Holder Section type. In the Contact Info Gadget, you can track information such as mailing / billing addresses, phone number, and fax number for that Resource.



Contacts

The Contacts Gadget may be used on any Section type to assign a contact (from the Contact Manager) to a Resource.

To assign a contact, search for and select the desired contact in the search box, then click "Assign". The Contact will show in the list below.



Contacts Action Menu:

The Action Menu (wrench icon) in the contacts gadget has three options:

Set Role: Opens a dialog box to select a role to assign to the contact, add a new role, or delete an existing role.



Unassign: Unassigns the contact from the resource

View: Redirects to the contact's detail page in the Contact Manager

Refer to the Contact Manager page in the documentation for more detail on working with the Contacts gadget and the Contact Manager.

A Note on Permissions

The ability to view and assign a contact to a resource is restricted by the Parent assigned to the contact upon creation. If a user does not have permissions for a contact's Parent Resource, that contact will not be able to be viewed / assigned via the gadget.

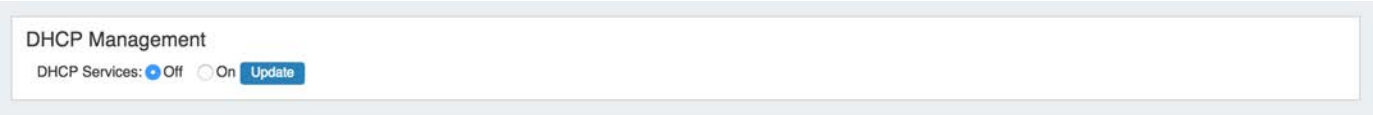
DHCP Server Management (Legacy)

The DHCP Server Management Gadget has functionally been replaced by the **DHCP** Tab updates in version 7.0.0.

Although the DHCP Management Gadget may still be used, we recommended checking out the **DHCP Groups**, **DHCP Servers**, and **DHCP Pools** pages instead for an improved user experience.

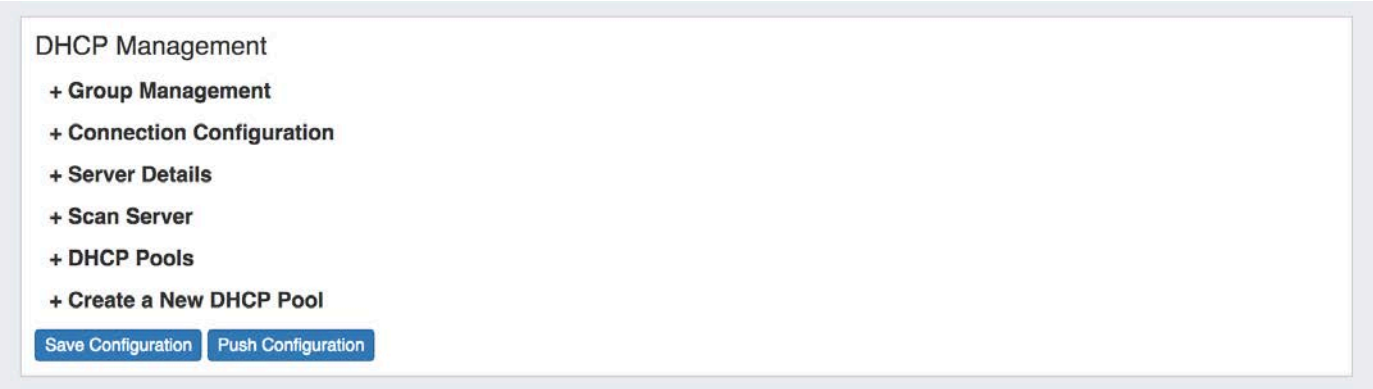
The DHCP Server Management Gadget should be used with the "Server" Section.

When initally added to the section, it will show in the "Off" configuration:



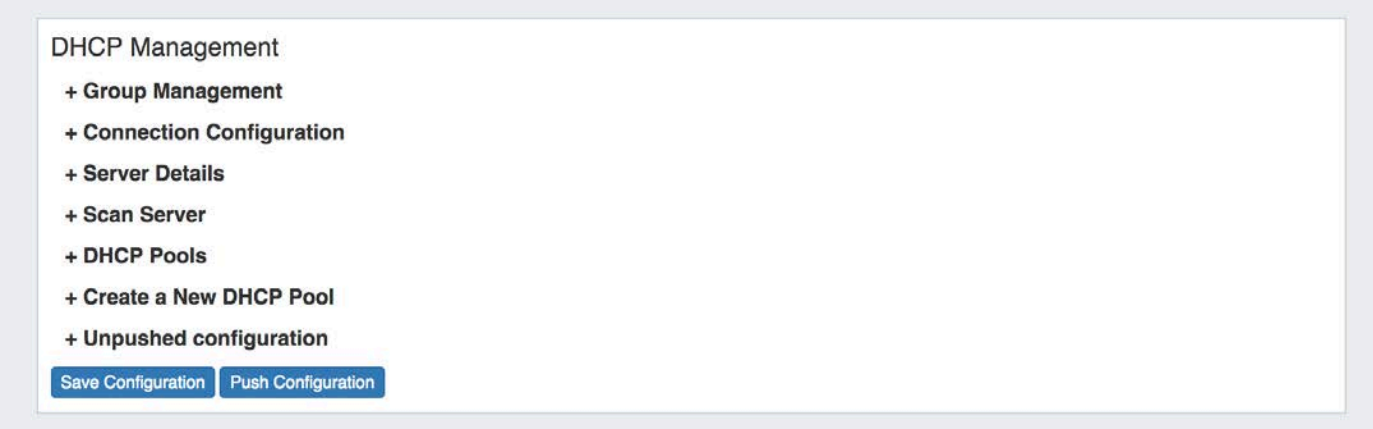
To enable the DHCP Management Gadget, select the "On" radio button next to "DHCP Services", and click the "Update" Button.

The DHCP Management Gadget in the "On" configuration:



Each of the sections in the DHCP Management Gadget - Group Management, Connection Configuration, Server Details, Scan Server, DHCP Pools, and Create a New DHCP Pool - may be expanded to change settings in each area.

Additional sections may appear in the Gadget once unpushed or pushed configurations are saved:



For detailed information on working with the DHCP Management Gadget, see DHCP Tab - Managing DHCP Server Configurations.

DHCP Customer Configuration

The DHCP Customer Configuration Gadget allows users to assign IP aggregates to the DHCP server pools and generates DHCP Server Configuration changes depending on the tag(s) and DHCP options selected.

These configurations are then sent to the associated DHCP server Management Gadget as "Unpushed Configurations", where they may be held until a manual or schedule push occurs.

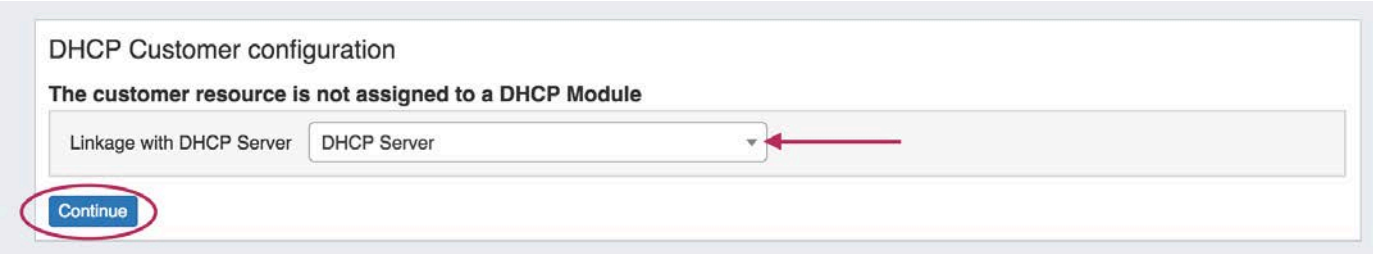
Before You Begin

Before using the DHCP Customer Configuration Gadget, the following should be set up in ProVision:

- The associated DHCP server should be created in ProVision and set up in the DHCP Management Gadget or **DHCP** Tab - **DHCP Servers** list (See: **DHCP** Tab).
- Add the DHCP Customer Configuration Gadget to the desired Section. You may want to create a specific "DHCP Customer" Section for DHCP customer entries (See: Customizing Sections).
- Have, or set up APNIC Aggregates for DHCP use from the **IPAM** Tab with the desired IP space type, Region(s), Tags, and any desired VLAN criteria. Regions are a required field when assigning IPs from the DHCP Customer Configuration Gadget. Any APNIC aggregate may be used for DHCP, so you may want to create a "DHCP" IPAM Tag to help denote the desired aggregate(s).
- If you are planning on Direct Assigning the DHCP blocks in the Customer Configuration Gadget, ensure that the desired blocks in the DHCP are split to individual /32 or /128 size. Smart Assign and Smart Browse will automatically split the block to assign the correct size.

Associate with DHCP Server

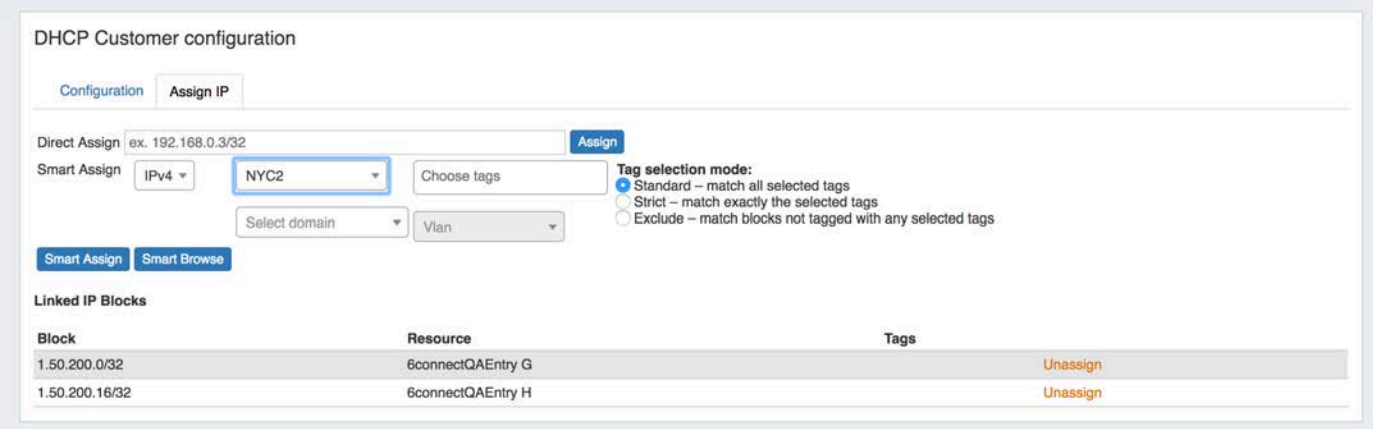
Once the Gadget is enabled, the first step is to associate it with an already-created DHCP server in ProVision. Select the DHCP Server to link to, and click "Continue".



Assign IP's

Next, Assign IP's for Pools from DHCP Aggregates. You may Direct Assign, Smart Assign, or assign from Smart Browse based on the selected criteria.

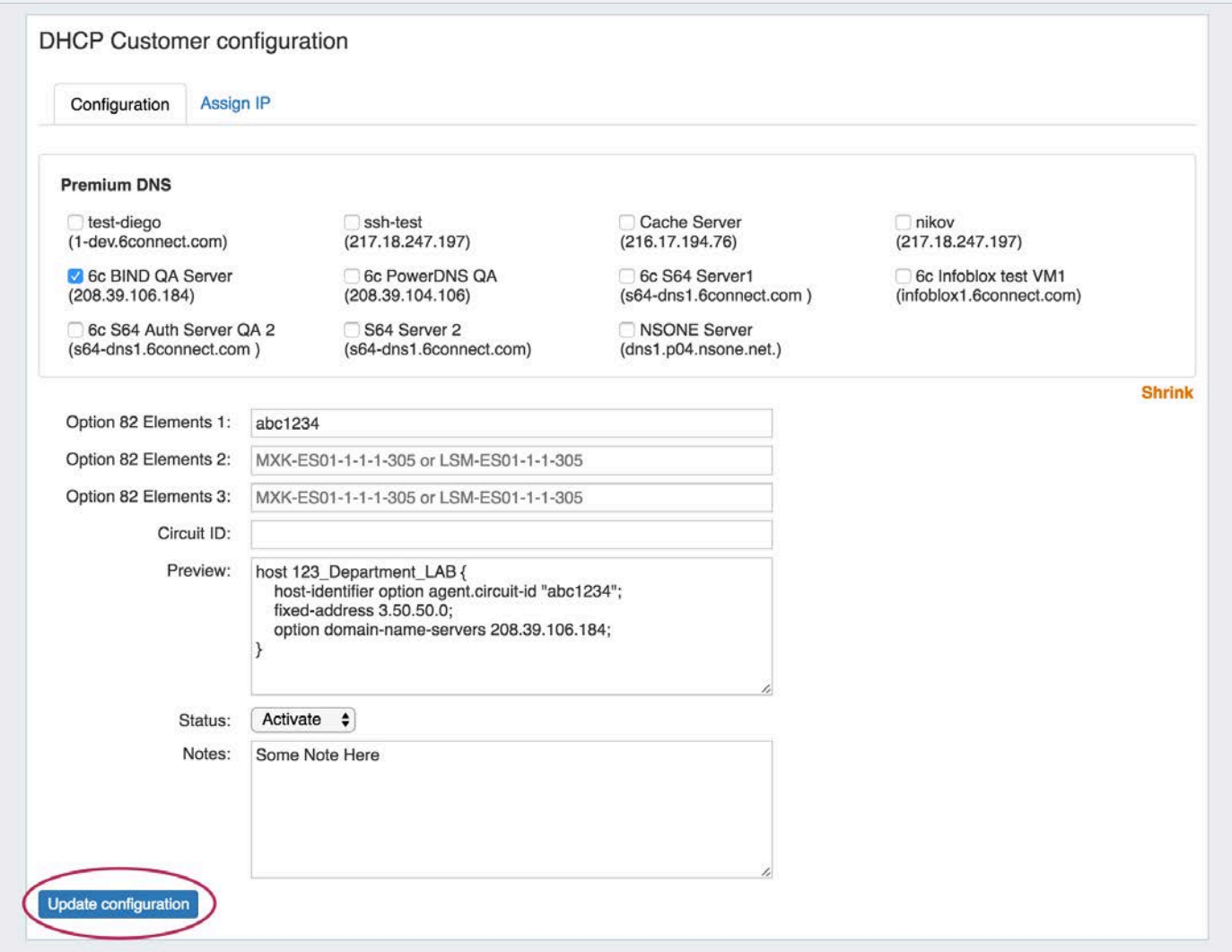
Once assigned, a Pool Name will be automatically generated under "Resource". Pools may be unassigned by clicking the "Unassign" button.



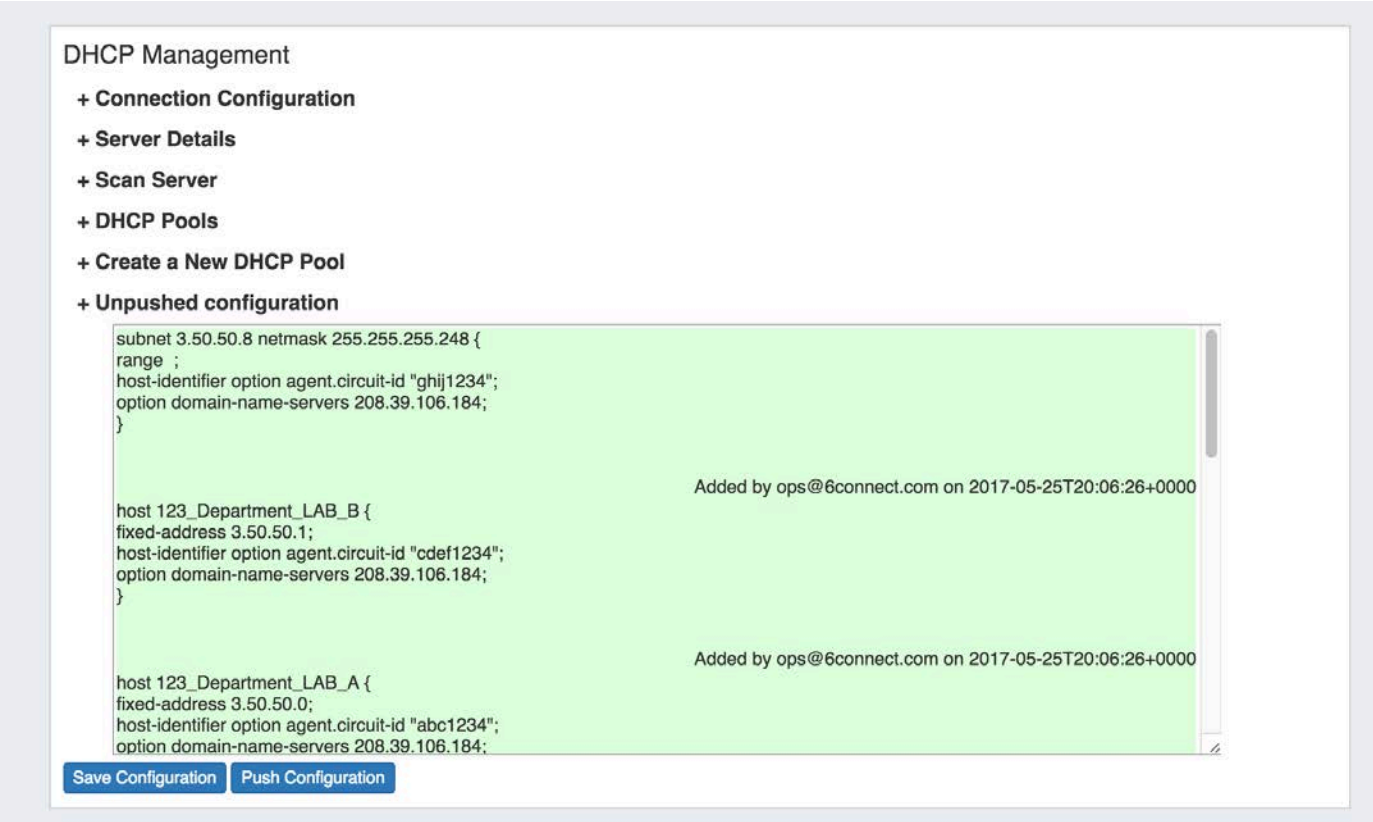
Set up Configuration

Once IP's have been assigned, the "Configuration" tab will appear. Click on the tab, and then select or deselect the DNS Server associated with the Customer, add Option 82 Elements, Add Circuit ID text, and add notes if desired. Use the "Preview" field to confirm the accuracy of the data, and select the status as "Activate" or "Terminate".

When done, click "Update Configuration". The configuration will be sent to the associated DHCP server's DHCP Management Gadget.



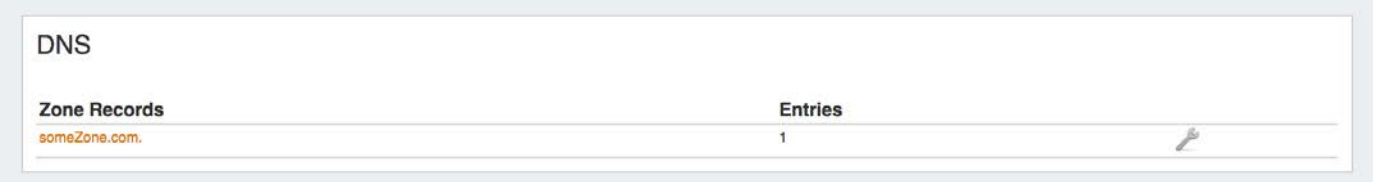
Reviewing and Pushing Configuration Updates
When a configuration has been saved from the DHCP Customer Configuration Gadget, it will appear as an "Unpushed Configuration" in the DHCP Management Gadget for the DHCP Server. If you expand that option, you will see the config elements highlighted – red are config elements that will be deleted, green are config elements that will be added.



At this point, an Admin user may manually push the updated configuration, or use a scheduled DHCP push task to automate the pushes.

DNS

This gadget is used on the Resource Holder Section type. The DNS Gadget shows zones that have the current Resource Holder set as their parent, and provides links to the zone's View Zone page.



To go to the View Zone page, you may either click on the zone name, or click on the Action Menu (wrench icon) and select "View Zone".

For more information on DNS functions and managing zones, refer to the documentation for the DNS Tab.

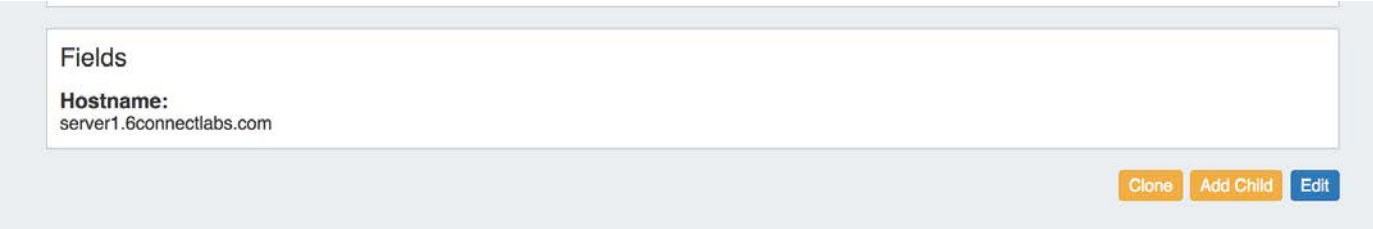
DNS Autogenerator

The DNS Autogenerator Gadget uses the Hostname field (6c-hostname-fqdn) of the Resource to generate a list of DNS forward and reverse zone entries based on the blocks assigned to the Resource in the IPAM Gadget It will generate a list of potential zone records - just select the ones you want to create, save the changes and you are ready to push the zones.

Enabling the DNS Autogenerator

To set up this Gadget, ensure that the Section of the Resource (typically, "Resource Holder"), has the "Hostname" field (6c-hostname-fqdn) and the DNS Autogenerator Gadget added to the Section. (See: Customizing Sections and Customizing Fields).

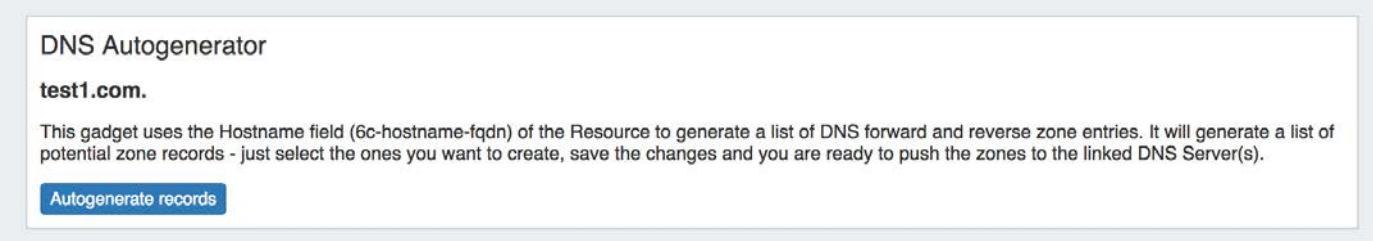
Then, check that the Resource itself has information entered into the "Hostname" field. If the field is already filled out, it will show in the "Fields" information area at the bottom of the Resource Entry page. To add or edit the hostname, click "Edit" at the bottom of the resource entry page, add the information to the Hostname field, and click "Save".



Once a hostname has been associated with a Resource, and the page refreshed, the DNS Autogenerator Gadget will be visible.

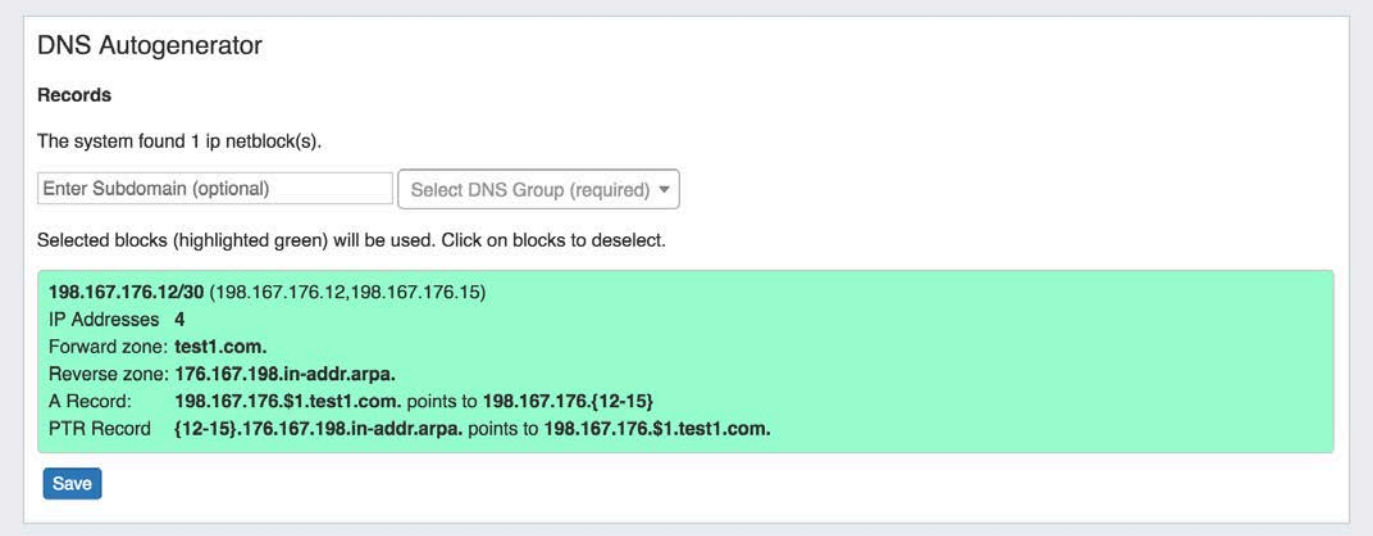
Working with the DNS Autogenerator

One enabled and visible, the DNS Autogenerator Gadget will initially show a short description, the hostname, and a button to "Autogenerate Records".

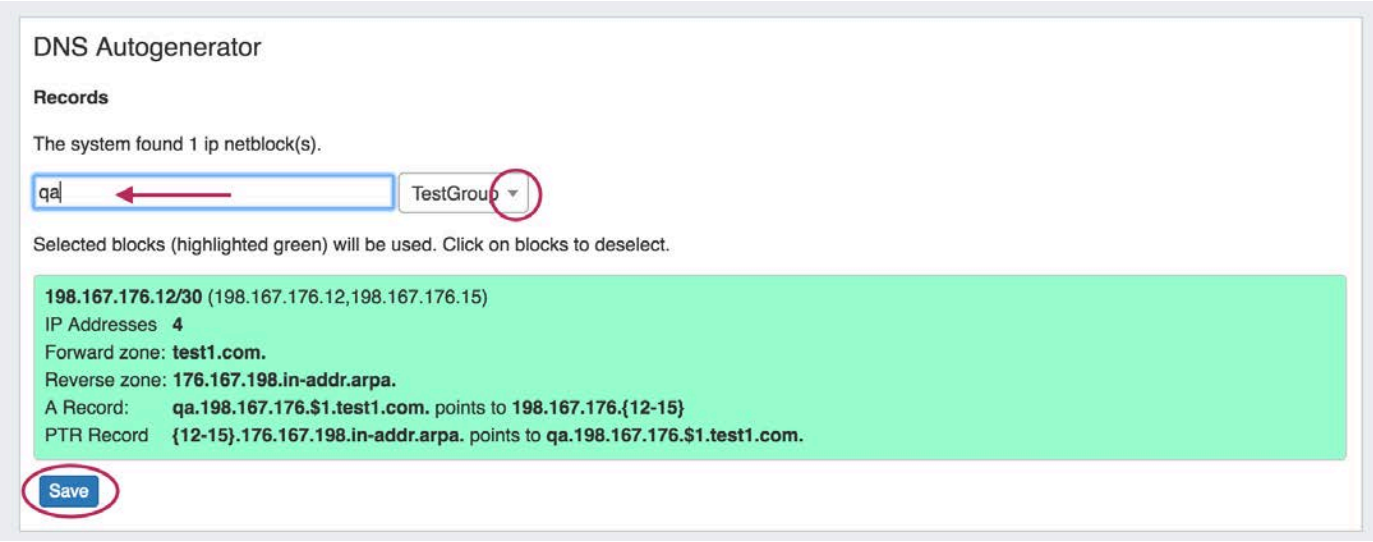


Before starting, check that the blocks assigned to the Resource in the IPAM Gadget are correct and up-to-date - the Autogenerator uses these blocks to create the records. If using a DNS Group other than "Default" to hold zones, ensure that the desired DNS Group to hold the generated records has been set up. (See: Working with DNS Groups).

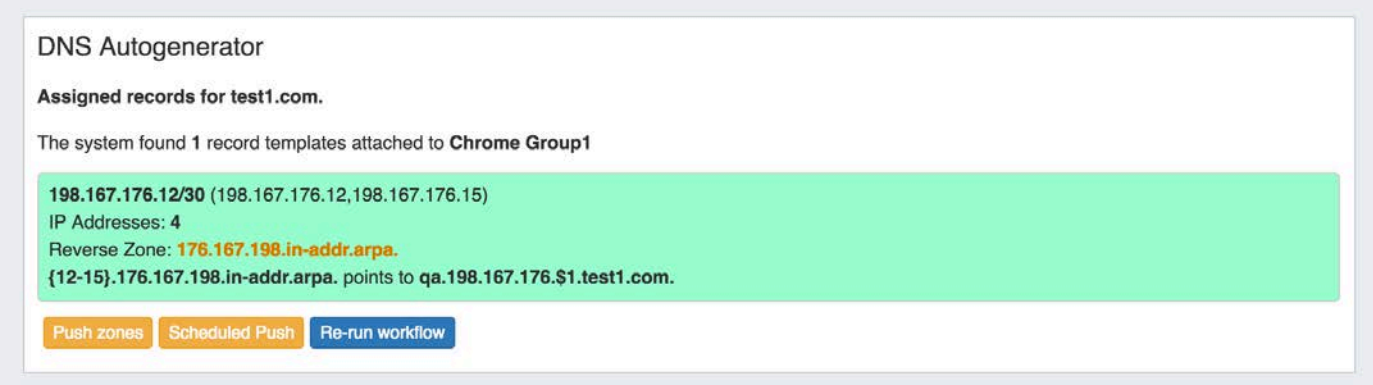
When ready, click the "Autogenerate Records" button. The Gadget will search through the blocks in the IPAM gadget and provide a list of found blocks.



Next, add in a subdomain if desired (optional), and select the DNS Group to hold the records (required). Click on the listed blocks to select (highlighted green) or deselect (grey) for DNS zone / record creation.



Finally, hit the "Save" button at the bottom of the gadget - the selected forward and reverse DNS records will be created and added into the DNS Group, viewable in the DNS Tab.



At this point, you may choose to immediately push the autogenerated zone(s), schedule a push, or re-run the workflow to edit the subdomain or group.

Document Storage

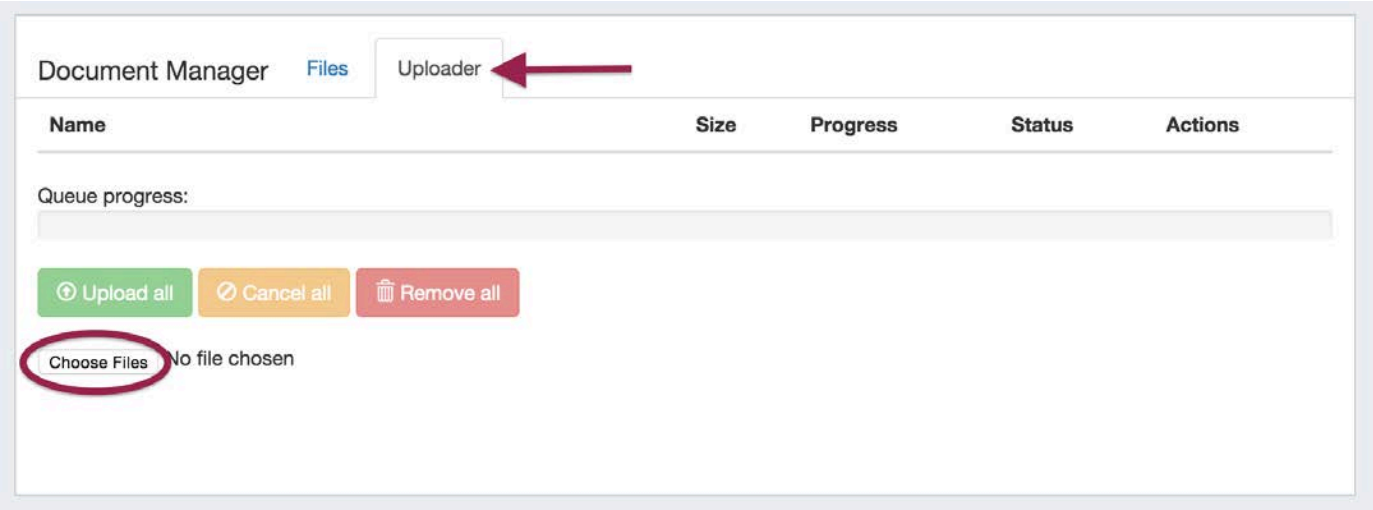
The Document Storage gadget allows you to upload documents to a resource, and have those documents accessible to download from the 6connect cloud to your local machine. It may be enabled on any Section type.

Note: The uploader has a file size limit of 14mb, files above this size will not successfully upload.

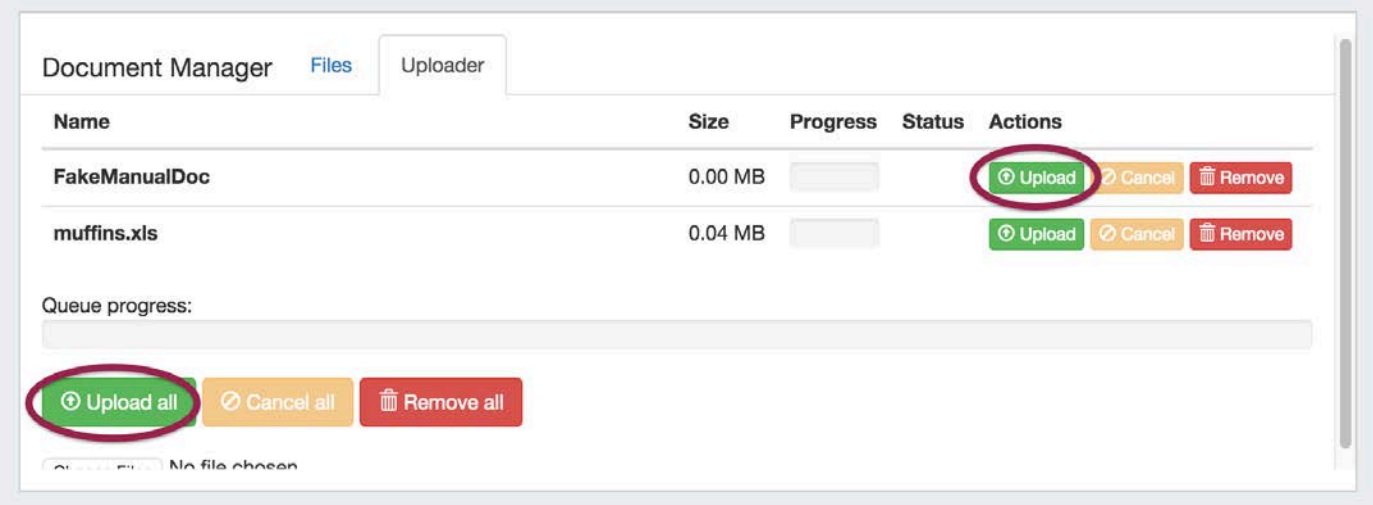
Upload Files

First, ensure the Document Storage gadget is enable for the Section type. Then, select the "Uploader" tab under the Document Manager gadget.

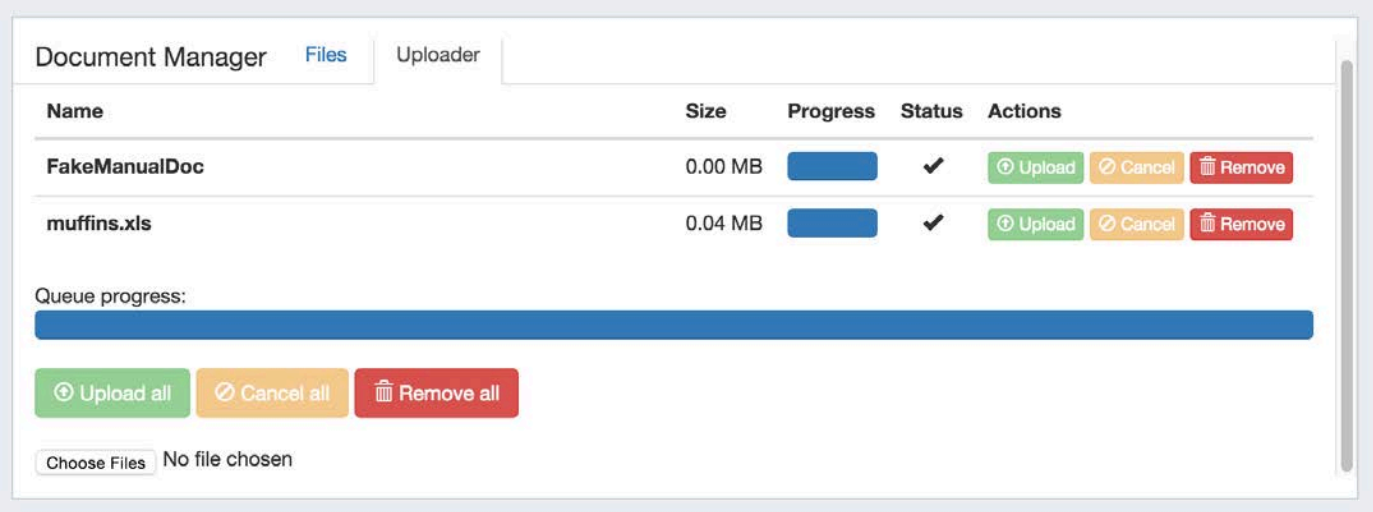
Click on the "Choose Files" button, and select the file(s) you would like to upload.



The selected files will show in a list under the Uploader tab. To upload the file(s), click on either the "Upload" button adjacent to the file to upload the individual file, or the "Upload All" button to upload all files listed. You may also choose to remove files from the upload list, or cancel.



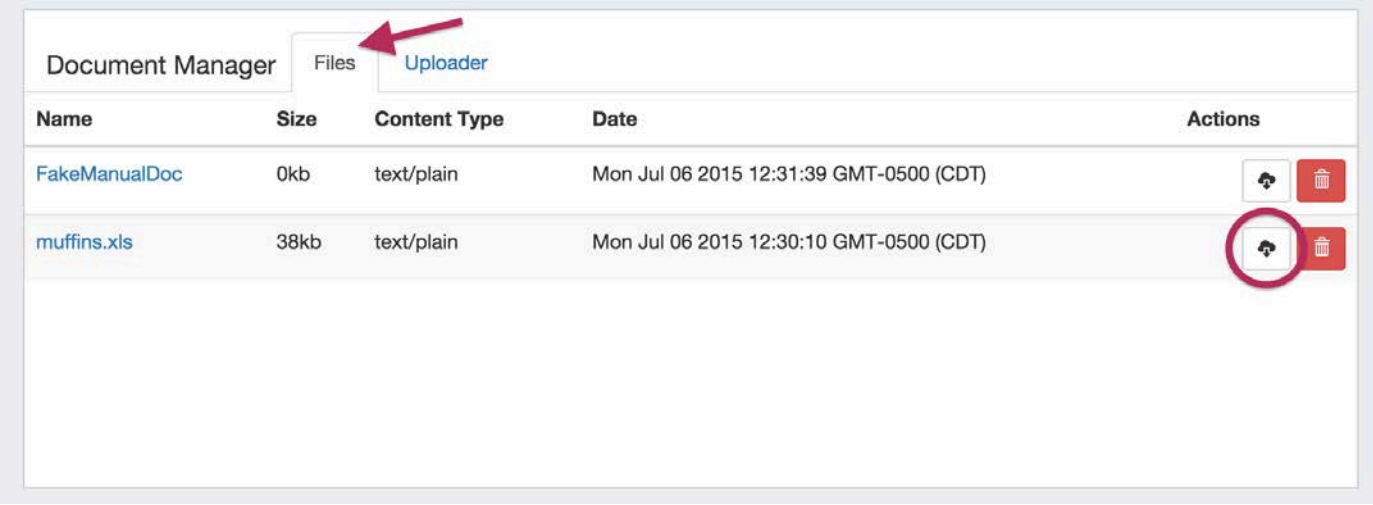
Once the progress bar is filled in, your upload is complete! Your files will show under the "Files" tab.



Download Available Files

Successfully uploaded files are listed under the "Files" tab of the Document Manager gadget. From here, you can see the file Name, Size, Type, Date, and available Actions.

To download a file to your local machine, ensure you are on the "Files" tab, then click on the "Download from Cloud" icon.



If a file is no longer needed, you may delete the file from the Document Manager by clicking on the Delete (trash can) icon.

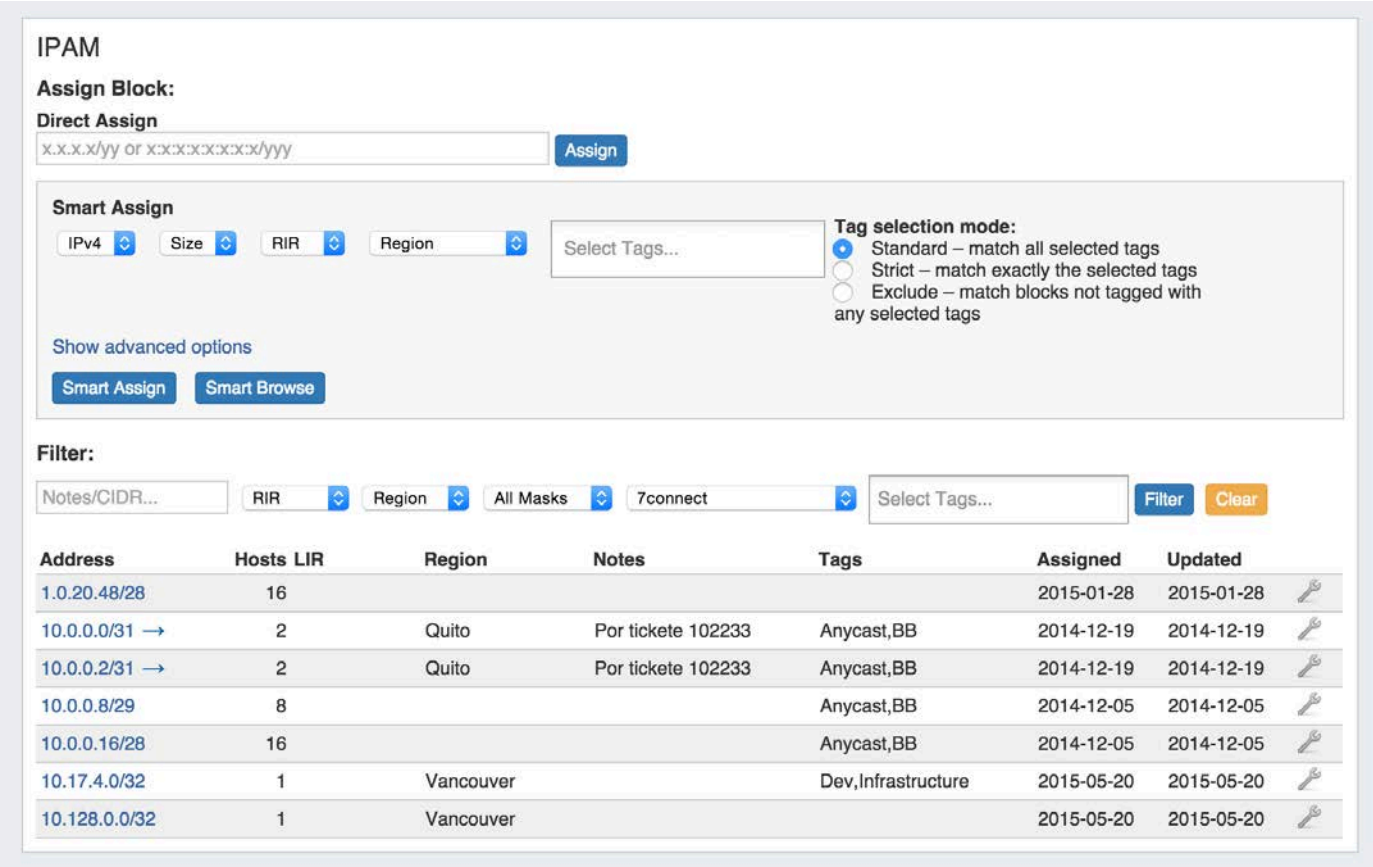
IPAM (Legacy)

The IPAM Gadget has functionally been replaced by the IPAMv2 Gadget in version 7.0.0.

Although the original IPAM Gadget is still able to be used by existing customers, we recommend switching to the IPAMv2 Gadget for improved speed, filters, and UI customization options.

This gadget is used on the Resource Holder Section type. IPAM Gadget allows you to view, assign, and manage blocks for that resource.

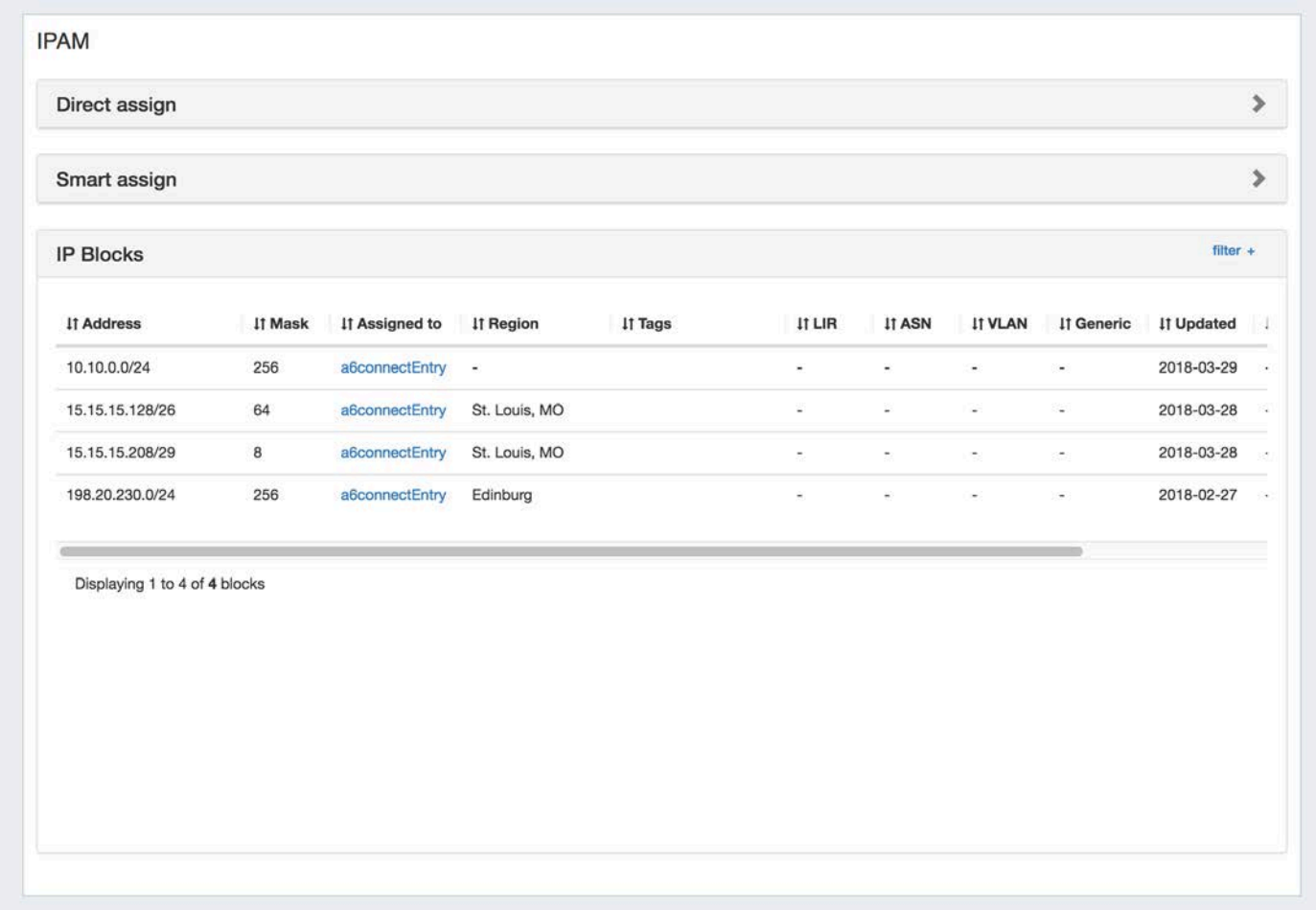
For more information on assigning and managing blocks, see Working with IP Blocks - Assigning IP Space.



IPAMv2

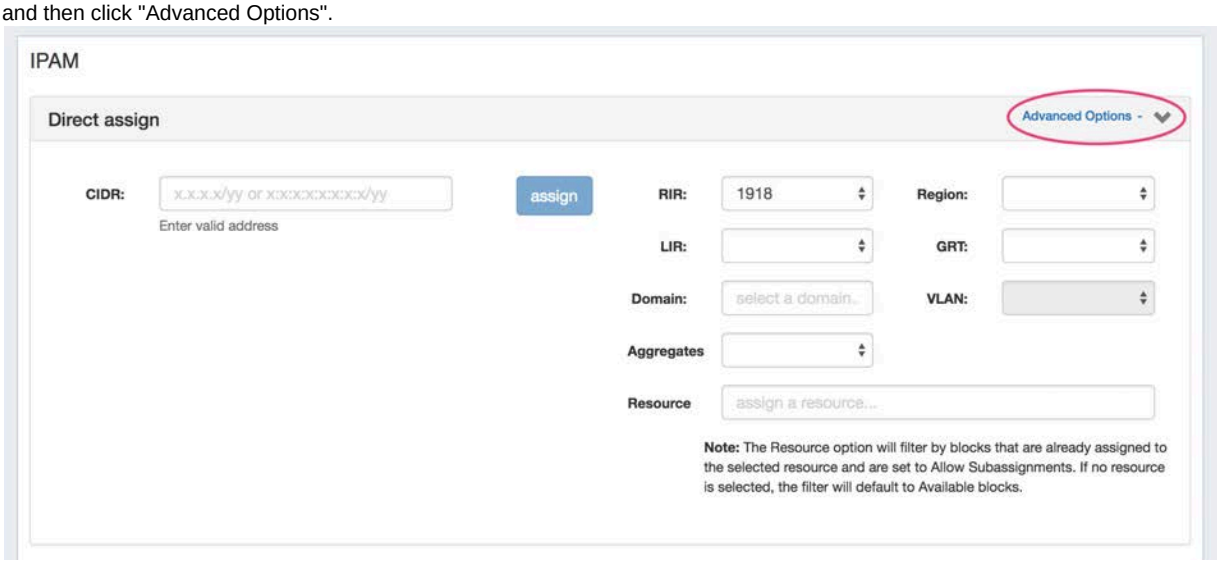
The IPAMv2 Gadget is used on the Resource Holder Section type, and allows you to view, assign, and manage blocks for that resource. As of ProVision 7.0.0, it replaces the previous IPAM Gadget.

It contains improvements over the original IPAM gadget, such as improved speed, pagination, advanced filters, sortable columns, reverse zone generation, customized column order/names, and custom data fields.



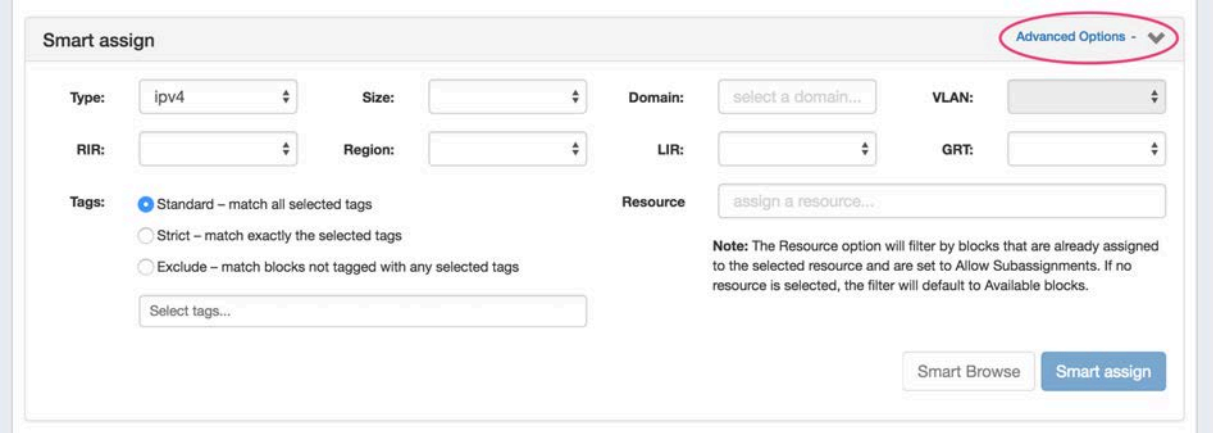
Using this gadget, you can:

- **Direct Assign** with or without additional filtering options. In cases where duplicate blocks exist, using the filter options with Direct Assign allows you to specifically control which of the dupilicate blocks is assigned.
  - To open the Direct Assign section of the Gadget and the advanced options, click the expansion caret in the upper right section, and then click "Advanced Options".

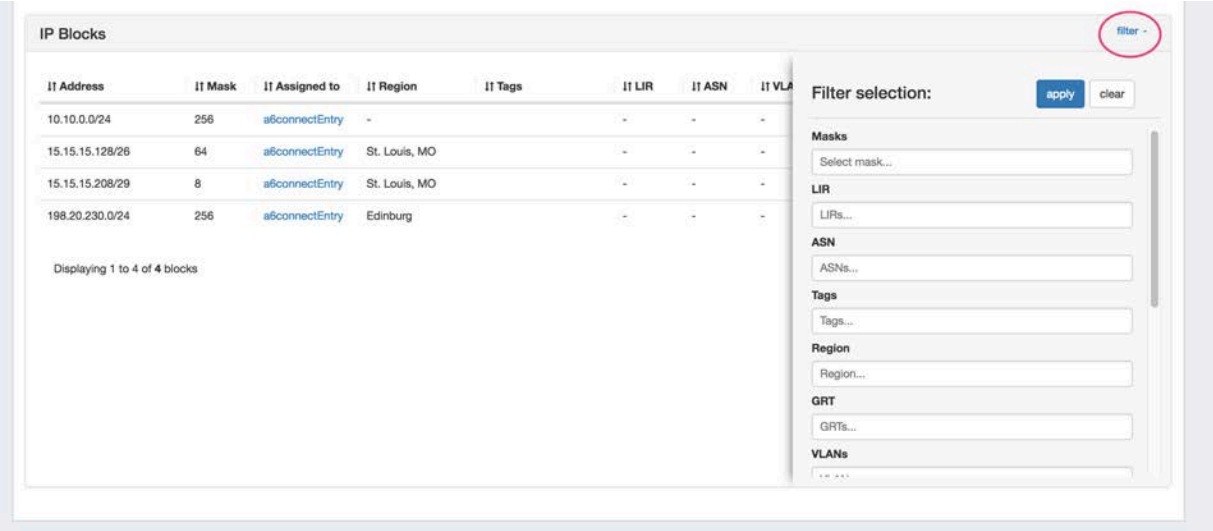


- **Smart Assign** with or without additional filtering options. Required fields for Smart assign are: "Type", "RIR", and "Size", but additional fields such as "Region", "VLAN", "LIR", "Resource" (for blocks that allow subassignments), and various "Tag" mode options.
- **Smart Browse** using smart assign filter options to bring up a browsable list of blocks meeting that critera, from which you may assign or assign out of.

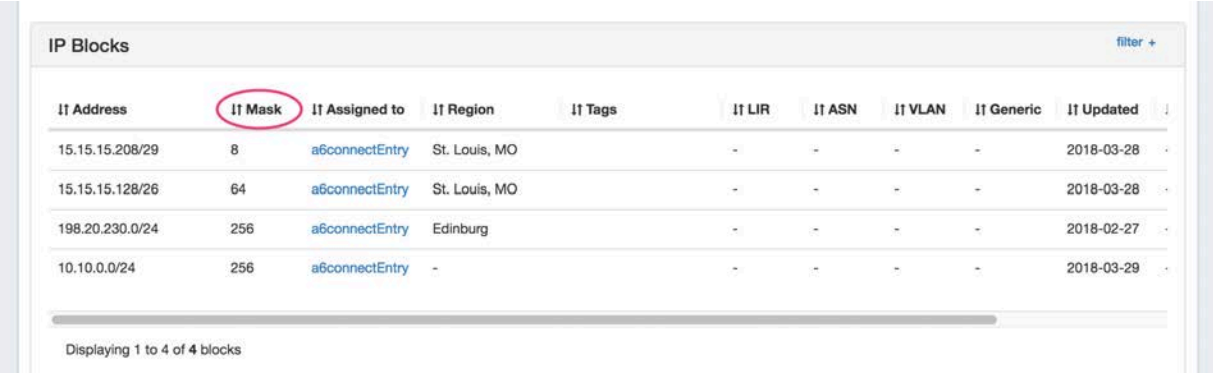
- To open the Smart Assign / Smart Browse section of the Gadget and the advanced options, click the expansion caret in the upper right Smart Assign section, and then click "Advanced Options".



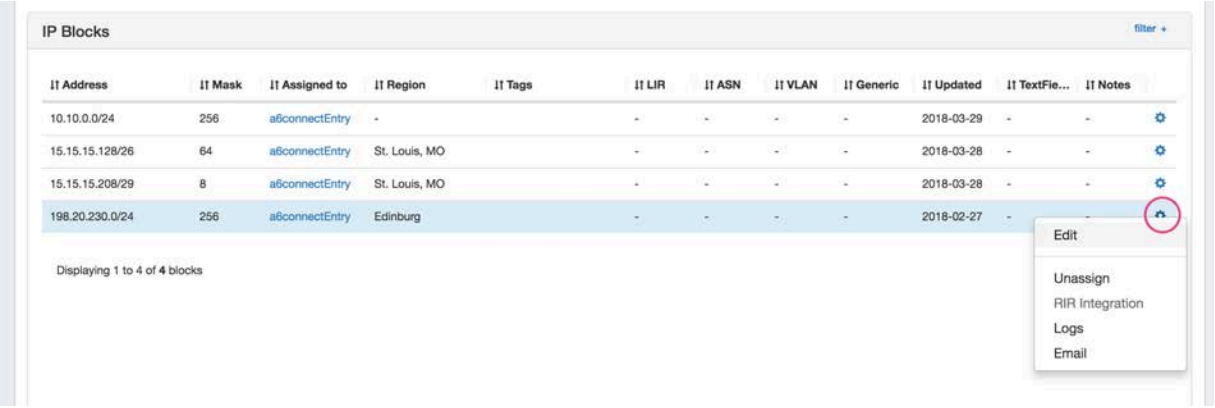
- **View, Filter, and Sort Assigned Blocks** for the resource via the "IP Blocks" list.
  - To Filter: click the "filter" link in the top right area of the header to open filter options. Enter your desired criteria and hit "Apply".



  - To Sort: Click on the header name you wish to sort by, and the field will toggle between ascending and descending order.



- **Manage Assigned Blocks** for the resource by going into the "IP Blocks" list, and using the Action Menu (Gear Icon) to Edit, Unassign, View RIR Integration, Logs, or Email.

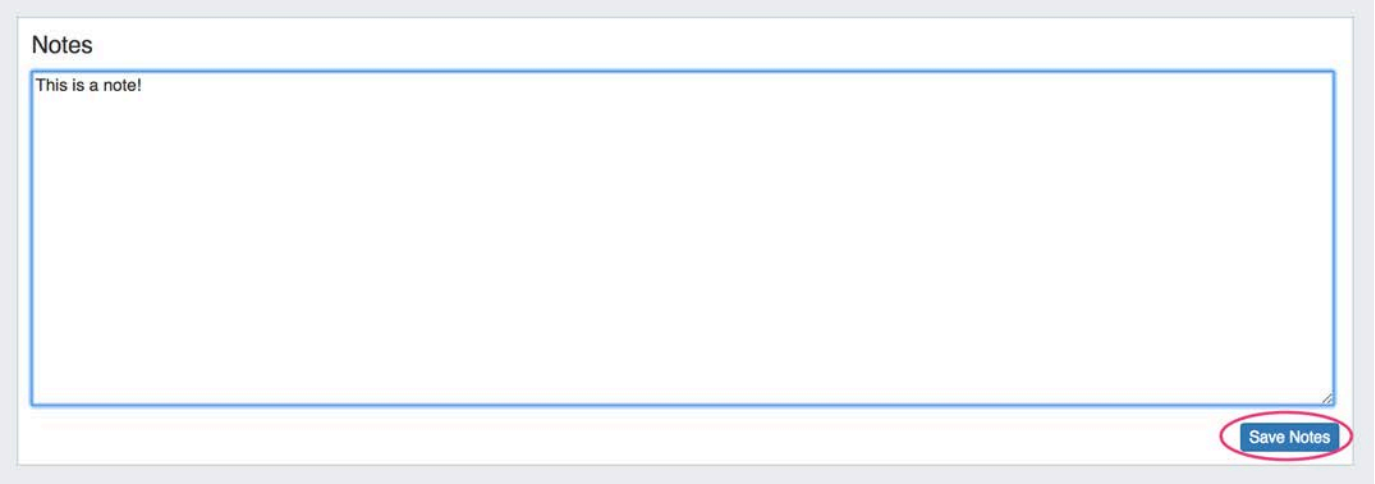


For more information on assigning and managing blocks, see Working with IP Blocks - Assigning IP Space.

Notes

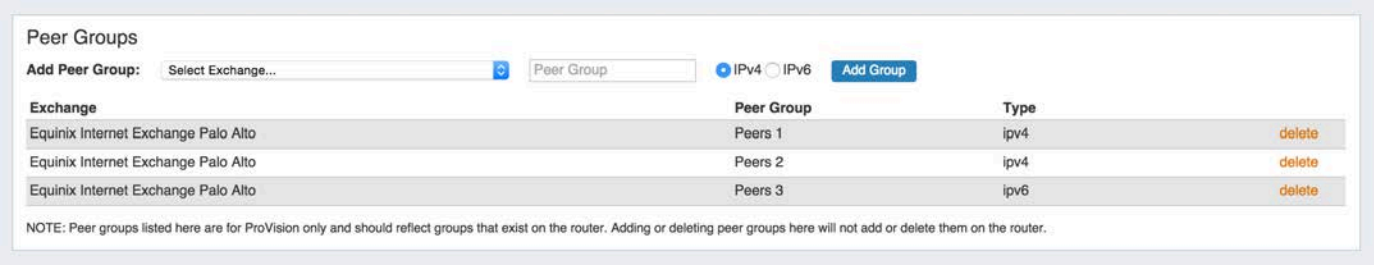
The Notes Gadget is a simple text area box, where you can save text notes relevant to the Resource.

To save a note, just click inside the note box, type the desired notes, and click the "Save Notes" button.



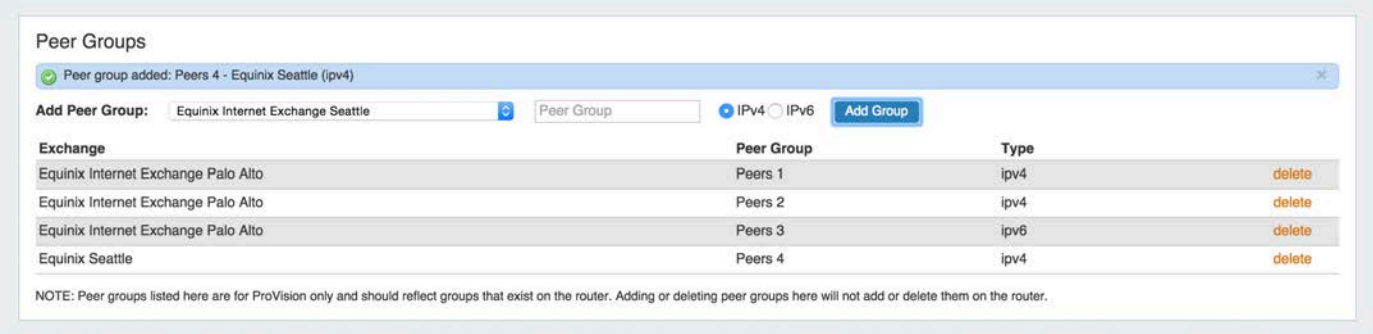
Peer Groups

The Peer Group Gadget allows you to add peer groups for IPv4 and IPv6 for a selected exchange from a router's Resource Entry page.



To do this, simply select the exchange, type in a Peer Group name in the text box, select IPv4 or IPv6, the click "Add Group".

Peer Groups added from this gadget will be then be available to select in the "Add Session" dialog box in the **Peering** tab.



Note

Peer groups listed in the Gadget are for ProVision only and should reflect groups that exist on the router.

Adding or deleting peer groups from the Gadget will not add or delete them on the router.

For additional information on Peering, see Peering v2.

Peering Sessions

This Gadget is used on the Router Section type. From here, new sessions may be added by clicking on the "Add Session" button, then entering session information into the "Create New Session" page.



Existing sessions may be managed by clicking on the "View" link for the session, bringing up the Peering v2 session details page.

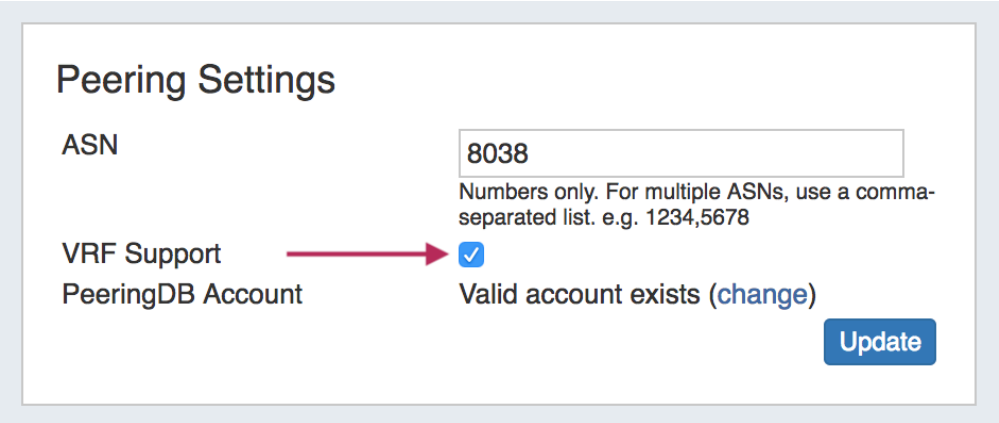


For additional information on Peering, see Peering v2.

Peering VRFs

The Peering VRF Gadget allows you to add VRFs from a router's Resource Entry page.

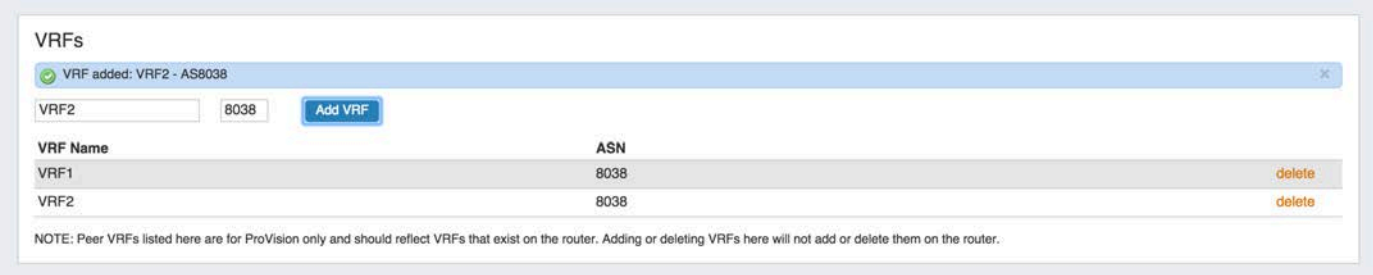
Enabling "VRF Support" in the Admin home page under "Peering Settings" will automatically add the VRF gadget to the router Section.



The VRF gadget will then be accessible in a router's Resource Entry page.



To add a VRF, type the VRF name and ASN, then hit "Add VRF".



To delete a VRF, click on "delete" next to the VRF entry in the gadget.

Once VRFs are set up for a router, the source ASNs for the associated VRFs will appear in the Source ASN dropdown when adding or editing a session for that router from the **Peering** tab.
Peering VRF currently only supports Cisco routers.

Resource Linkage Gadget

The Resource Linkage Gadget allows you link a Resource to other Resources, and display the links in one place, without needing a hierarchy relationship.This gives you more flexibility to centralize data without requiring direct relationships.

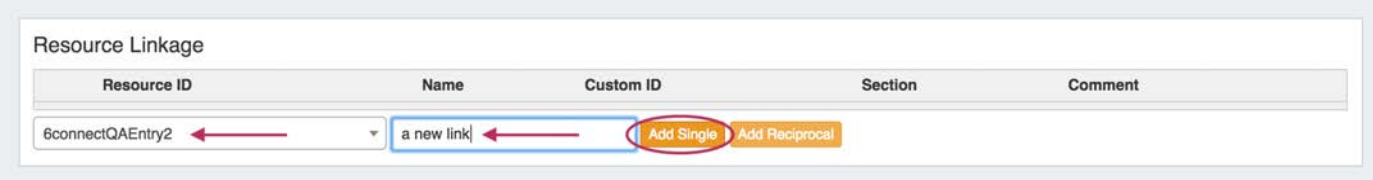
To use the Resource Linkage Gadget, first, ensure that "Resource Linkage" is enabled under "Manage Gadgets" for the Section type of the Resource being used.

At this step, an empty Resource Linkage Gadget is displayed on the Resource's Entry page.



Add a new link by clicking the "Add New" button.

Search for the desired Resource to add, select, type in notes if desired, and then click the "Add Single" or "Add Reciprocal" button. The linked Resource will show in the Resource list.



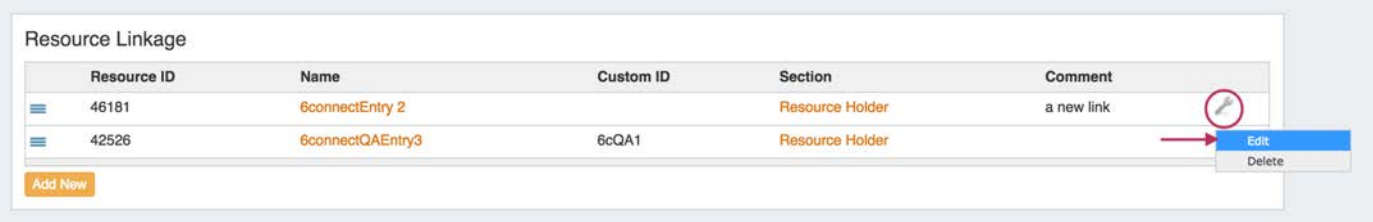
Save your changes by clicking the "Save Changes" button to finalize your selection.



Once a resource link has been saved, you may click on the link to go to the linked Resource's Entry page, or use the Action Menu (wrench icon) to Edit or Delete the link.

Edit a Resource Link

Editing a link allows for comments to be added to a link inside the Resource Linkage Gadget. Click on the link's Action Menu (wrench icon), and select "Edit" from the menu.

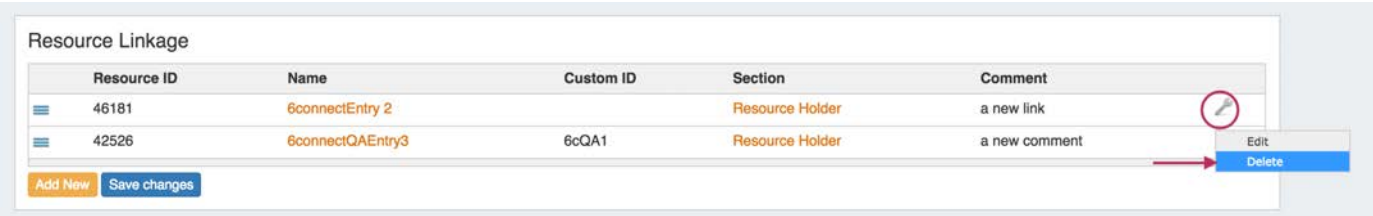


Type in the desired comment for the link, and click "Done" when complete and then hitting "Save Changes", or you may exit out by clicking "Cancel".



Delete a Resource Link

To delete a Resource Link, click the Action Menu (wrench icon) for the desired link in the list, then select "Delete" from the menu.



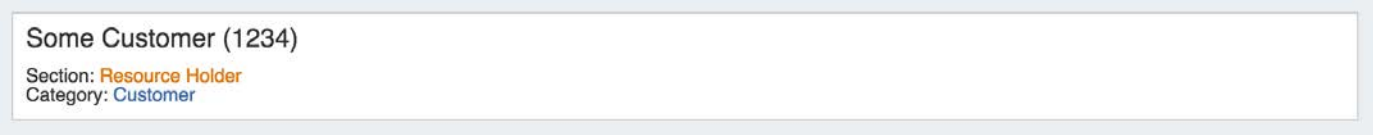
Reorder Resource Links

The Resource list in the Resource Linkage Gadget may be manually reordered. To move a link to a different position in the list, click the List icon at the beginning of the row, then click and drag to the desired position and release. After reordering, click "Save Changes" at the bottom right corner of the gadget to finalize your changes.



Resource View

This Gadget is used on the Resource Holder Section type. The Resource view displays and provides links for the Section and Category for the Resource.



Reverse API Console

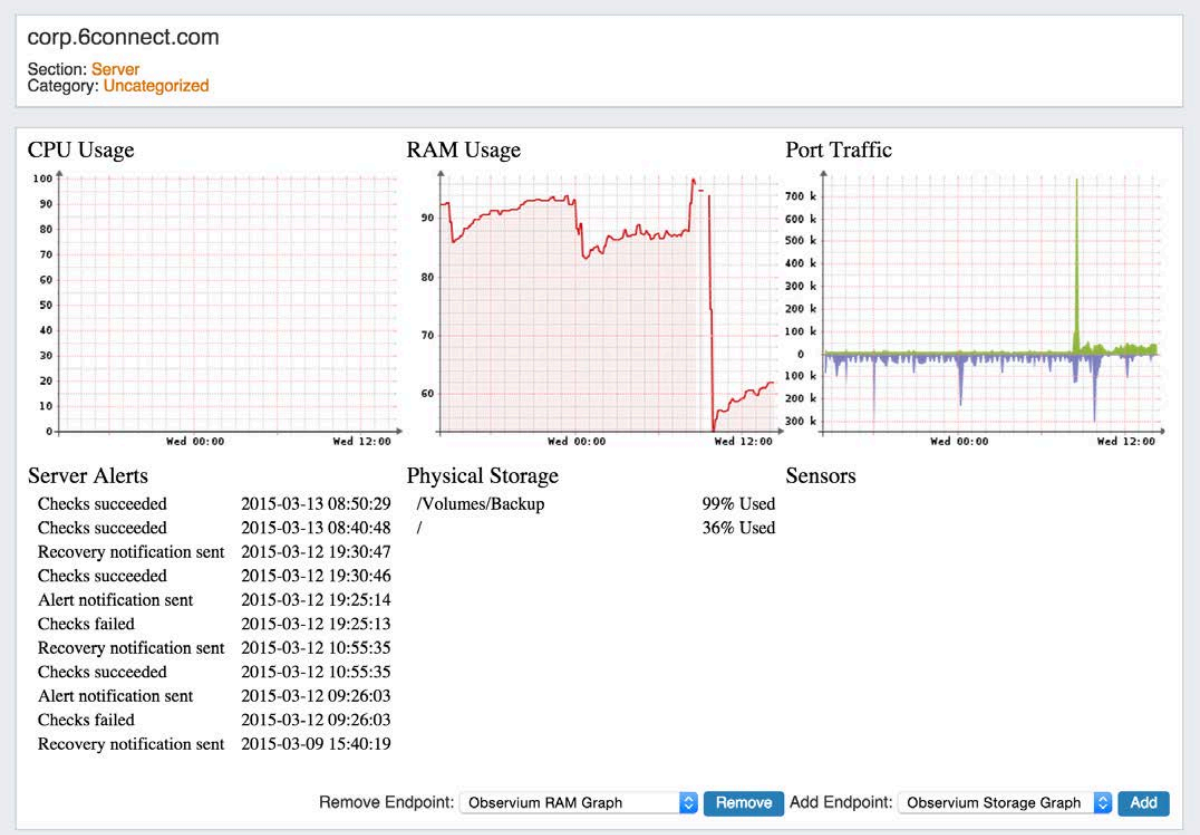
The Reverse API Console Gadget allows you to add endpoints from a Resource Entry page, and view customizable javascript displays set for the selected endpoint. The Reverse API Console may be added to any type of Section / Resource, and must be enabled for a Resource for certain rAPI processes to successfully complete.

After adding the gadget to a section, the Reverse API console will appear as whitespace with the option to add an endpoint display. Endpoints must already be created via rAPI or through the Admin Reverse API interface in ProVision to be selectable via the gadget.



Selecting an Endpoint and clicking "Add" will allow the javascript display code (added via rAPI or the Admin Reverse API interface) associated with that endpoint to display on the Resource Entry page.

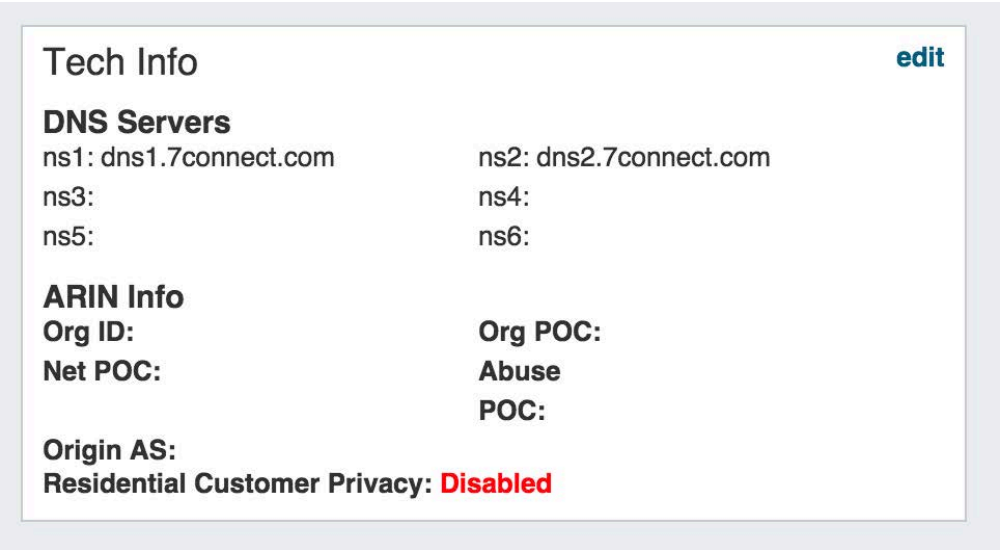
An example display might be integrating Observium to display server status modules and alerts for Resources with the "Server" Section:



As a completely customizable area, displays can be designed to meet individual Resource needs - create charts, alerts, command buttons, or any other type of data that you wish to view.

Tech Info

This visual element is used on the Resource Holder Section type. This Gadget allows you to list DNS servers, ARIN information, and enable /disable customer privacy.



Creating your own Gadgets

6connect provides XML specifications for users interested in creating their own gadgets for ProVision. See the XML Specifications section linked below for more information.

User created gadgets are not supported at this time and the specification below could change without notice. If you want to make your own gadget, please get in touch so we can help you

- XML Specifications

Additional Information

Continue on to the following pages for additional information on Working with Resources in Provision:

- Contact Manager

XML Specifications

XML Specifications

THIS IS AN EXPERIMENTAL FEATURE

User created gadgets are not supported at this time and the specification below could change without notice. If you want to make your own gadget, please get in touch so we can help you.

- XML Specifications
  - XML Specification
    - Implemented Tags
    - Example
    - Fields

XML Specification

The XML gadget specification is based on the Atlassian Gadgets.

Implemented Tags

The implemented tags and corresponding attributes are:

- ModulePrefs

  Description
  - title
  - width - "full" or "half" are the only options for now
- ContentSources
  - type - "file" uses the file given in src, "html" uses the content in the tag (eg. <Content type="html">This is the content</Content>)
  - src - relative filename or url
- Source
  Fields
  - type - "css" or "javascript"
  - src - relative filename or url
- Field
  - slug

Example

```
<?xml version="1.0" encoding="UTF-8" ?>
<Module>
    <ModulePrefs title="Contact Info" width="half" />
    <Description>This gadget adds a field editor for fields related to contact info (phone, address,
ect).</Description>
    <Content type="file" src="template.html" />
    <Sources>
       <Source type="javascript" src="script.js" />
    </Sources>
    <Fields>
       <Field slug="6c-resourceholder-phone-main" />
       <Field slug="6c-resourceholder-phone-fax" />
    </Fields>
</Module>
```

Fields

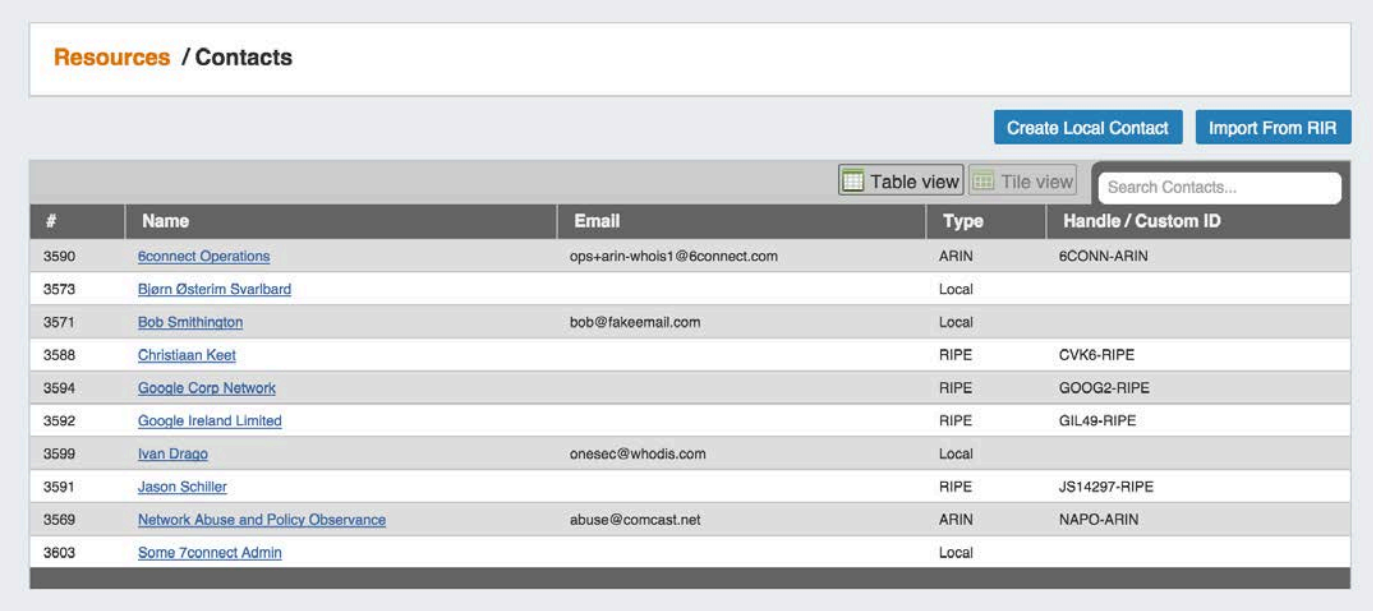
If a gadget uses fields, you can optionally add the slug of the field in this section to hide it from the main field list.

This can be very useful and make your Resource Sections easier to work with. If the fields are not hidden, this can lead to long lists of redundant data in multiple places and can cause confusion. However, all viewing and editing of the field will have to be done through the gadget. If your gadget uses a field in a read-only manner, then you should **not** add it to the gadget's manifest because that would prevent users from editing the field data through the standard edit page.

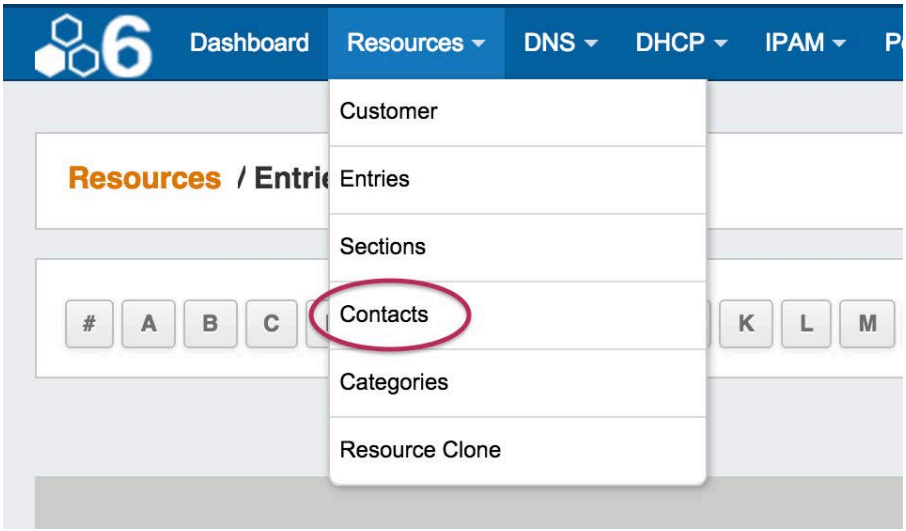
Contact Manager

Contact Manager

The Contact Manager lists and imports contacts into ProVision. You can create new contacts manually, or import contacts from RIR. Contacts may then be assigned to ProVision resources through the Contacts Gadget.



The Contact Manager is accessed from the Resource tab dropdown menu, under "Contacts"



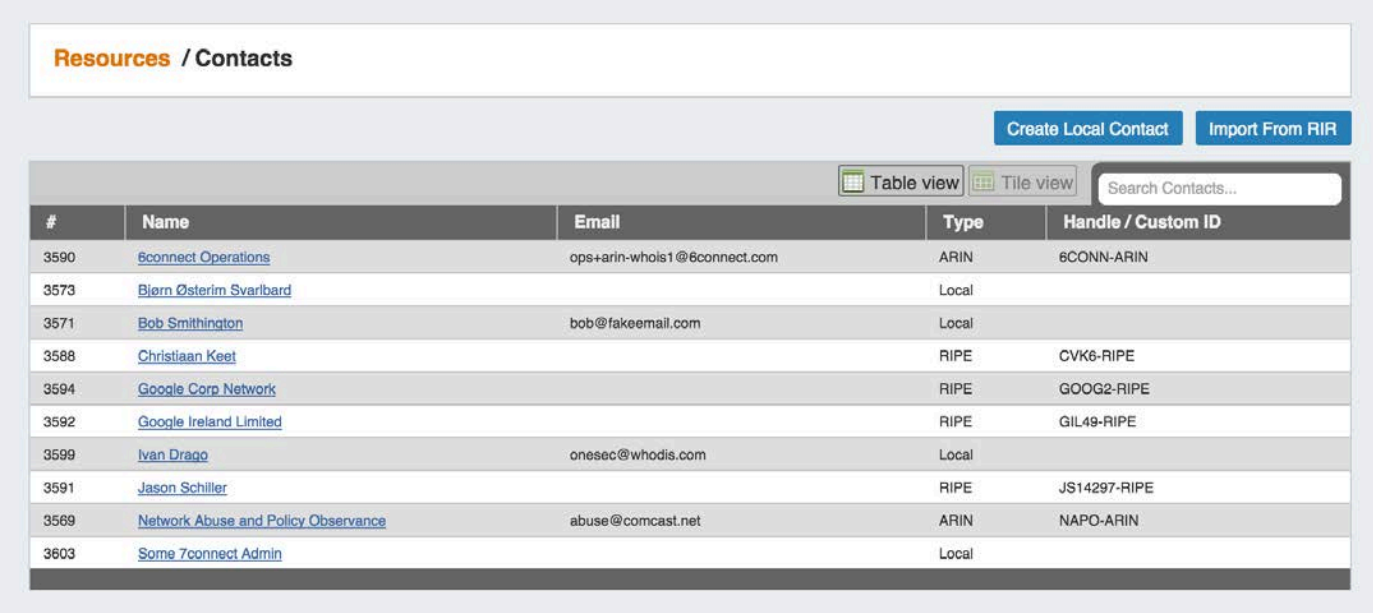
- Contact Manager
  - UI Overview
    - Table View
    - Tile View
    - View Contact Details
  - Create New Contact
  - Import Contact from RIR
    - 1) Select RIR / Type
    - 2) Enter Point of Contact Handle
    - 3) Verify and Add
  - Edit or Delete Contact
- Working with the Contacts Gadget

- Assigning a Contact to a Resource
  - Assign the Contact
  - Create a Contact Role
  - Assign a Role to a Contact
  - Delete a Role from the Roles List
- Unassign a Contact
- View the Contact Details Page

UI Overview

Table View

Table View shows contacts in list form, their ID number, email, contact type, and their handle /custom ID. If the contact was manually created through the "Create New Contact" button, it will show as type "Local", whereas a contact imported from RIR will show of type "RIPE" or "ARIN".



Tile View

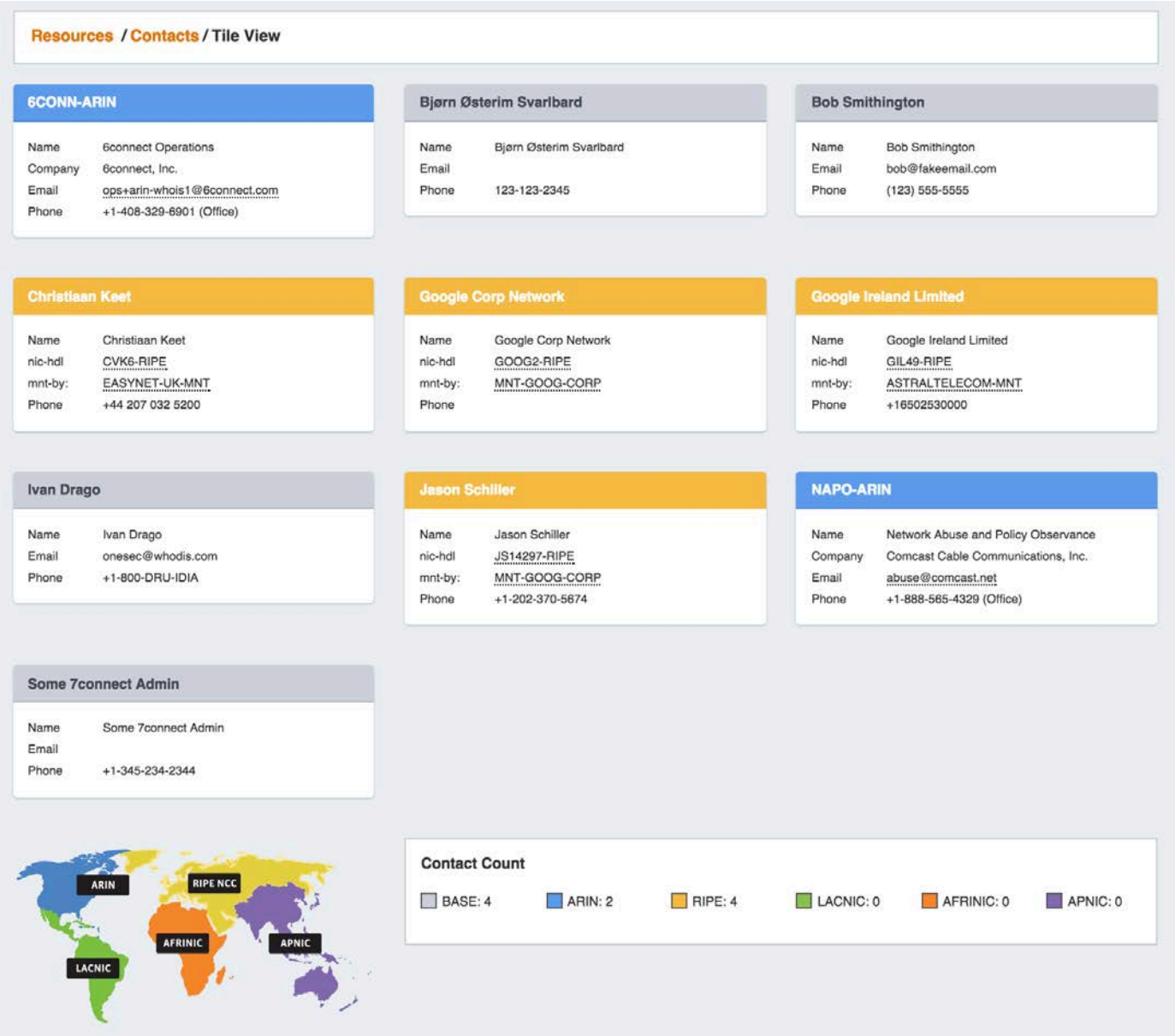
Tile view allows you to see all current contacts as a snapshot of basic information, based on contact type:

Base Contacts: Shows Contact Name, Email, and Phone

ARIN Contacts: Shows Contact Name, Company, Email, and Phone (if provided in ARIN)

RIPE Contacts: Shows Contact Name, nic-handle, maintained by, and phone (if provided in RIPE).

Clicking on the contact name on the header of the tile takes you to the contact's detail information page.



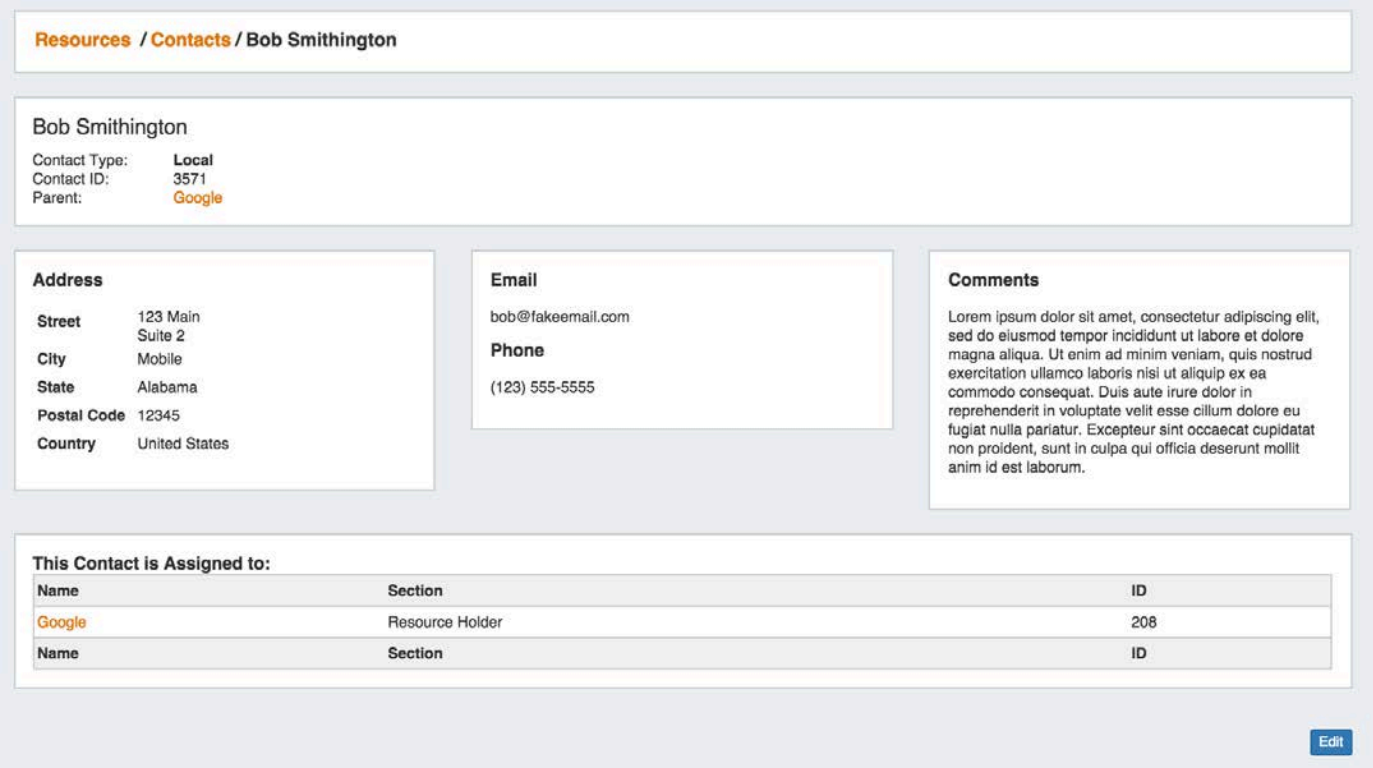
At the bottom of the Tile View page, counts of current contacts by RIR as well as an image map are provided. At this time, only Base (Local), ARIN, and RIPE contacts are supported.

View Contact Details

To view the contact details, click on the contact's name in either list or tile view. You will be taken to the contact details page.

A Local contact created in ProVision will have modules shown for Address, Email/Phone, and Comments, as well as a list of any resources that contact is assigned to via the Contacts gadget.

You may edit or delete a Local contact by clicking the "Edit" button at the bottom of the page.



A contact imported from a RIR will have its details shown as determined by the imported RIR module - this data is not editable in ProVision.

You may delete the contact by clicking the "Delete" button at the bottom of the page.

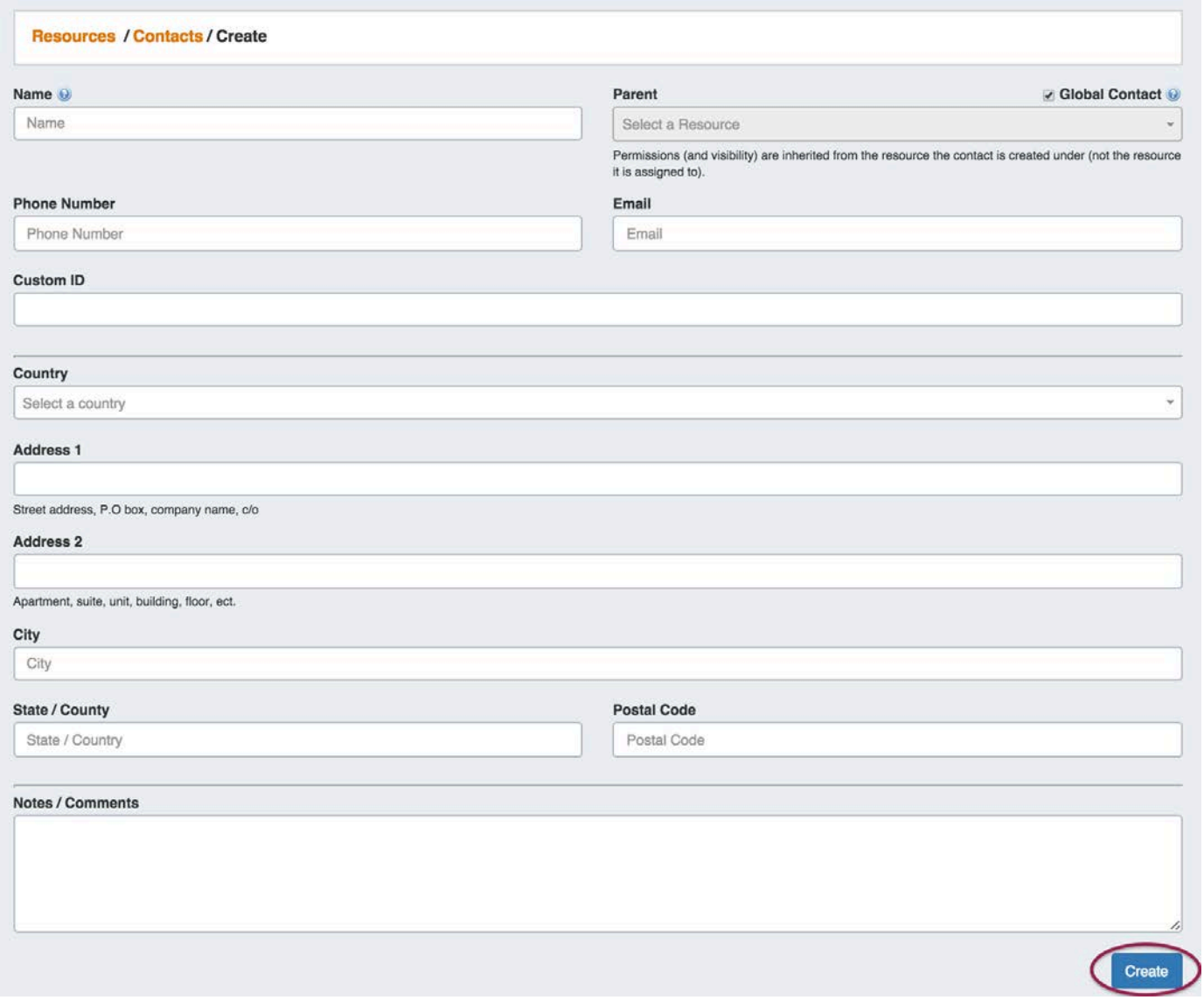


Create New Contact

You can manually create a new contact in ProVision by clicking on "Create Local Contact" under the contacts page.



Fill in the contact's information for Name, Parent Resource (if desired), Phone, Custom ID, Email, Address, and Comments if desired.



When done filling out the information fields, hit the "Create" button to save your contact.

Parent

The parent field relates the contact to a resource's permissions structure, only allowing users with permissions for that resource to view and assign the affiliated contact(s). To allow the contact to be viewed globally in ProVision, check the "Global Contact" box above the Parent Field.

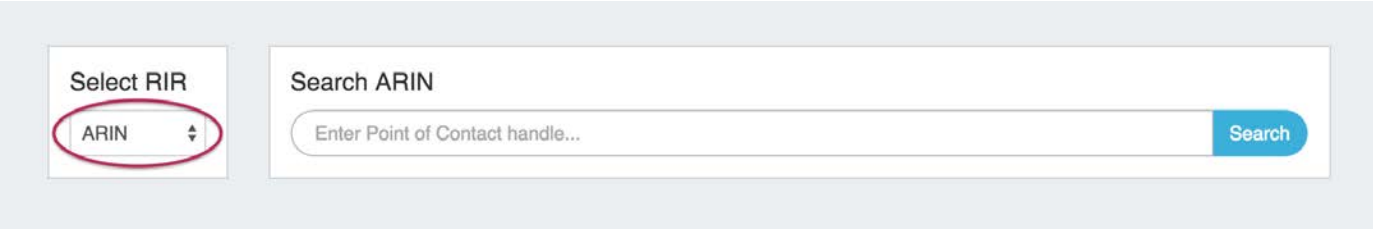
Import Contact from RIR

To import a contact from ARIN or RIPE, click the "Import from RIR" button from the contacts page.



1) Select RIR / Type

Once on the the Import from RIR page, on the left hand side of the page select the RIR from which you wish to import - ARIN or RIPE.



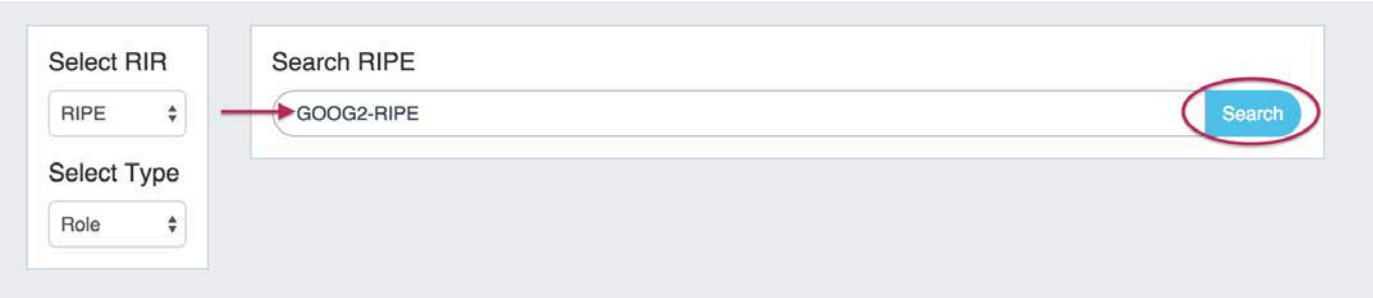
If you select RIPE, you will also be prompted to choose whether you are importing a "Person" or "Role".



2) Enter Point of Contact Handle

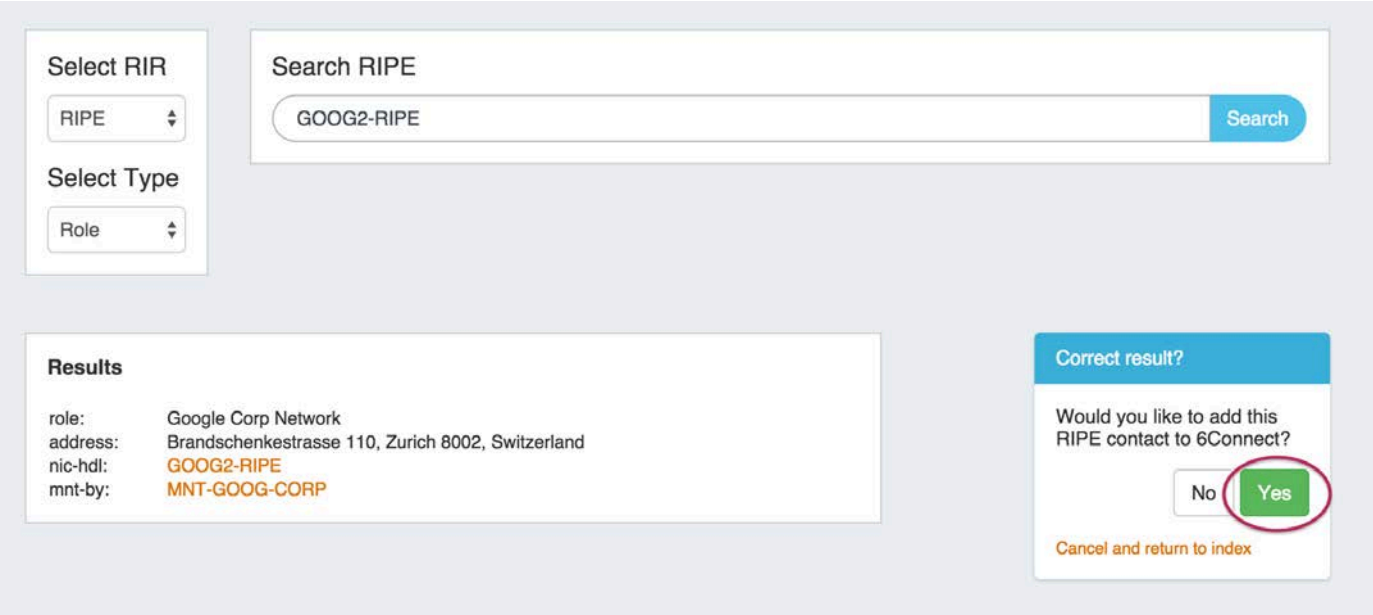
After selecting the RIR / Type (if applicable), enter the ARIN or RIPE Point of Contact Handle for the contact you are importing, then click "Search".

Point of Contact Handle is the only search term that is valid. Attempting to search by name or keyword is not valid, and will return an error.



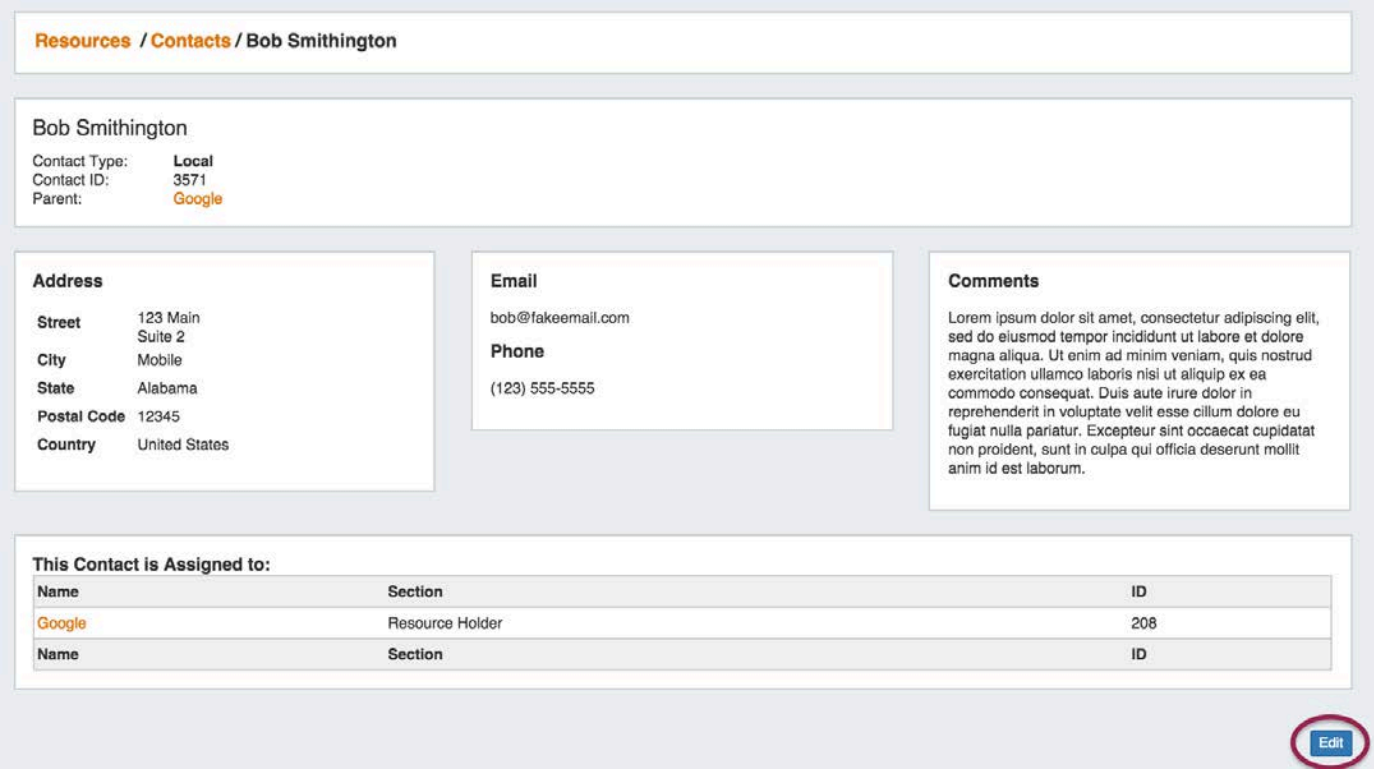
3) Verify and Add

If the search is successful, the found contact information will display at the bottom of the screen. If this is the information you wish to add into ProVision, click the "Yes" button on the right hand side of the page under "Would you like to add this RIPE contact to 6connect?". Otherwise, you may hit "No" to decline adding, or cancel and return to the index.

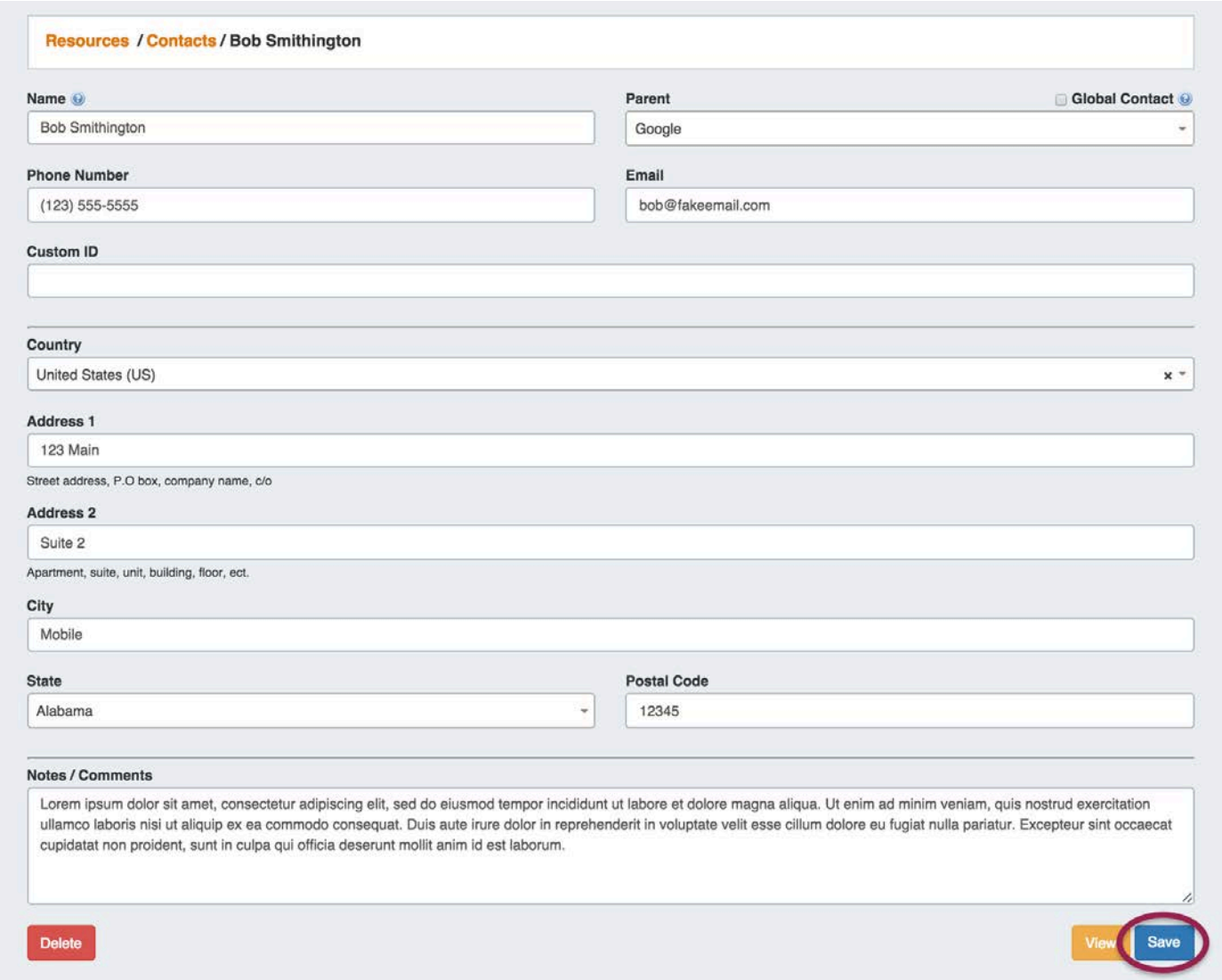


Edit or Delete Contact

To Edit or Delete a base contact, click on the contact's name in the contact list to go to its detail view page. From there, click "Edit".



This will take you to the detail information page, where you may edit fields as desired. When done editing, hit "Save" to save your changes, or "View" to cancel without saving and view the contacts list.



To delete the contact, click the "Delete" button while in the contact details page.



Working with the Contacts Gadget

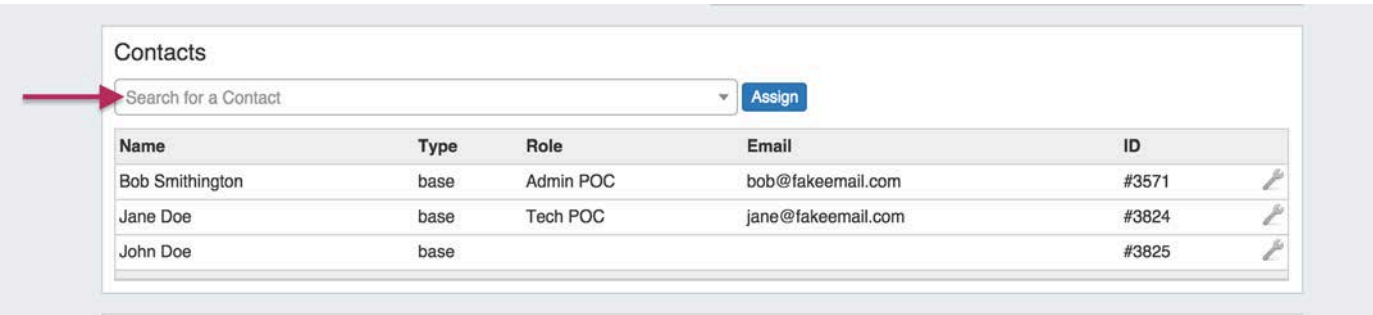
Assigning a Contact to a Resource

Contacts may be assigned to a resource through the Contacts gadget. To add a gadget to a Resource's Entry page, see Customizing Sections and Gadgets.

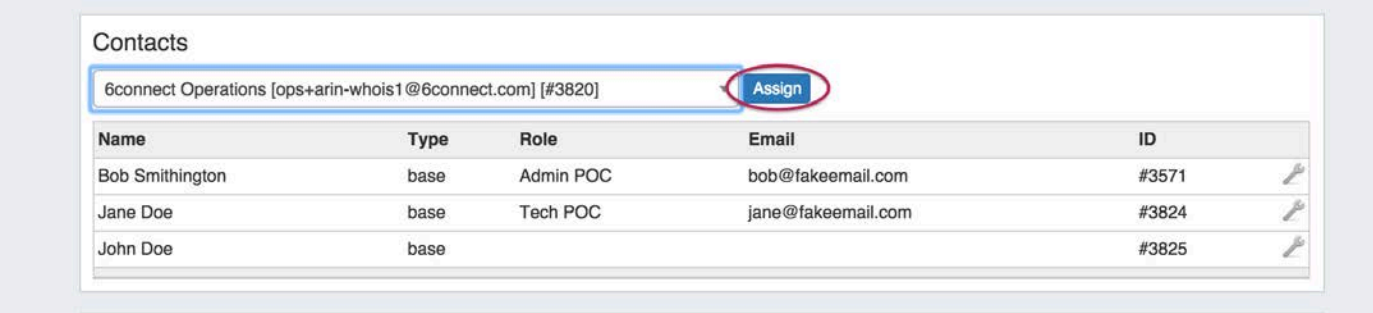
Once the Contacts gadget has been added to the desired Section, go to the Resource Entry page for the Resource you wish to assign the contact to, and navigate to the Contacts gadget.

Assign the Contact

In the Contact Gadget, search for a contact by typing the first few letters of the contact name in the search box, then click on the desired contact.



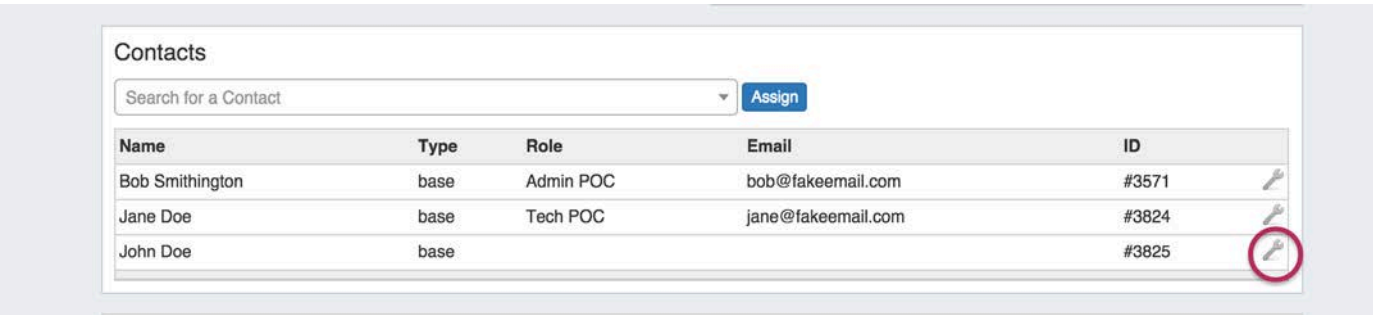
Then hit "Assign".



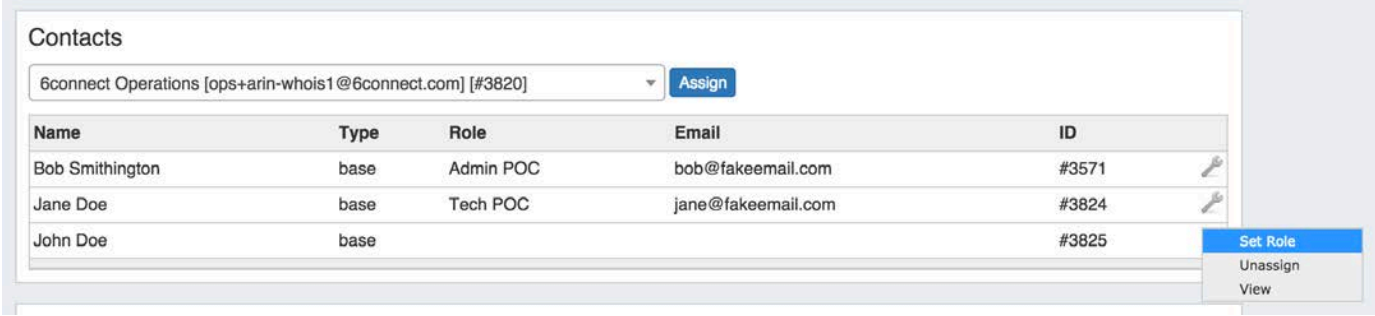
The Contact will show in the list below.

Create a Contact Role

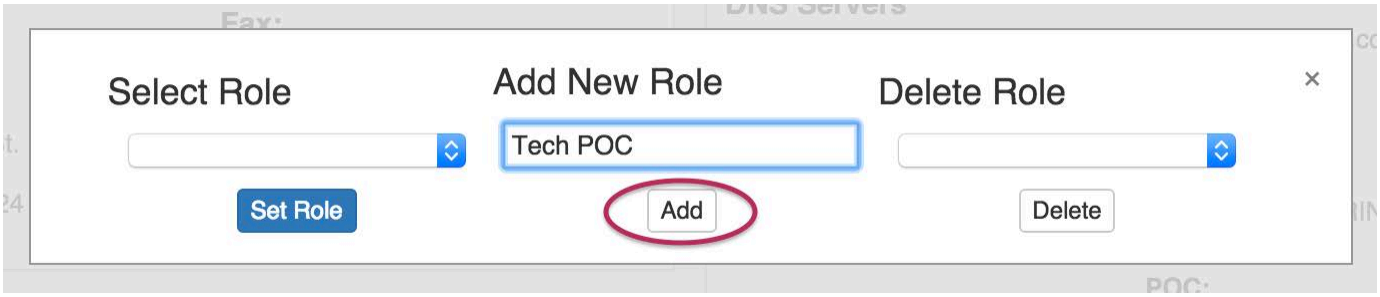
Once a contact has been assigned, you can create and set a role for the contact by clicking on the Action Menu (wrench icon).



Select "Set Role"

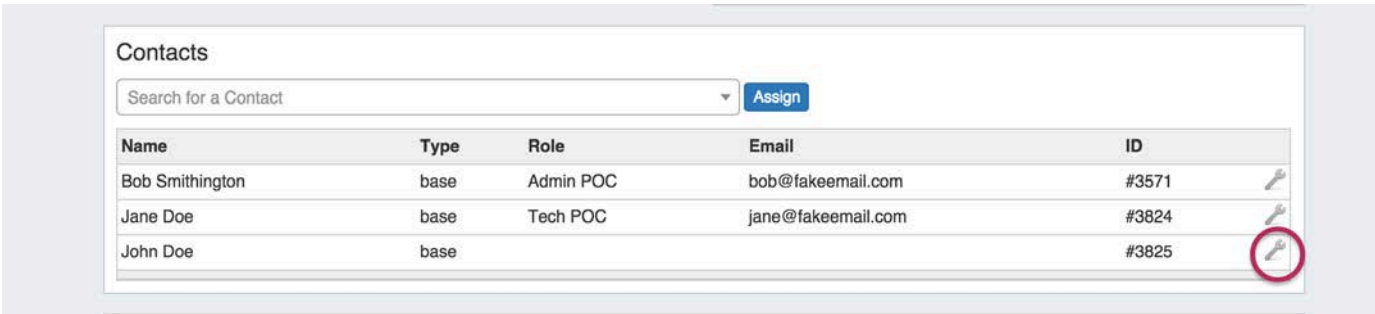


Under the "Add New Role" section, type in the desired tole name and click the "Add" button.

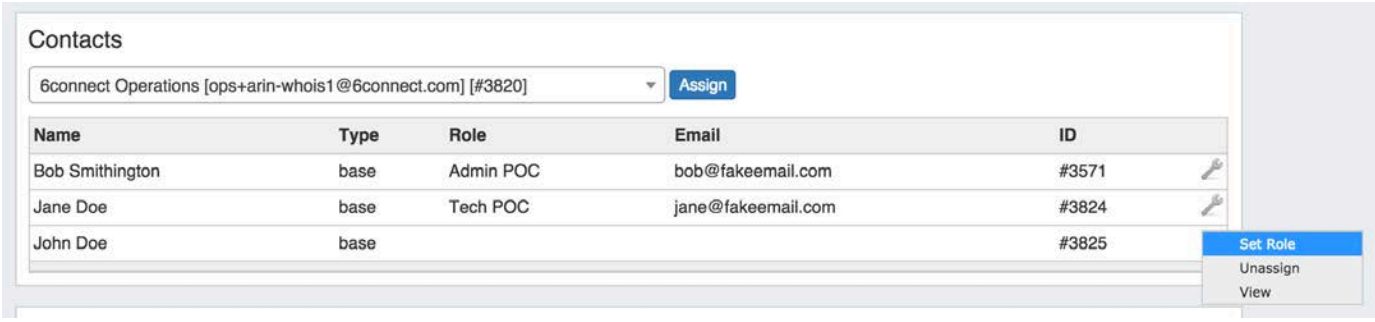


Assign a Role to a Contact

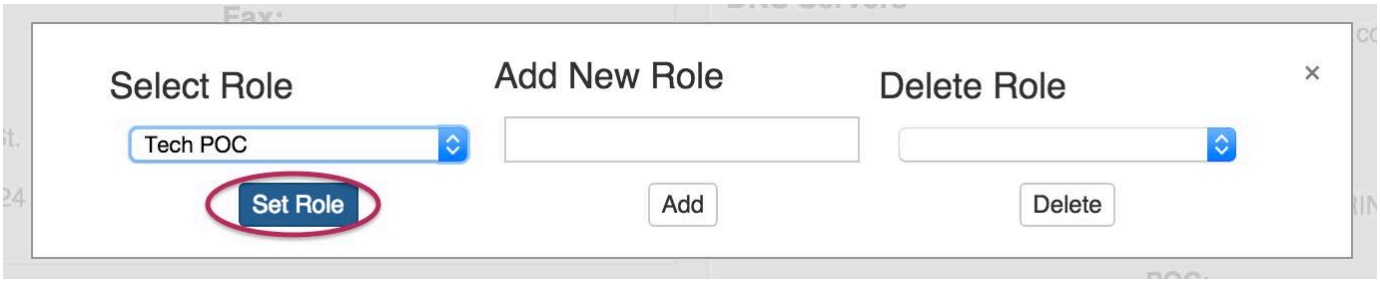
Once a contact has been assigned, you can set a role for the contact by clicking on the Action Menu (wrench icon).



Select "Set Role"

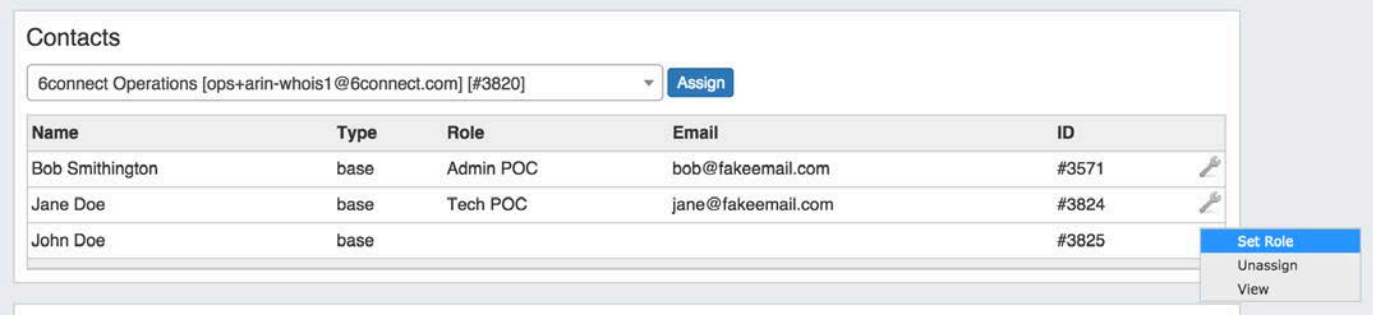


Under the "Select Role" section, choose the desired Role from the dropdown list, and click the "Set Role" button.

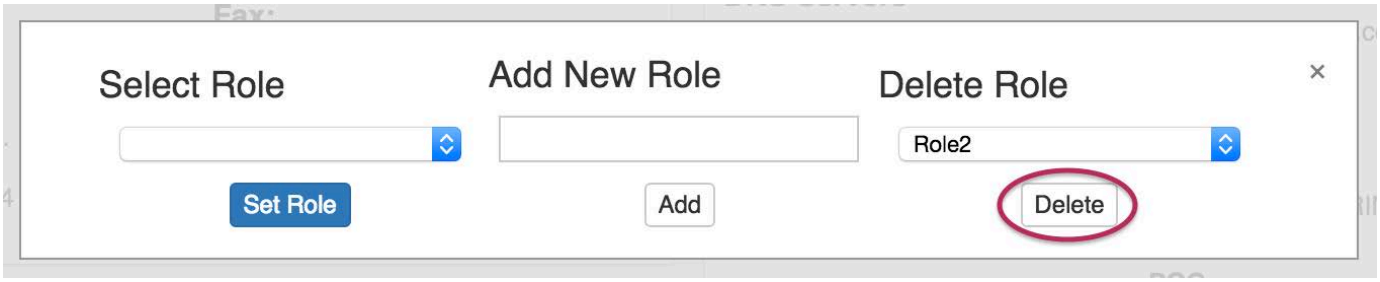


Delete a Role from the Roles List

Select "Set Role"



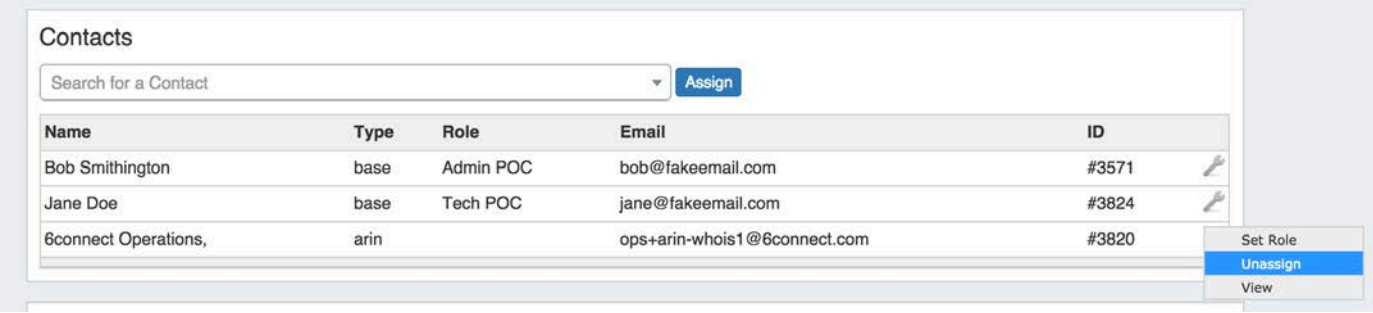
Under the "Delete Role" section, choose the desired Role from the dropdown list, and click the "Delete" button.



This removes the role from the "Select Role" list.

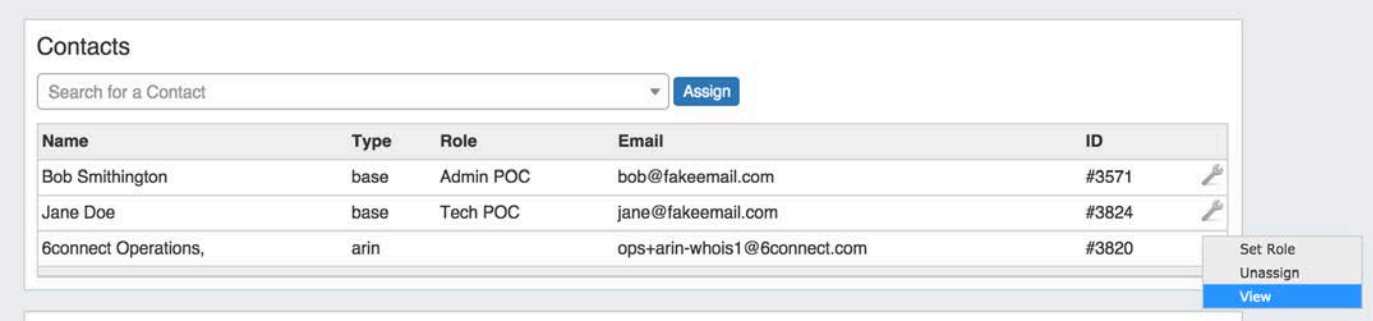
Unassign a Contact

To unassign a contact from a resource, in the Contacts gadget click the Action Menu, then select "Unassign".



View the Contact Details Page

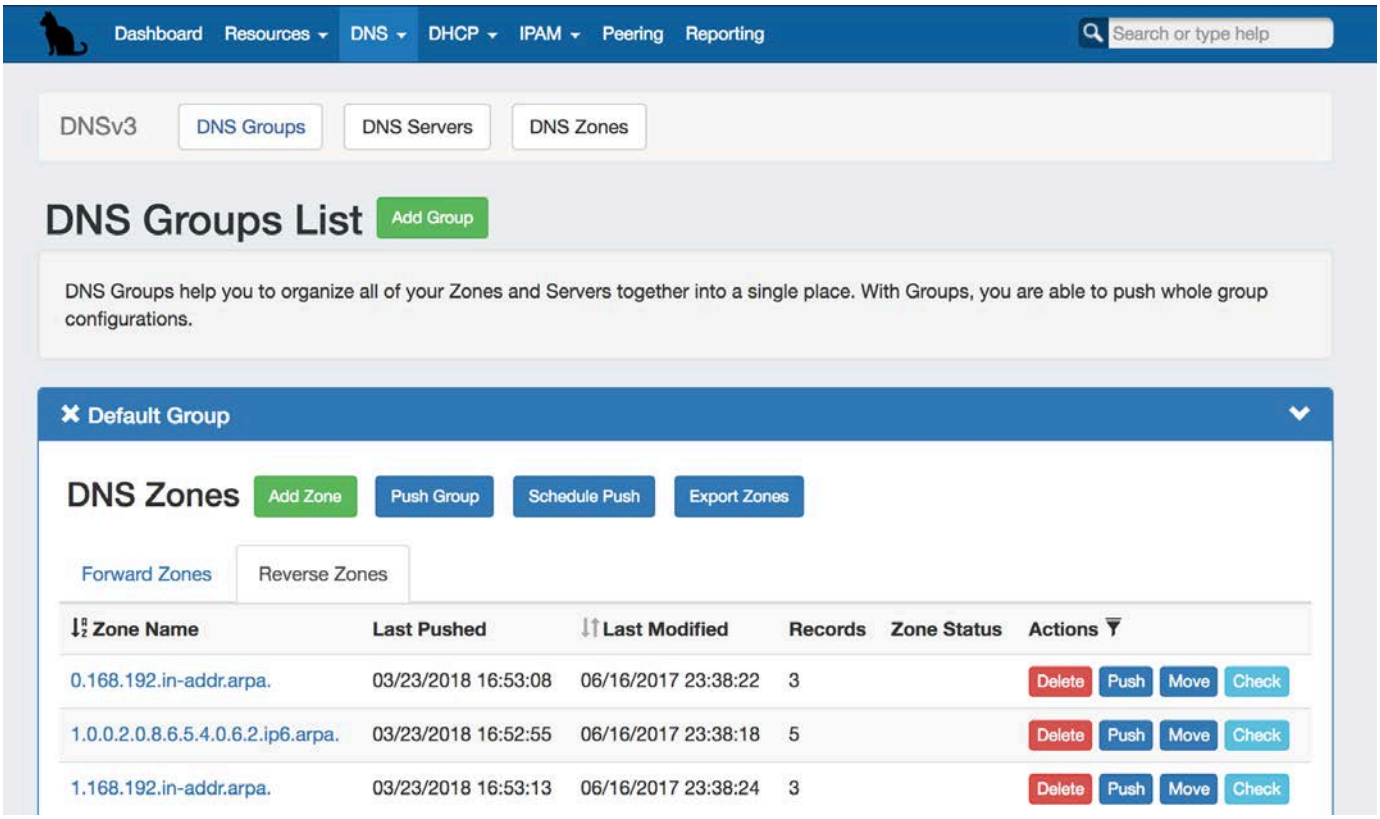
To view the contact's details, click the Action Menu, then select "View". You will be redirected to the contact's detail page.


Contacts
Search for a Contact
Assign
Name
Type
Role
Email
ID
Bob Smithington
base
Admin POC
bob@fakeemail.com
#3571
Jane Doe
base
Tech POC
jane@fakeemail.com
#3824
6connect Operations,
arin
ops+arin-whois1@6connect.com
#3820
Set Role
Unassign
View

DNS Tab

DNSv3

ProVision's DNSv3 combines server management, group organization, and zone management under the DNS tab.



The DNS tab contains three sub-tabs: **DNS Groups, DNS Servers, and DNS Zones**.

The **DNS Groups** tab is where DNS Groups, zones, ACL's, and default SOA values are managed. The DNS tab **Groups List** allows you to create, view, and manage groups of DNS Servers and Zones. Using groups, you can configure and push selected combinations of zones at one time.

The **DNS Servers** tab is only accessible to Admin users, and contains functions for adding, updating, and managing DNS servers as well as scheduling server tasks. For Admin-level DNS tasks, see DNS Administration and Working with DNS Servers.

The **DNS Zones** tab lists all the zones in ProVision able to be viewed by the user, in the same format as on the **DNS Groups** page, but without the Group organization. The Zone List allows you to view, add, delete, and push individual DNS Zones.

- DNSv3
  - DNSv3 Overview
    - DNSv3 Permissions
  - DNSv3 Workflow
  - Additional Information

DNSv3 Overview

DNSv3 reorganizes ProVision's DNS system into a more unified and accessible interface, combining both admin and non-admin DNS tasks together under the DNS tab.

In DNSv3, zones are gathered under DNS Groups, servers are attached to those Groups, and Nameservers, Default SOA values, and ACLs are managed at a per-Group level. Users can then view and manage Groups, individual zones, default SOA values, ACLs, attach servers, and perform pushes all on the same page.

DNSv3 Permissions

DNSv3 (6.0.0 release), restructures DNS to incorporate DNS zones and Groups into ProVision's Resource System. Zones and Groups are Resources just like Customers, Servers, Routers, or Contacts (See Resource Concepts for a more detailed explanation of Resources in ProVision).

This allows for DNS zones and Group permissions to be managed similarly to other ProVision resources, where users with Resource permissions (Create / Read / Update / Delete) on the parent resource of the DNS Group can create groups and zones, manage those groups and zones, push (if a server is attached), and delete.

A user with full Resource permissions on a DNS Server, as well as the parent resource of a Group, may view and attach that server to a Group.

Users with Admin permissions can access the **DNS Servers** area under the **DNS** tab manage DNS server creation, edits, and deletion.

For more information on setting up permissions groups in ProVision, see Users & Permissions.

DNSv3 Workflow

DNSv3 revolves around Groups. Zones are gathered under Groups, servers attached to Groups, and pushes may be done on a per Group level. Thus, the first workflow step in DNSv3 is to set up one or more DNS Groups. A "Default Group" is automatically provided in ProVision, but other Groups may be desired to organize zones and default values.

To create a new DNS Group, click the "Add Group" button from the **DNS Groups** tab. Enter the desired default values for the Group, and save. If only using the Default Group, ensure the default parameter values are set as needed. For more information, see Working with DNS Groups.

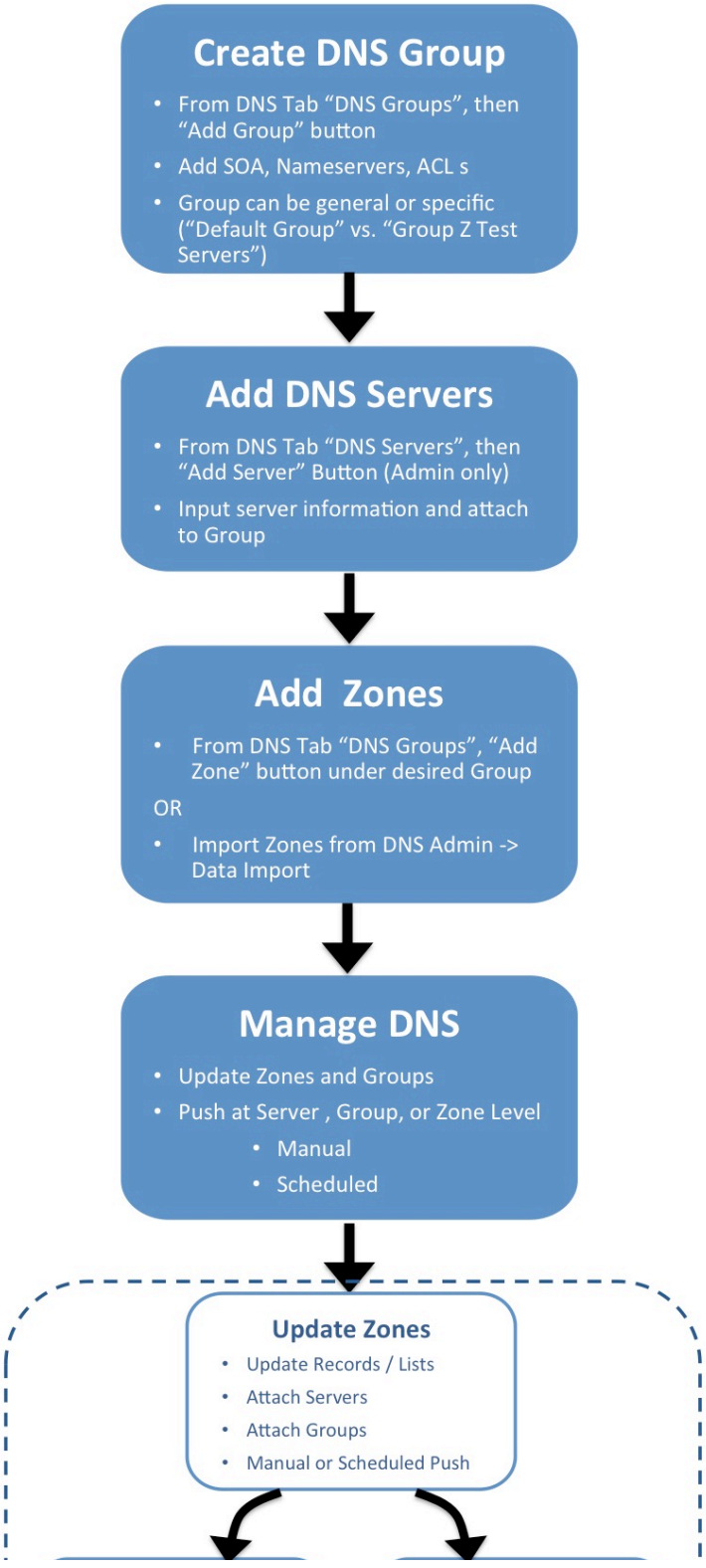
After Groups have been set up, DNS servers should be added or settings verified. Admin users may add DNS servers from the **DNS Servers** tab "Add Server" button. Input the server information and save. Existing servers may be reviewed and edited by clicking on the server name in the DNS Server List. Once a server is created in ProVision, it may be attached to any DNS Group under the Group's "Attached Servers" module. Attaching a server to a Group will allow for zones in that group to be pushed to the attached server(s). See Working with DNS Servers.

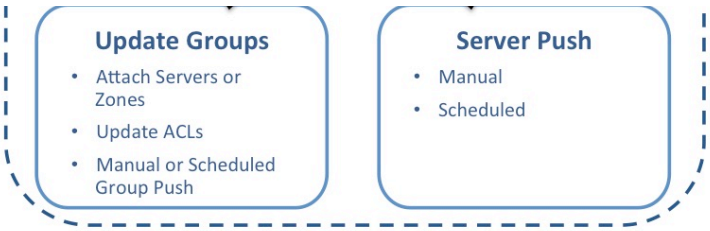
Next, add zones to your groups. Zones may be manually added under each group by clicking the "Add Zone" button in the DNS Groups or DNS Zones tabs, or it may be imported via DNS Importers into a selected Group. Add the zone and record information, and save. See Working with DNS Zones - Common Tasks for additional information.

Zones may be only exist once per Group, but may be duplicated under multiple Groups. Zones may also be moved from Group to Group as needed.

At this point, all major components of the ProVision DNS system have been added - from here management tasks take over. Zones may be updated and moved to or from Groups; Groups may be edited with different default values or servers, and pushes maybe be performed for an individual zone, a full Group, or for an entire server. Pushes may be manual or scheduled for a future time through the Scheduler.

DNS WORKFLOW



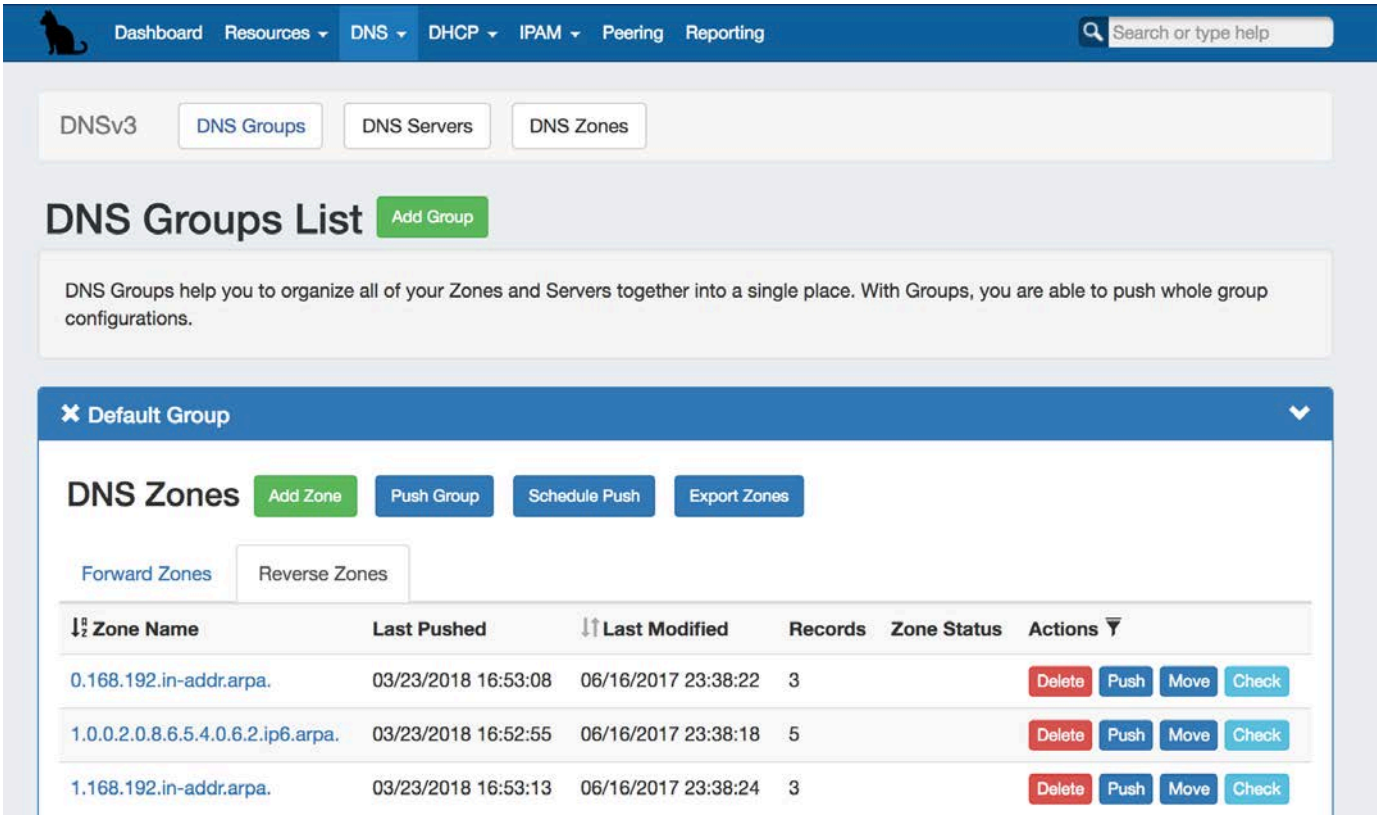


Additional Information

- Working with DNS Groups
- Working with DNS Zones - Common Tasks
- Editing DNS Zones

Working with DNS Groups

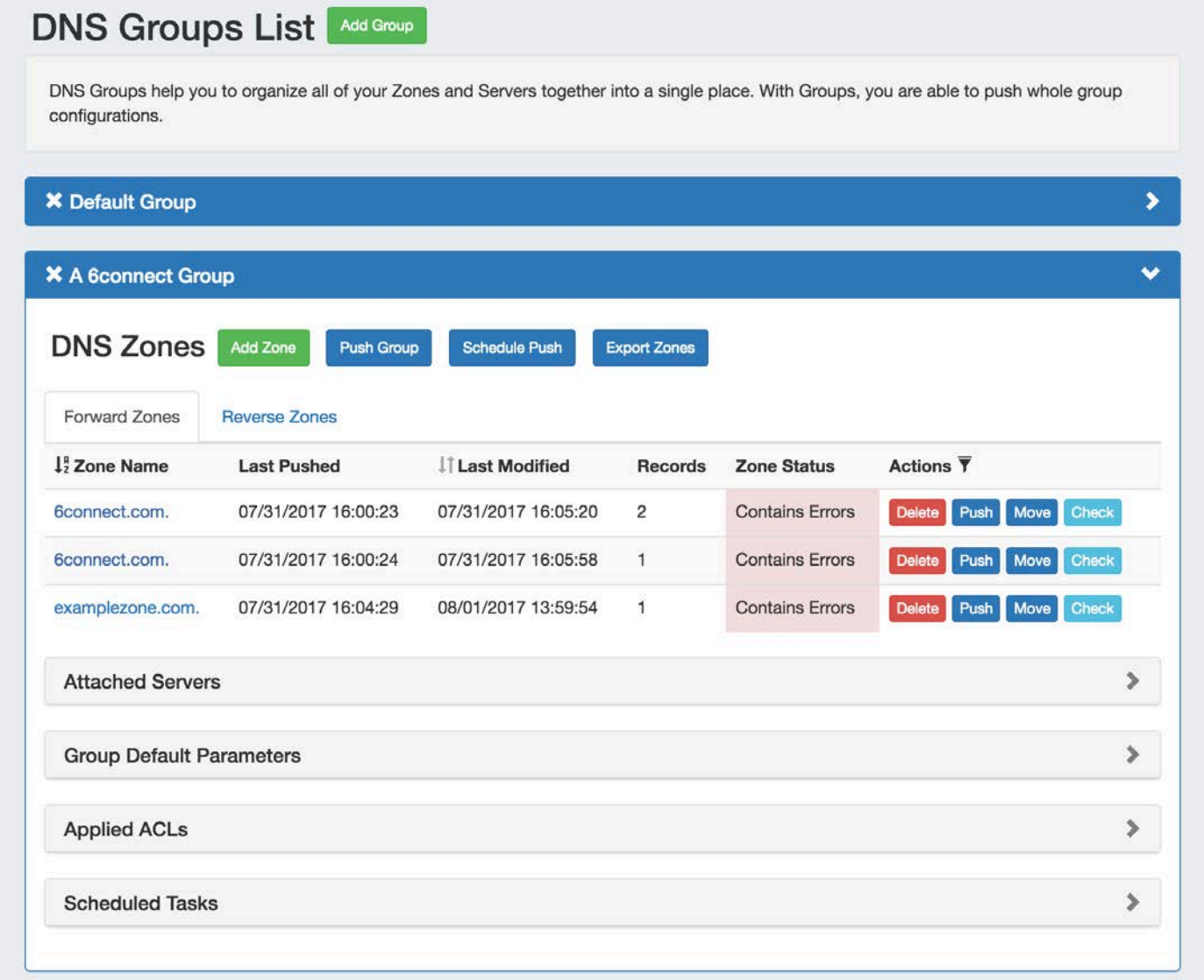
DNS Groups



The DNS tab **Groups List** allows you to create, view, and manage groups of DNS Servers and Zones. Using groups, you can configure and push selected combinations of zones at one time.

- DNS Groups
  - DNS Groups List Interface
- Working with DNS Groups
  - Add a Group
  - Edit a Group
    - Edit a Group Name:
    - Edit a Group's Default Parameter Values:
      - Edit Nameservers
      - Edit Push Policy
      - Edit SOA Values
      - Save Changes
  - Add Zones to a Group
  - Pushing a Group
    - Manual Group Push
    - Scheduled Group Push
  - Delete a Group
  - Export a Group
  - Using the Templates Group / Cloning Zones to a Group
    - Create your template zone
    - Clone Template to a Different Group
  - Additional Information

DNS Groups List Interface



1) Add Group Button: Opens a dialog for creating a DNS Group

2) Group Container: The Group container header shows the group name. Expand or close the group container by clicking on the expansion arrow (>) on the far right of the header. Clicking on the X will delete the Group and all zones included in the group. Rename the group by clicking on the name, then typing the desired changes and clicking outside of the name.

Group Zone Actions:

3) Add Zone Button: Opens a dialog for creating a DNS zone under that group.

4) Push Group Button: Pushes all zones in the group.

5) Schedule Push Button: Schedules a push of all zones in the group to the selected day and time. Requires the scheduler tab / cron tasks to be enabled.

6) Export Zones: Exports all zones in the group into a .zip file.

7) Zone List: Forward and reverse zones included in the Group.

8) Forward / Reverse Zones Tabs: Shows zone list containing only forward or reverse zones, respectively.

9) Zone Name (Sortable): The name of the zone. Clicking on the zone name will open the zone details window, showing individual records. Clicking on the "Zone Name" header will sort the zone list by ascending or descending by name.

10) Last Pushed: The date and time the zone was last pushed.

11) Last Modified (Sortable). The date and time the zone was last modified. Clicking on the "Last Modified" header will sort the zone list by most / least recent modification date.

12) Records: How many records are in the zone.

13) Zone Status: Will notify if the zone or server connection contains errors, after a check has been performed (by clicking on the "Check" button). Clicking on the "Contains Errors" message will provide additional details on the error.

14) Actions: The actions that may be performed on each zone:



15) Delete: Deletes the zone from ProVision and removes the entry in ProVision conf file on the remote server(s) (the user will also receive a prompt to confirm they wish to complete the action)

16) Push: Pushes the zone to the associated server.

17) Move: Moves the zone to a different Group.

18) Check: Performs a check on the zone, notifying the user if errors or warnings are present for the zone.

19) Attached Servers: Click on the header title or > to expand or collapse. View, attach, or detach a server from the group.

20) Group Default Parameters: Click on the header title or > to expand or collapse. Edit default group settings including nameservers, SOA values, and Push Policy.

21) Applied ACLs: Click on the header title or > to expand or collapse. Add or detach ACLs to the group.

22) Scheduled Tasks: If a scheduled push has been set up for the group, the task information will show in a "Scheduled Tasks" module. Click on the header title or > to expand or collapse. Review or delete scheduled group pushes.

Working with DNS Groups

DNS Groups allow for organizing multiple zones together and linking those zones with one or more servers. Default nameservers, SOA values, and ACL's are set at the Group level, so different Groups may have their own default values. Zones may be pushed individually, as a Group, or at the server level.

ProVision automatically designates a Default Group for zones to be imported or added to, as well as a "templates" group for holding pre-DNSv3 templates or to hold new zones created as templates. Creating additional Groups is completely optional.

Add a Group

DNSv3 automatically contains two Groups - one Default Group to contain existing or imported zones, and a Templates Group, containing zones mimicking DNS Templates. Additional Groups may be added or removed at any time, and zones may be freely moved or duplicated between different groups as needed.

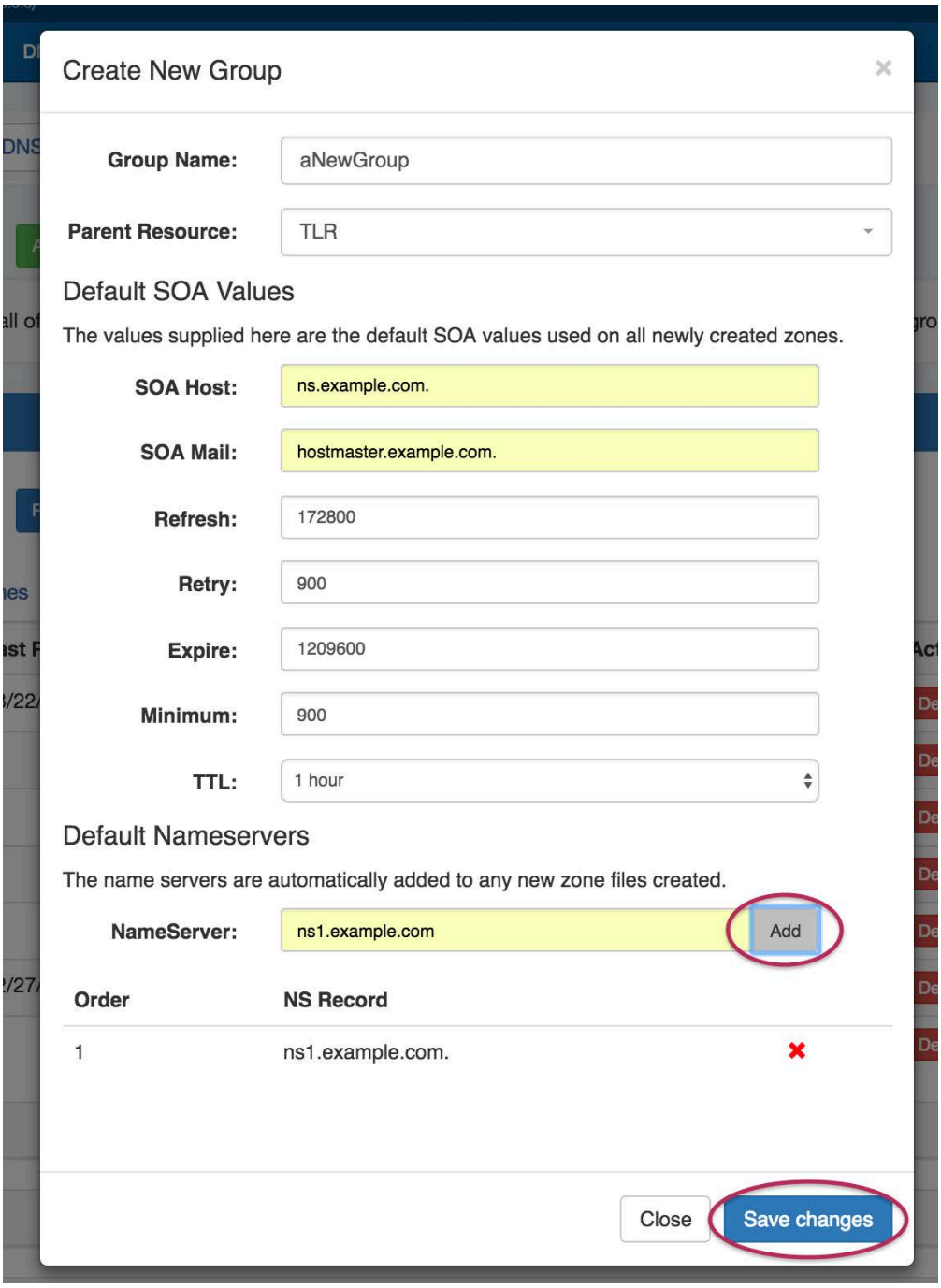
To create a new Group, from the DNS Tab, select the **DNS Groups** sub menu. Then, click the "Add Group" button next to "DNS Groups List".



This will open the "Create New Group" dialog.

Enter the desired Group Name, select a parent resource, enter the default SOA values, and add a default Nameserver (type the NS, then click "Add").

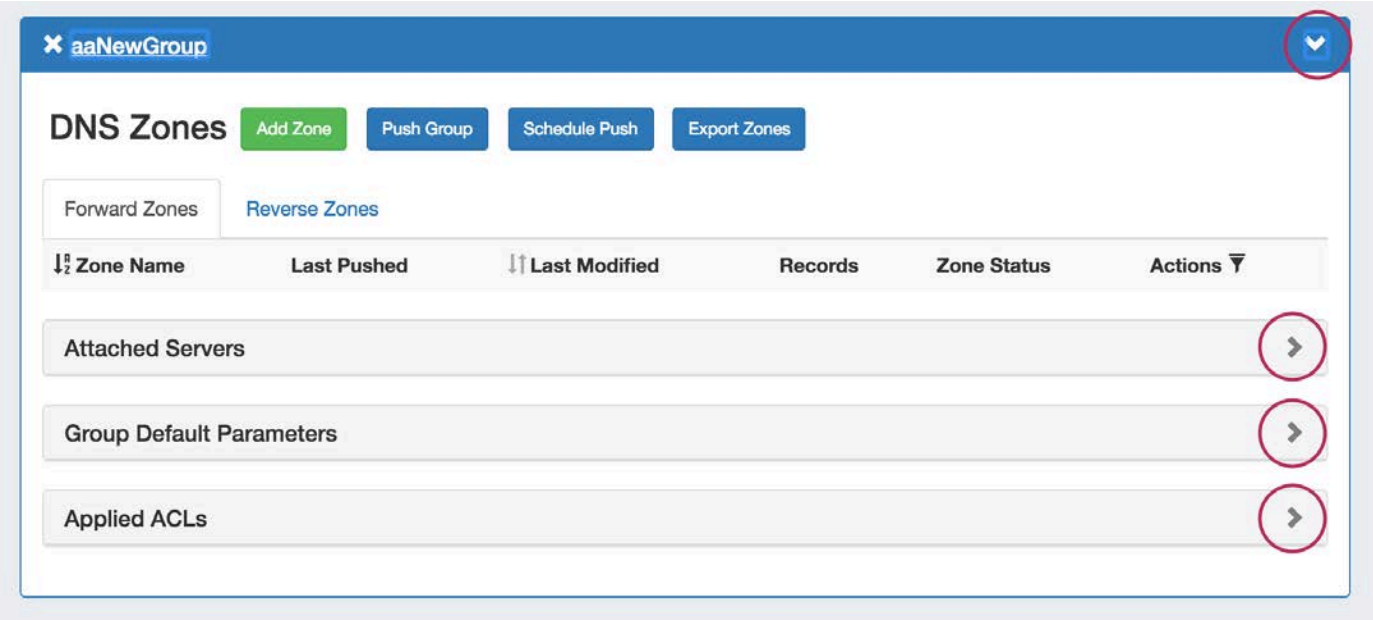
Click the "Save Changes" button when complete.



The new group will be added to the DNS Groups List.

Open the Group container for the new Group by clicking on the expansion arrow at the top right corner of the header.

Information for Attached Servers, Group Default Parameters, and Applied ACLs may also be viewed and edited by clicking on the expansion arrows for each section.



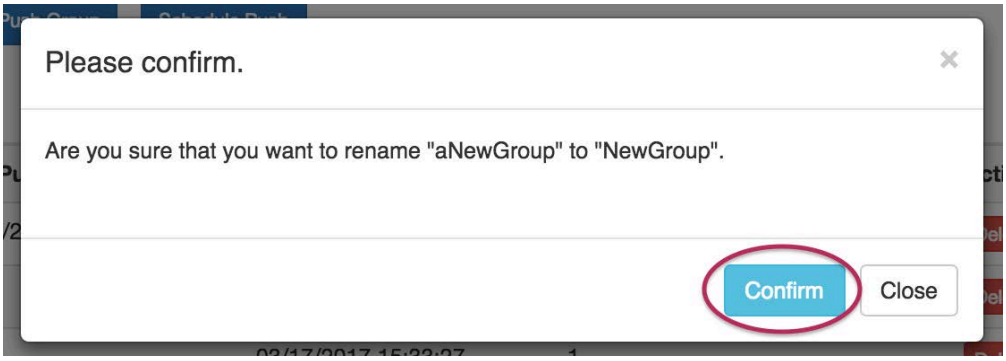
Edit a Group

Edit a Group Name:

Edit the name for a Group by clicking inside the Group's Name. A highlight box will appear around the name, and a cursor inside the box.



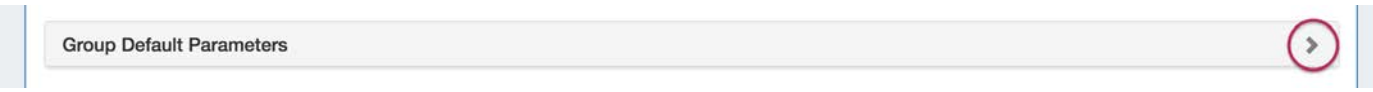
Edit the text as desired using your mouse or arrow keys to navigate through the text. When complete, click anywhere outside of the name box, and a confirmation dialog will appear.



Click "Confirm" to save your changes, or "Close" to exit without saving.

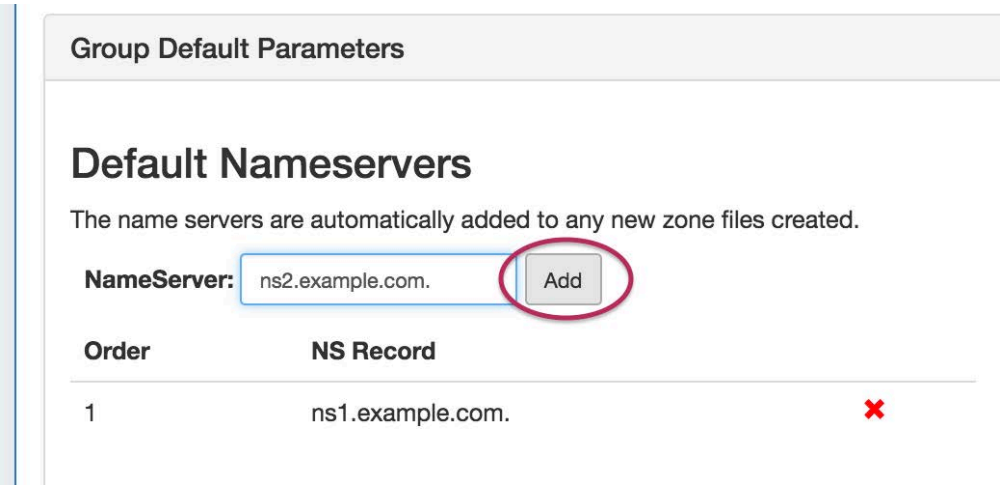
Edit a Group's Default Parameter Values:

Under the desired Group, expand the "Group Default Parameters" module by clicking on the expansion arrow.

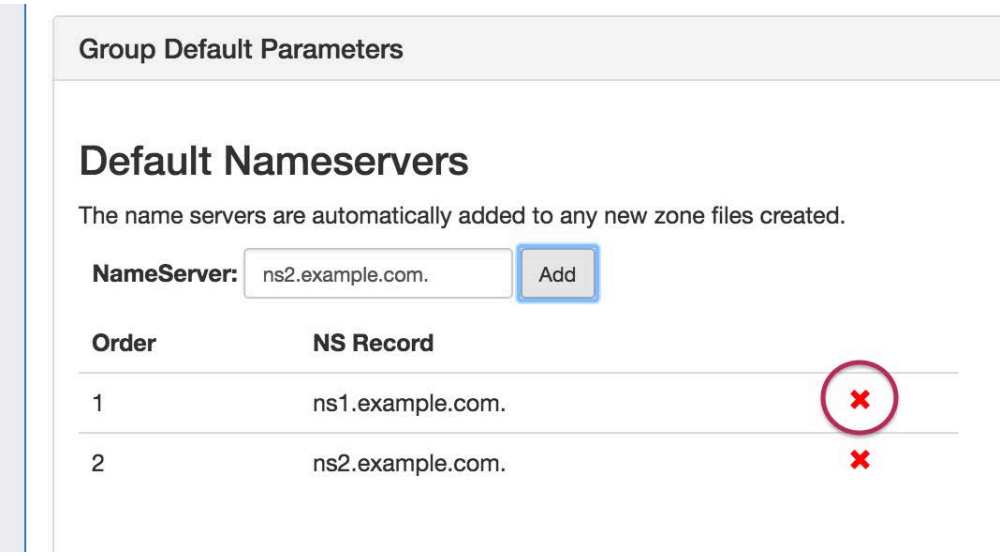


Edit Nameservers

Change a Nameserver by typing the new NS record into the Nameserver box, then click "Add".

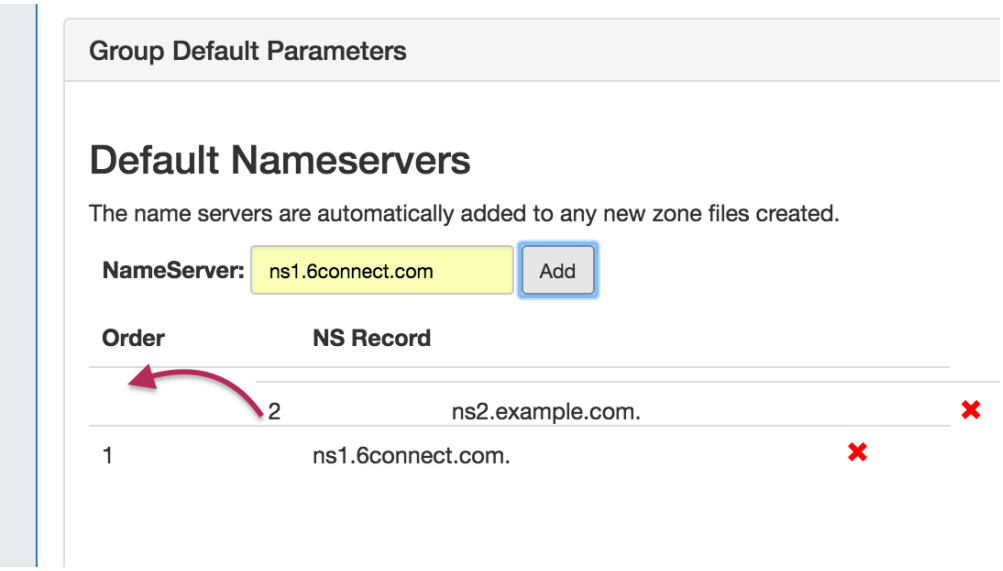


Delete the previous Nameserver by clicking on the red "X" to the right of the NS Record.



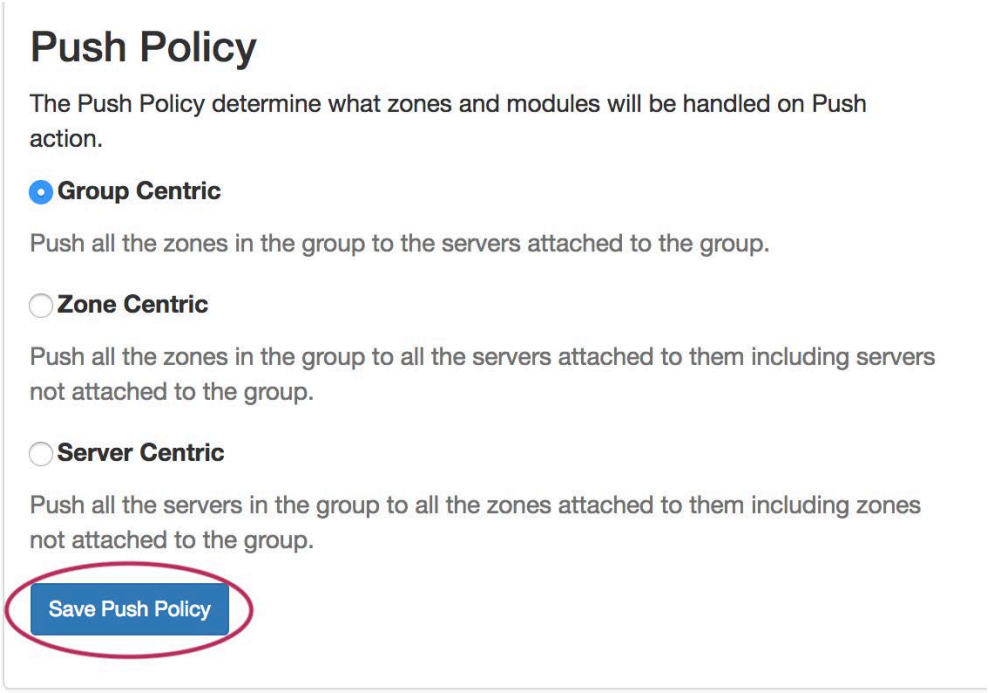
Change the order of multiple Nameservers by left-clicking (holding the mouse button down) over the NS Record to move, dragging to its desired placement, then releasing the mouse button.

The NS record will drop into its new location, with the order updated.



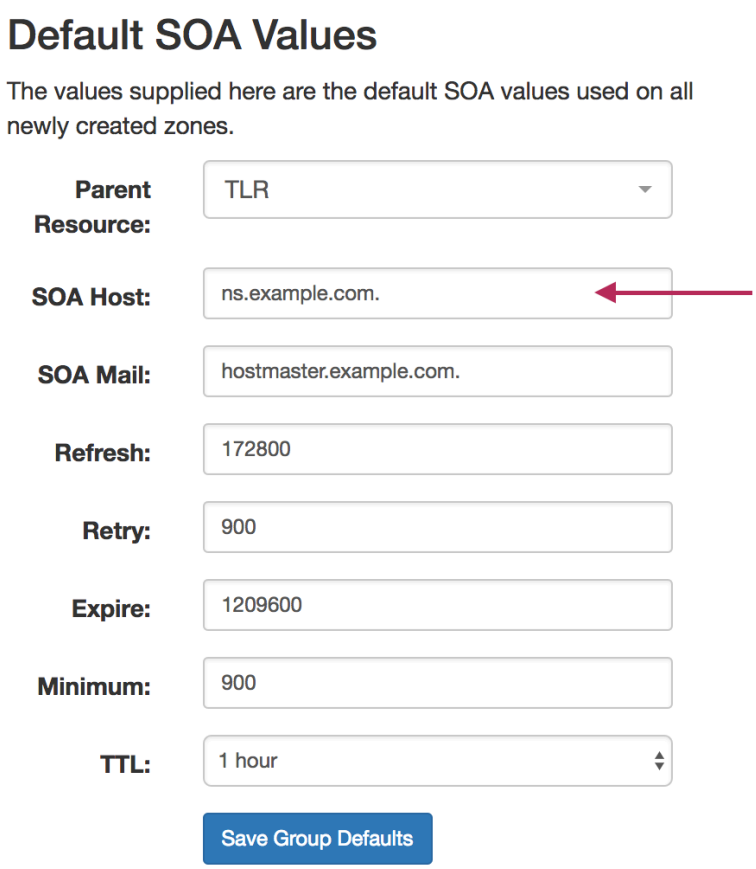
Edit Push Policy

Edit the push policy by selecting the radio next to the desired push behavior ("Group Centric", "Zone Centric", or "Server Centric") and click "Save Push Policy".



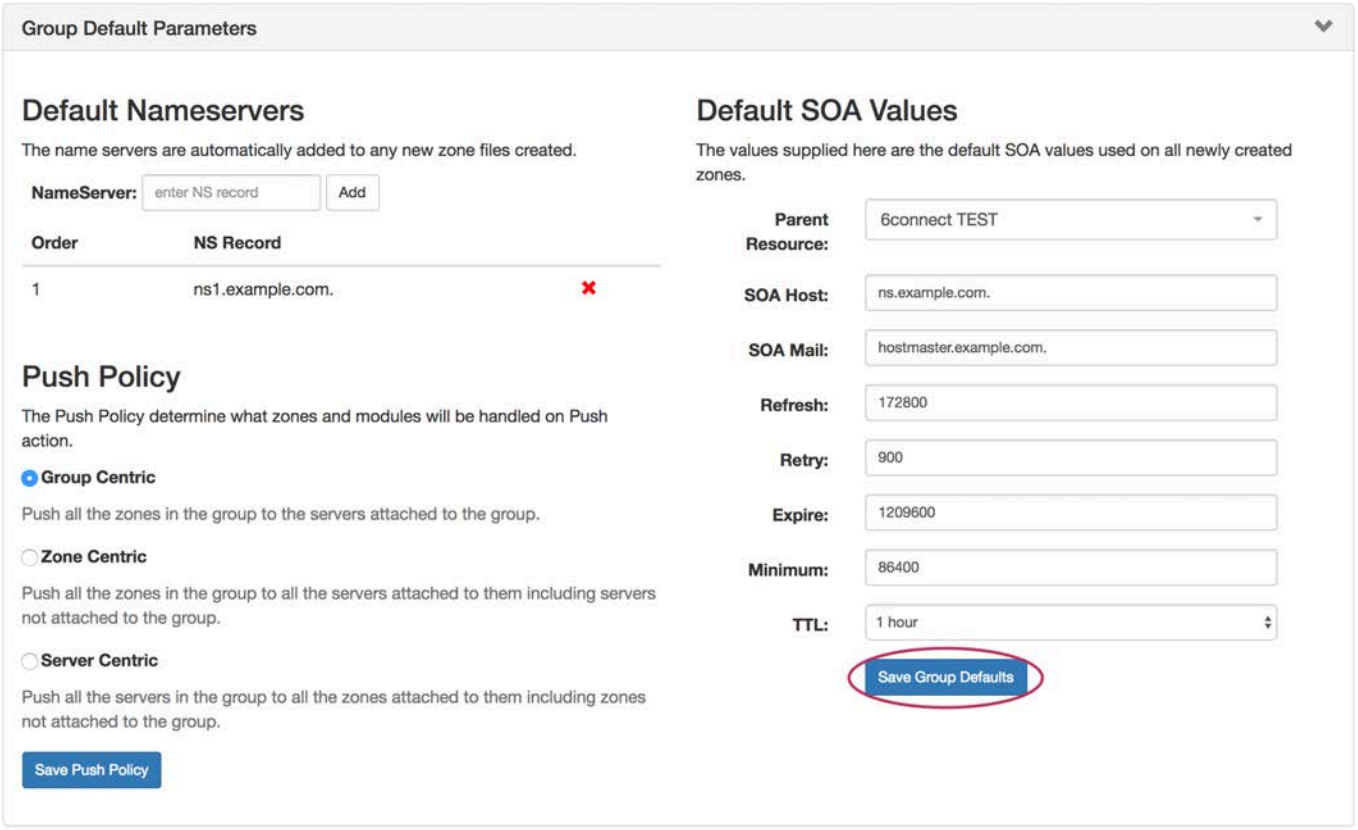
Edit SOA Values

To edit the Default SOA values, simply click inside the value to be changed and type or select your new value. Be sure to also verify the desired Parent Resource for the Group, as the resource selected determines the Group's permissions.



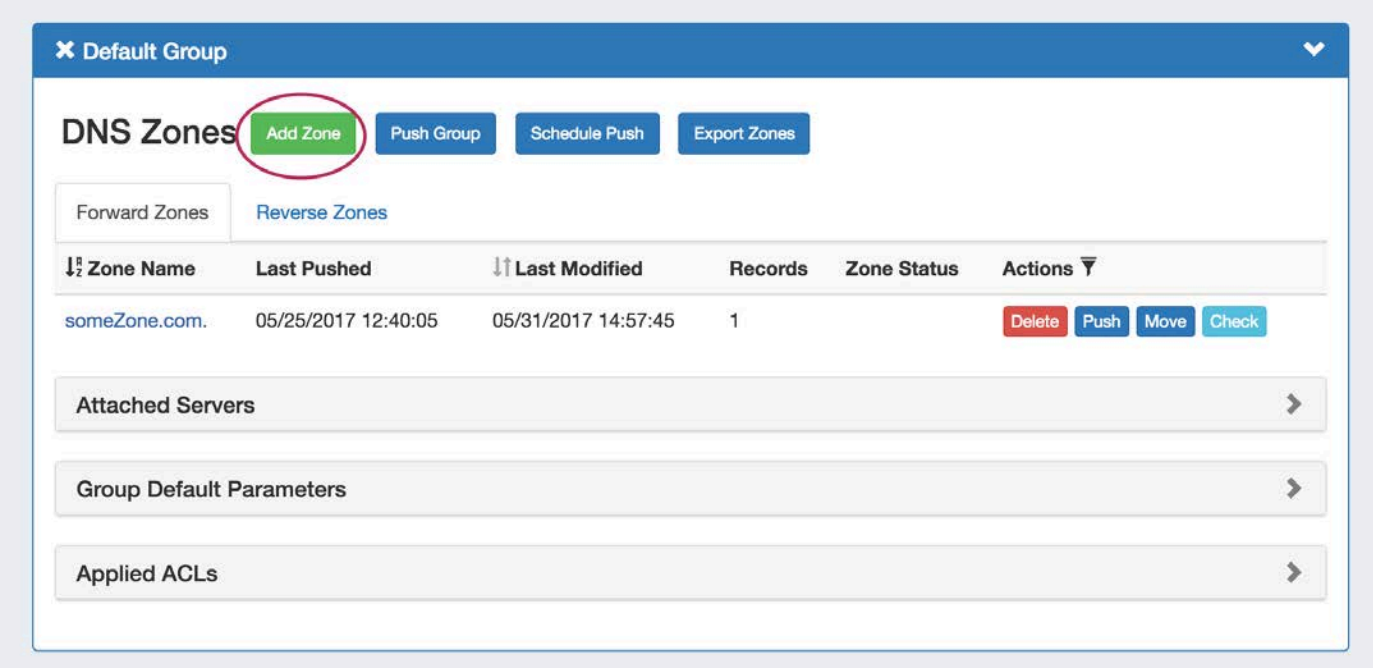
Save Changes

Save all Nameserver and SOA changes by clicking on the "Save Group Defaults" button at the bottom of the section. Closing without saving will cause all changes to be lost.



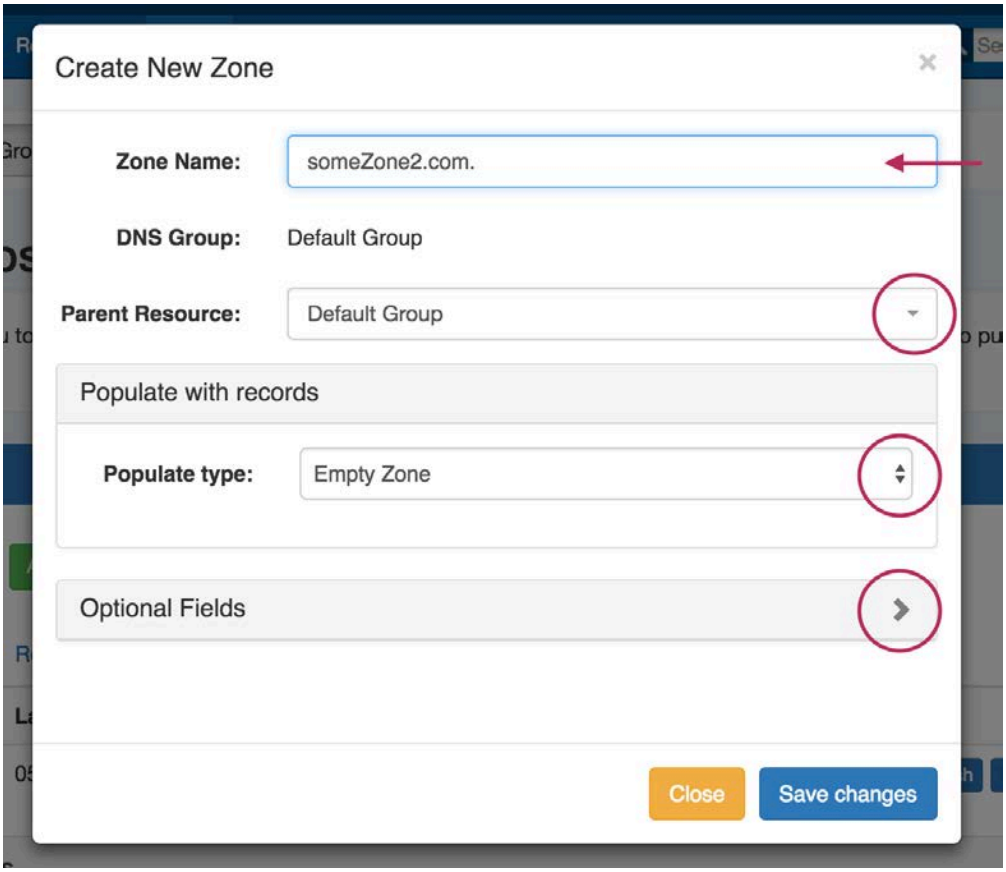
Add Zones to a Group

To add a new zone to a Group, click the "Add Zone" button inside the desired Group.



Fill in the zone name, select a parent resource (by default, the Group is selected), select a populate type, and fill in any optional fields.

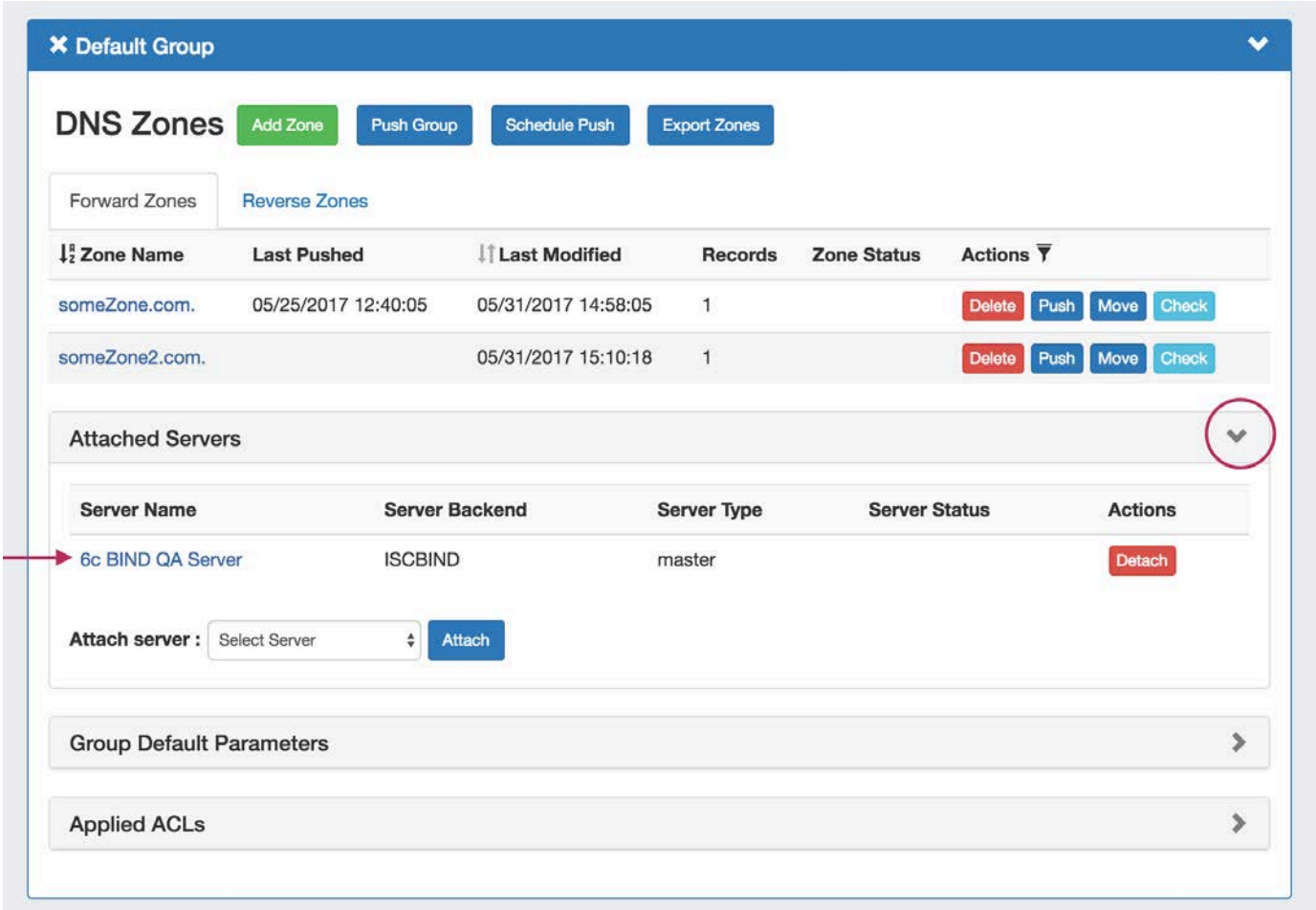
For more detail on each of these options, see Working with DNS Zones - Add a Zone.



When complete, hit the "Save Changes" button.

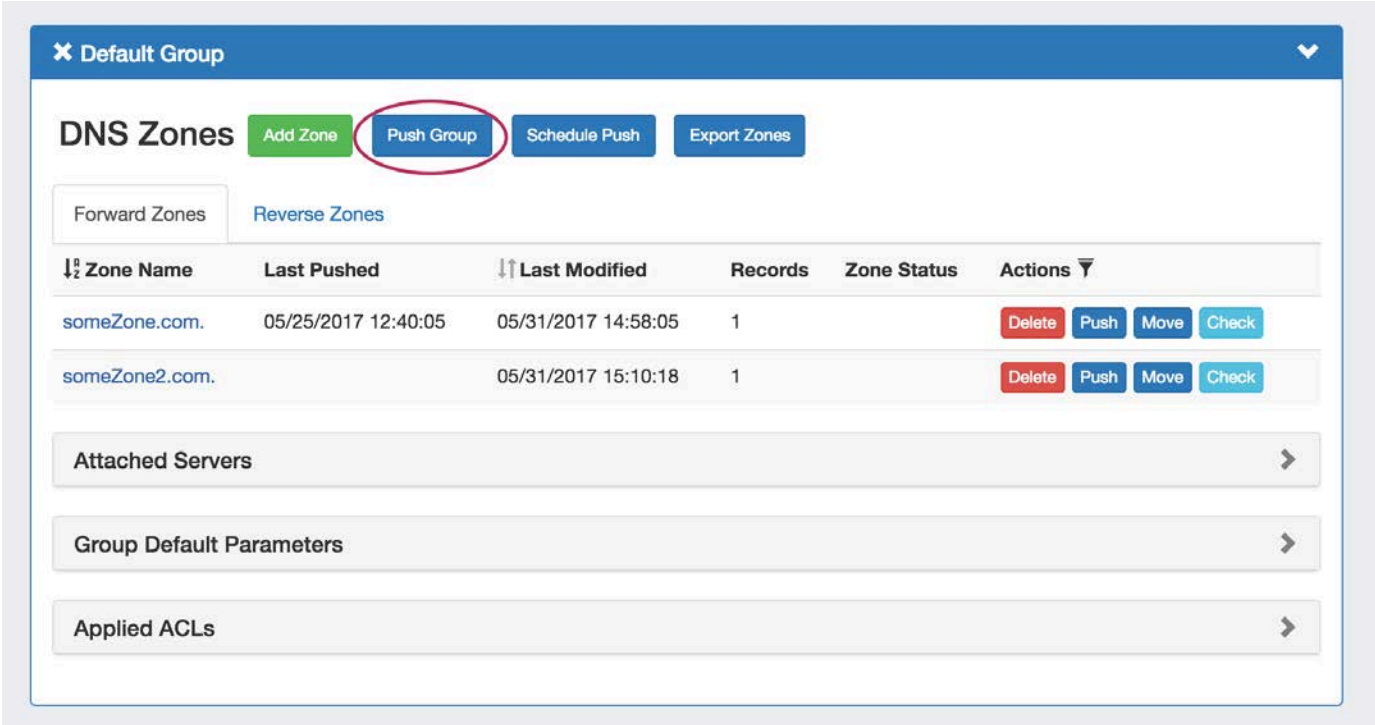
Pushing a Group

Before pushing a group, expand the "Attached Servers" section and verify that the correct server is attached to the DNS Group.

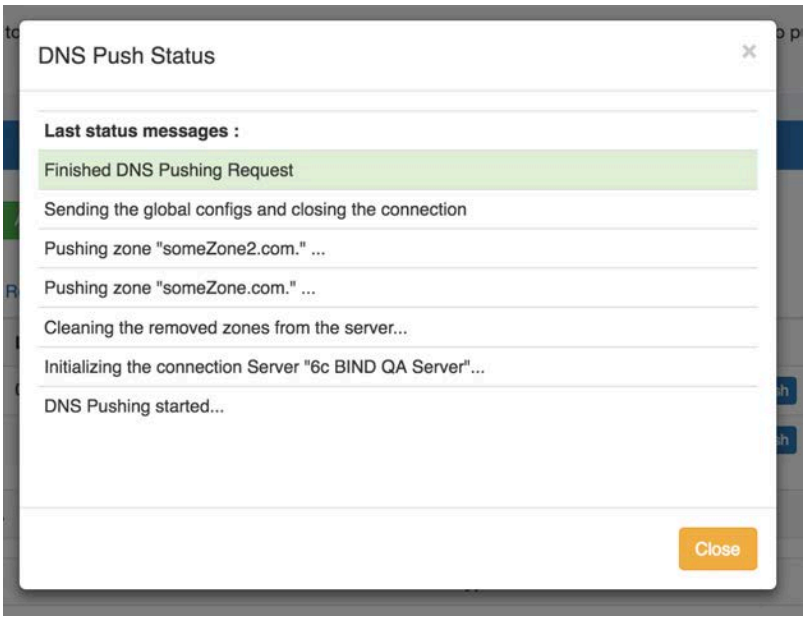


Manual Group Push

After verifying the attached server, manually push the Group by clicking the "Push Group" button under the desired Group.

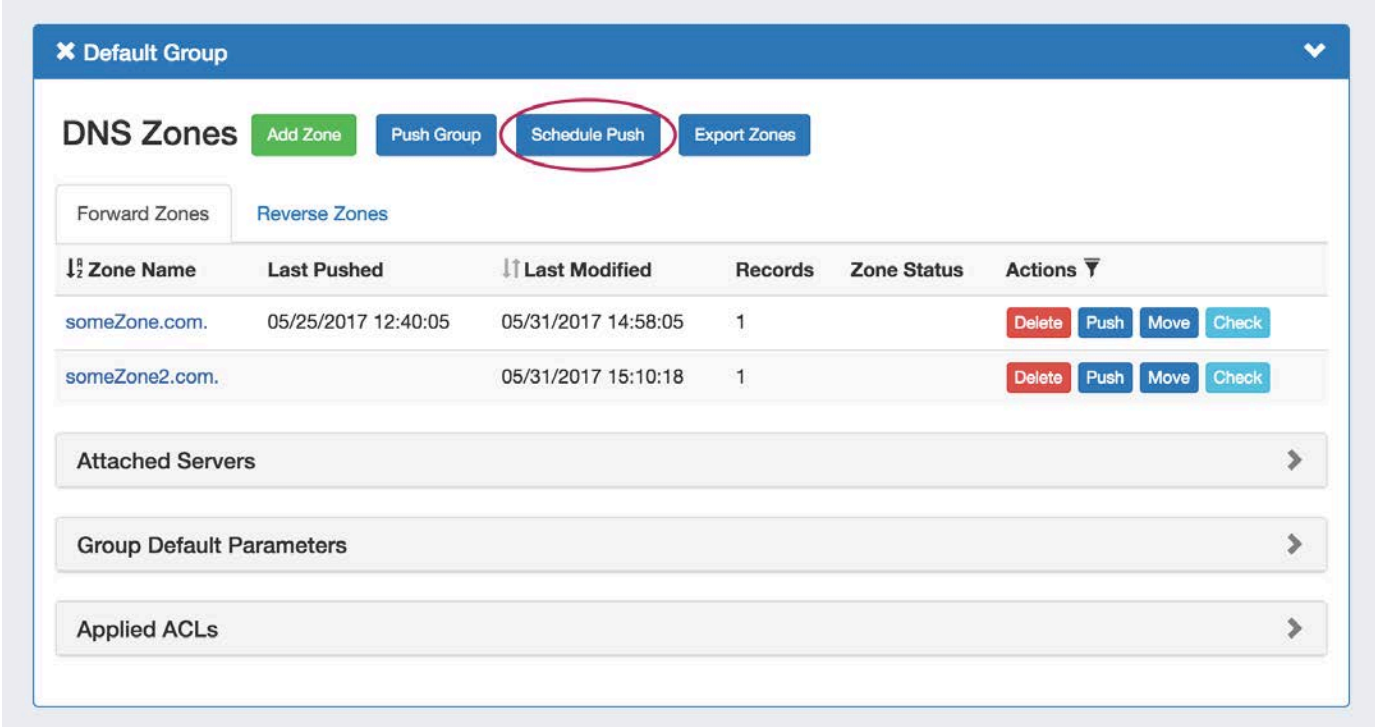


A DNS Push Status window appears, showing each zone in the Group as it is pushed to the attached server. If successful, a green "Finished DNS Pushing Request" will appear as the final message. If errors occur, an error message will show with details on the issue. When complete, hit the "Close" button.

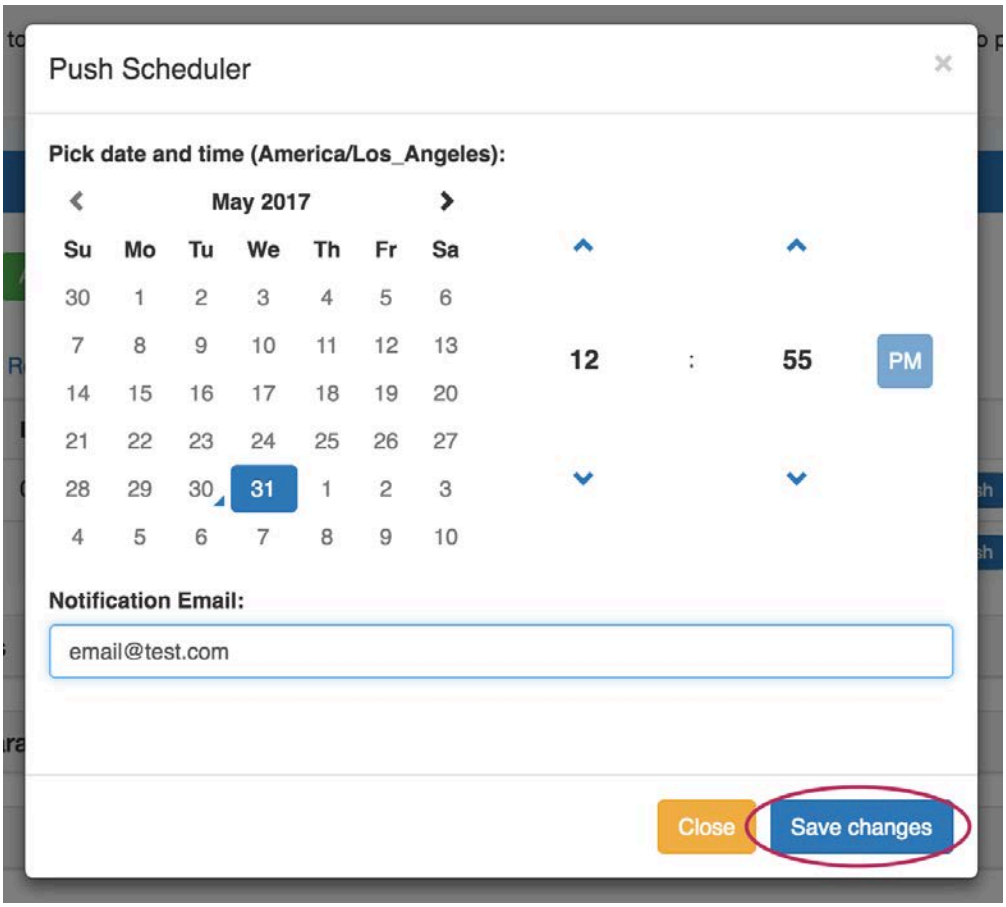


Scheduled Group Push

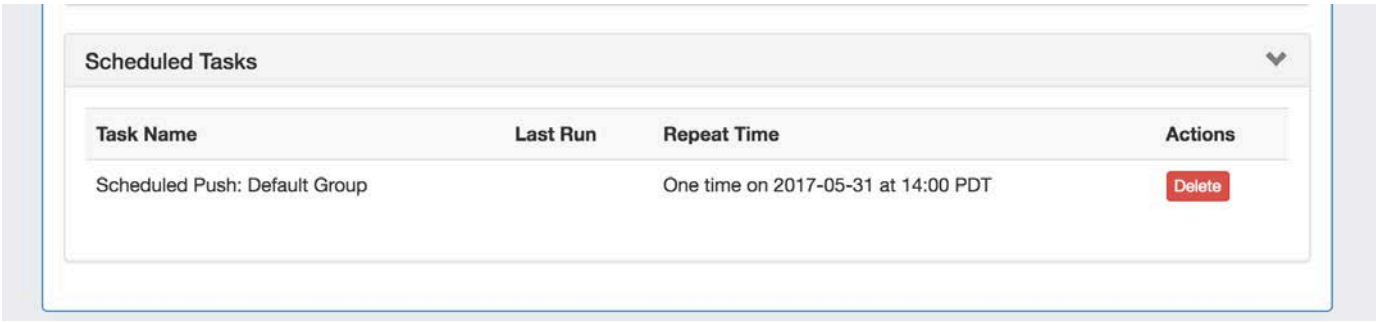
Scheduled pushes may be set up through the "Schedule Push" button for the Group.



Click "Schedule Push" for the Group, then select a date, time (12 hour, with AM / PM toggle), add a notification email address, and click "Save changes".



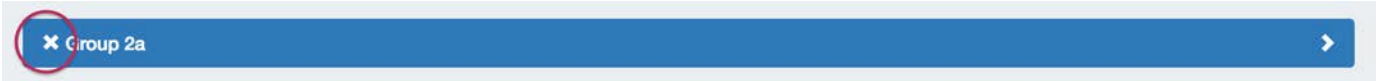
The Scheduled task will be shown in both the Group details (under "Scheduled Tasks") and the Admin Scheduler task list.



The task may be deleted prior to running by clicking the "delete" button, but will automatically be removed once completed.

Delete a Group

Delete a non-Default Group by clicking on the "X" to the left of the Group's name in the DNS Groups List.



A confirmation message will appear, click "Confirm" to save your changes, or "Close" to exit without saving.

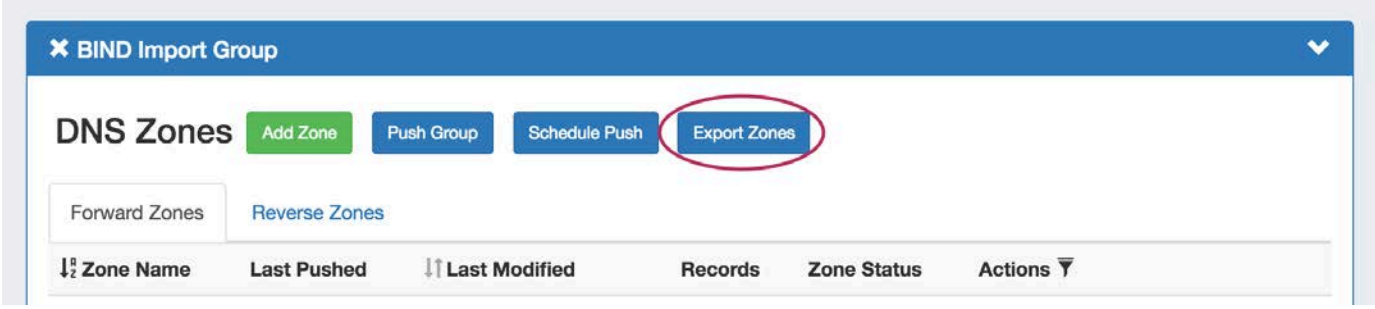


The Default Group may not be deleted.

Export a Group

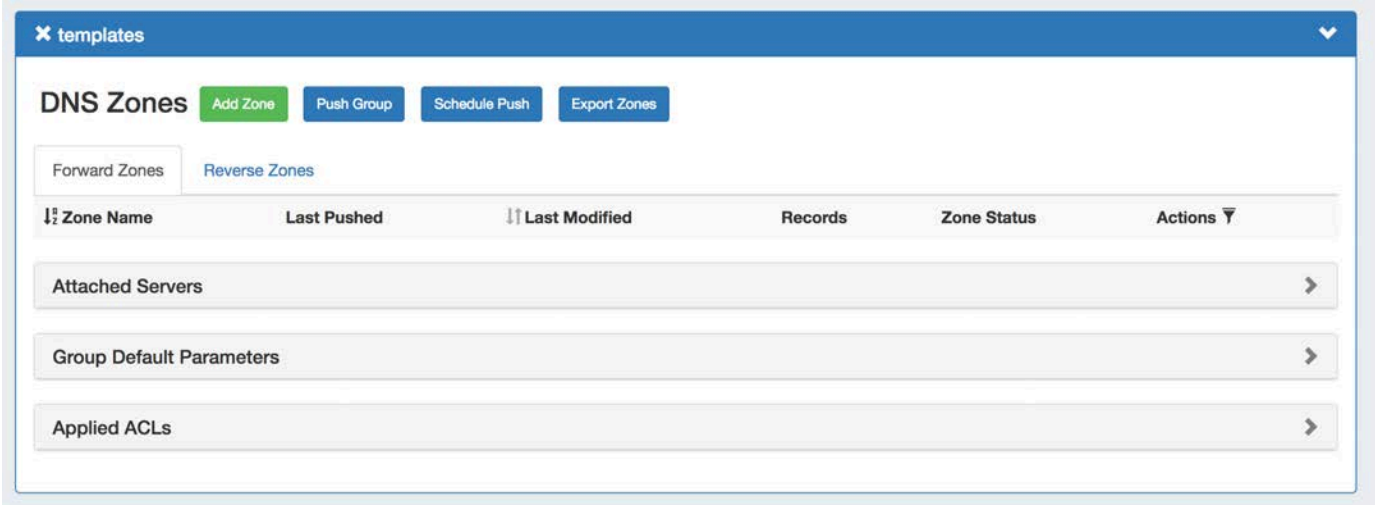
All zones under a DNS Group may be exported to a .zip file by clicking the "Export Zones" button at the top of the Group module.

The exported .zip file may be used with ProVision's BIND DNS Zone Upload / Import tool.



Using the Templates Group / Cloning Zones to a Group

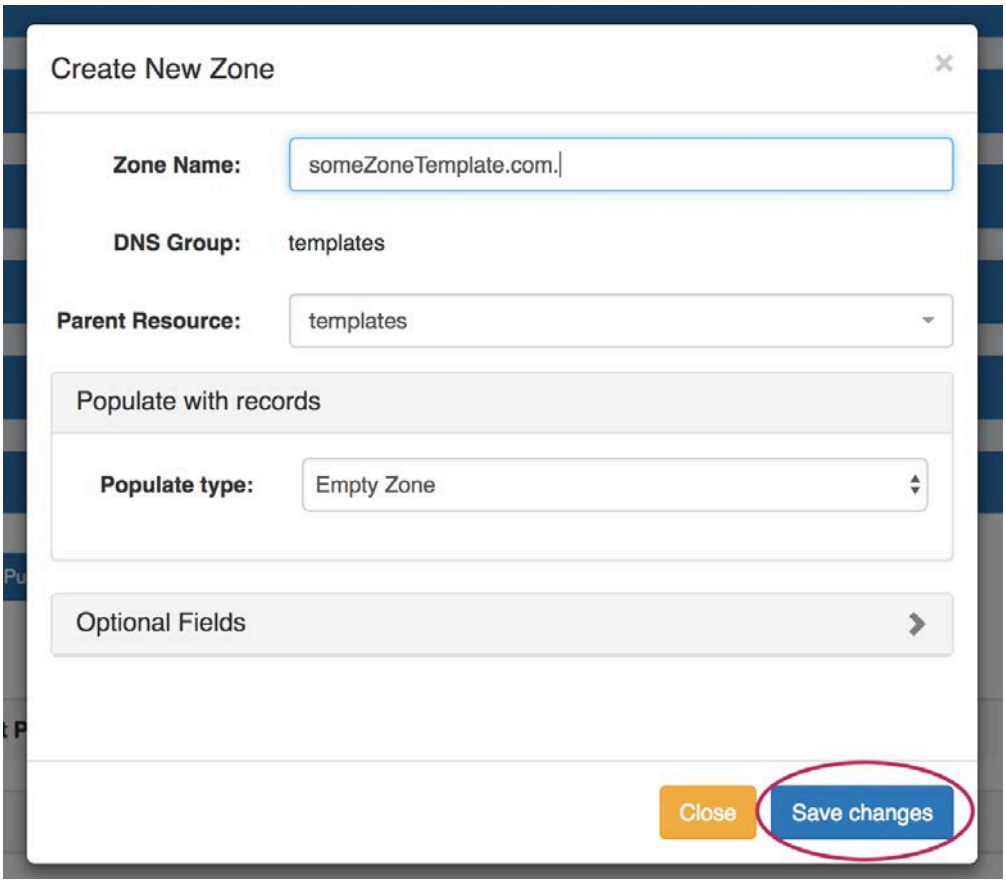
By default, a "templates" group is included in DNSv3. Any pre-DNSv3 DNS Templates will be included in this group, as well as any newly created zones to be used as template zones.



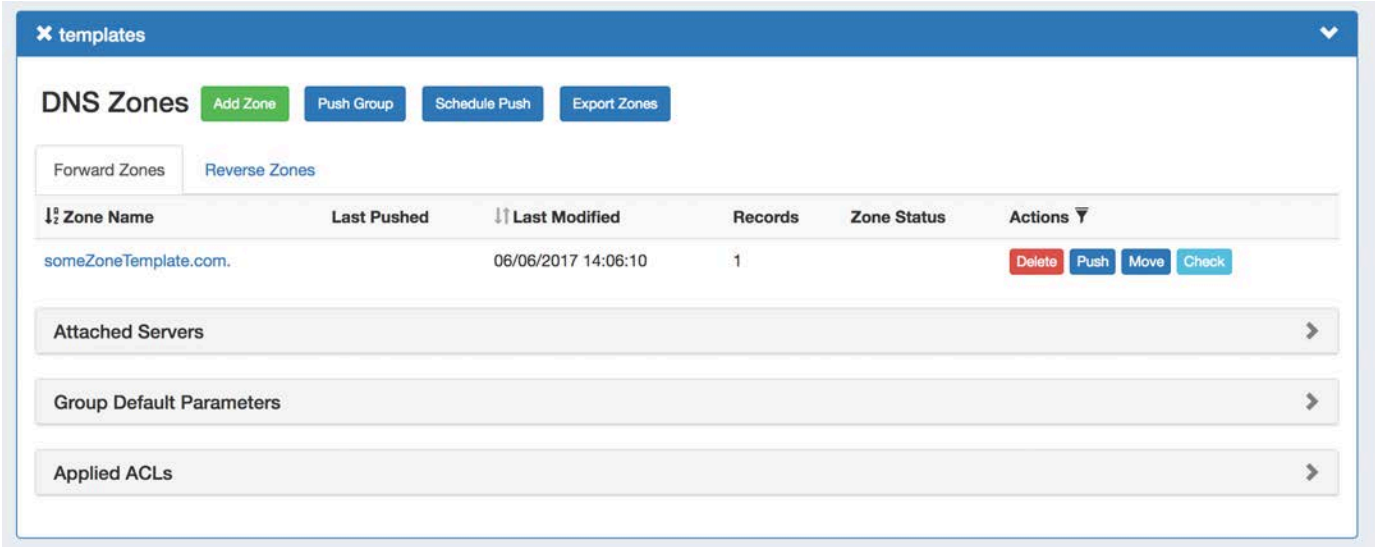
The template group functions the same as any other DNS Group, but gives a holding place for zones designed as templates separate from other zones.

Create your template zone

Add a template zone by clicking the "Add Zone" button, giving the zone a descriptive name to clearly identify as a template, select the populate type, and fill in any optional data. (see Working with DNS Zones - Add a Zone.)



Once the zone has been created, it will show in the templates group zone list.

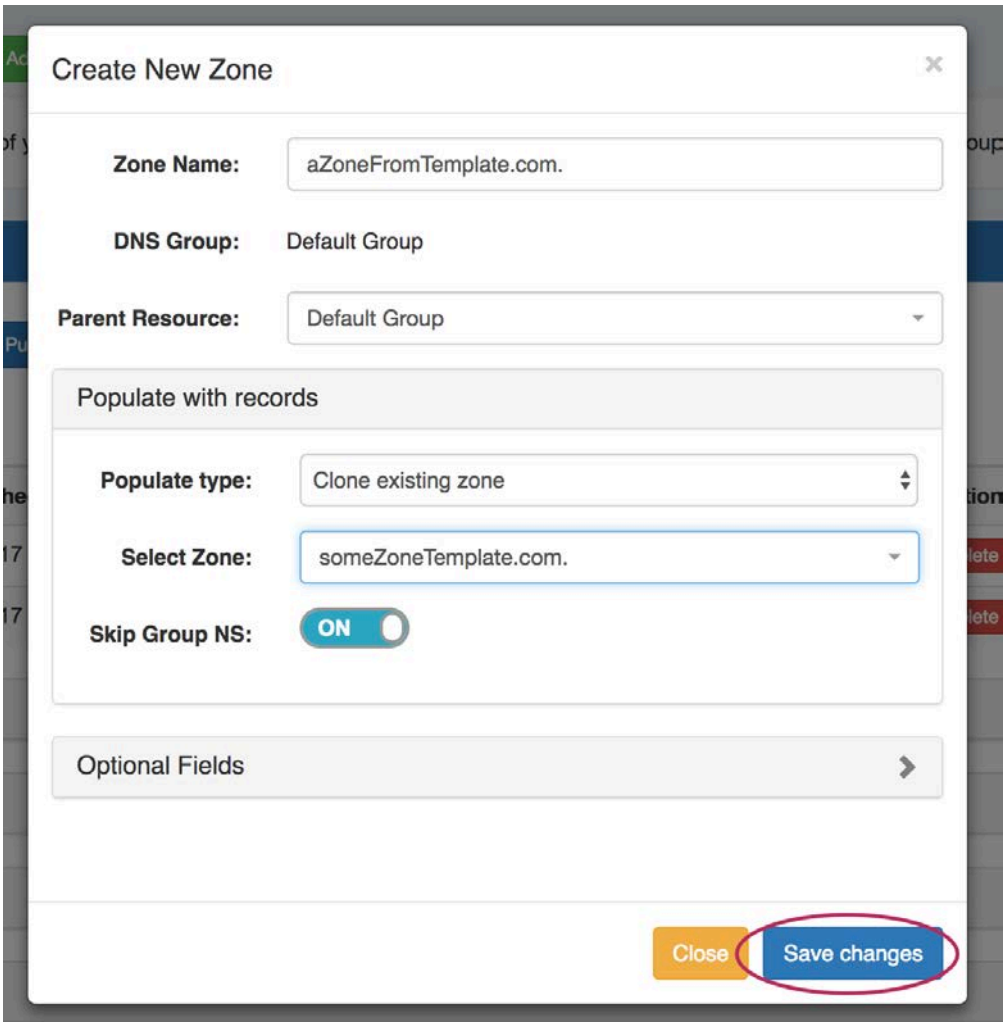


This zone, like any other, is now selectable as a base for the "clone existing zone" option when creating a new zone.

Clone Template to a Different Group

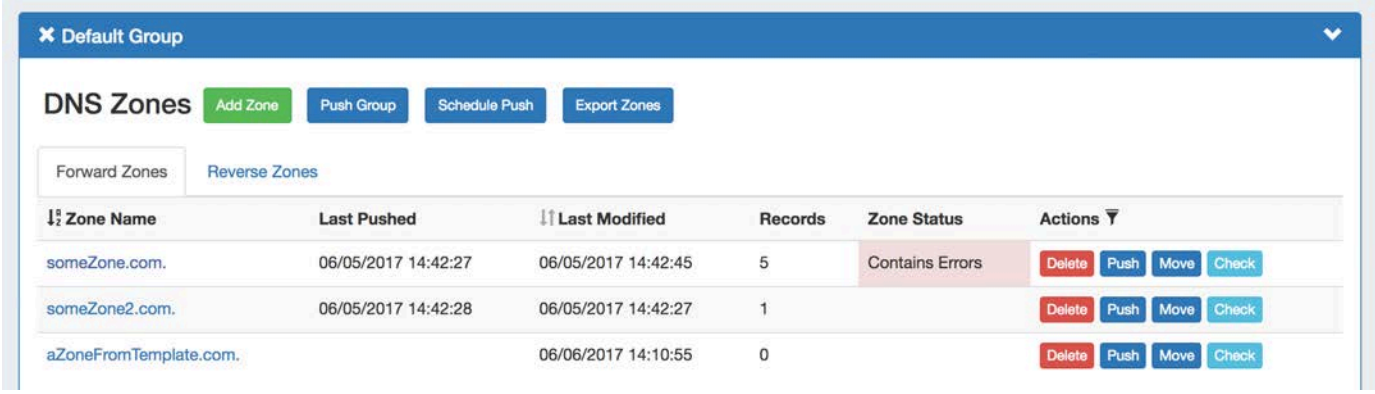
Go to a different group, and click "Add Zone". Enter in the new zone name and parent.

Under "Populate Type", select "Clone existing zone" and select /search for the template zone you created.



Edit the optional fields as needed, and click "Save changes".

Your cloned zone has been added to the Group's zone list.



Additional Information

For more information on DNSv3 tasks, see:

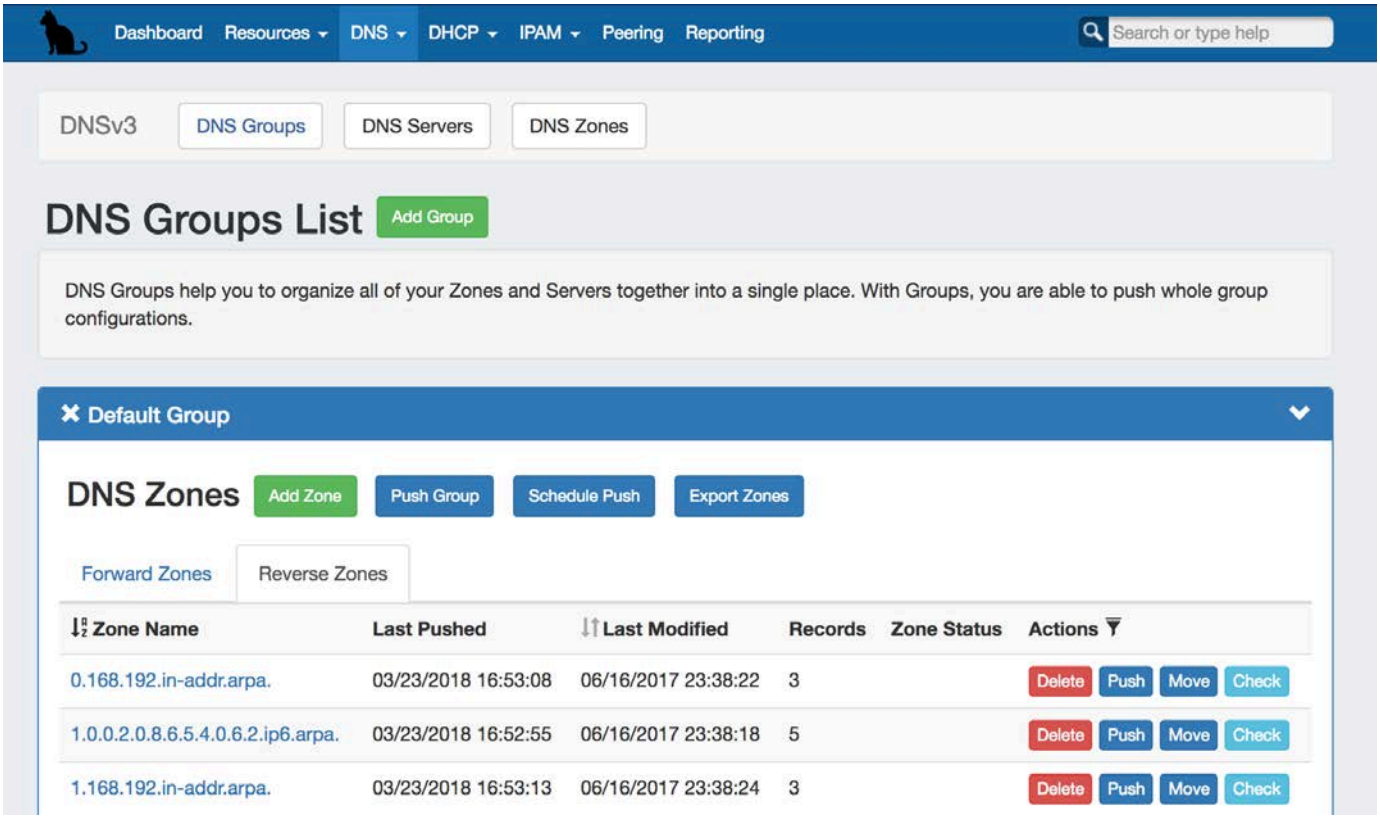
Working with DNS Zones - Common Tasks

Editing DNS Zones

DNS Administration

Working with DNS Zones - Common Tasks

Working with DNS Zones



This page details some common DNS tasks performed from the **DNS** tab and DNS Gadget. DNS Zones are accessed under the **DNS** tab, **DNS Groups** or **DNS Zones** sub-tabs. Separate zone lists exist under each DNS Group.

DNSv3 Zone Directory

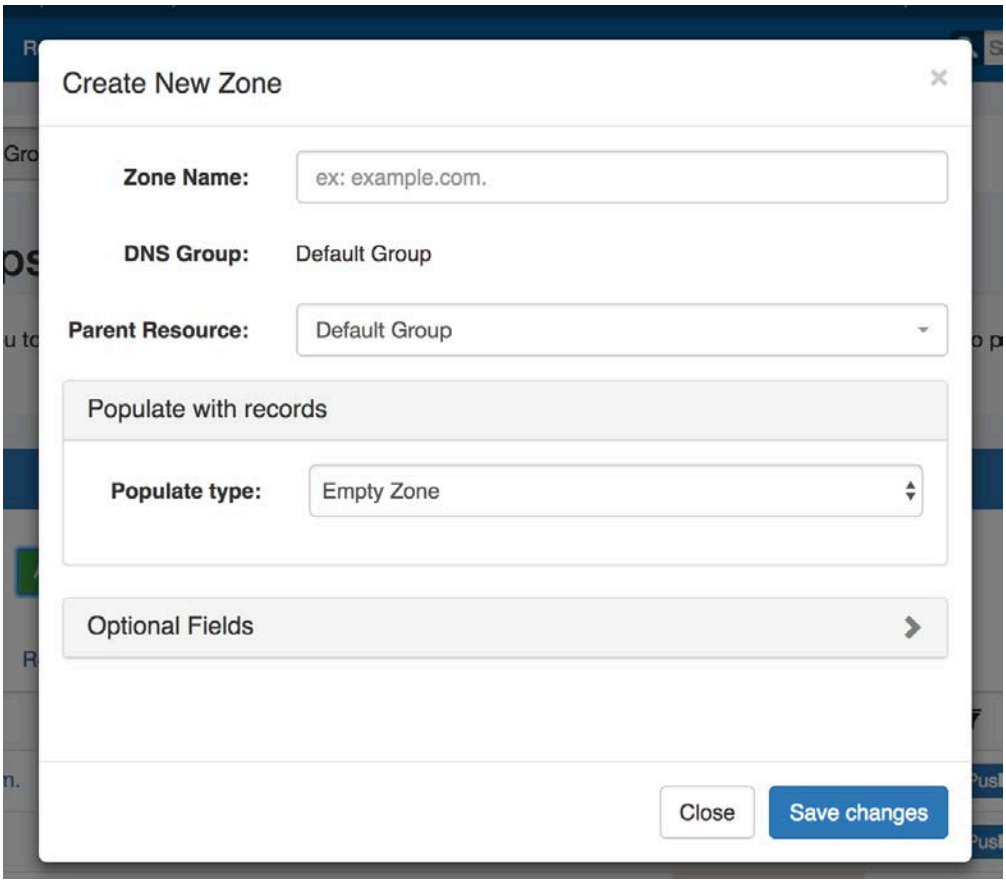
In DNSv3, zone files in the server directory will have a number appended, similar to 'example.com._123.zone'. This number is the Resource ID associated with the zone, and is used in ProVision to identify the zone and differentiate zones that may have the same name, but under different DNS Groups. This append does not affect any zone functionality otherwise.

- Working with DNS Zones
- Working with DNS Zones
  - Add a Zone
    - Optional Fields
  - Edit / Manage a Zone (View Zone page)
    - Add a comment:
    - Edit Advanced Settings:
    - Add or Edit Zone Records
    - Delete Zone Records
    - Attach / Detach Servers to a Zone
  - Move Zones to a Different Group
  - Pushing Zones
    - Push Now (from Group List):
    - Push Now (from View Zone page):
    - Schedule Push (from View Zone page):
    - Schedule Push (Admin Scheduler):
    - Group Push:
    - Server Push:
  - Export a Zone
  - Delete a Zone
  - Addiitonal Information:
-

Working with DNS Zones

Add a Zone

Add a new zone while in the **DNS Groups** tab by selecting "Add Zone" under the desired Group. You may also add new zones from the **DNS Zones** page, also by the "Add Zone" button. The "Create New Zone" dialog will pop up.



Enter the name of your new zone and select a parent resource, if desired. The DNS group will already be selected.

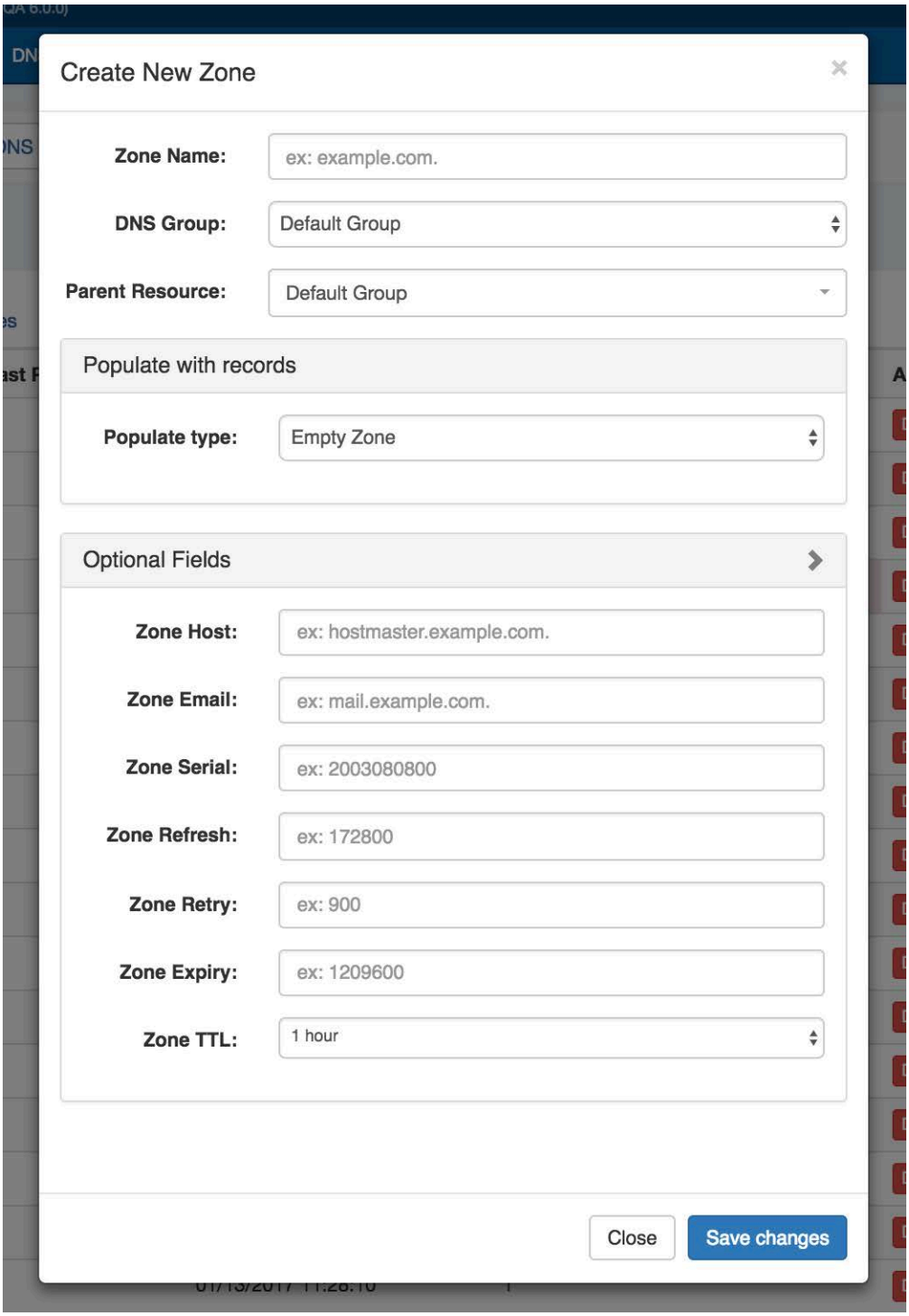
DNSv3 allows you to populate the zone with three options:

- Empty Zone - create the zone as an empty zone, and manually create records
- Populate by AXFR Transfer - transfer zones from an outside server
- Clone an existing zone - use an existing zone as a template, and manually edit as needed

From there, you may either save your changes and complete zone creation, or add in optional field details.

Optional Fields

Expand the "Optional Fields" section by clicking on the chevron (>) or the "Optional Fields" header.



Under this section, add in optional values for Zone Host, Email, Serial , Refresh, Retry, Expiry, or TTL.

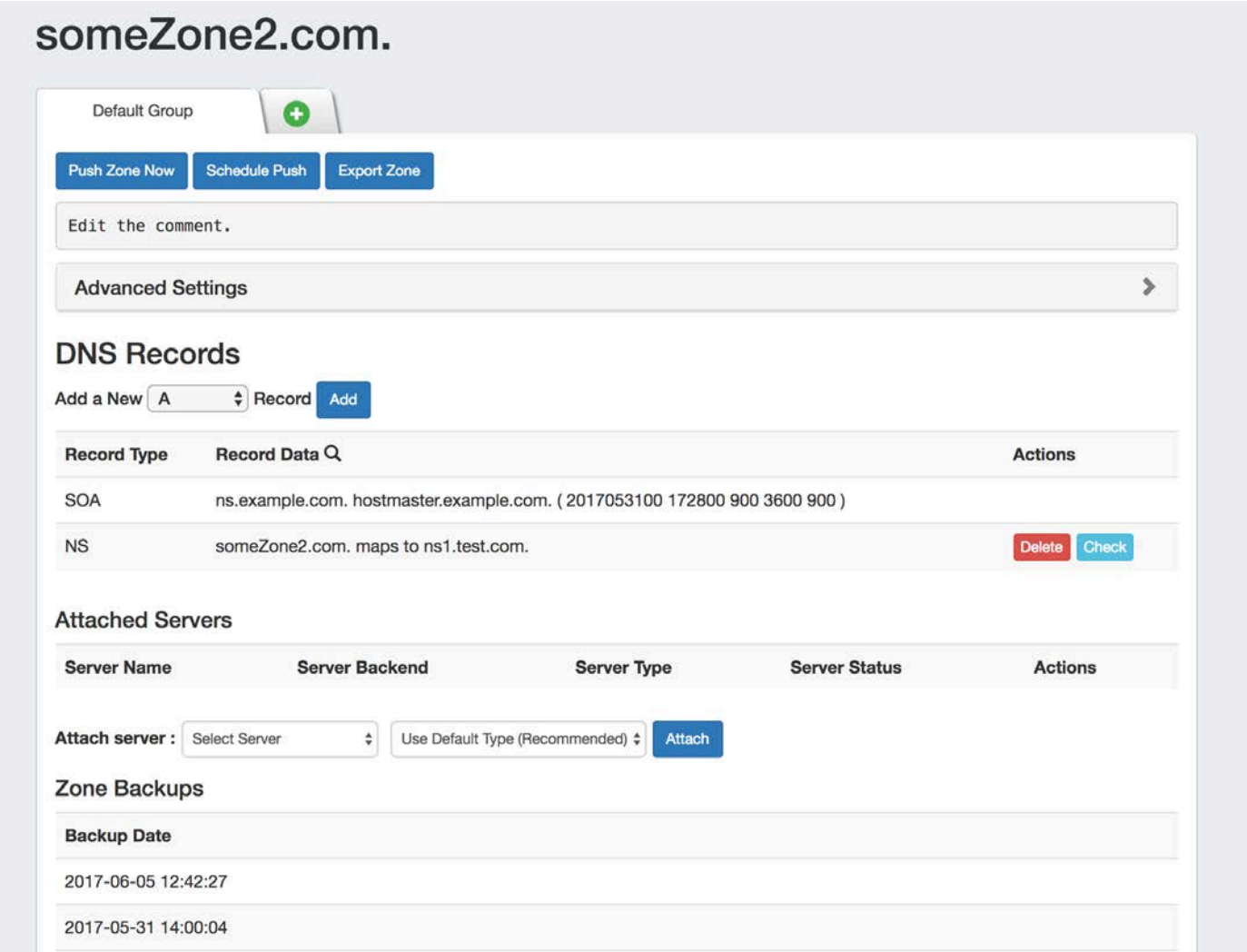
When finished, click the "Save Changes" button.

To exit without saving, simply click "Close".

Edit / Manage a Zone (View Zone page)

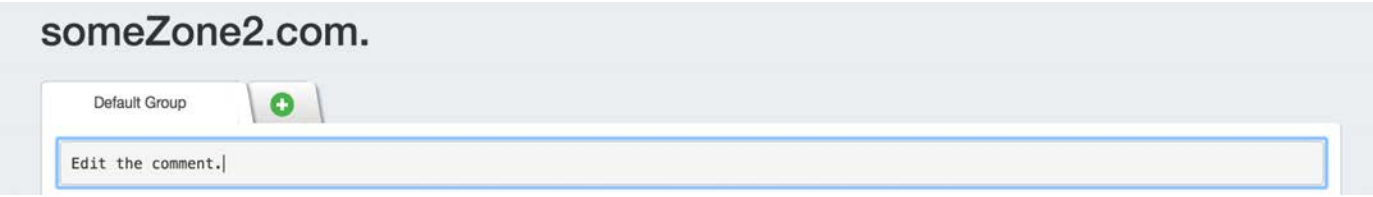
Zones may be editing by clicking on the Zone name in any zone list in which it appears - the Zone List, zones listed under DNS Groups, or zones listed as attached to a server under DNS Servers.

Clicking on the zone name opens up the View Zone page, from where comments may be added, records added or updated, servers attached to the zone, advanced settings edited, backups restored or the zone pushed / exported. For details on the View Zone page, see Editing DNS Zones.



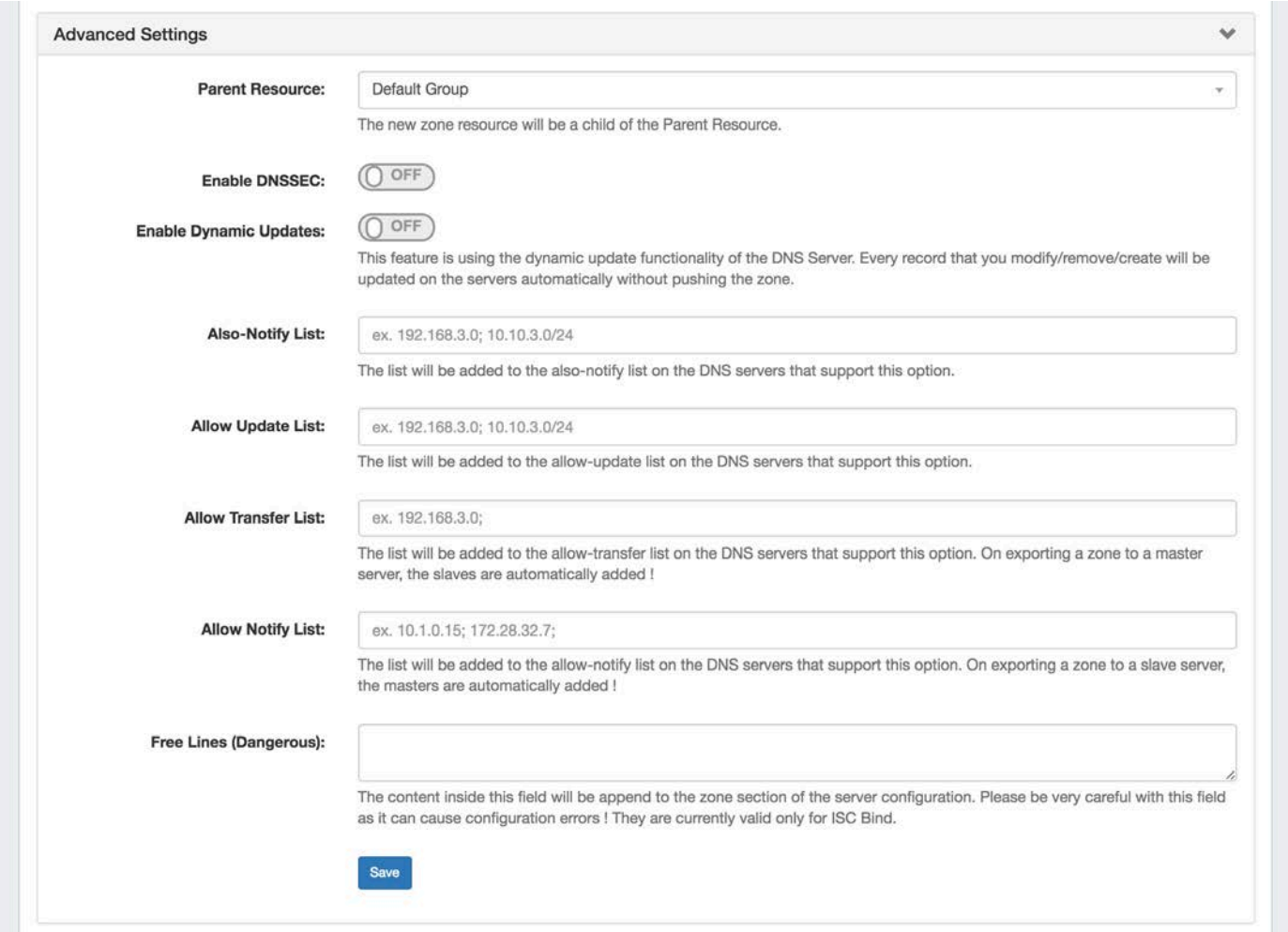
Add a comment:

To add a comment to a zone, click anywhere inside the comment area and begin typing. The comment will be saved when an area outside of the comment box is clicked, deselecting the comment area.



Edit Advanced Settings:

Open the zone "Advanced Settings" section by clicking on the header or the expansion arrow (>). Edit fields as needed, and click "Save" when complete.



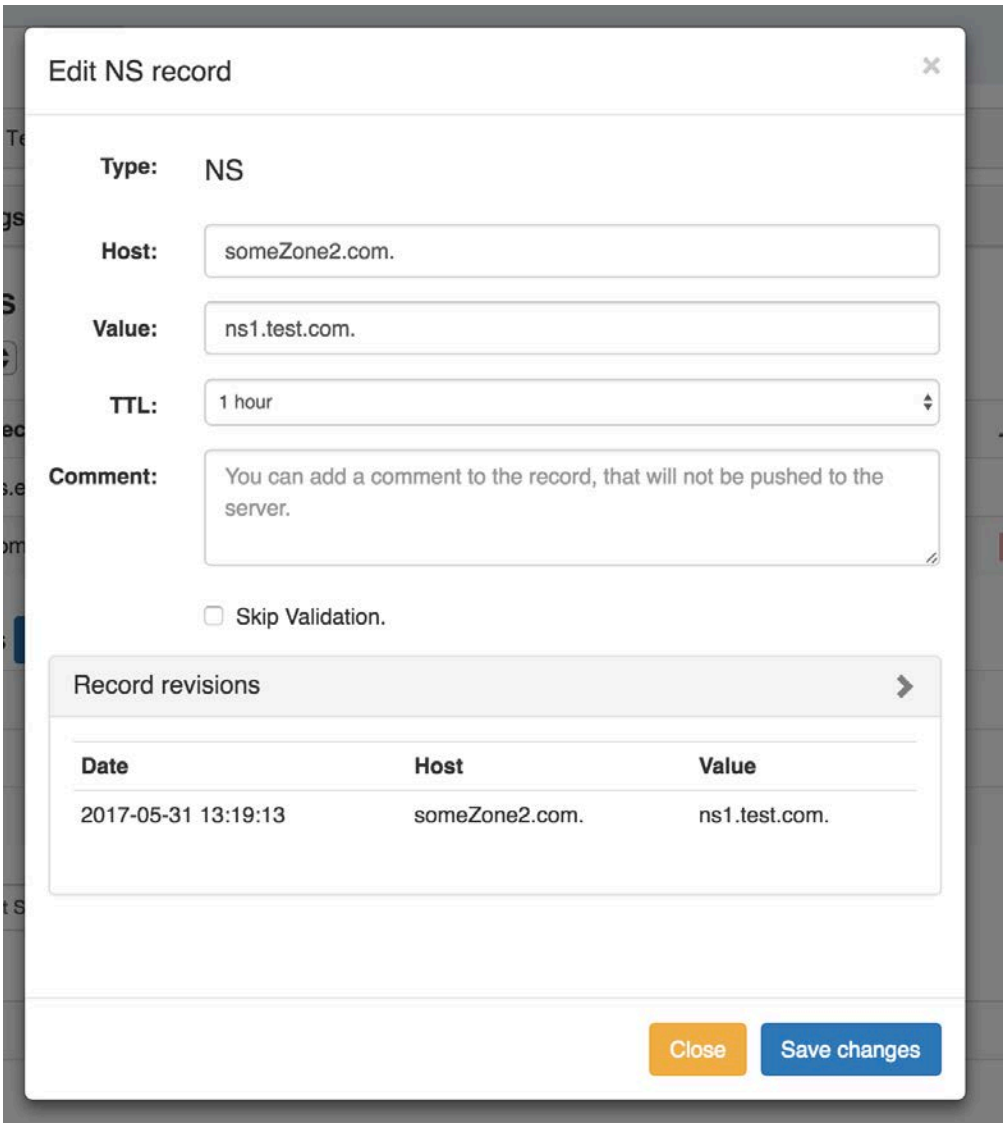
Add or Edit Zone Records

From the View Zone page, under DNS Records, look for the line "Add a new ___ Record". Select the desired record type, then click "Add".

To edit an existing zone record, click on the record line in the DNS Records list.



The Edit Record dialog will open, allowing you to fill in record details.



Fill in the Host, Value, TTL, and comment if desired.

An option exists to "Skip Validation" - check to select if you wish to bypass validation for the record.

If previous versions of the record exist, the revision log will be visible under "Record revisions".

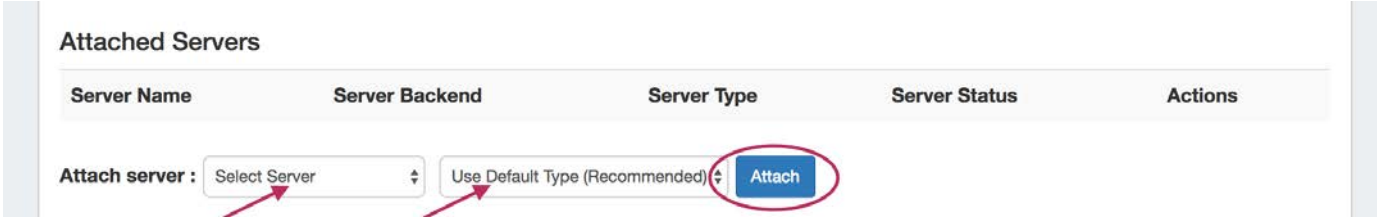
When edits or reviewing is complete, click "Save changes", or "Close" to exit without saving.

Delete Zone Records

Delete a zone record by clicking the "Delete" button under "Actions" for the desired record entry.

Attach / Detach Servers to a Zone

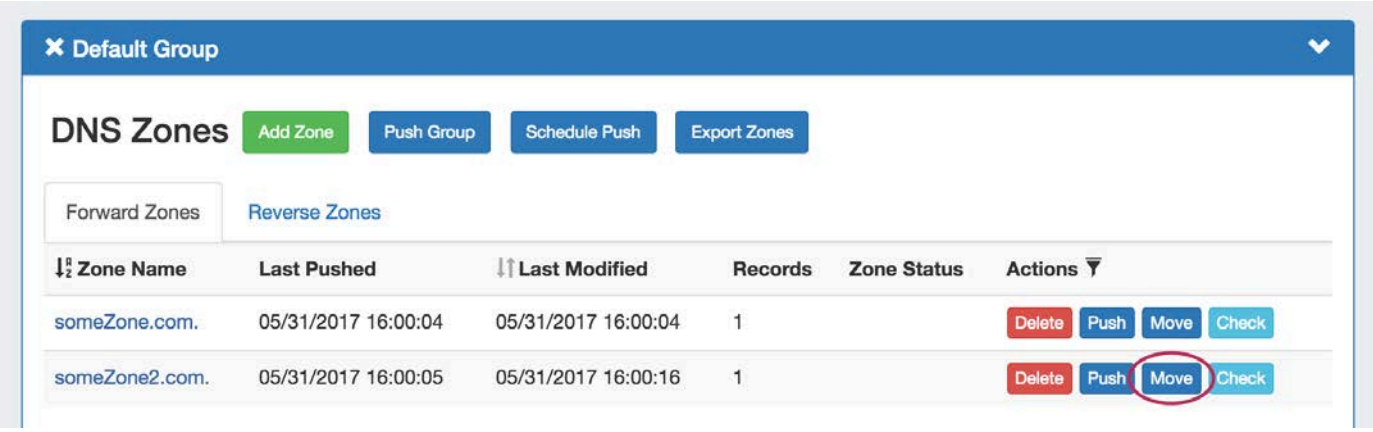
Under the "Attached Servers" section of the View Zone page, next to "Attach Server:", select the desired server to attach and choose the server type (Default, Master, Slave), then click the "Attach" button.



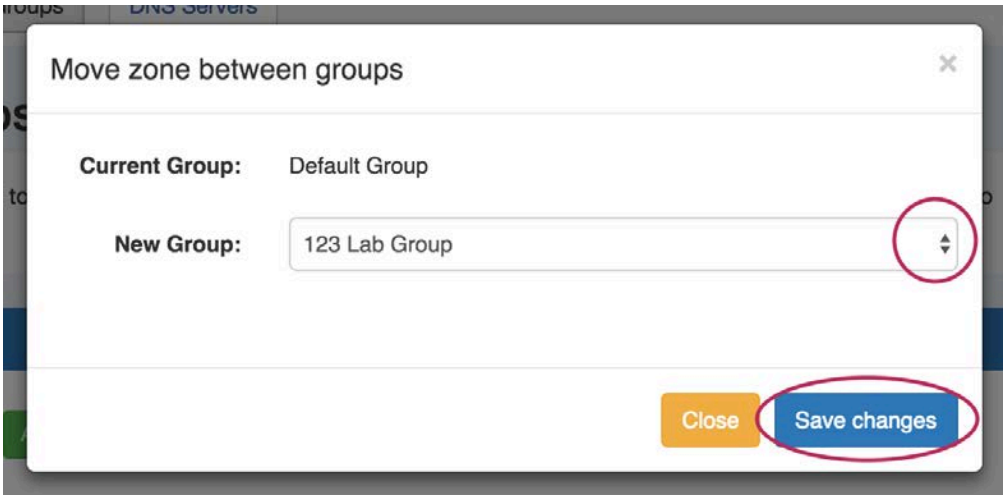
To detach a server from a zone, look for the server under the "Attached Servers" list, then click "Detach" under actions for that server.

Move Zones to a Different Group

Zones may be moved to a different Group than the one created under, by clicking the "Move" button in the zone's Actions menu.



The "Move Zone" modal will appear - select the new Group to move to the zone under, and click "Save Changes" to complete.

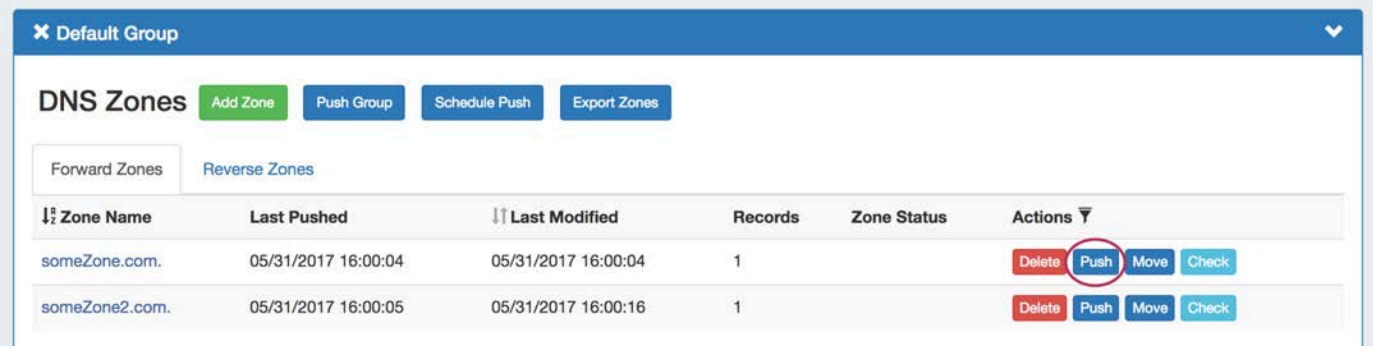


Pushing Zones

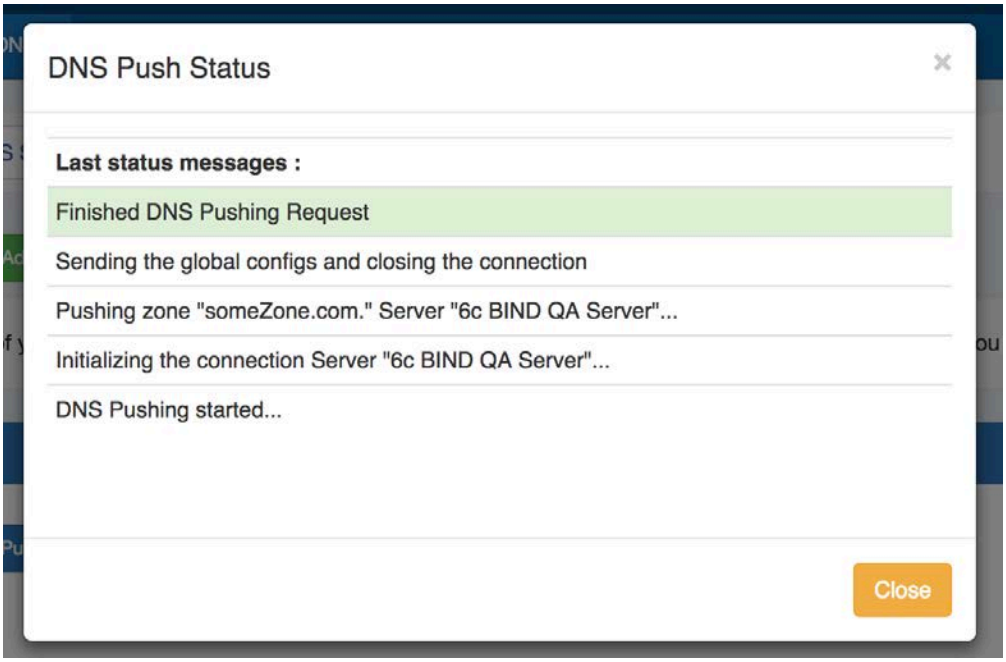
Zones may be pushed individually ("Push Zone Now"), as a scheduled future push ("Schedule Push"), or pushed as part of a Group or Server Push.

Push Now (from Group List):

To immediately push a single zone, navigate to the DNS Group the zone is under, then click the "Push" button in the Actions column for the zone. You may also look for the zone in the **DNS Zones** tab Zone List, and then likewise, click the "Push" button in the zone row.



The "Push Status" modal will appear, showing status and any errors that occur. When a successful zone push is complete, a green "Finished DNS Pushing Request" message will appear, and the window may be closed.



Push Now (from View Zone page):

The View Zone page, accessed from clicking on the zone name in the Group zone list, includes a "Push Zone Now" button that immediately attempts to push the zone to the attached server(s).

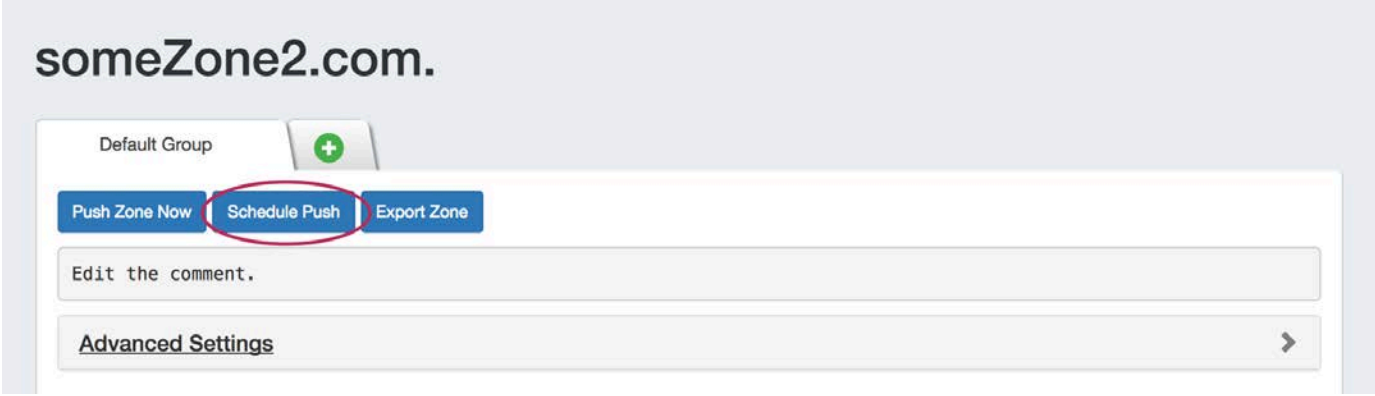


Click on "Push Zone Now", and the "Push Status" modal will appear, showing status and any errors that occur.

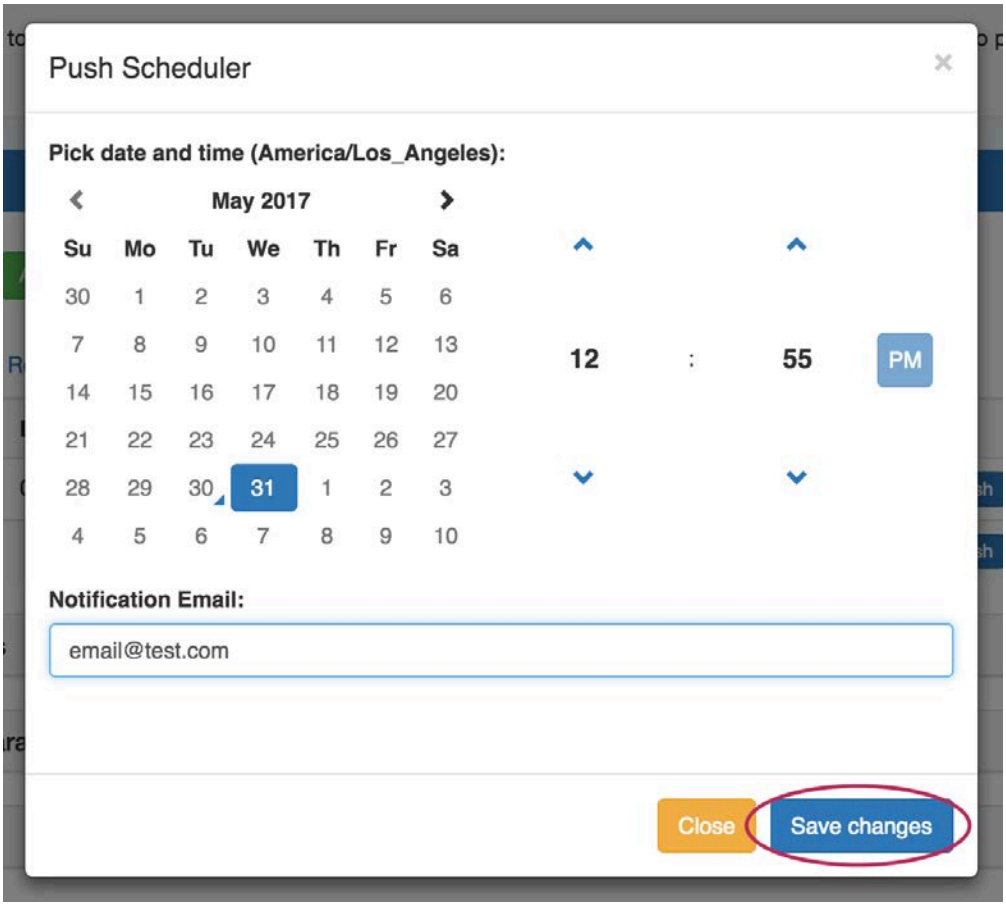
When a successful zone push is complete, a green "Finished DNS Pushing Request" message will appear, and the window may be closed.

Schedule Push (from View Zone page):

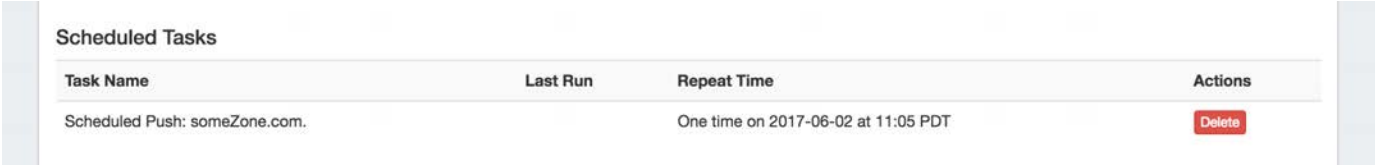
The View Zone page includes an option to schedule a zone push, similar to scheduling a Group push.



Click "Schedule Push" for the zone, then select a date, time (12 hour, with AM / PM toggle), add a notification email address, and click "Save changes".



The Scheduled task will be shown in both the Zone View page (under "Scheduled Tasks") and the Admin Scheduler task list.



The task may be deleted prior to running by clicking the "delete" button, but will automatically be removed once completed.

Schedule Push (Admin Scheduler):

Scheduled zone and server pushes may also be set up by Admin users from the **Scheduler** tab. For more information, see the Scheduler documentation.

Group Push:

All zones contained in a DNS Group may be pushed at the same time, by pushing the DNS Group.

For information on working with DNS Groups, including performing a manual or scheduled Group push, see Working with DNS Groups - Pushing a Group.

Server Push:

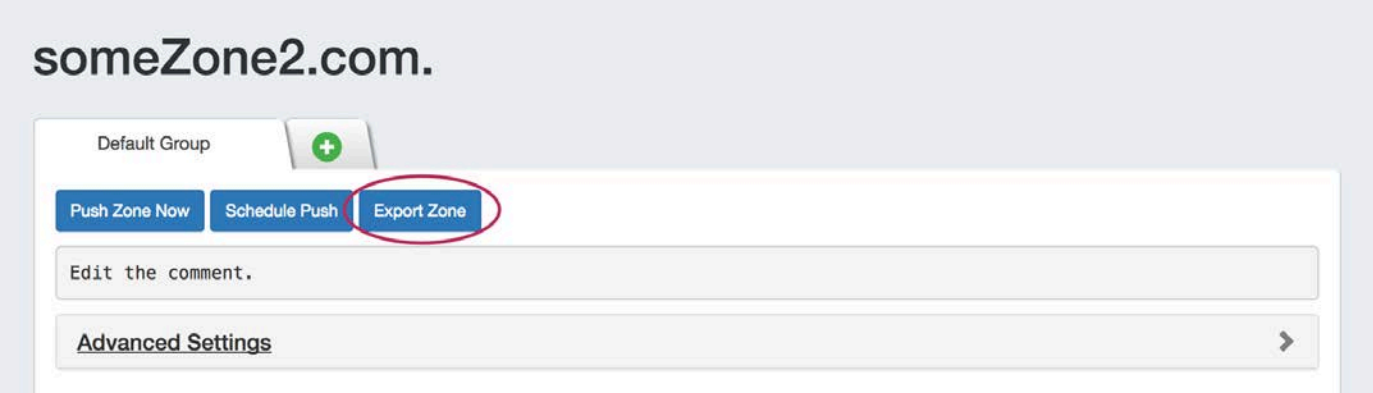
Admin users may push all zones on a server from the **DNS** Tab -> **DNS Servers** page.

For information on working with DNS Servers, including performing a manual or scheduled server push, see Working with DNS Servers - Pushing a Server.

Export a Zone

Individual zones may be exported to a .zone file by clicking the "Export Zone" button on the View Zone page.

Individual .zone files may later be compressed into a .zip, and used with ProVision's BIND DNS Zone Upload / Import tool.



Delete a Zone

Users with resource delete permissions over the zone may delete a zone record by clicking the "Delete" button under "Actions" for the desired zone entry in any Zone List.

Addiitonal Information:

For more information on working with DNS Zones, Groups, and Servers, see the following sections:

DNS Tab

Working with DNS Groups

Editing DNS Zones (The View Zone Page)

DNS Administration

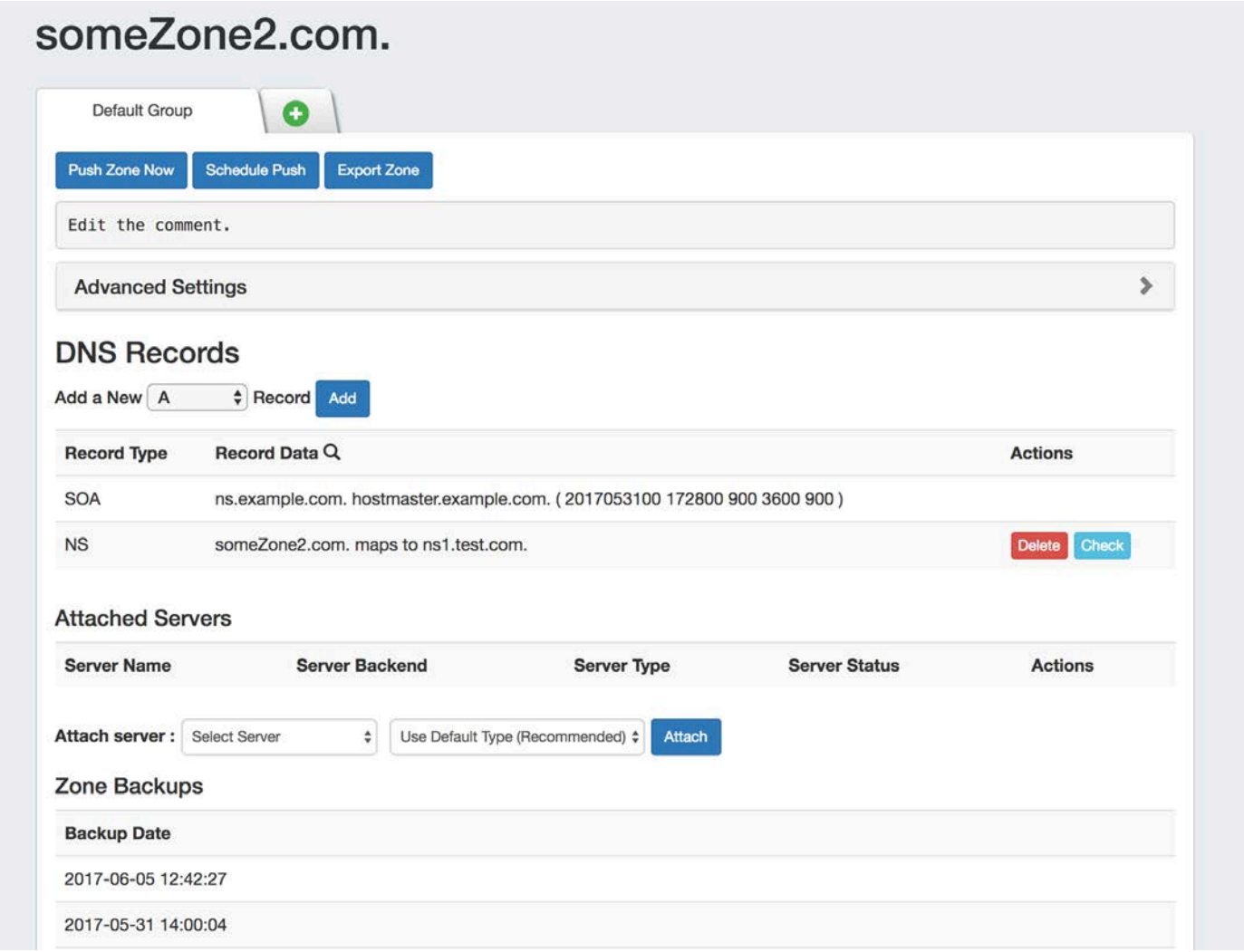
Import DNS Zones

Editing DNS Zones

Editing DNS Zones - The View Zone Page

Zones may be editied by clicking on the Zone name in any zone list in which it appears - zones listed under DNS Groups, under DNS Server attached zones list, or from the DNS Gadget.

Clicking on the zone name opens up the View Zone page, from where comments may be added, records added or updated, servers attached to the zone, advanced settings edited, backups restored or the zone pushed / exported.



Areas of the Zone View page include:

- Editing DNS Zones - The View Zone Page
    - Error Notification
    - Zone Actions
    - Comments
    - Advanced Settings
    - DNS Records
        - Add a Record
        - Edit a Record
        - Delete a Record
        - Check a Record
    - Scheduled Tasks
    - Attached Servers
    - Zone Backups
- Addiitonal Information:

Error Notification

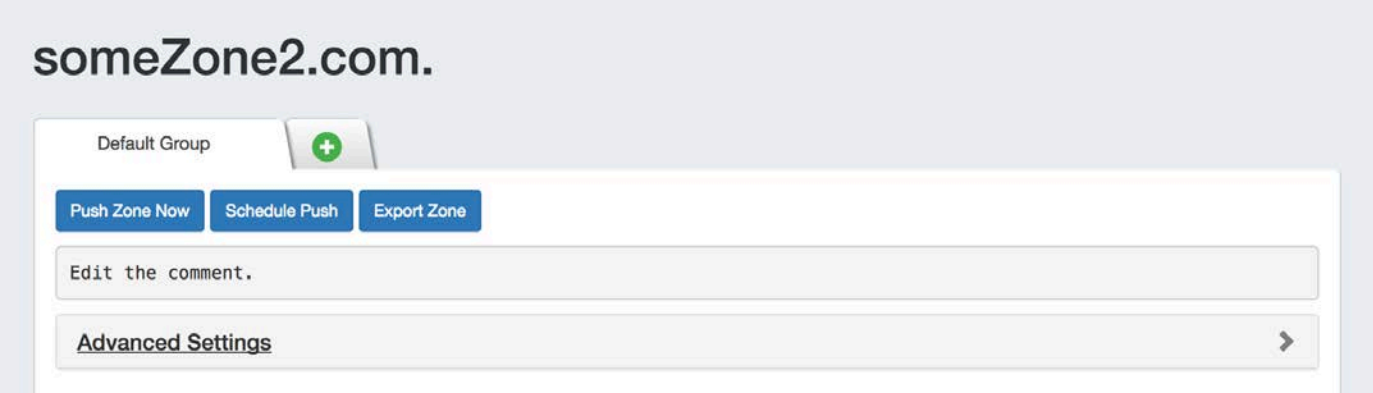
If errors are detected for the zone, an error notice will be given at the top of the group tab for the zone. To review the error, click on the message, and it will list the number of errors present, and details if available.

Records will errors will be highlighted in red in the records list.



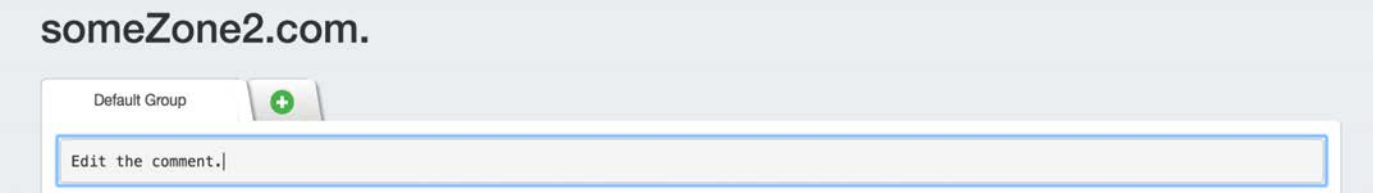
Zone Actions

Zone actions that may be taken include pushing the zone ("Push Zone Now"), scheduling a zone push ("Schedule Push") or exporting the zone to a .zone file ("Export Zone").



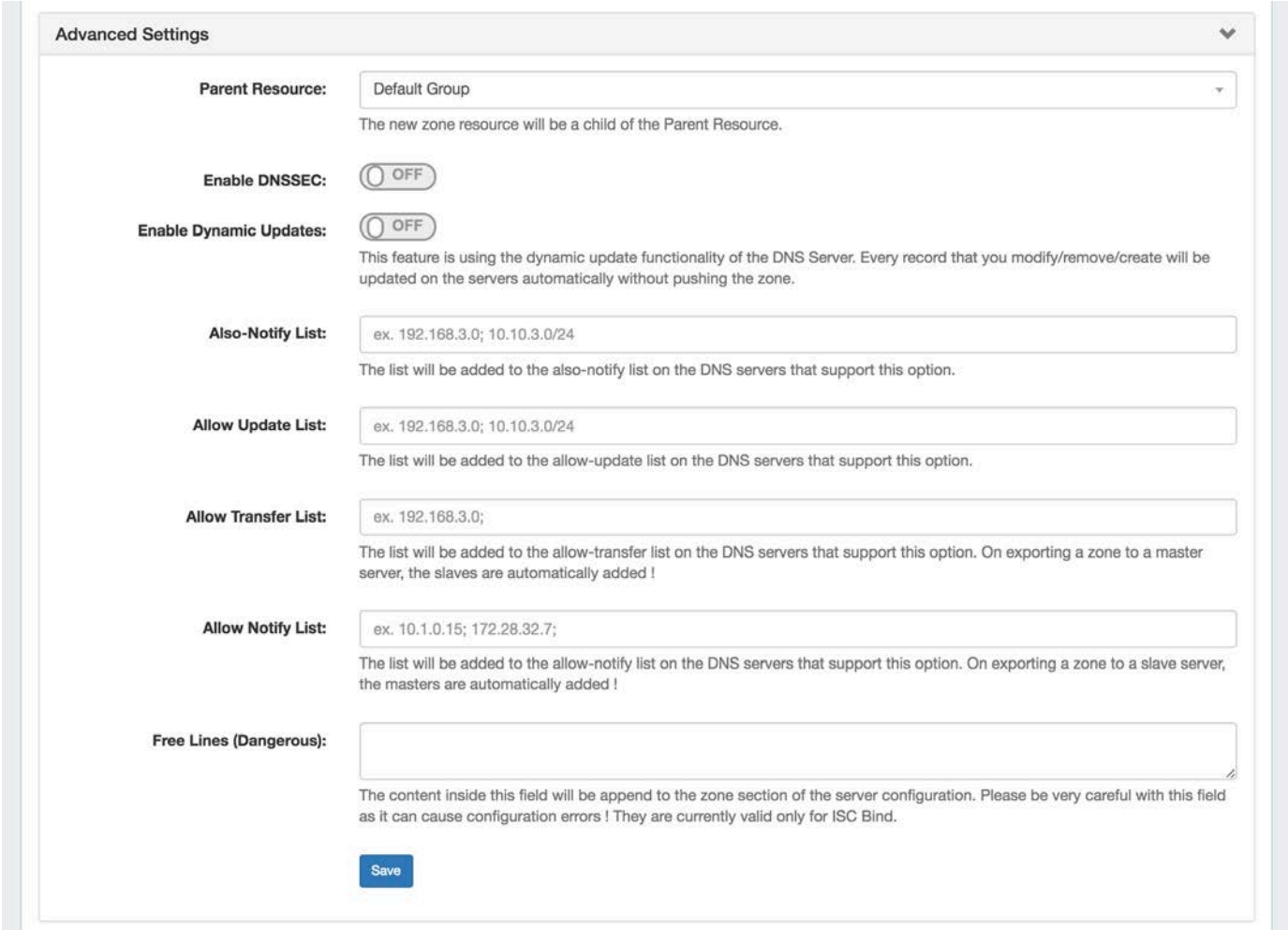
Comments

To add a comment to a zone, click anywhere inside the comment area and begin typing. The comment will be saved when an area outside of the comment box is clicked, deselecting the comment area.



Advanced Settings

Open the zone "Advanced Settings" section by clicking on the header or the expansion arrow (>).



Here, you will find options for:

Parent Resource: Sets the zone's Parent Resource, which affects the permissions access for the zone. By default, the parent is set as the DNS Group the zone is in.

Enable DNSSEC: Toggle "On" or "Off" to enabled DNSSEC for the zone, if supported by the attached server.

Enable Dynamic Updates: Toggle "On" or "Off" to enabled Dynamic Updates for the zone, if supported by the attached server.

Also-notify List: Add IPs to also-notify list of the attached DNS sever, if supported.

Allow Update List: Add IPs to the allow updates list of the attached DNS sever, if supported.

Allow Transfer List: Add IPs to the allow transfer list of the attached DNS sever, if supported. Slaves are automatically added if exporting a zone to a master server.

Allow Notify List: Add IPs to the allow notify list of the attached DNS sever, if supported.

Free Lines: Enter free lines to append to the zone section of the server configuration for the attached server (ISC BIND only). Use with caution - this can cause configuration errors if mistakenly included.

Save Button: Be sure to click "save" when edits are complete, or changes will not be saved.

DNS Records

The DNS Records list contains options to add a new record of the selected type, edit record information, delete a record, or check a record.



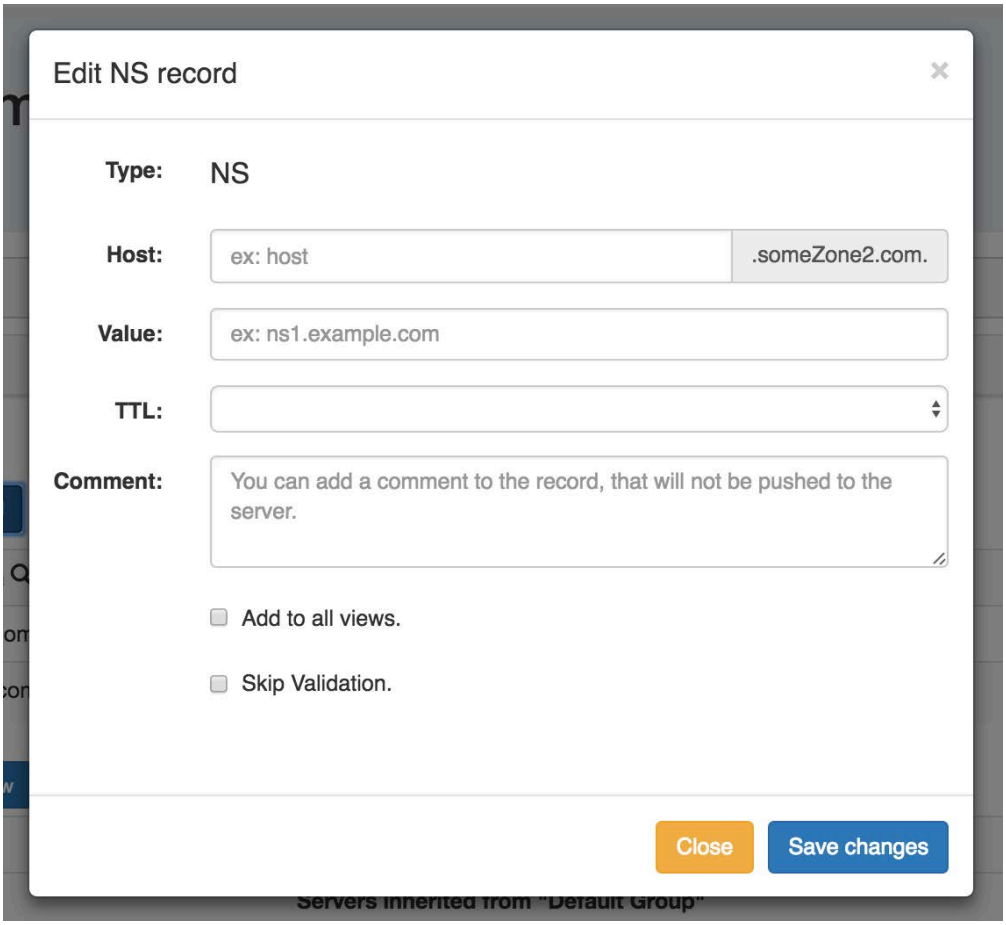
Add a Record

From the View Zone page, under DNS Records, look for the line "Add a new ___ Record".

Select the desired record type, then click "Add".



Fill out the record host, value, select TTL, add a comment (optional), and check "Add to All Views" or "Skip Validation" if desired.

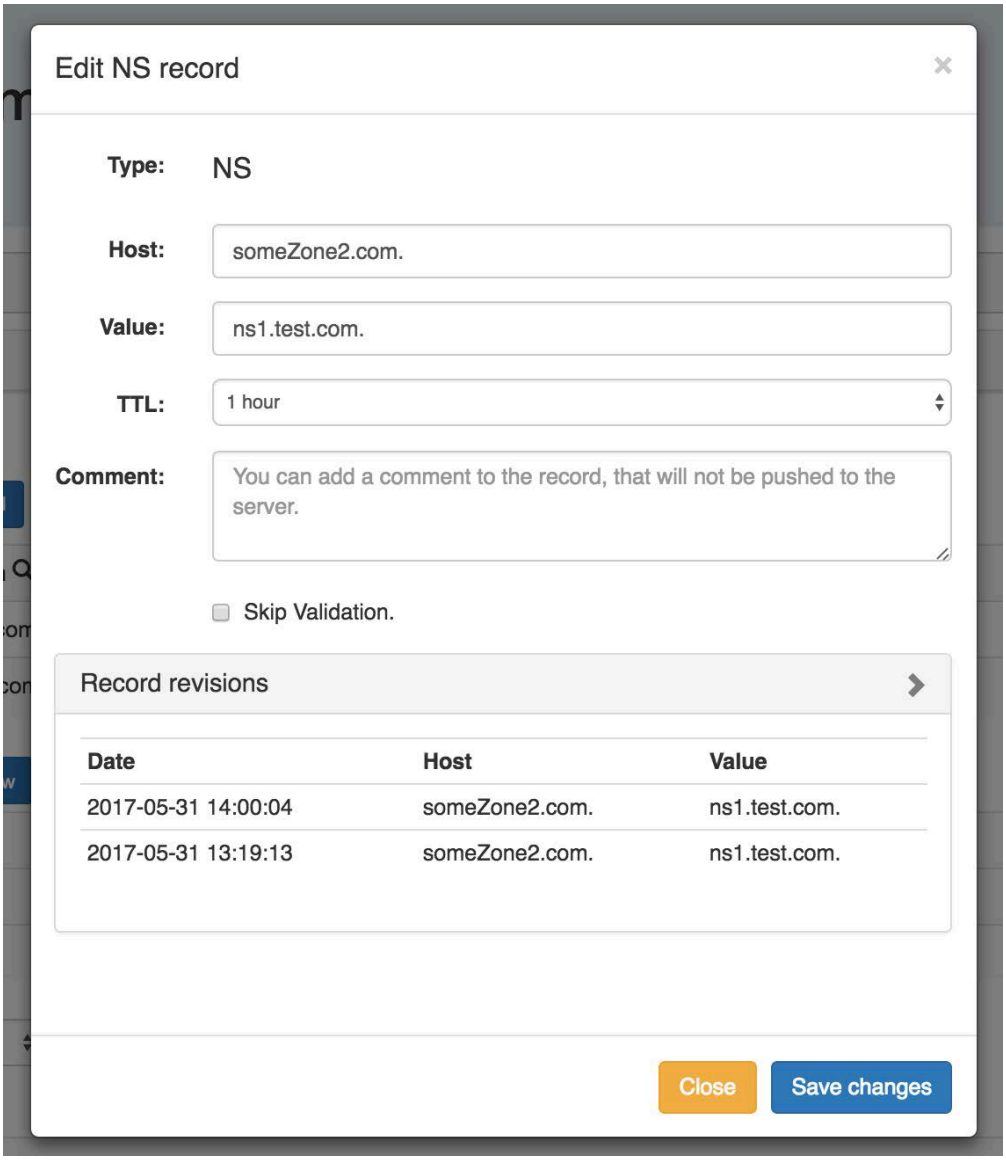


When complete, click "Save changes".

Edit a Record

To edit an existing zone record, click on the record line in the DNS Records list.

The Edit Record dialog will open, allowing you to fill in record details.



Fill in the Host, Value, TTL, and comment if desired.

An option exists to "Skip Validation" - check to select if you wish to bypass validation for the record.

If previous versions of the record exist, the revision log will be visible under "Record revisions".

When edits or reviewing is complete, click "Save changes", or "Close" to exit without saving.

Delete a Record

Delete a zone record by clicking the "Delete" button under "Actions" for the desired record entry.

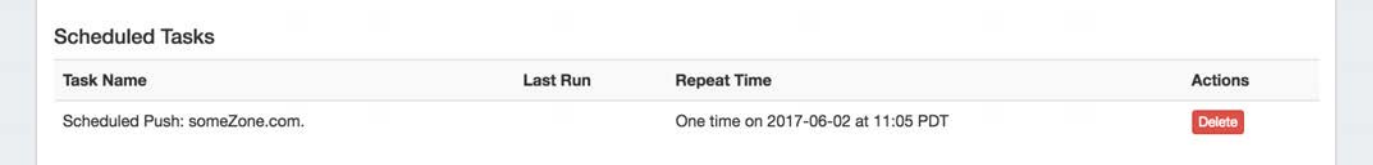
Check a Record

Check a record for errors by clicking the "Check" button under "Actions" for the desired record entry.

Scheduled Tasks

If scheduled tasks (zone pushes) have been set up for the zone, they will appear under a "Scheduled Tasks" section.

Tasks here will also show in the Admin Scheduler task list.



The task may be deleted by clicking the "delete" button.

Attached Servers

Servers attached directly to the zone, or inherited from the zone's parent DNS Group will be shown here.

Additional servers may be attached by selecting the server name, type (default, master, slave), and clicking the "Attach" button, as long as the user has permissions on both the zone and the server to attach.



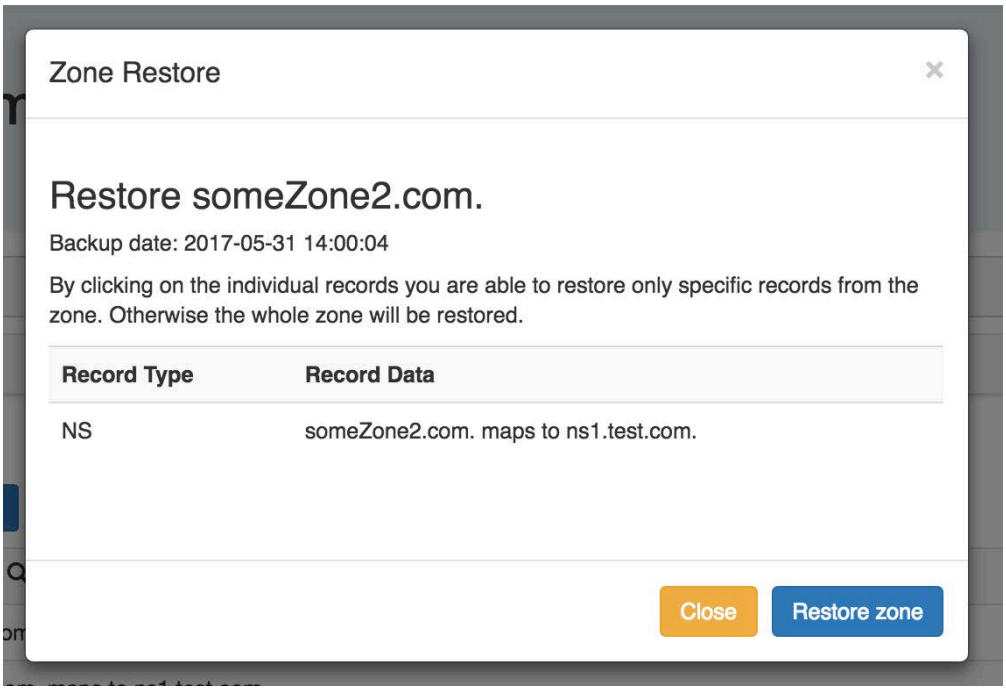
To detach a server from a zone, look for the server under the "Attached Servers" list, then click "Detach" under actions for that server.

Zone Backups

If a zone has had changes sucessfully pushed, a "Zone Backups" section appears near the bottom of the page, showing the date and time of the backup.



To view details, or restore the zone to the previous version, click on the row of the desired date/time backup, and the "Zone Restore" module will open.



From here, you may restore individual records by clicking on them, or restore the entire zone by clicking the "Restore Zone" button.

To exit without saving, click the "Close" button.

Addiitonal Information:

For more information on working with DNS Zones, Groups, and Servers, see the following sections:

Working with DNS Groups

Working with DNS Zones - Common Tasks

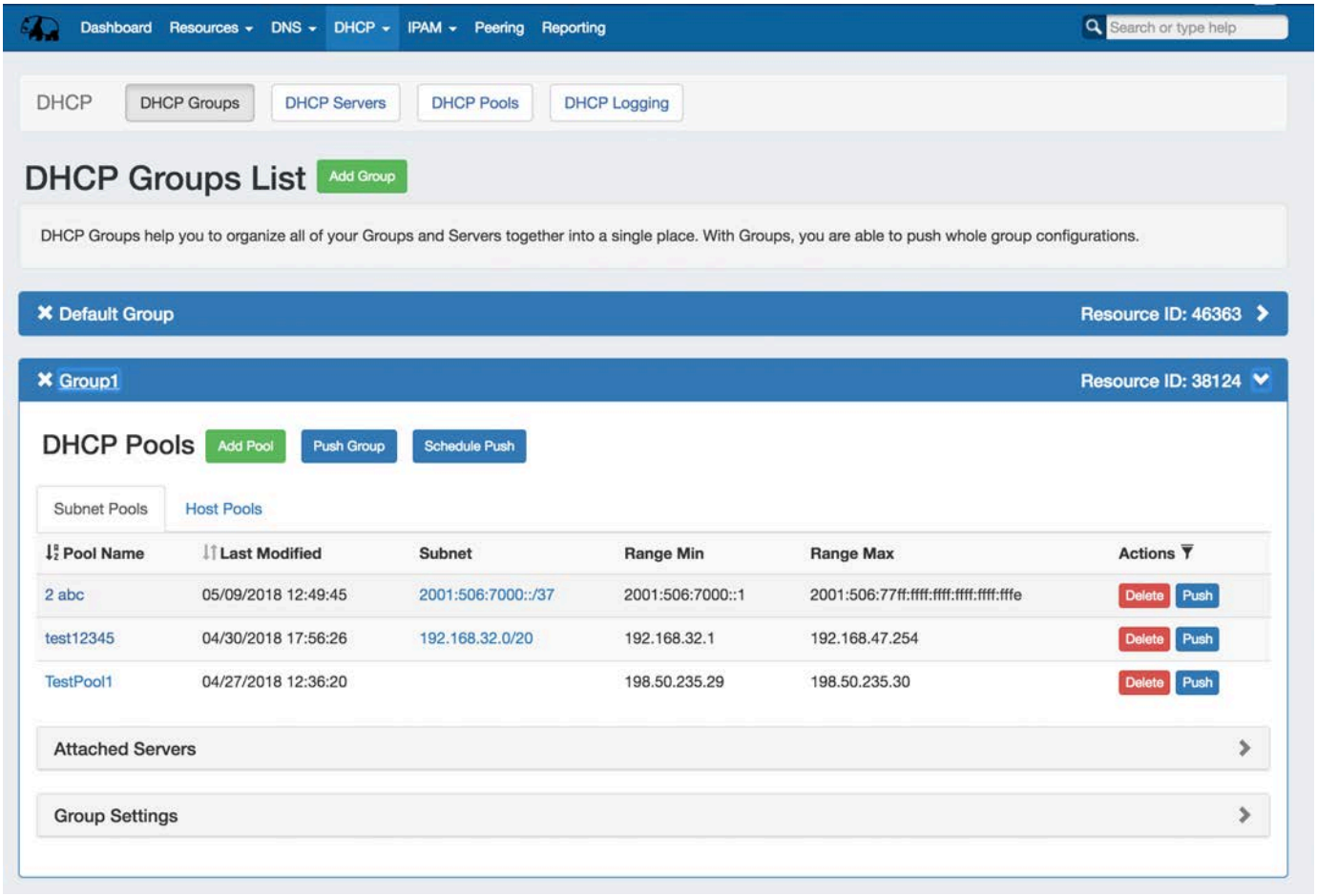
DNS Administration

Import DNS Zones

DHCP Tab

The DHCP Tab

ProVision's **DHCP** Tab combines server management, group organization, and pool management under one tab.



The **DHCP** tab contains four sub-tabs: **DHCP Groups, DHCP Servers, DHCP Pools, and DCHP Logging**.

The **DHCP Groups** tab is where DHCP servers and pools are managed as a part of DHCP Groups. The **DHCP** tab **DHCP Groups List** allows you to create, view, and manage groups of DHCP servers and pools. Using groups, you can configure and push selected combinations of pools at one time.

The **DHCP Servers** tab is only accessible to Admin users, and contains functions for adding, updating, and managing DHCP servers as well as scheduling server tasks. For Admin-level DHCP tasks, see Working with DHCP Servers.

The **DHCP Pools** tab lists all the pools created in ProVision able to be viewed by the user, in the same format as on the **DHCP Groups** page, but without the Group organization. The Pool List allows you to view, add, delete, and push individual DHCP Host and Subnet Pools.

The **DHCP Logging** tab provides a link to ProVision's logging system, where you can filter to view DHCP-only logs.

- The DHCP Tab
  - Permissions
  - Additional Information

Permissions

DHCP Management integrates with ProVision's resource and permissions hierarchy, as well as the IP Management system. Individual DHCP servers can be assigned via Resource Permissions to different internal user groups, to be managed by only the appropriate parties.

Additional Information

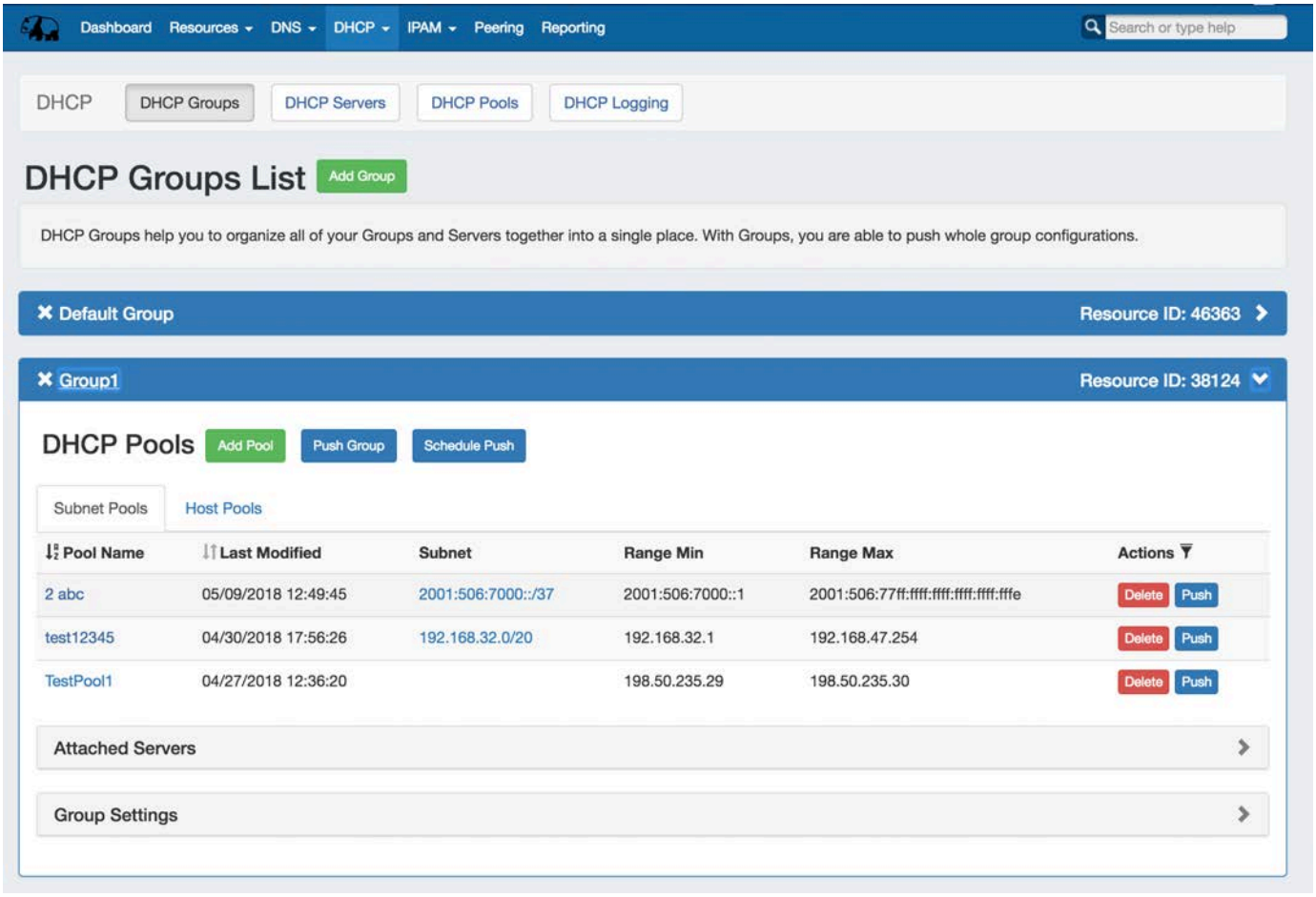
See the following sections for detail on working within the DHCP Tab:

- Working with DHCP Groups
- Working with DHCP Pools
- Copy of DHCP Tab

For more detail on advanced DHCP topics and the DHCP API, see API Module - DHCP.

Working with DHCP Groups

DHCP Groups

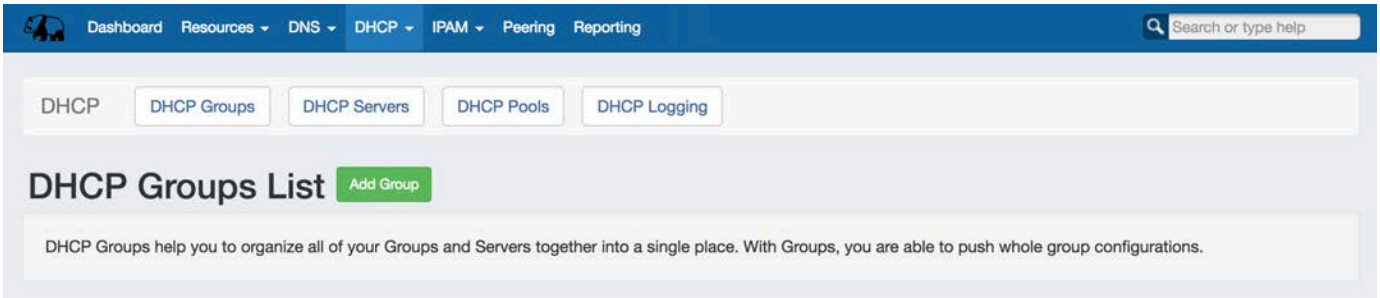


The DHCP tab **DHCP Groups List** allows you to create, view, and manage groups of DHCP Servers and Pools. Using groups, you can configure and push selected combinations of zones at one time.

- DHCP Groups
  - DHCP Groups List Interface
- Working with DHCP Groups
  - Add a Group
  - Edit a Group
    - Edit a Group Name:
    - Edit a Group's Settings:
      - Edit Parent Resource
      - Edit Failover Name
  - Add Pools to a Group
    - Add a Subnet Pool
      - Smart Assign
      - Direct Assign
    - Add a Host Pool
      - Smart Assign
      - Direct Assign
  - Pushing a DHCP Group
    - Manual Group Push
    - Scheduled Group Push
  - Delete a Group
  - Additional Information

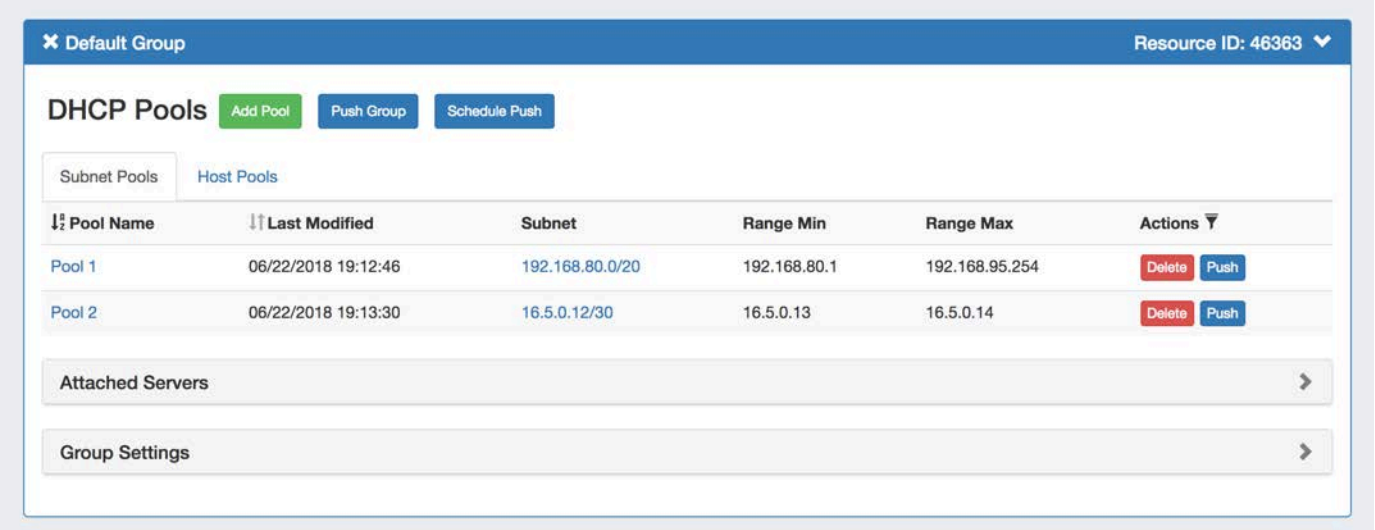
DHCP Groups List Interface

Navigation: Move between the DHCP Groups, DHCP Servers, DHCP Pools, and DHCP Logging pages via the subtab buttons below the main navigation bar.



1) Add Group Button: Opens a dialog for creating a DHCP Group

2) Group Container: The Group container header shows the group name. Expand or close the group container by clicking on the expansion arrow (>) on the far right of the header. Clicking on the X will delete the Group and all zones included in the group. Rename the group by clicking on the name, then typing the desired changes and clicking outside of the name.



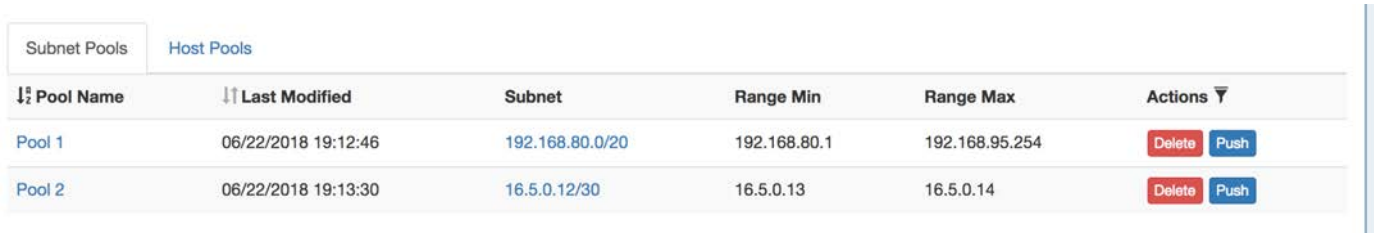
Group Pool Actions:

3) Add Pool Button: Opens a dialog for creating a DHCP Pool under that group.

4) Push Group Button: Pushes all pools in the group to the associated DHCP server.

5) Schedule Push Button: Schedules a push of all pools in the group to the selected day and time. Requires the scheduler tab / cron tasks to be enabled.

6) Pool List: Subnet and Host Pools included in the Group.



7) Subnet / Host Pools Tabs: Shows zone list containing only forward or reverse zones, respectively.

8) Pool Name (Sortable): The name of the pool. Clicking on the pool name will open the pool details page. Clicking on the "Pool Name" header will sort the zone list by ascending or descending by name.

9) Last Modified (Sortable): The date and time the pool was last modified. Clicking on the "Last Modified" header will sort the list by most / least recent modification date.

10) Subnet (or IP): The subnet or IP of the subnet/host pool, respectively.

11) Range Min (or MAC): The starting range address for subnet pools, or the MAC for host pools.

12) Range Max: For the subnet list, the ending range address for the pool.

13) Actions: Clicking on the "Actions" header will reveal sortable filters for selected fields. Buttons under the Actions column indicate actions that may be performed on each pool:

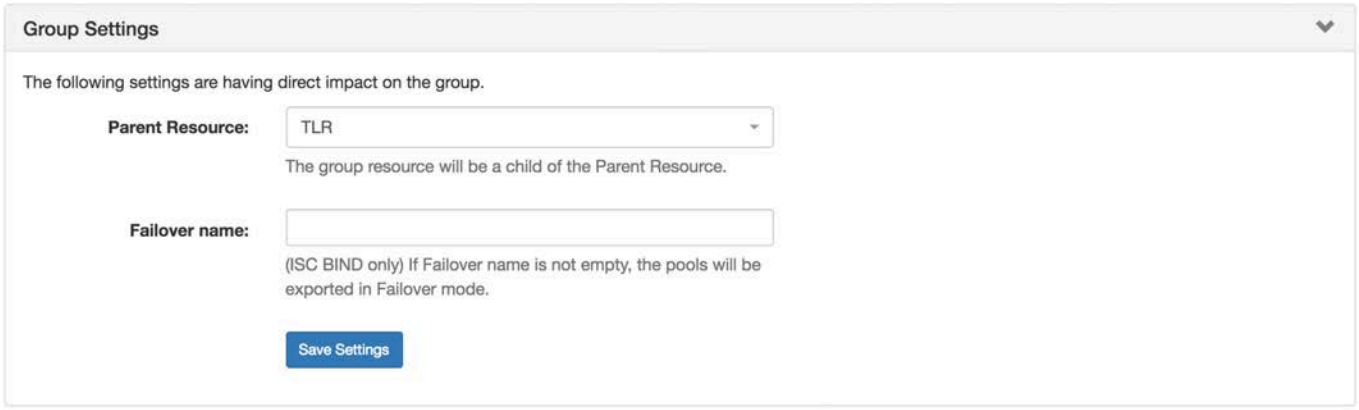
14) Delete: Deletes the pool from ProVision.

15) Push: Pushes the pool to the associated server.

16) Attached Servers: Click on the header title or > to expand or collapse. View, attach, or detach a DHCP server from the group.

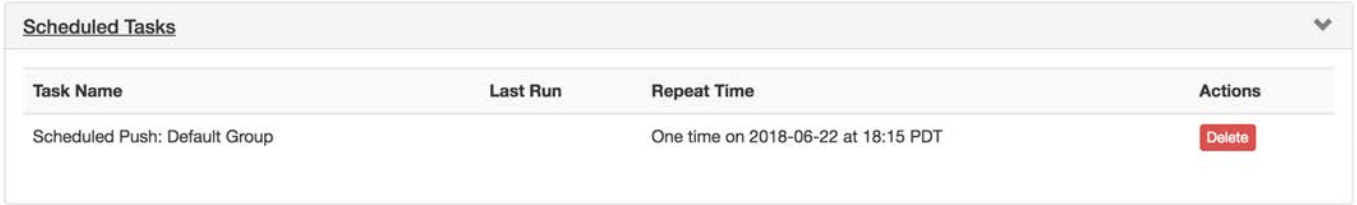


17) Group Settings: Click on the header title or > to expand or collapse. Edit default group settings including the Parent Resource and Failover name.



18) Save Settings Button: Saves any changes made to the DHCP Group Settings options.

22) Scheduled Tasks: If a scheduled push has been set up for the group, the task information will show in a "Scheduled Tasks" module. Click on the header title or > to expand or collapse.



From here, you may review or delete scheduled group pushes.

Working with DHCP Groups

DHCP Groups allow for organizing multiple pools together and linking those pools with one or more servers. Pools may be pushed individually, as a Group, or at the server level.

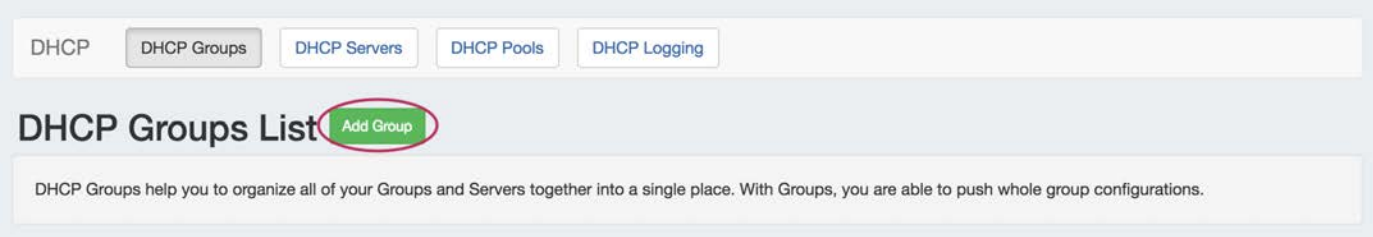
ProVision automatically designates a Default Group for pools to be imported or added into. Creating additional Groups is completely optional.

Add a Group

Create DHCP Groups by clicking the "Add Group" Button, then add subnet or host pools under the group by clicking "Add Pool" for the desired Group.

Pools may be pushed individually, as a part of a DHCP Group, or through pushing an entire DHCP Server.

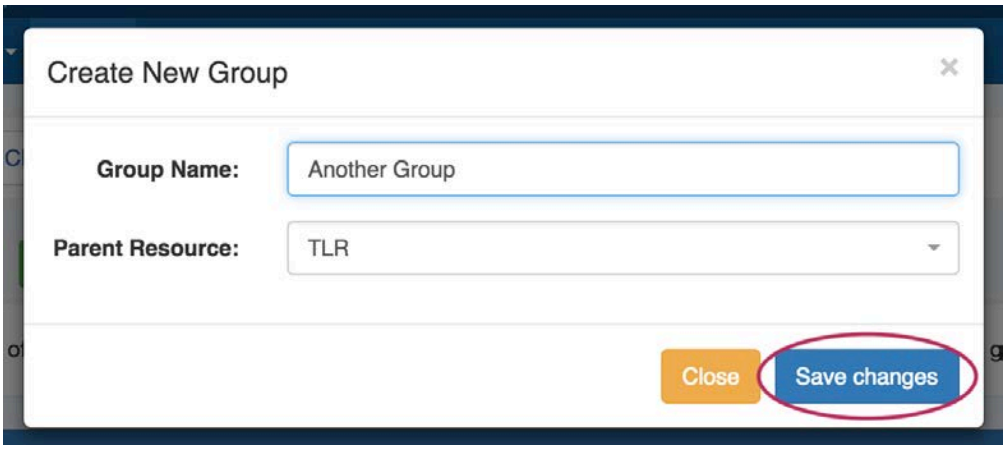
To create a new Group, from the **DHCP** Tab, select the **DHCP Groups** sub menu. Then, click the "Add Group" button next to "DNS Groups List".



This will open the "Create New Group" dialog.

Enter the desired Group Name and select a parent resource.

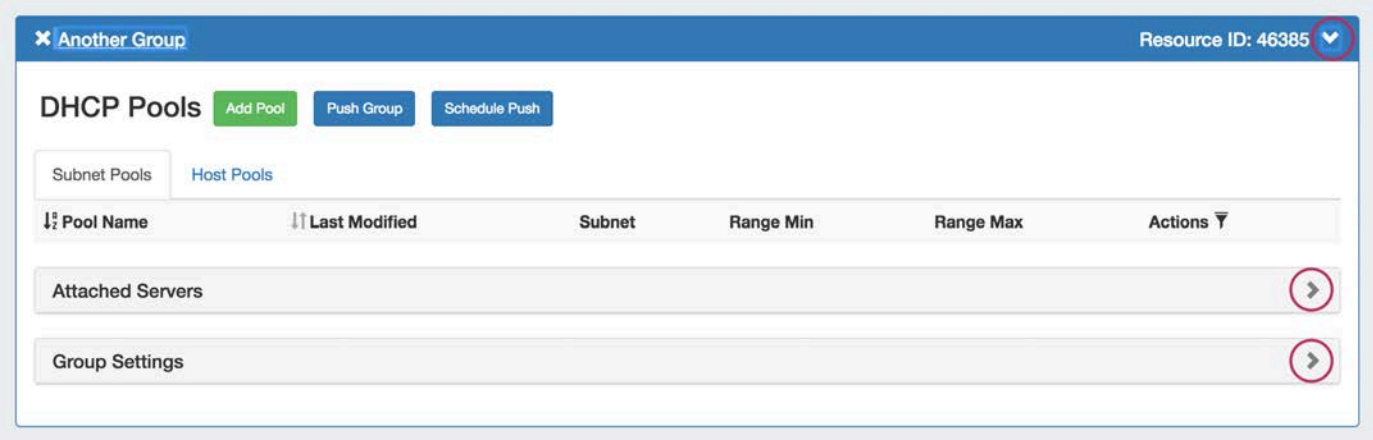
Click the "Save Changes" button when complete.



The new group will be added to the DHCP Groups List.

Open the Group container for the new Group by clicking on the expansion arrow at the top right corner of the header.

Information for Attached Servers and Group Settings may also be viewed and edited by clicking on the expansion arrows for each section.



Edit a Group

Edit a Group Name:

Edit the name for a Group by clicking inside the Group's Name. A highlight box will appear around the name, and a cursor inside the box.



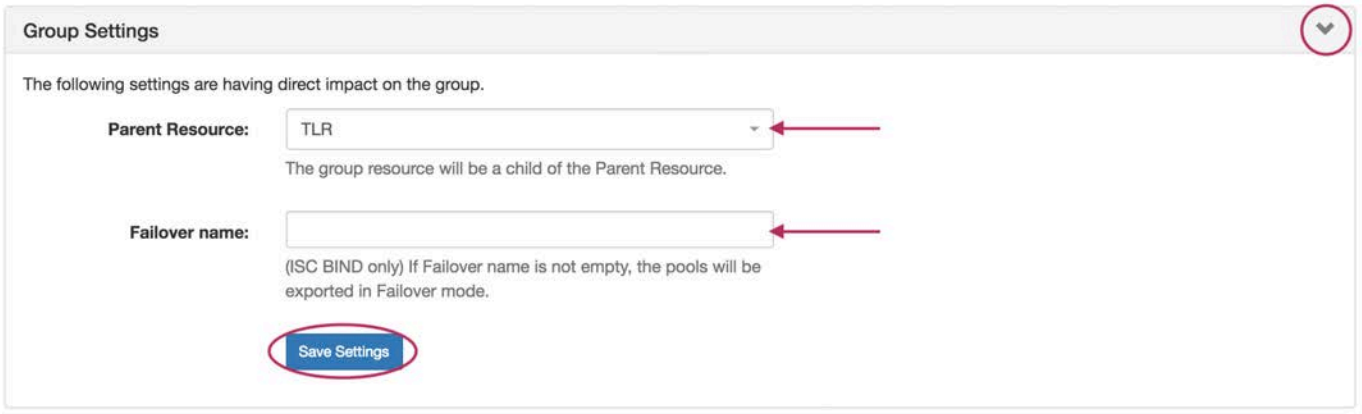
Edit the text as desired using your mouse or arrow keys to navigate through the text. When complete, click anywhere outside of the name box, and a confirmation dialog will appear.



Click "Confirm" to save your changes, or "Close" to exit without saving.

Edit a Group's Settings:

Under the desired Group, expand the "Group Settings" module by clicking on the expansion arrow in the upper right corner of the header.



Edit Parent Resource

Change the parent resource for the DHCP Group by searching and selecting the desired resource from the dropdown box.

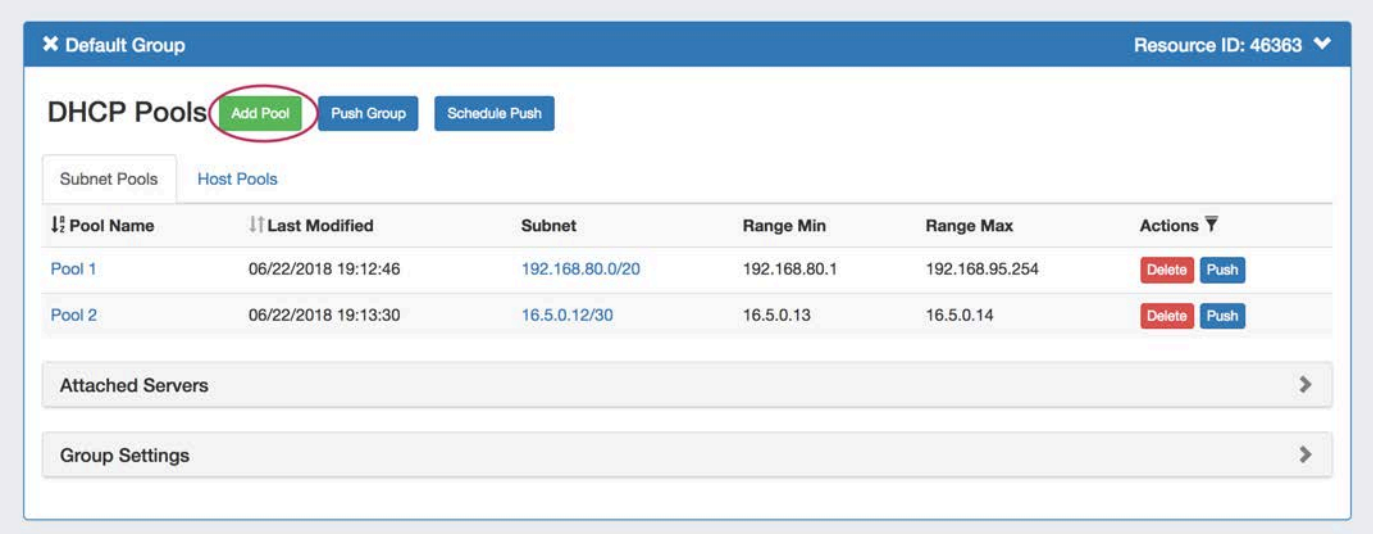
Edit Failover Name

Edit the Failover name by typing the name directly in the Failover Name input. If the Failover name is not empty, the pools will be exported in Failover mode. NOTE: This is for ISC BIND only.

When done with your edits, be sure to click the "Save Settings" button.

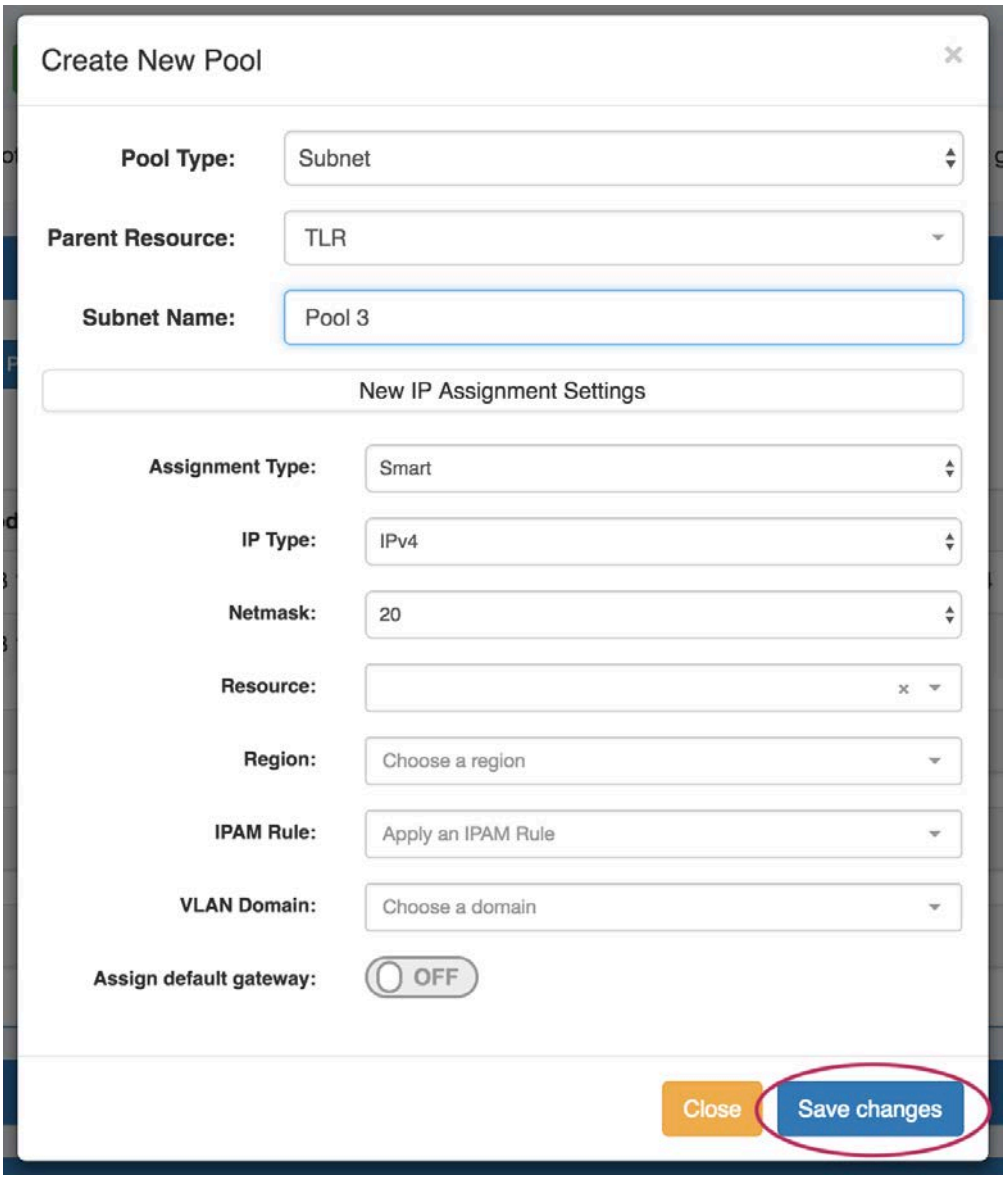
Add Pools to a Group

To add a new pool to a DHCP Group, click the "Add Pool" button inside the desired Group.



Add a Subnet Pool

To add a new subnet pool to a DHCP Group, select "Subnet" as the Pool Type. Select the parent resource and type in a Subnet Name.



Next, choose an assignment type - you may either have ProVision smart assign a block for the pool that matches your provided criteria, or directly assign a specific block.

Smart Assign

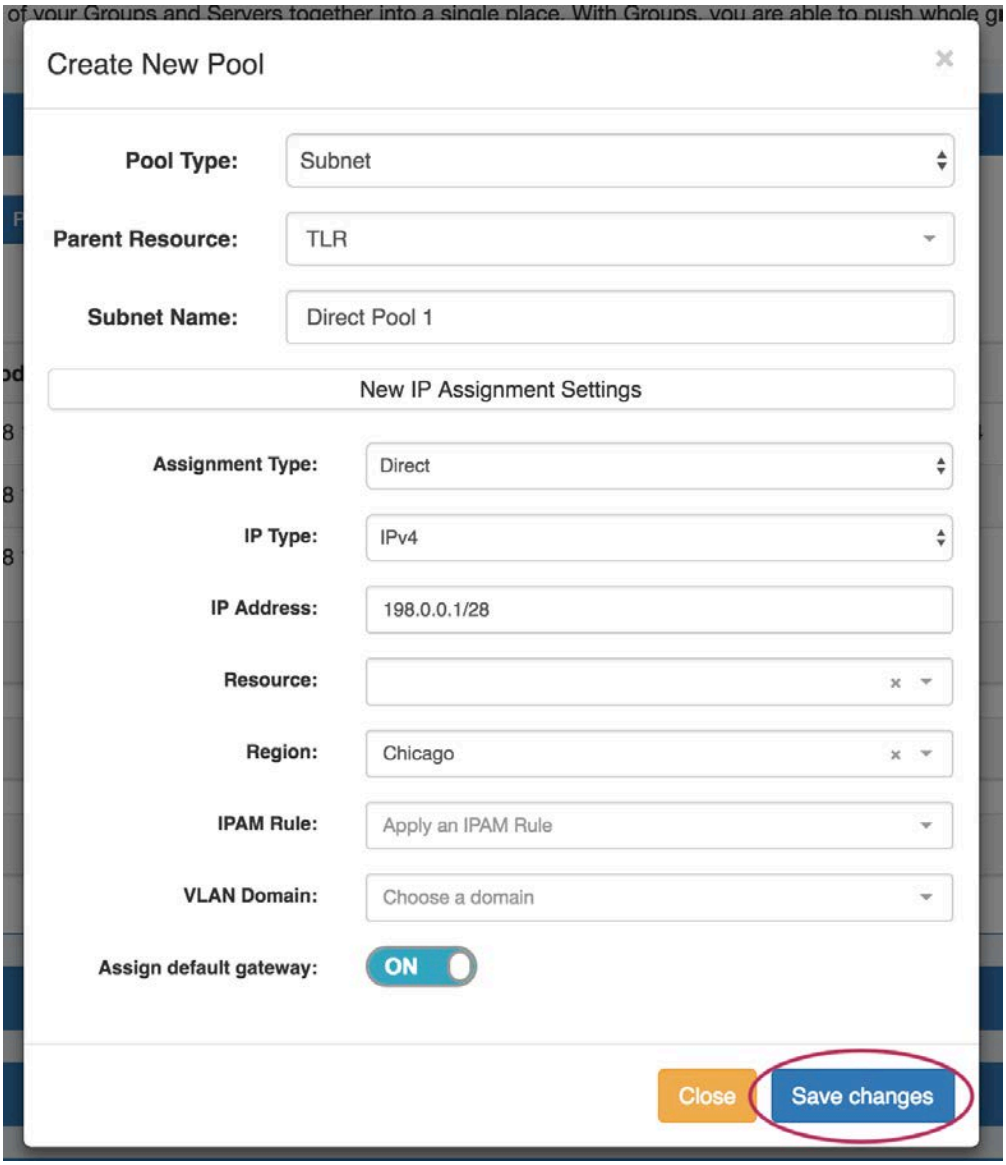
To Smart Assign a DHCP pool, select "Smart" for the Assignment type, choose IPv4 or IPv6 for the IP type, then select a netmask.

All other fields are optional - Resource, Region, IPAM Rules, and VLAN may be selected to further narrow the block criteria. If desired, toggle "Assign Default Gateway" on.

When done, click "Save Changes". ProVision will choose the first block matching all criteria to assign to the pool.

If no block matches the given criteria, a message will state so. Try either removing some criteria, or ensure that blocks exist in IPAM (with subassignments enabled!) with the given attributes.

Direct Assign



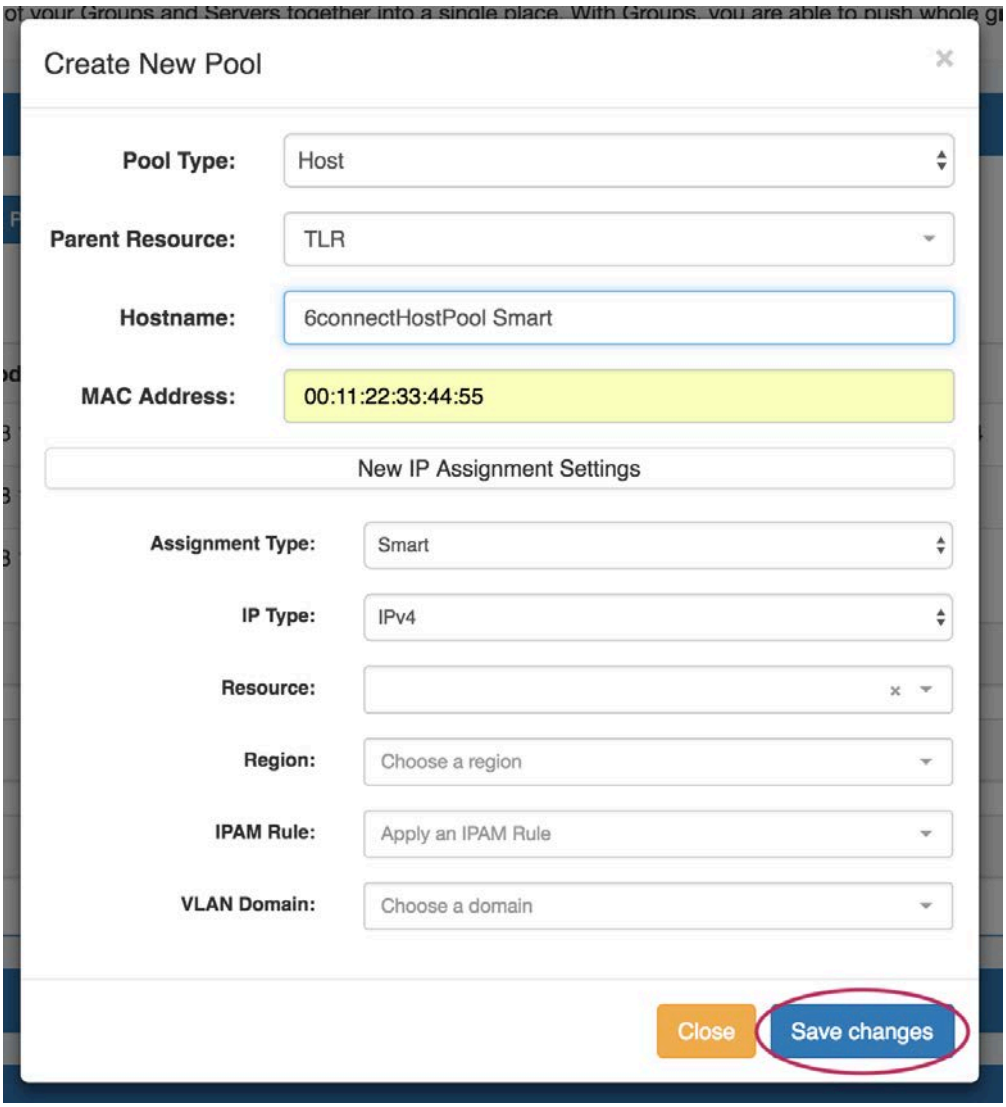
To Direct Assign a DHCP pool, select "Direct" for the Assignment type, choose IPv4 or IPv6 for the IP type, then type in the desired IP Address.

All other fields are optional - Resource, Region, IPAM Rules, and VLAN may be selected to further narrow the block criteria, if duplicate blocks exist. If desired, toggle "Assign Default Gateway" on.

When done, click "Save Changes". If the typed block exists in ProVision with subassignments enabled, and meets any additional criteria, that block will be assigned to the pool.

Add a Host Pool

To add a new host pool to a DHCP Group, select "Host" as the Pool Type. Select the parent resource, and add a Hostname and MAC Address.



Next, choose an assignment type - you may either have ProVision smart assign a block for the pool that matches your provided criteria, or directly assign a specific block.

Smart Assign

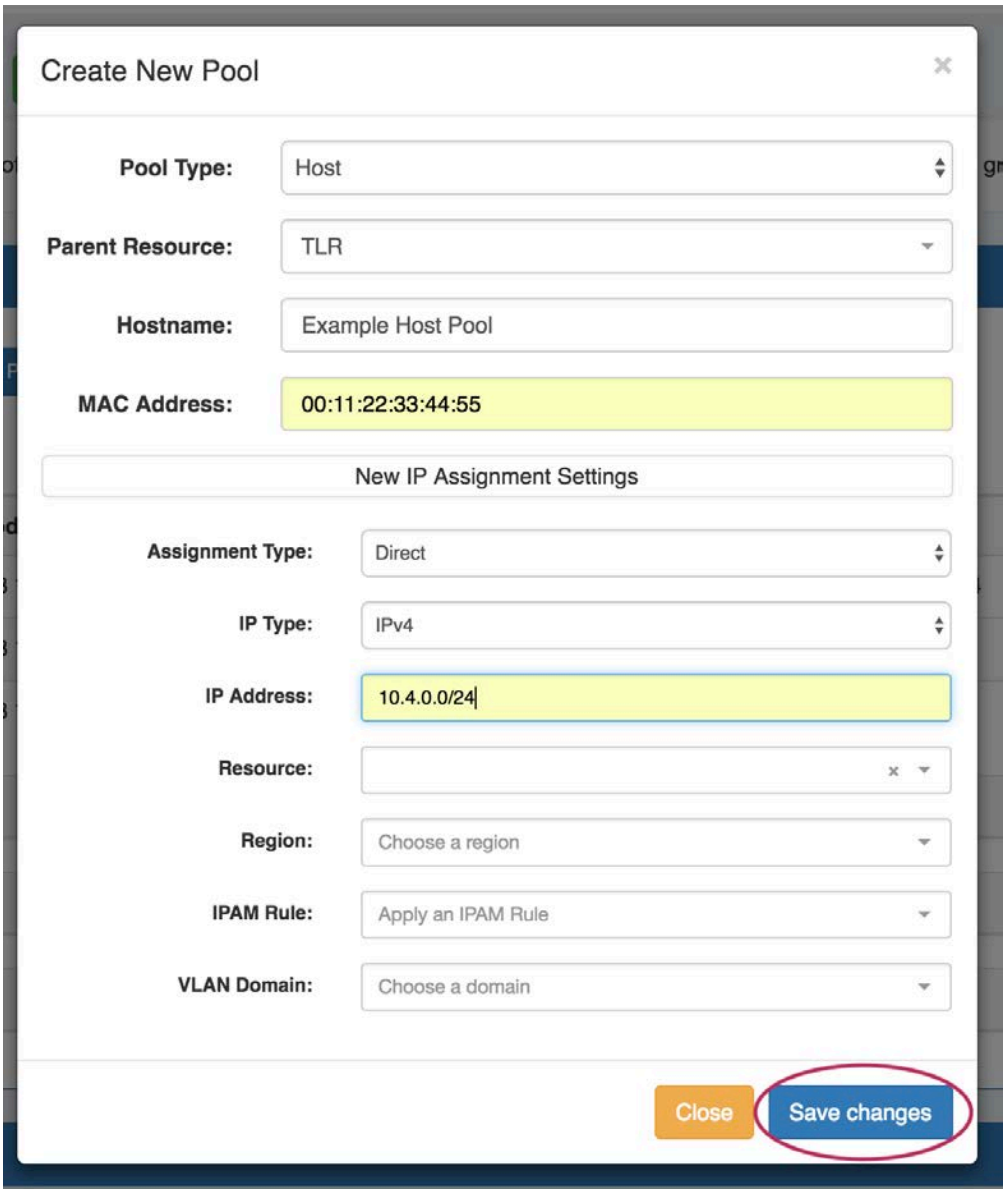
To Smart Assign a DHCP pool, select "Smart" for the Assignment type and choose IPv4 or IPv6 for the IP type.

All other fields are optional - Resource, Region, IPAM Rules, and VLAN may be selected to further narrow the block criteria.

When done, click "Save Changes". ProVision will choose the first block matching all criteria to assign to the pool.

If no block matches the given criteria, a message will state so. Try either removing some criteria, or ensure that blocks exist in IPAM (with subassignments enabled!) with the given attributes.

Direct Assign



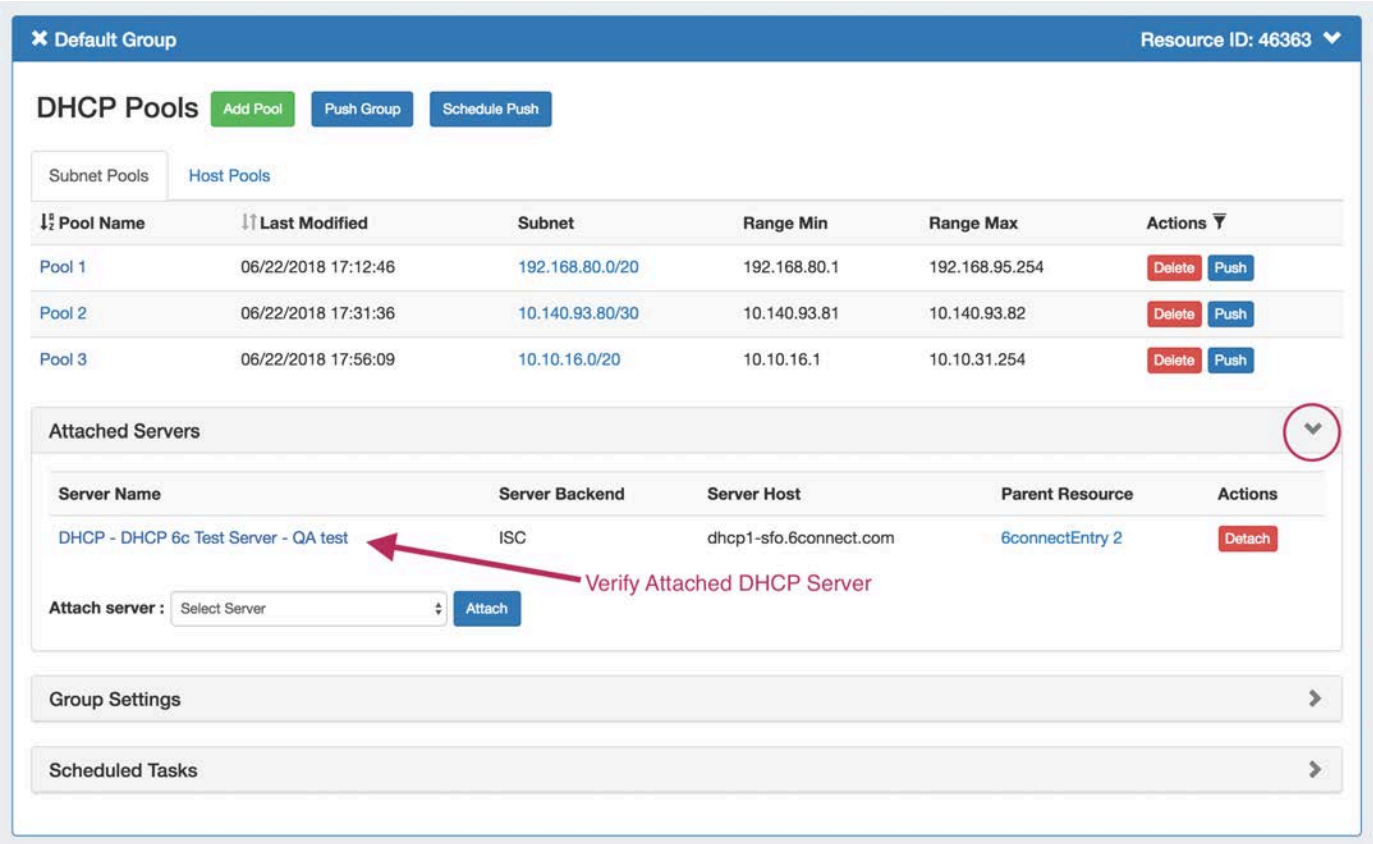
To Direct Assign a DHCP pool, select "Direct" for the Assignment type, choose IPv4 or IPv6 for the IP type, then type in the desired IP Address.

All other fields are optional - Resource, Region, IPAM Rules, and VLAN may be selected to further narrow the block criteria, if duplicate blocks exist.

When done, click "Save Changes". If the typed block exists in ProVision with subassignments enabled, and meets any additional criteria, that block will be assigned to the pool.

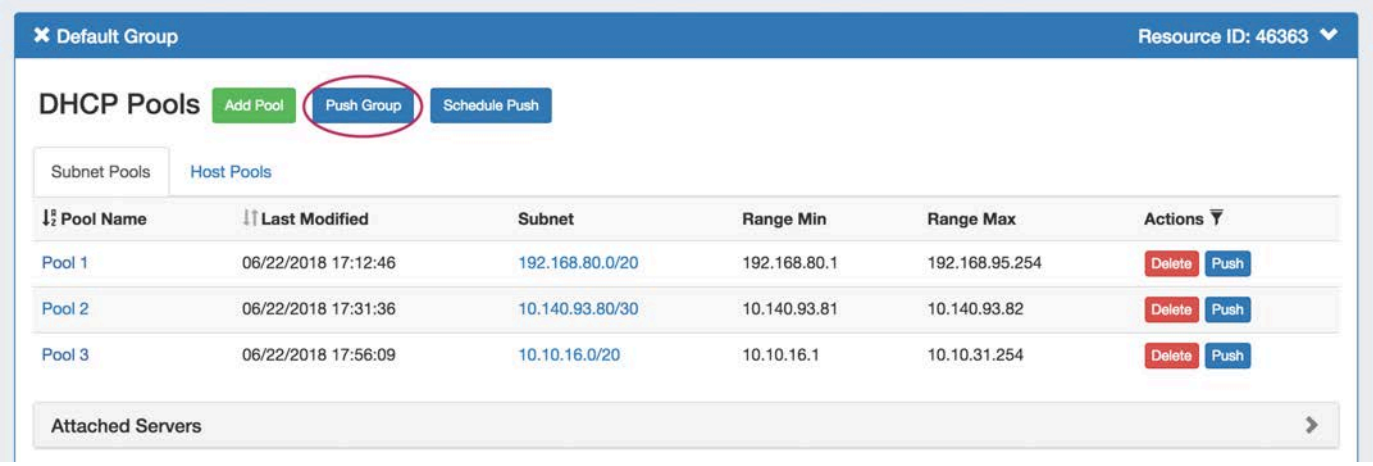
Pushing a DHCP Group

Before pushing a group, expand the "Attached Servers" section and verify that the correct server is attached to the DHCP Group.



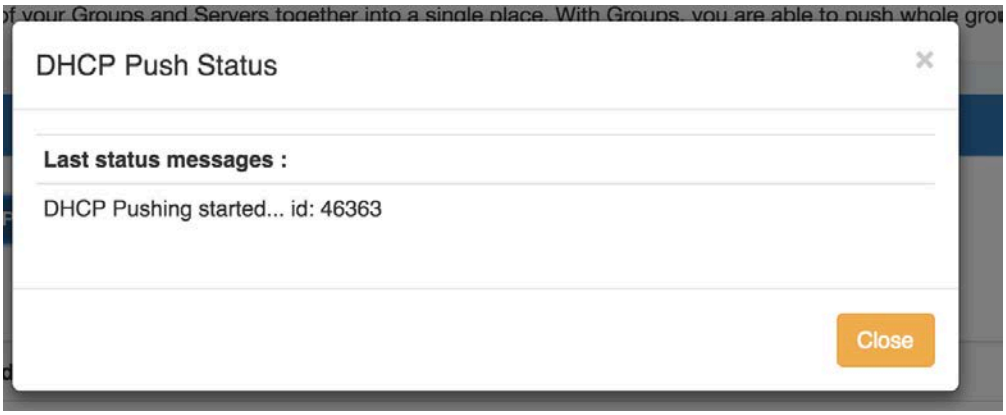
Manual Group Push

After verifying the attached server, manually push the Group by clicking the "Push Group" button under the desired DHCP Group.



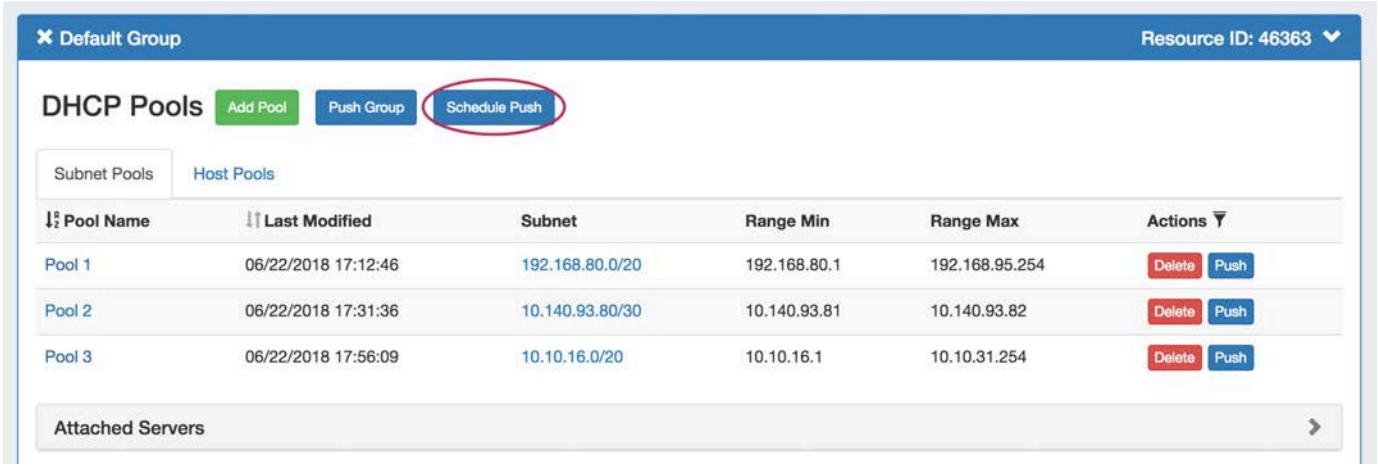
A DHCP Push Status window appears. If successful, a green "Finished DHCP Pushing Request" will appear as the final message.

If errors occur, an error message will show with details on the issue. When complete, hit the "Close" button.

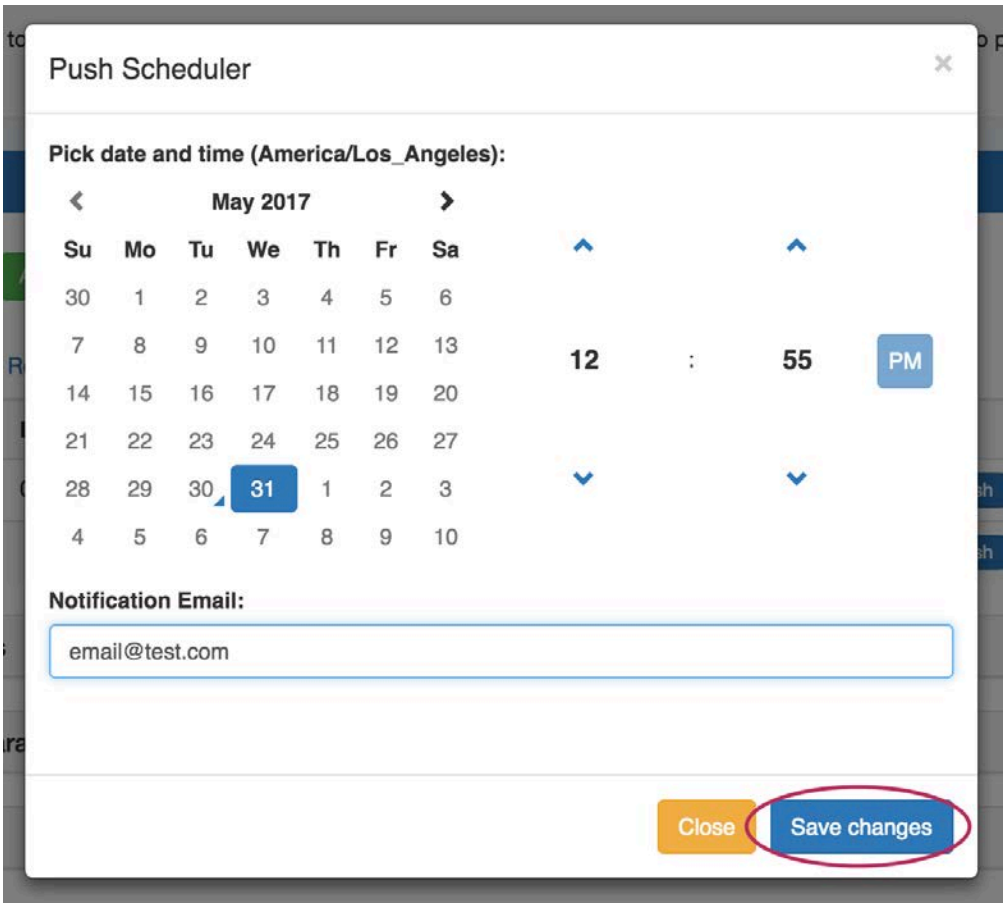


Scheduled Group Push

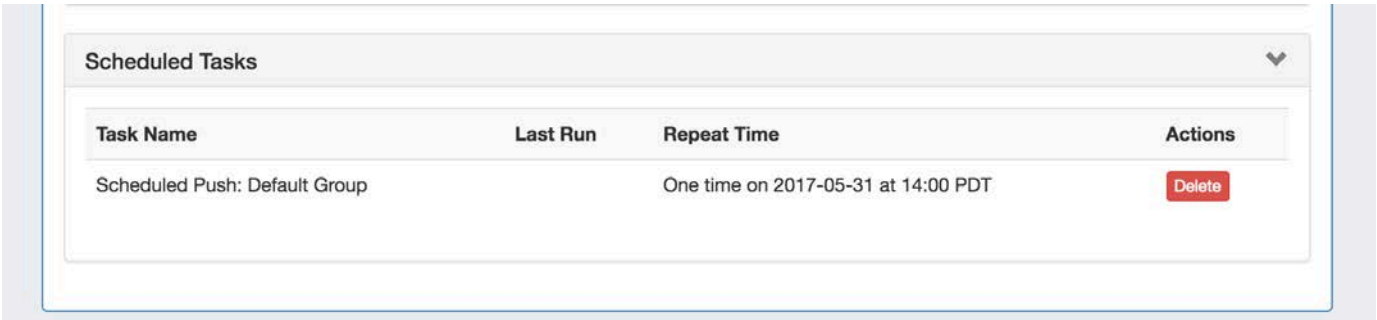
Scheduled pushes may be set up through the "Schedule Push" button for the Group.



Click "Schedule Push" for the Group, then select a date, time (12 hour, with AM / PM toggle), add a notification email address, and click "Save changes".



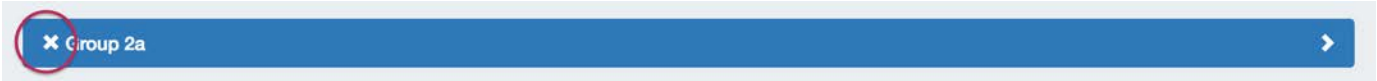
The Scheduled task will be shown in both the Group details (under "Scheduled Tasks") and the Admin Scheduler task list.



The task may be deleted prior to running by clicking the "delete" button, but will automatically be removed once completed.

Delete a Group

Delete a non-Default Group by clicking on the "X" to the left of the Group's name in the DHCP Groups List.



A confirmation message will appear, click "Confirm" to save your changes, or "Close" to exit without saving.



The Default Group may not be deleted.

Additional Information

For more information on DHCP tasks, see:

DHCP Tab

Working with DHCP Pools

Copy of DHCP Tab

Administrative Tasks:

DHCP Administration

Working with DHCP Servers

Working with DHCP Pools

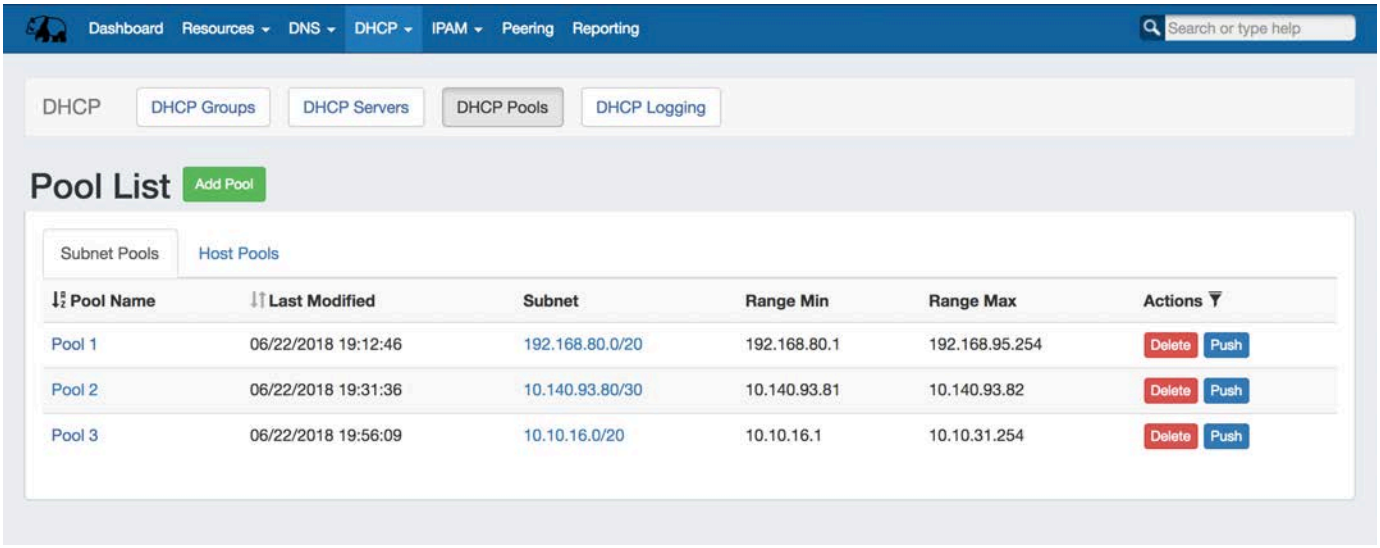
DHCP Pool Management

DHCP Pool specific tasks are managed from two main areas in ProVision - the **DHCP Pools** list, and the **Pool Details** page.

The DHCP Pools List is accessed from the **DHCP** tab, then clicking on the **DHCP Pools** subtab button.

Clicking on the Pool name from any DHCP Pool List will open the **Pool Details** page for the specific pool.

DHCP Pools List



The **DHCP Pools** list provides the same pool-level functionality as the DHCP Groups pool lists, but includes all pools regardless of Group attachment.

From this page, you may create, view, edit, and push subnet or host pools.

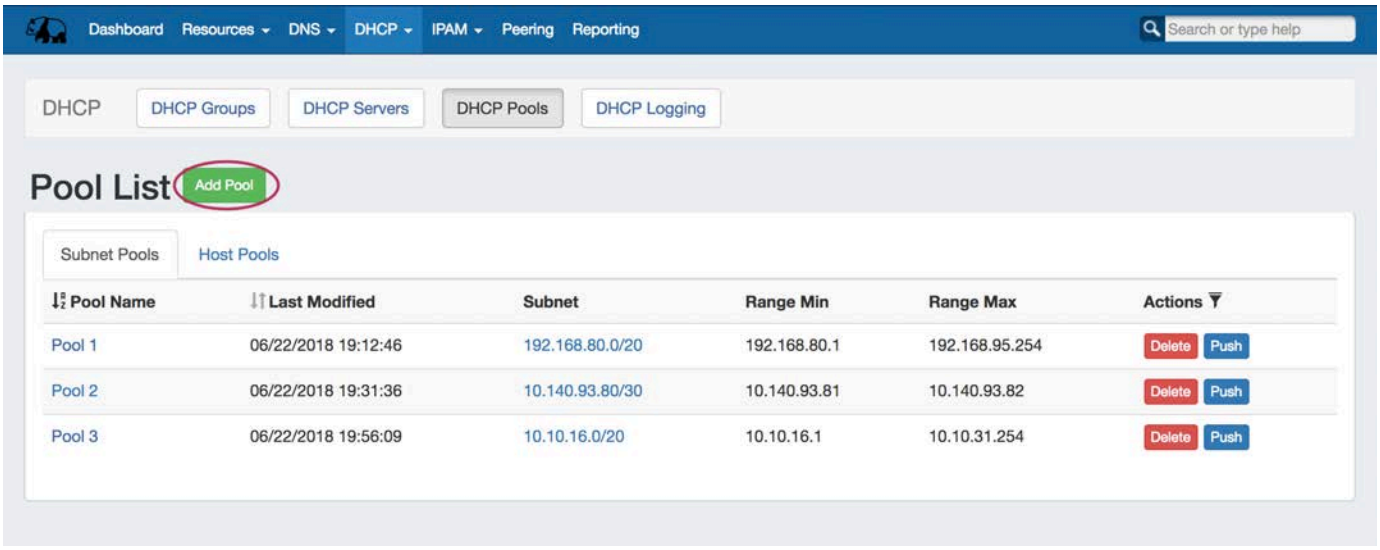
Clicking on the pool name will open the Pool Details page, and clicking on the Subnet will open IPAM Mange for the assigned pool block.

- DHCP Pool Management
- DHCP Pools List
  - Add a DHCP Pool
    - Create Subnet Pool
      - Smart Assign
      - Direct Assign
    - Create a Host Pool
      - Smart Assign
      - Direct Assign
  - Edit / Manage a Pool (Pool Details page)
    - Add / Edit a comment:
    - Edit Pool Range
    - Edit IP Assignment
    - Add / Remove a Pool Reservation
    - Edit Advanced Settings:
    - Detach Server from a Pool
  - Pushing DHCP Pools
    - Manual Pool Push (DHCP Pools List)
    - Manual Pool Push (Pool Details page):
    - Schedule Push (from Pool Details page):
    - Schedule Push (Admin Scheduler):
    - DHCP Group Push:
    - DHCP Server Push:
  - Delete a Pool

- Addiitonal Information:
-

Add a DHCP Pool

Add a new DHCP Pool from the Pool List by clicking the "Add Pool" button at the top of the page.

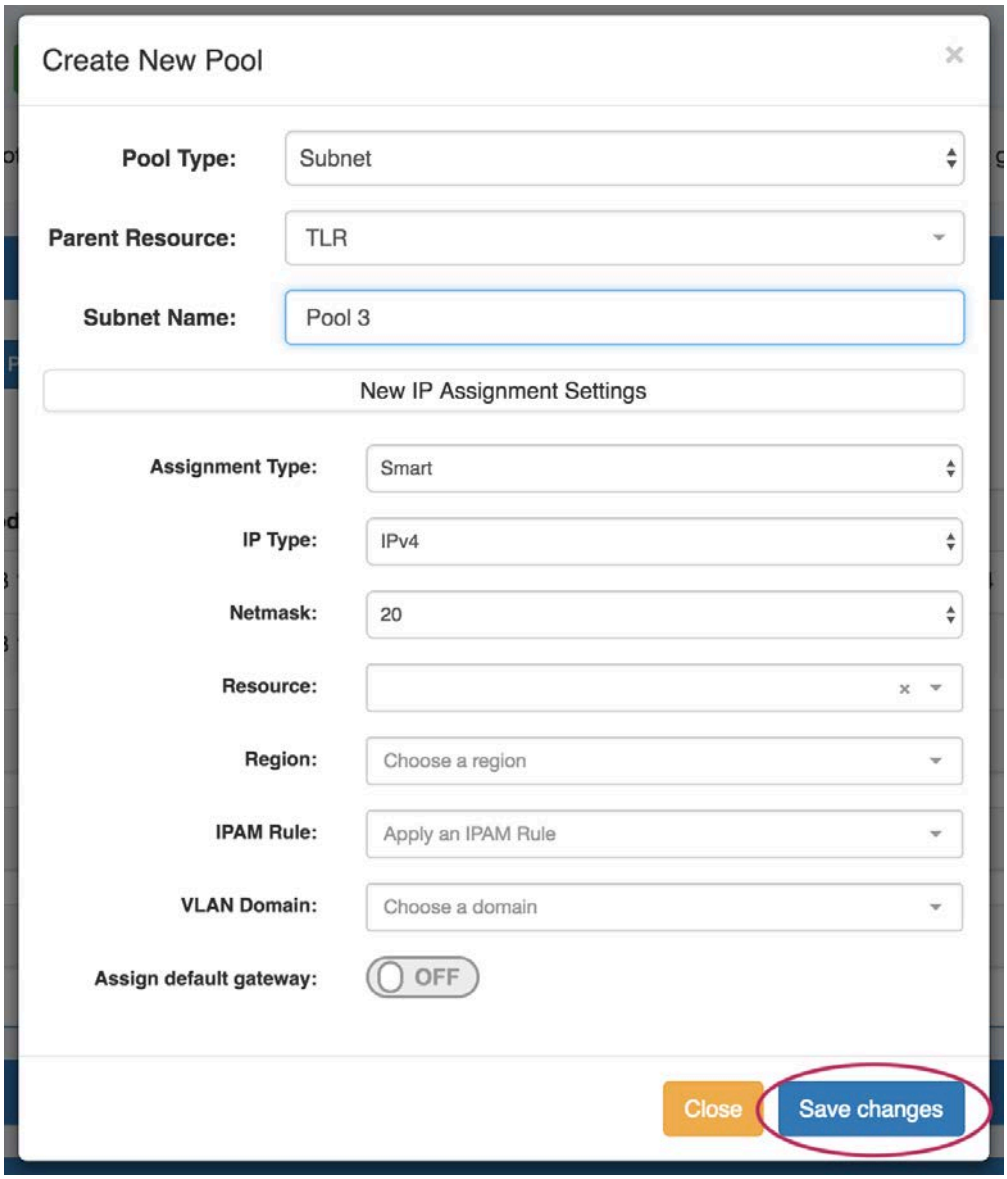


When the "Create New Pool" dialog appears, you may choose to create either a Subnet or Host Pool. Either pool type may have a block smart assigned or manually direct assigned.

Create Subnet Pool

Click here to expand...

To add a new subnet pool to a DHCP Group, select "Subnet" as the Pool Type. Select the parent resource and type in a Subnet Name.



Next, choose an assignment type - you may either have ProVision smart assign a block for the pool that matches your provided criteria, or directly assign a specific block.

Smart Assign

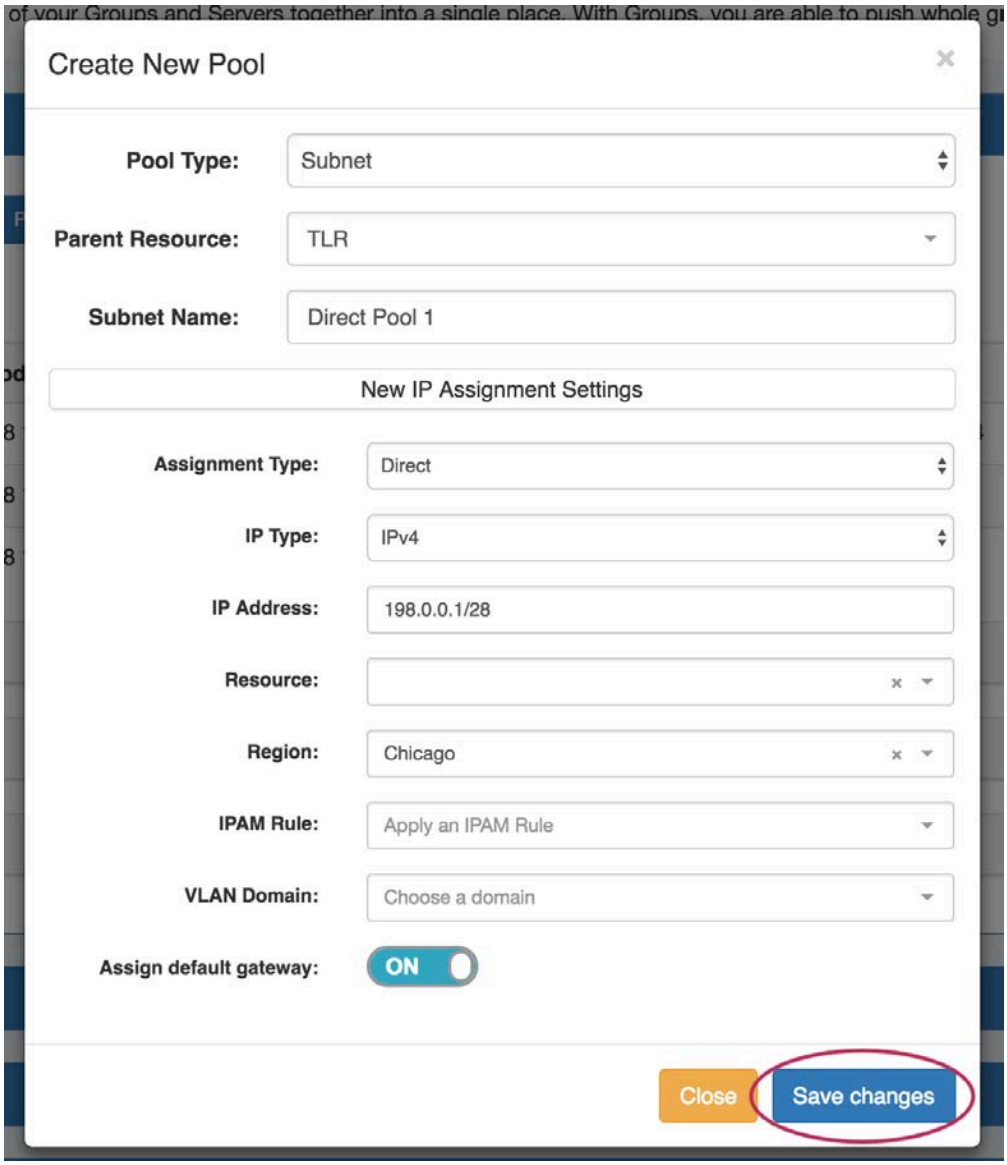
To Smart Assign a DHCP pool, select "Smart" for the Assignment type, choose IPv4 or IPv6 for the IP type, then select a netmask.

All other fields are optional - Resource, Region, IPAM Rules, and VLAN may be selected to further narrow the block criteria. If desired, toggle "Assign Default Gateway" on.

When done, click "Save Changes". ProVision will choose the first block matching all criteria to assign to the pool.

If no block matches the given criteria, a message will state so. Try either removing some criteria, or ensure that blocks exist in IPAM (with subassignments enabled!) with the given attributes.

Direct Assign



To Direct Assign a DHCP pool, select "Direct" for the Assignment type, choose IPv4 or IPv6 for the IP type, then type in the desired IP Address.

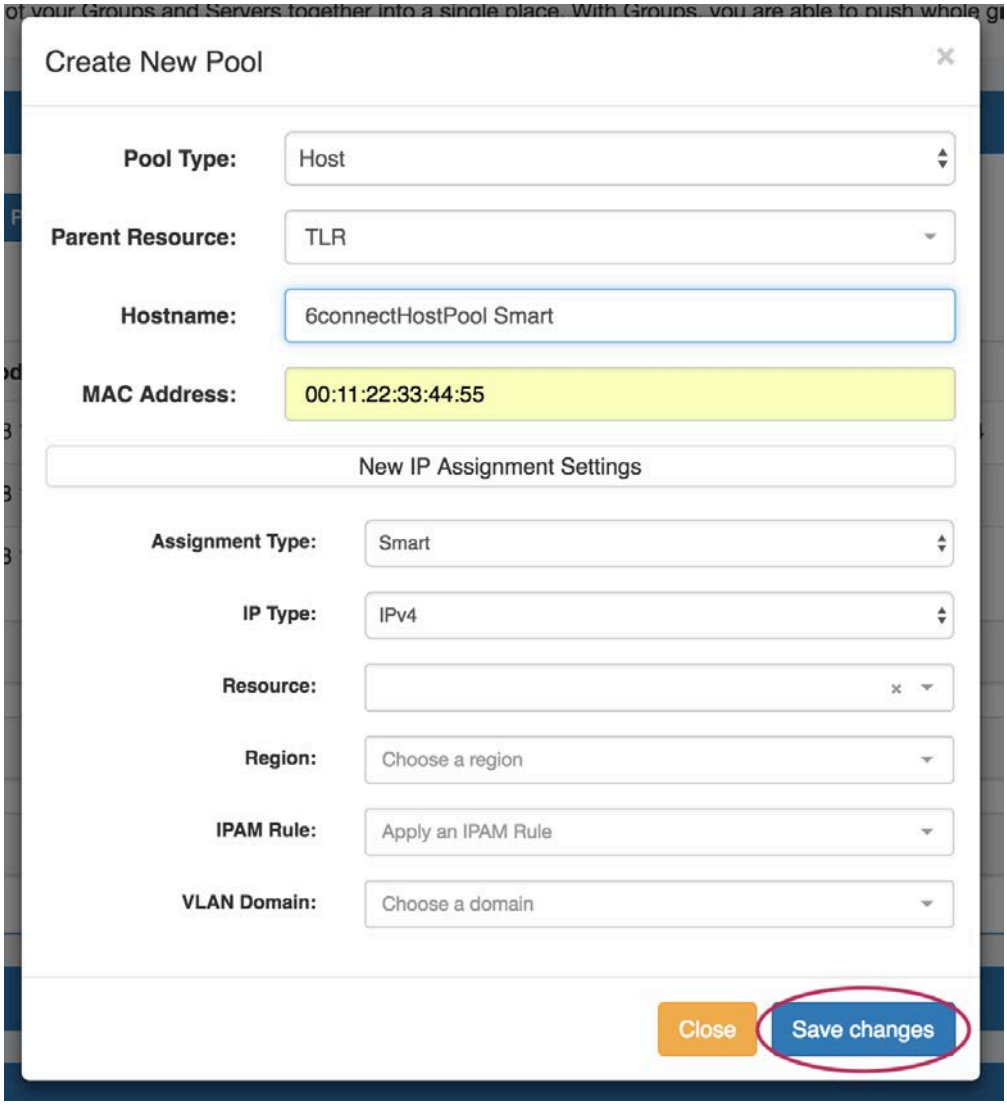
All other fields are optional - Resource, Region, IPAM Rules, and VLAN may be selected to further narrow the block criteria, if duplicate blocks exist. If desired, toggle "Assign Default Gateway" on.

When done, click "Save Changes". If the typed block exists in ProVision with subassignments enabled, and meets any additional criteria, that block will be assigned to the pool.

Create a Host Pool

Click here to expand...

To add a new host pool to a DHCP Group, select "Host" as the Pool Type. Select the parent resource, and add a Hostname and MAC Address.



Next, choose an assignment type - you may either have ProVision smart assign a block for the pool that matches your provided criteria, or directly assign a specific block.

Smart Assign

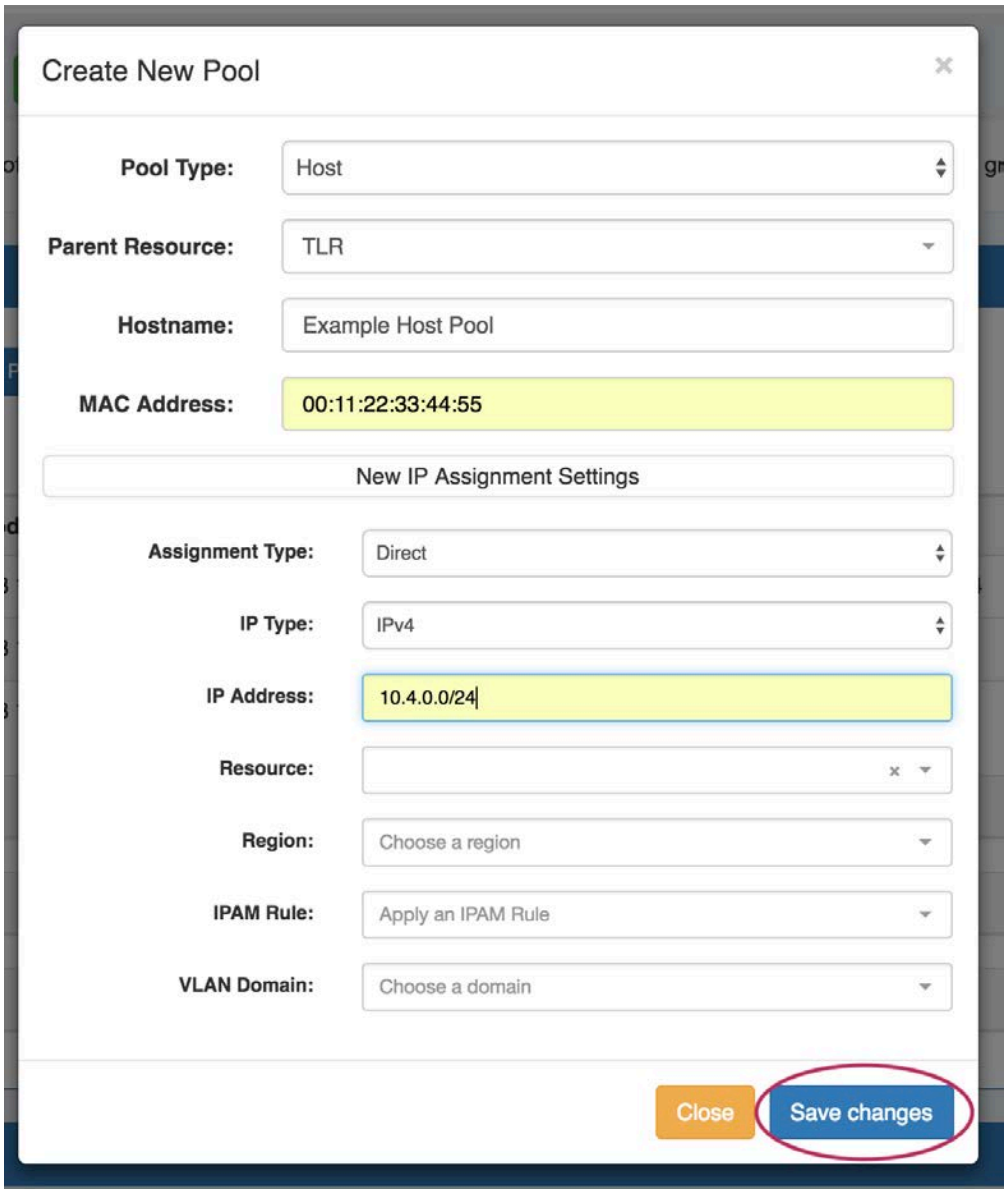
To Smart Assign a DHCP pool, select "Smart" for the Assignment type and choose IPv4 or IPv6 for the IP type.

All other fields are optional - Resource, Region, IPAM Rules, and VLAN may be selected to further narrow the block criteria.

When done, click "Save Changes". ProVision will choose the first block matching all criteria to assign to the pool.

If no block matches the given criteria, a message will state so. Try either removing some criteria, or ensure that blocks exist in IPAM (with subassignments enabled!) with the given attributes.

Direct Assign



To Direct Assign a DHCP pool, select "Direct" for the Assignment type, choose IPv4 or IPv6 for the IP type, then type in the desired IP Address.

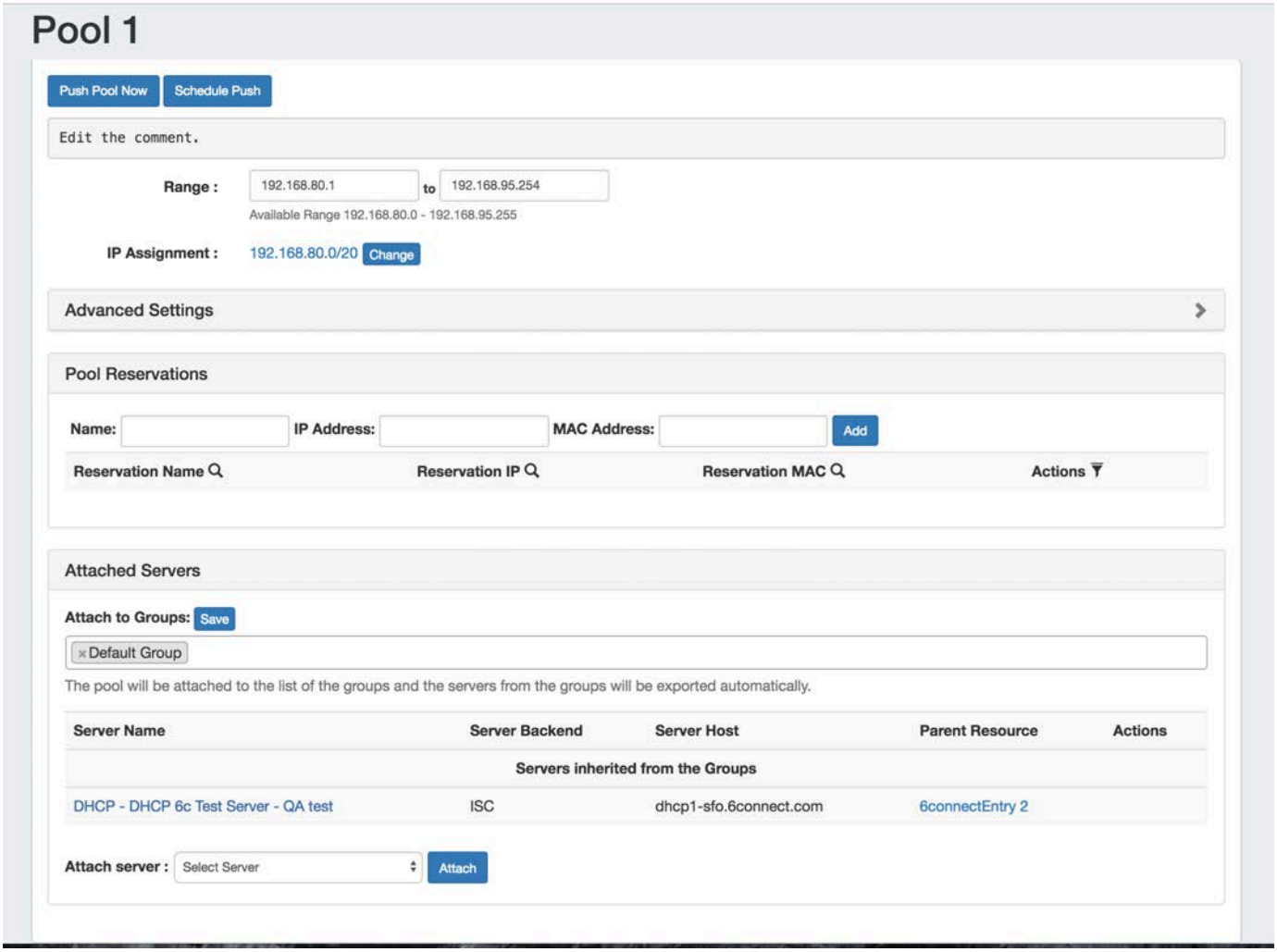
All other fields are optional - Resource, Region, IPAM Rules, and VLAN may be selected to further narrow the block criteria, if duplicate blocks exist.

When done, click "Save Changes". If the typed block exists in ProVision with subassignments enabled, and meets any additional criteria, that block will be assigned to the pool.

Edit / Manage a Pool (Pool Details page)

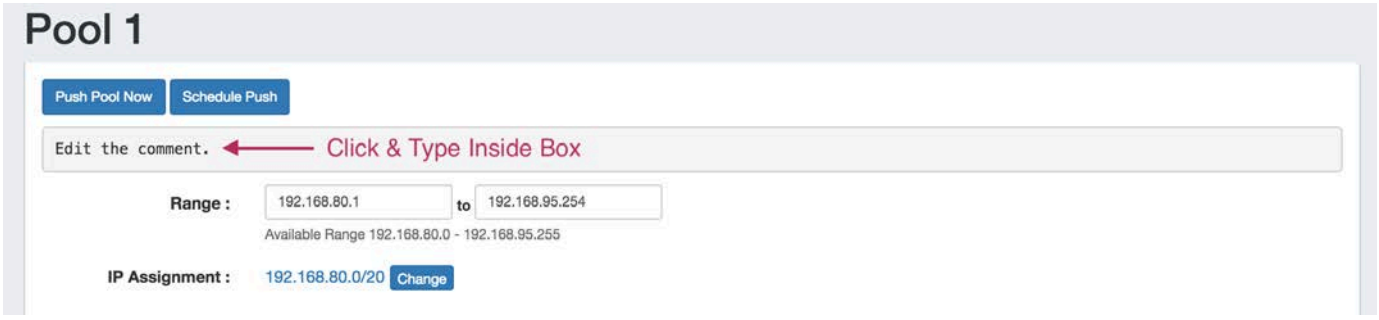
Additional Pool Management options are available from the **Pool Details** page. Pools may be edited by clicking on the Pool name in any **DHCP Pools** list in which it appears.

From this page, users may edit assignment, range, settings, reservations, or servers for the pool.



Add / Edit a comment:

To add a comment to a pool, click anywhere inside the comment area and begin typing. The comment will be saved when an area outside of the comment box is clicked, deselecting the comment area.



Edit Pool Range

The pool range may be manually edited by typing the desired range start and end values into the "Range" section, then click the "Save" button.

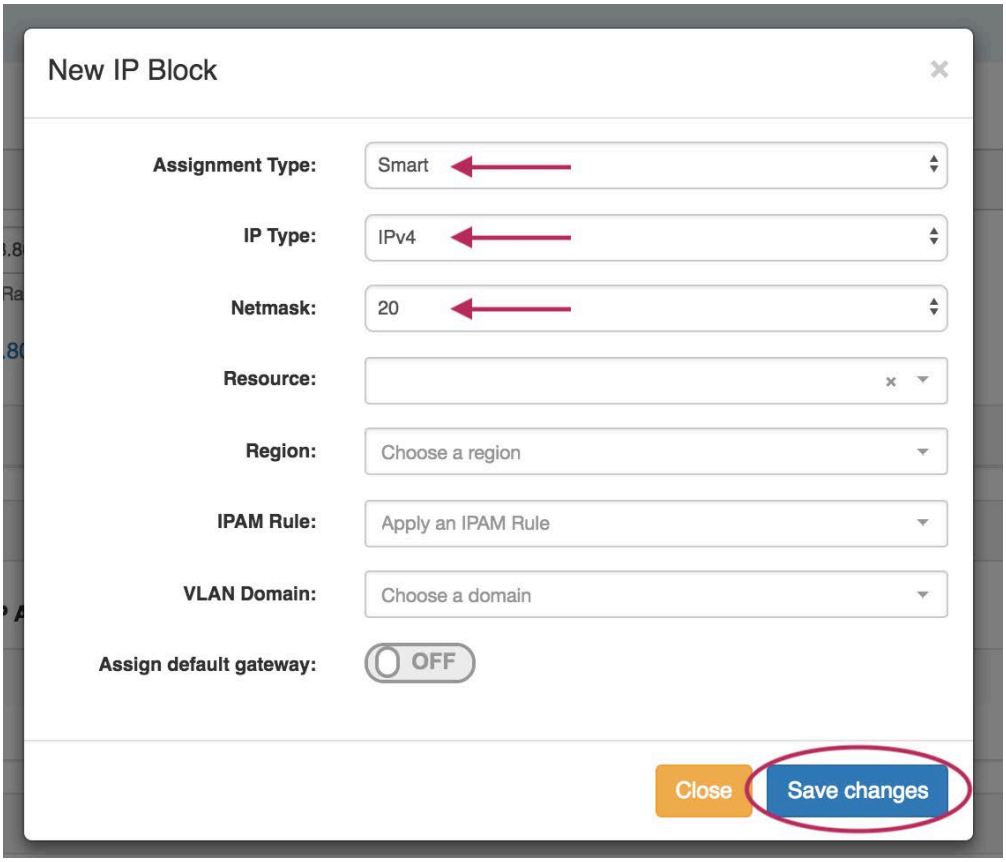


Edit IP Assignment

To change the current IP assignment for the pool, click the "Change" button next to "IP Assignment".



In the "New IP Block" dialog, select the Assignment Type, IP Type, and Netmask. Optional fields include Resource, Region, IPAM Rule, and VLAN as filtering criteria if desired.



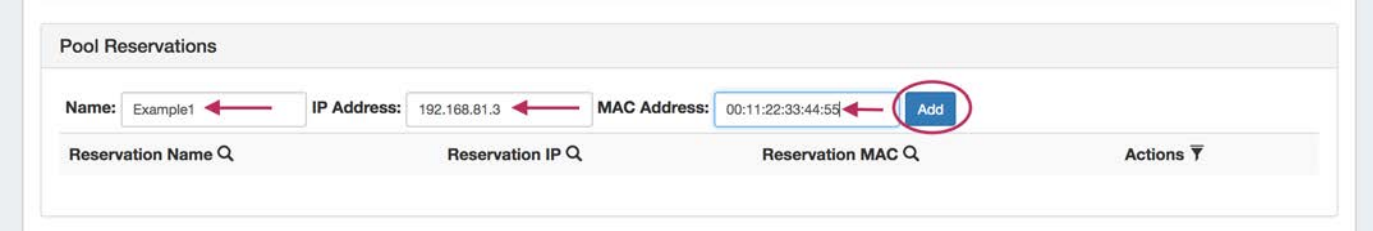
When complete, click "Save changes".

If no block meets the selected criteria, an error message will appear - try removing filter criteria, or verify that subassignable blocks meeting the criteria have been created in IPAM.

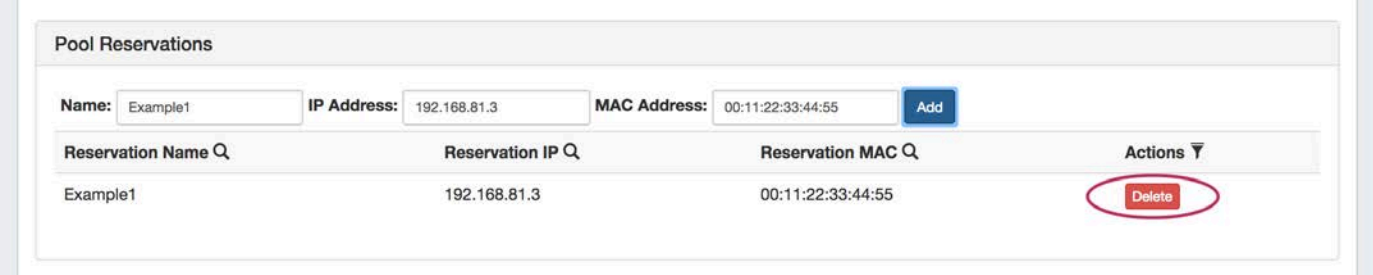
If successful, the IP assignment will change to a block meeting the provided attributes.

Add / Remove a Pool Reservation

Add a new pool reservation by entering the Name, IP, and MAC Address into the fields under "Pool Reservations", and click the "Add" Button.

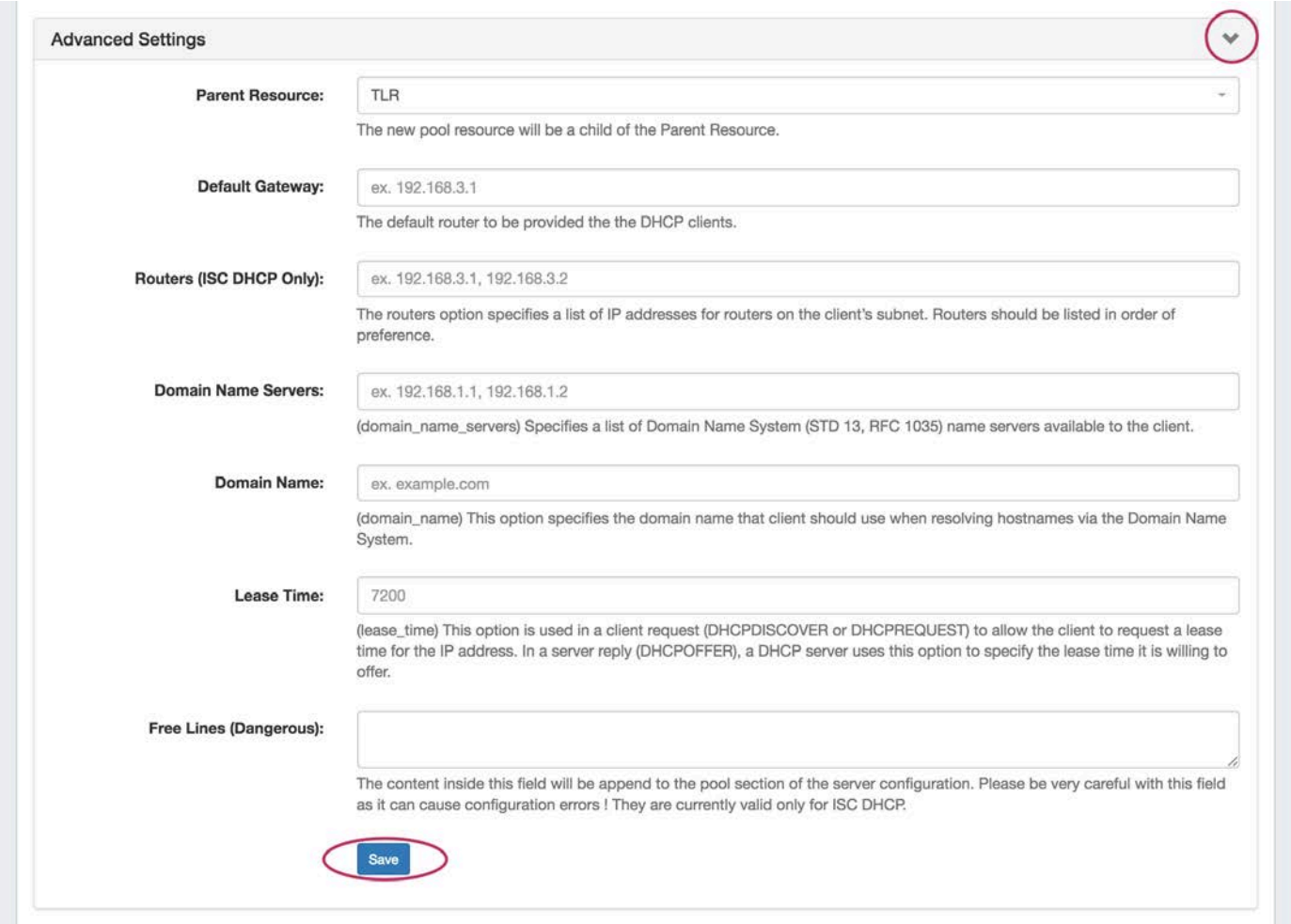


Once created, a reservation may be removed by clicking the "Delete" button next to it.



Edit Advanced Settings:

Open the zone "Advanced Settings" section by clicking on the header or the expansion arrow (>). Edit fields as needed, and click "Save" when complete.

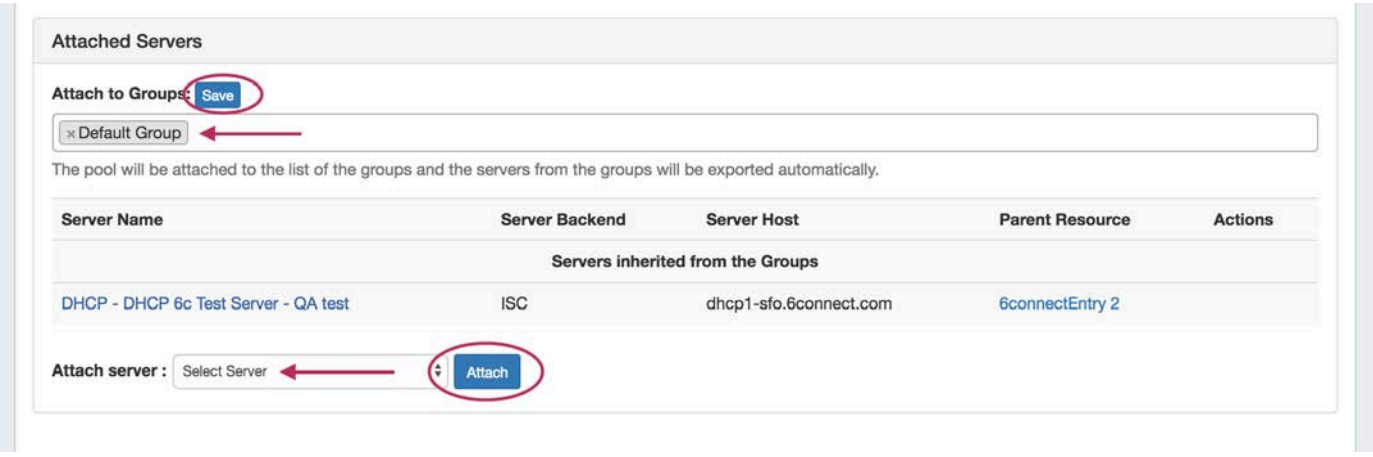


Attach Group / Server to a Pool

Servers may be attached to a pool either from a DHCP Group association, or directly from the Pool Details Page.

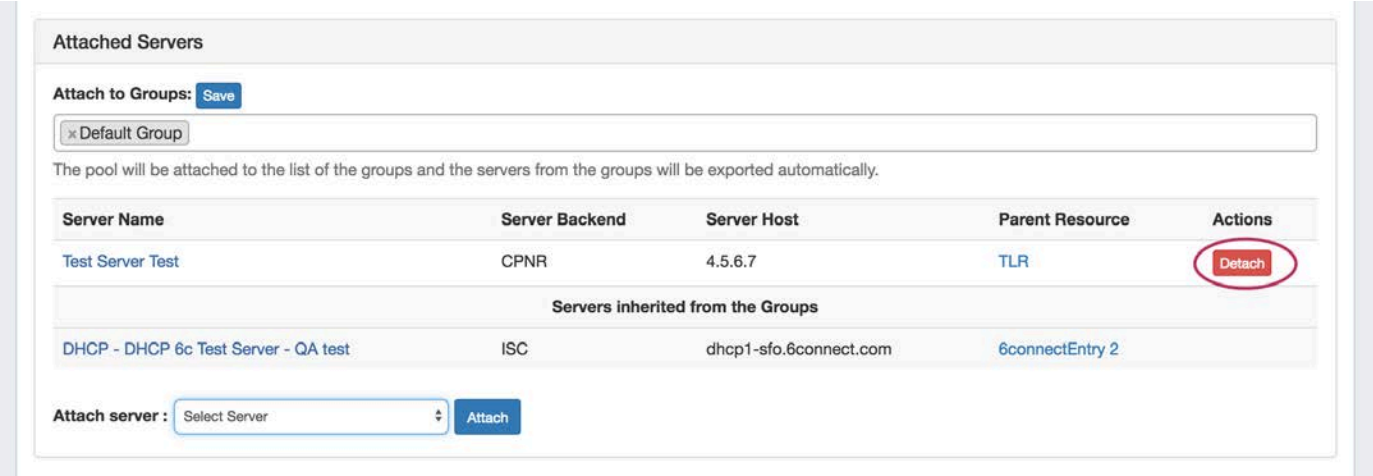
Attach the pool to a DHCP Group by clicking in the DHCP Group search box, selecting a Group, then clicking "Save" next to "Attach to Groups". The pool will inherit the DHCP server associated with that group.

To directly attach a DHCP server to a pool, select the server from the "Attach server" dropdown, then click the "Attach" button.



Detach Server from a Pool

To detach a server from a pool, look for the server under the "Attached Servers" list, then click "Detach" under actions for that server.

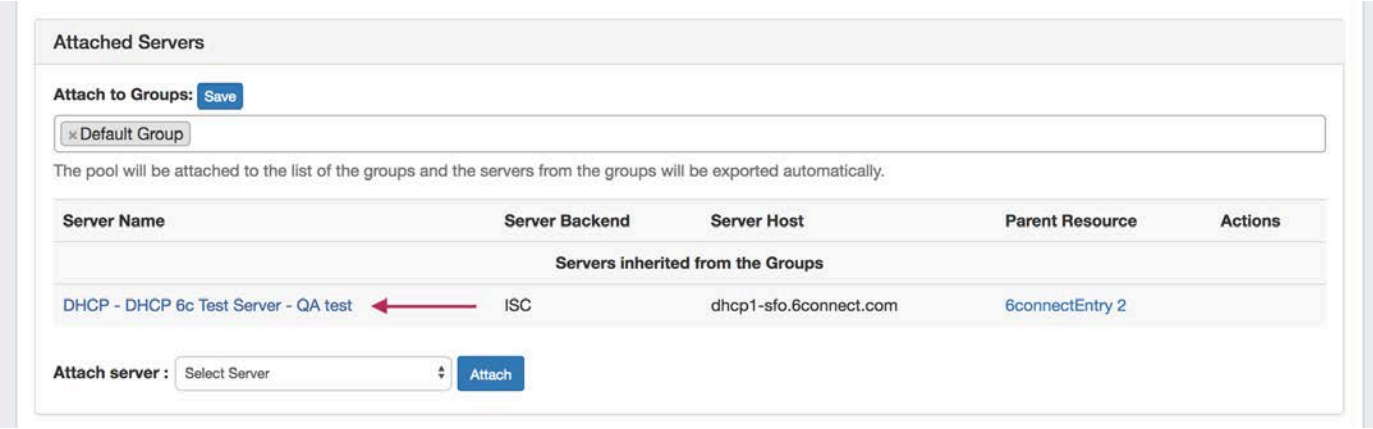


To detach a server inherited from a Group, the pool must be removed from the group. To do that, click the "x" next to the group name in the "Attach to Groups" box, then click "Save".

Pushing DHCP Pools

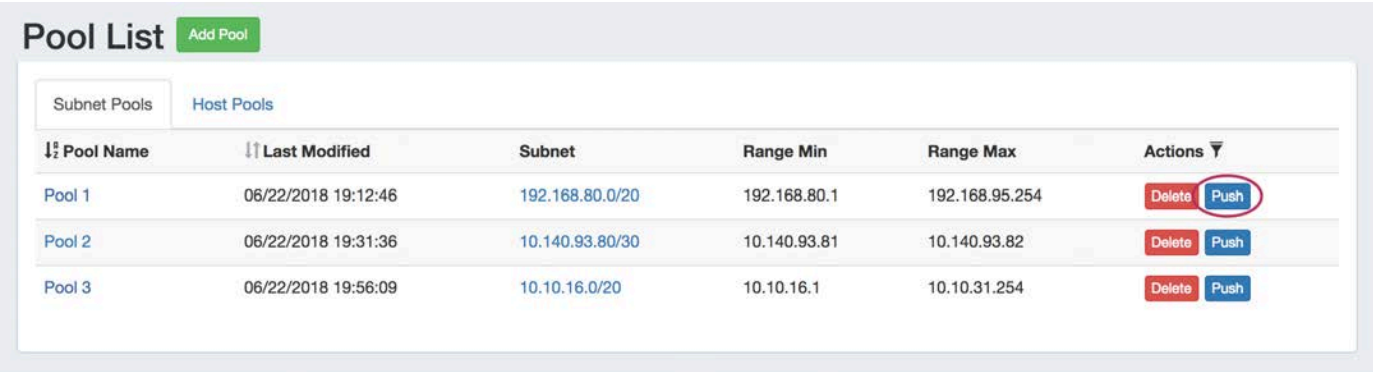
Pools may be pushed manually ("Push", "Push Pool Now"), as a scheduled future push ("Schedule Push"), or pushed as part of a DHCP Group or DHCP Server Push.

Before pushing a pool for the first time, view the "Attached Servers" section in the **Pool Details** page and verify that the correct server is attached to the DHCP Pool, either directly or via a DHCP Group.

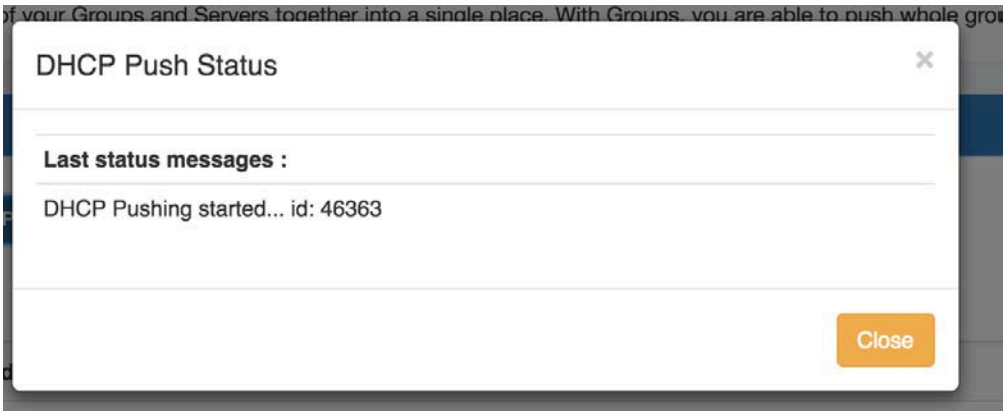


Manual Pool Push (DHCP Pools List)

Pools may be manually pushed by clicking the "Push" button for the pool from any **DHCP Pools** list.



The "DHCP Push Status" modal will appear, showing status and any errors that occur.



When a successful push is complete, a green "Finished DHCP Pushing Request" message will appear, and the window may be closed.

Manual Pool Push (Pool Details page):

The Pool Details page, accessed from clicking on the zone name in the Group zone list, includes a "Push Pool Now" button that immediately attempts to push the pool to the attached server(s).



Click on "Push Pool Now", and the "DHCP Push Status" modal will appear, showing status and any errors that occur.

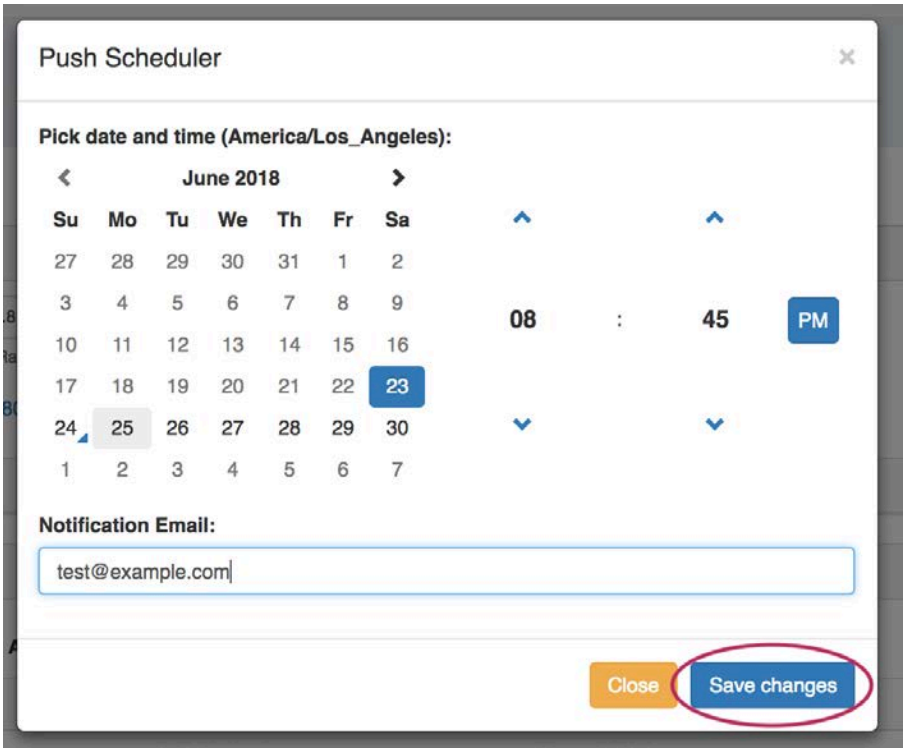
When a successful push is complete, a green "Finished DHCP Pushing Request" message will appear, and the window may be closed.

Schedule Push (from Pool Details page):

The Pool Details page includes an option to schedule a push, similar to scheduling a Group push.



Click "Schedule Push" for the pool, then select a date, time (12 hour, with AM / PM toggle), add a notification email address, and click "Save changes".



The Scheduled task will be shown in both the Pool Details page (under "Scheduled Tasks") and the Admin Scheduler task list.



The task may be deleted prior to running by clicking the "delete" button, but will automatically be removed once completed.

Schedule Push (Admin Scheduler):

Pools may also be pushed via scheduled DHCP Group or DHCP Server pushes. Server pushes may be set up by Admin users from either the **DHCP Servers** section, or the **Scheduler** tab.

For more information, see the Working with DHCP Servers or Scheduler documentation.

DHCP Group Push:

All pools contained in a DHCP Group may be pushed at the same time, by pushing the DHCP Group.

For information on working with DHCP Groups, including performing a manual or scheduled Group push, see Working with DHCP Groups - Pushing a Group.

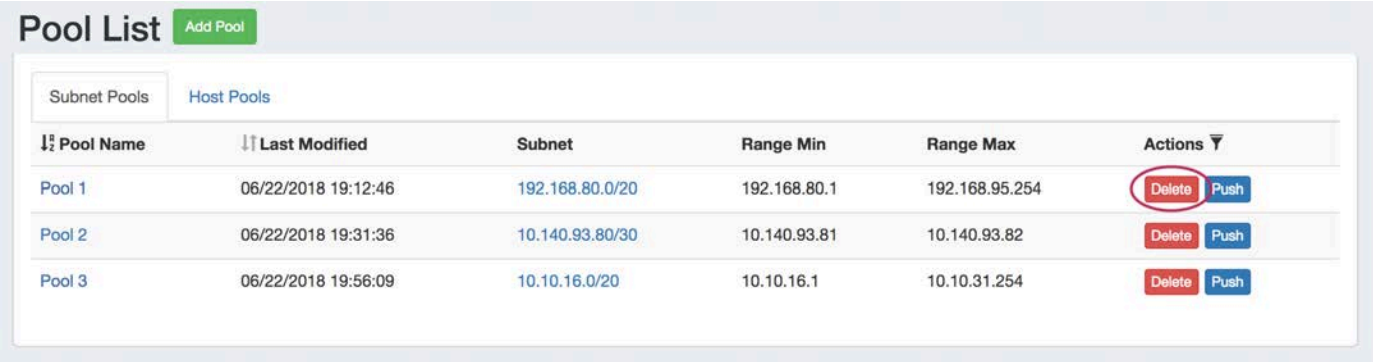
DHCP Server Push:

Admin users may push all pools on a DHCP server from the **DHCP** Tab -> **DHCP Servers** page.

For information on working with DHCP Servers, including performing a manual or scheduled server push, see Working with DHCP Servers - Pushing a Server.

Delete a Pool

Users with delete permissions over the pool may delete it by clicking the "Delete" button under "Actions" from any DHCP Pool List.



Addiitonal Information:

For more information on working with DNS Zones, Groups, and Servers, see the following sections:

DNS Tab

Working with DNS Groups

Editing DNS Zones (The View Zone Page)

DNS Administration

Import DNS Zones

Copy of DHCP Tab

Working with DHCP Gadgets

There are two DHCP Gadgets available to integrate onto a Resource Entry Page - the DHCP Management Gadget, and the DHCP Customer Configuration Gadget.

For an overview of Gadgets and how to attach them to Resource pages, see Gadgets and Adding a Gadget to a Section .

- Working with DHCP Gadgets
- DHCP Management Gadget (Legacy)
    - Before You Begin:
        - Attach / Enable the DHCP Gadget
        - Managing DHCP Server Configurations
        - Group Management
            - Working with DHCP Groups
        - Connection Configuration
        - Server Details
        - Scan Server
        - DHCP Pools
            - DHCP Pool Search:
        - Create a New DHCP Pool - Subnets
        - Create a New DHCP Pool - Host
        - Current Pushed Configuration
        - Saving/Pushing DHCP Server Configurations
- Working with the DHCP Customer Configuration Gadget
        - Setting up the DHCP Customer Configuration Gadget
        - Using the DHCP Customer Configuration Gadget
    - Additional Information

DHCP Management Gadget (Legacy)

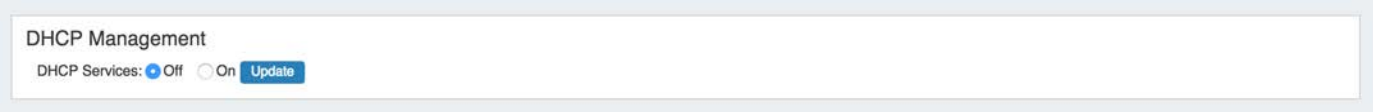
Note: As of ProVision 7.0.0, the DHCP Management Gadget is a legacy feature.

While it is still available to be used, we recommend checking out the updated **DHCP** Tab and using the DHCP Groups, DHCP Servers, and DHCP Pools pages as the primary DHCP management system.

Before You Begin:

Attach / Enable the DHCP Gadget

Verify that the DHCP Gadget is attached to the Resource Section. Then, you can enable the gadget by selecting the radio button next to "On", and click "Update".



Once enabled, the DHCP Management Gadget will show sections for "Group Management", "Connection Configuration", "Server Details", "Scan Server", "DHCP Pools", and "Create a New DHCP Pool". Once pushes or unpushed configuration changes occur, a "Current Pushed Configuration" or "Unpushed Configuration" section will be available for review.



These sections will be reviewed in detail further on in this section.

A Note on DHCP Aggregates

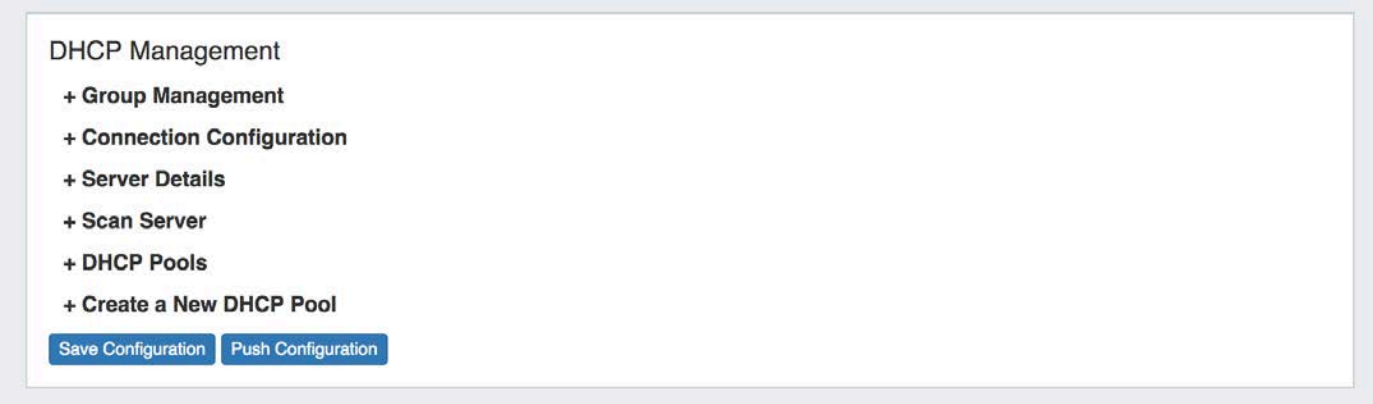
ProVision versions prior to 7.0.0 required creating or identifying specific IPAM blocks as "DHCP Aggregates" using the "DHCP Available" status to assign DHCP pools from.

As of 7.0.0, these statuses have been removed, and any IPAM block may be used to assign DHCP pools from, as long as they are subassignable and meet the assignment criteria provided by the user.

This applies to all DHCP Management functions - the DHCP Management Gadget, DHCP Customer Configuration Gadget, and all DHCP Tab areas.

Managing DHCP Server Configurations

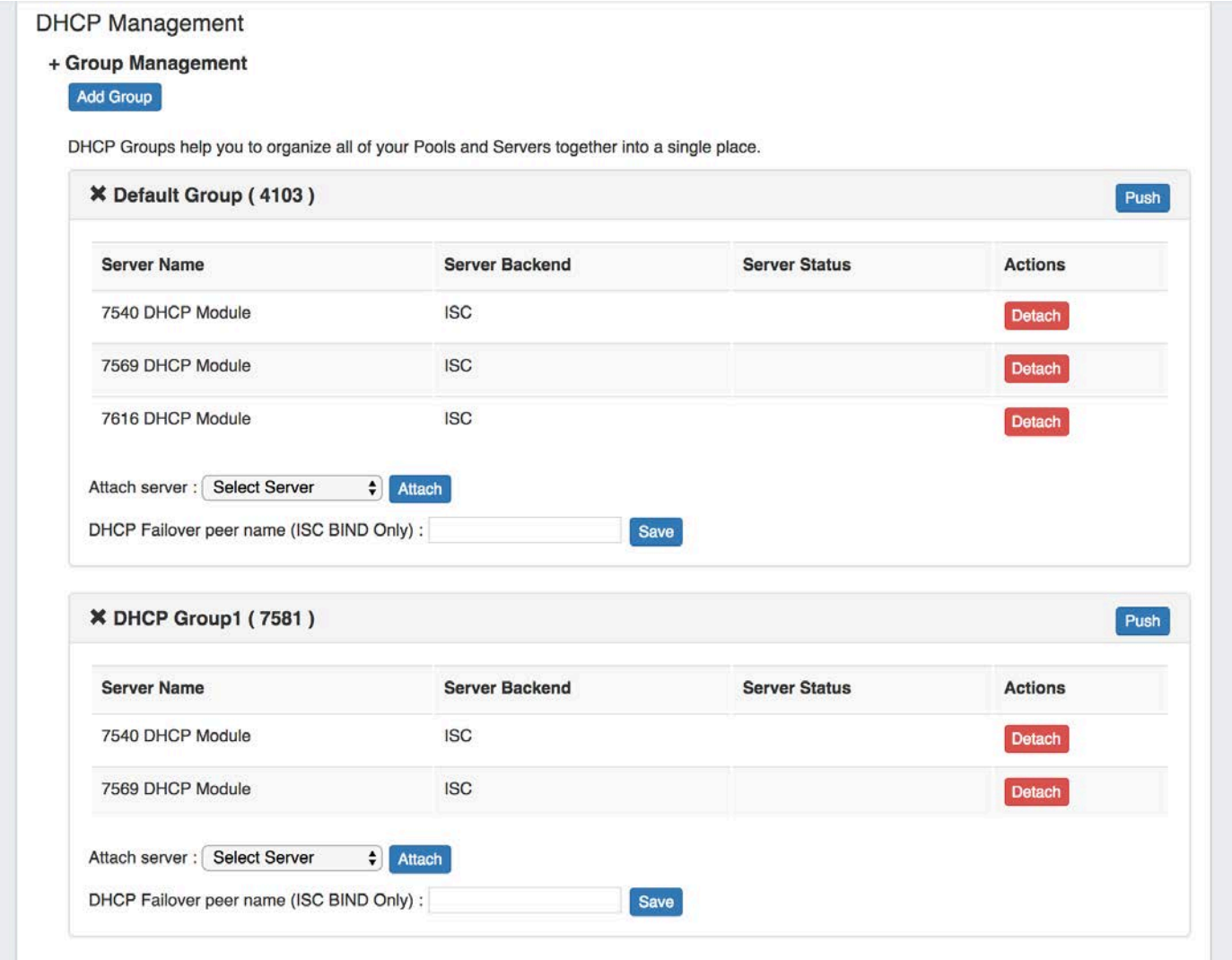
Once DHCP functions are enabled for a Resource Section, you will be able to manage configurations per Resource by expanding the relevant areas on the Resource's Entry page.



Group Management

This section of the DHCP Management Gadget allows you to create and manage DHCP Groups.

DHCP groups help you to organize all of your Pools and Servers together into a single place, and push pools linked to a DHCP Group to all of the servers attached to that Group. It also allows for the designation of a DHCP Failover Peer for ISC BIND servers.



Working with DHCP Groups

Step 1: Create DHCP Servers

Click here to expand...

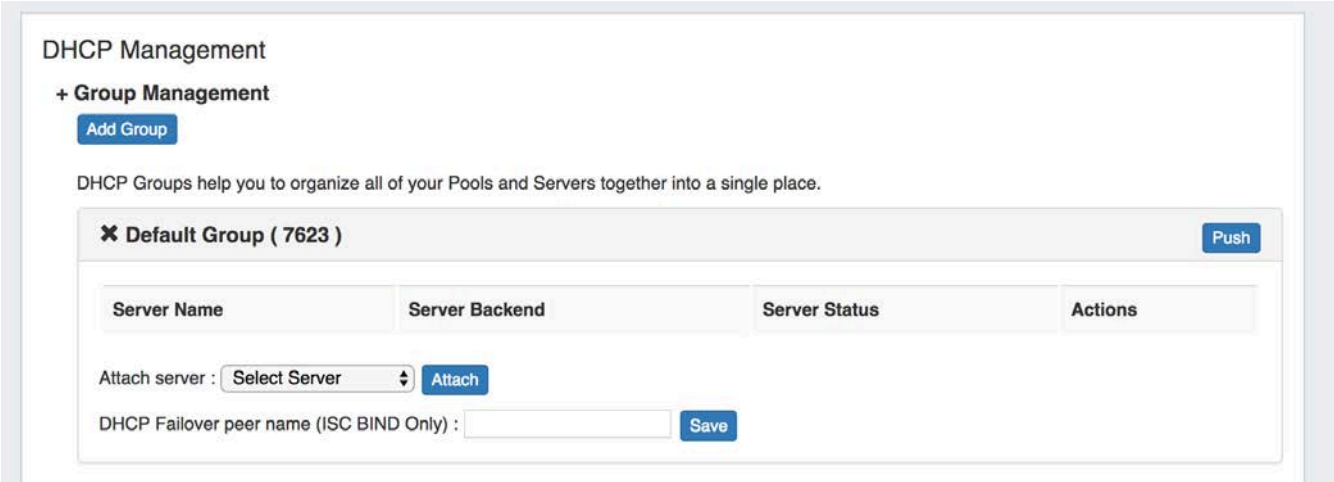
Ensure that all necessary DHCP servers are created via the **DHCP** Tab **DHCP Servers** page. See Working with DHCP Servers for additional information.

All DHCP servers listed under the DHCP tab will be available to select to attach to DHCP Groups (in the format of "ResourceID - DHCP Module").

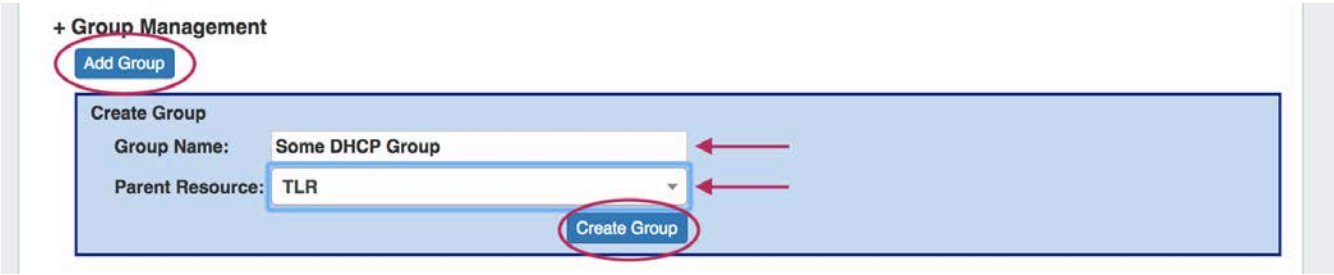
Step 2: Create DHCP Groups

Click here to expand...

Select a DHCP server page with the DHCP Management Gadget from, and expand the Group Management section. Note: Any DHCP server page with the Manangment Gadget is fine to use, as the Group Management section will globally update to be the same in any DHCP Management Gadget. As of ProVision 7.0.0, DHCP Groups may also be created and managed from the DHCP Groups page under the **DHCP** Tab.



A "Default Group" should be available immediately, or you can click the "Add Group" button to create a new DHCP Group. Type in the desired Group Name and select a Parent Resource for the Group, then click "Create Group".



Your new Group will be available in the DHCP Management Gadget. To delete the DHCP Group, click the "X" next to the Group name.

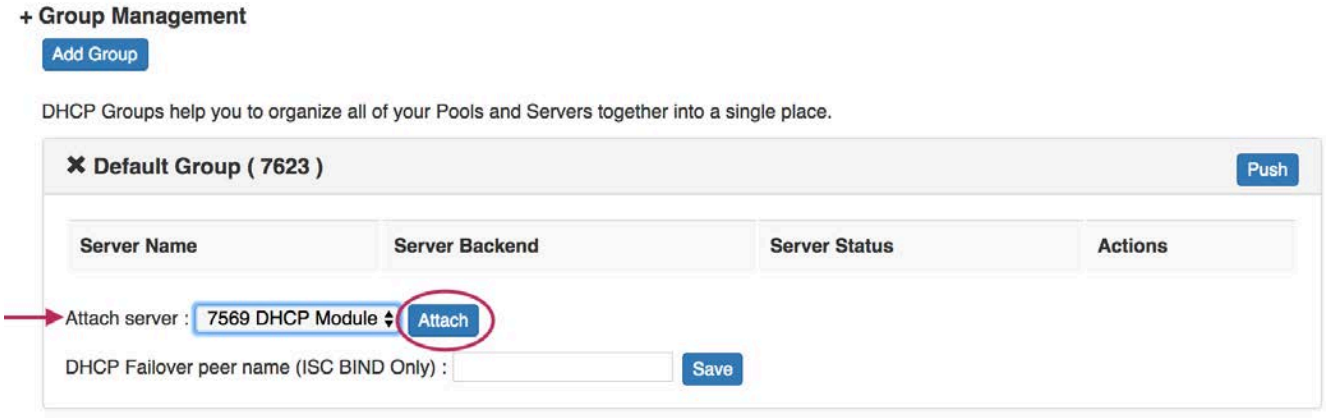
Step 3: Attach DHCP Servers to Groups

Click here to expand...

All DHCP servers listed under the ProVision DHCP Tab will be available to select and attach to the Group. They are identified by their resource ID number, so if necessary, check the server's resource ID number in the resource information Gadget.



Select the desired server from the "Attach Server" selector, then click "Attach". The server is now attached to the group and available for pushes.



If needed, a server may be removed from a DHCP Group by clicking the "Detach" button.

Step 4: Identify Failover Peer (Optional - ISC Bind Only)

Click here to expand...

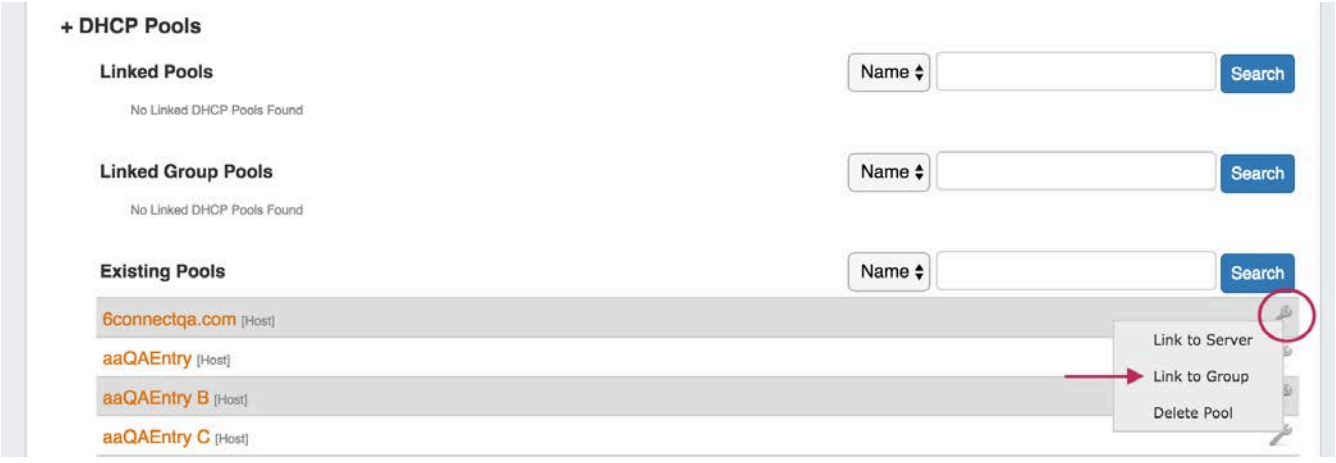
If a DHCP failover peer is configured for the server, enter the server name and click the "Save" button. In the event that the primary server fails to push, the push will be sent to the failover peer instead.



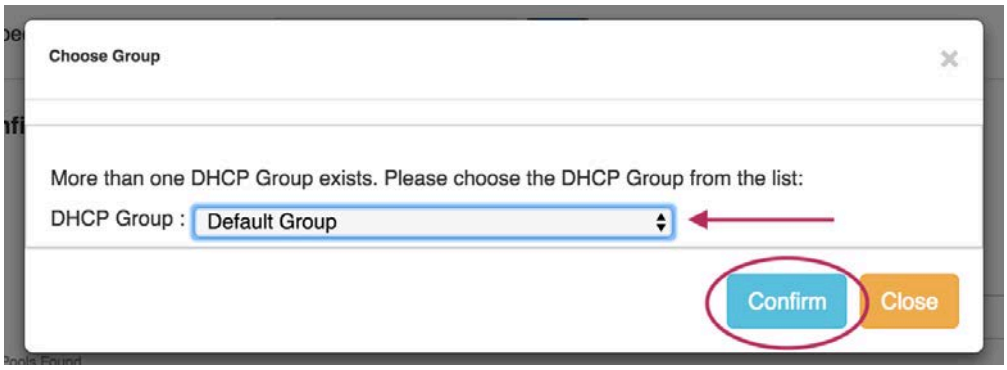
Step 5: Link Pools to DHCP Groups

Click here to expand...

From the Existing Pools list under the DHCP Pools section, click the Action Menu for the pool you wish to link to a DHCP Group. From there, click "Link to Group".



If multiple DHCP Groups exist, a dialog will pop up for you to select the desired Group to link the pool to. Select the DHCP Group, then click "Confirm" to add the pool to the Group.



Once linked, the DHCP Pool will show under the "Linked Group Pools" area of the DHCP Pool section. From here, if desired, you may remove the pool from the group by clicking the Action Menu and "Remove Link".



Step 6: Push the DHCP Group

Click here to expand...

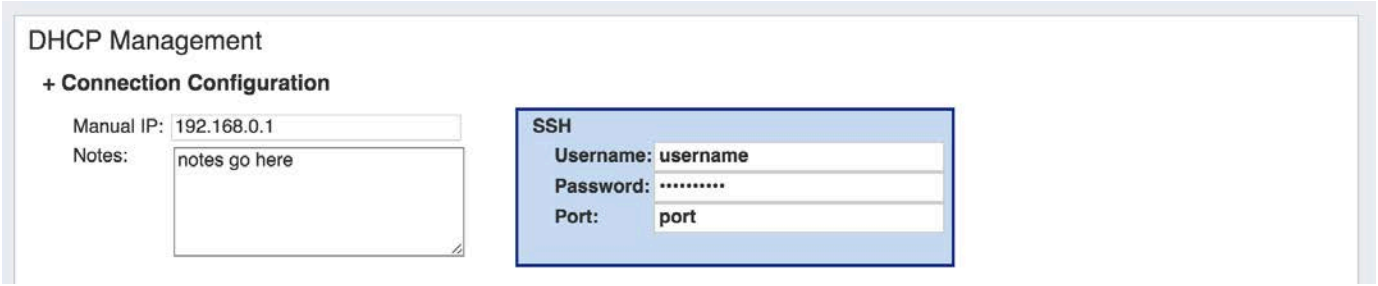
Lastly, click the "Push" button for the DHCP Group to push the linked pool(s) for the Group to the linked DHCP Module, and then push all DHCP servers linked to the Group.



While pushing, a status box will appear to show status and any errors, if applicable.

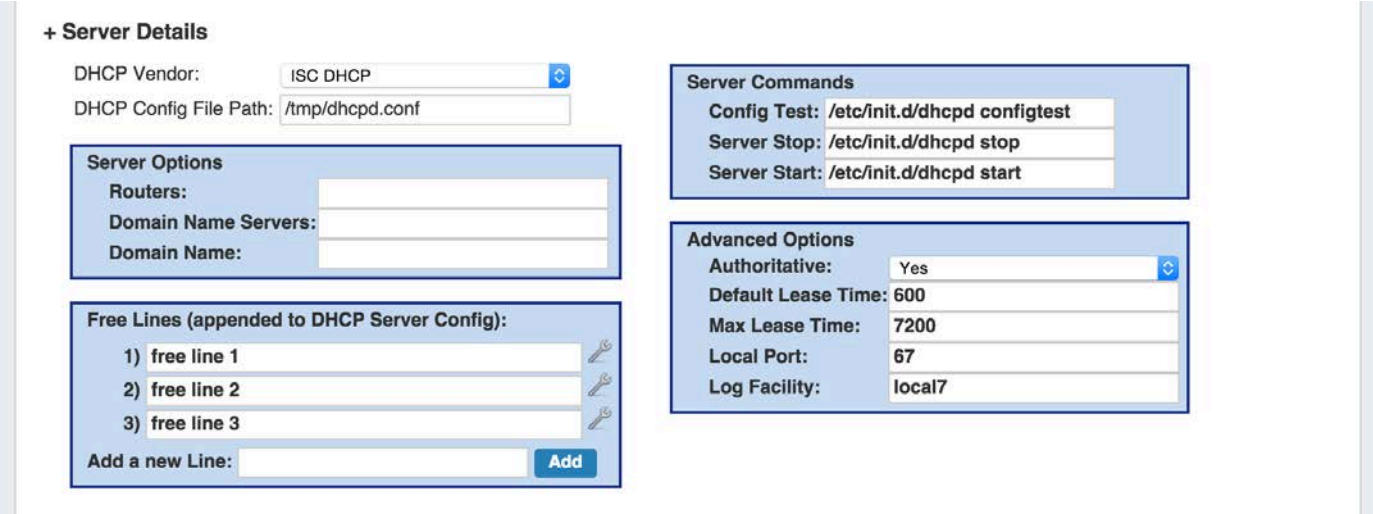
Connection Configuration

In this gadget area, you may enter in the information that will be used for ProVision to communicate to the DHCP Server.



Server Details

Server details and advanced options may be entered under this portion of the gadget.



Server Command: Config Test

A note on the Server Command: Config Test option:

This command is run after the newly-written DHCP Config file has been transferred to the server but before the server is restarted. Due to the wide variety of DHCP install configurations that can arise on different systems, it is best if the Config Test option is as explicit as possible. For example:

sudo /usr/sbin/dhcpd -t -cf /etc/dhcp/dhcpd.conf

This command explicitly requests the DHCP daemon start in test-only mode (-t) with a specific config file (-cf). This config file should be the location supplied in the "DHCP Config File Path" section.

Scan Server

Scan Server provides a "Scan Now" button, which scans the DHCP server and returns found IPs under DHCP Pools.



The returned IPs are divided by Pools and IPv4/IPv6. Returned fields include MAC address, IP address, and name. The percentage utilization of IP space out of the available DHCP Pool space is also included at the top of the list.

+ Scan Server

Scan Now

Scan Results

Blah: 0 Hosts Found. 256 Hosts Available. **100.00% Free**

No Hosts Found

Miscellaneous Addresses: found addresses which do not match a linked DHCP Pool

Hostname	MAC	IPv4	IPv6
Unknown Host	08:d8:33:93:87:70	10.1.10.35	
sep58bfea1164be	58:bf:ea:11:64:be	10.1.10.136	
Unknown Host	00:0e:8f:88:5b:8f	10.1.10.82	
philips-hue	00:17:88:24:2e:8b	10.1.10.85	
Unknown Host	00:0c:29:62:6c:57	10.1.10.59	
directv-hr44-b1550683	10:77:b1:55:06:84	10.1.10.30	
Unknown Host	00:80:a3:91:47:ae	10.1.10.55	
lifx bulb	d0:73:d5:00:06:a6	10.1.10.86	
sonoszb	00:0e:58:1c:e0:0a	10.1.10.108	
lifx bulb	d0:73:d5:00:ae:e2	10.1.10.96	
lifx bulb	d0:73:d5:00:a8:43	10.1.10.87	
Unknown Host	f4:b8:5e:00:7b:f3	10.1.10.89	
Unknown Host	74:29:af:d6:ba:8d	10.1.10.32	
Unknown Host	ec:e0:9b:b7:f3:15	10.1.10.51	
lifx bulb	d0:73:d5:00:20:51	10.1.10.81	
sonoszp	00:0e:58:ff:8e:82	10.1.10.109	
npi4305e9	4c:0f:6e:90:7a:da	10.1.10.75	
Unknown Host	74:75:48:08:bb:b1	10.1.10.77	
Unknown Host	a0:02:dc:1c:c4:aa	10.1.10.103	
Unknown Host	94:10:3e:4c:82:c9	10.1.10.79	
Unknown Host	00:25:90:02:ff:42	10.1.10.58	
living-room	c8:69:cd:2c:38:56	10.1.10.44	
kitchen	c8:69:cd:37:c9:bc	10.1.10.66	
iphone6splus	7c:01:91:d1:34:f3	10.1.10.104	
macbook	a8:66:7f:15:6b:85	10.1.10.39	
iphone1	d8:bb:2c:7a:61:8e	10.1.10.37	
mac-mini	20:c9:d0:96:4e:9d	10.1.10.107	
bedroomappletv2	6c:94:f8:e1:b0:14	10.1.10.54	
mac-air	9c:f3:87:b1:5b:b6	10.1.10.90	
livingroom	a4:67:06:64:b3:97	10.1.10.49	
aarons-imac	b8:09:8a:ba:dd:4f	10.1.10.34	
iphone-88	5c:ad:cf:03:3b:66	10.1.10.45	
ipad2	2c:1f:23:49:1f:93	10.1.10.84	
Unknown Host	34:46:6f:0a:c5:a1	10.1.10.33	
02aa01ac351309sc	18:b4:30:0c:c5:27	10.1.10.31	
Unknown Host	00:11:32:16:29:ef		2607:fae0:6000::514c
Unknown Host	20:61:73:79:6e:ff		2607:fae0:6000::5e35
Unknown Host	88:1f:a1:2a:d7:71		2607:fae0:6000::5065

If the scan is unable to be completed, an error message will appear in the DHCP Management section.

DHCP Pools

In this area, the admin can specify what DHCP Pools are linked to the DHCP server or DHCP Groups.

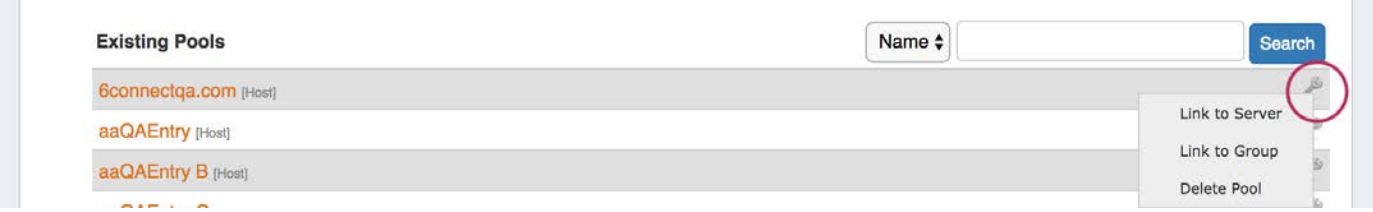
This includes any host reservations as well as DHCP Pools as defined in the next section.

Use the Action menu on Existing Pools to Link to Server, Link to Group, or Delete Pool

The Action menu for Linked Pools will show the option to Remove Link from Pools.







Additional edits may be made to pools by clicking on the Pool name. This allows the Pool name, assignment, and lines to be changed. Be sure to hit "Save" after making changes.



DHCP Pool Search:

Both Linked and Existing DHCP pools have field-specific search options to filter the DHCP pool list(s). Select the search term type from the dropdown list, type the search term in the search box, then hit "Search".



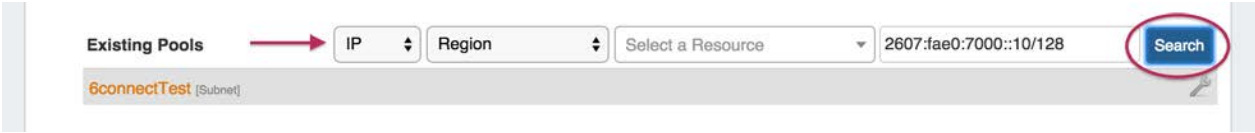
For both Linked and Existing pools, search terms include:

Name: The full or partial name of the pool (ex: "Lab")

MAC: The full (ex: 00:11:22:33:44:55:66:77) or partial (:22) MAC address, for pools that were created as "Host" type. MAC address searches must include a colon.

IP: Search by IP Block by providing the CIDR assigned to the pool. *(ex: 10.0.0.0/24)*

For Existing Pools, additional filter terms may be included to further filter the IP search by Region or assigned Resource.



After entering your search terms, click "Search", and the pool list will filter to the results.

Create a New DHCP Pool - Subnets

When Assigning a Subnet (via dropdown) the IP Assignment selection will pull the data from any IPAM block meeting your defined criteria. You can use either a Smart or Direct assignment depending on your preference. For Smart Assign pool creation, existing IP Rules may be applied to reserve additional addresses out of the pool range. To create an IP Rule, see IPAM Rules.

DHCP Pools and IP Rules

For DHCP Pools, ProVision automatically reserves the first and last address of the pool for Gateway and Broadcast addresses, respectively.

If an additional IP Rule is applied, the rule will begin with the second address in the block.

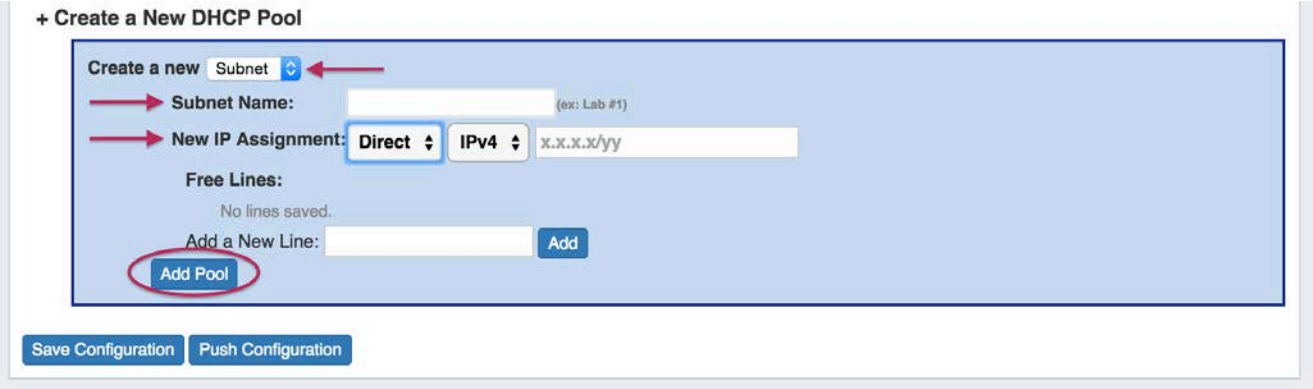
For example: if a DHCP Pool is created using 10.0.0.64/29 with an IP Rule of "Reserve First Three", the resulting pool range would be 10.0.0.68 through 10.0.0.70, as the first four as well as the final address would be reserved.

Enter the name of the Pool / Subnet, and select the assignment criteria:

For **Smart Assign**: select IPv4 / IPv6, Mask, then optional Region, Resource assignment, Domain - VLAN, and IP Rules criteria, add free lines if desired, then click "Add Pool".



For **Direct Assign**: select IPv4 / IPv6 and enter the block in CIDR format, add free lines if desired, then click "Add Pool".

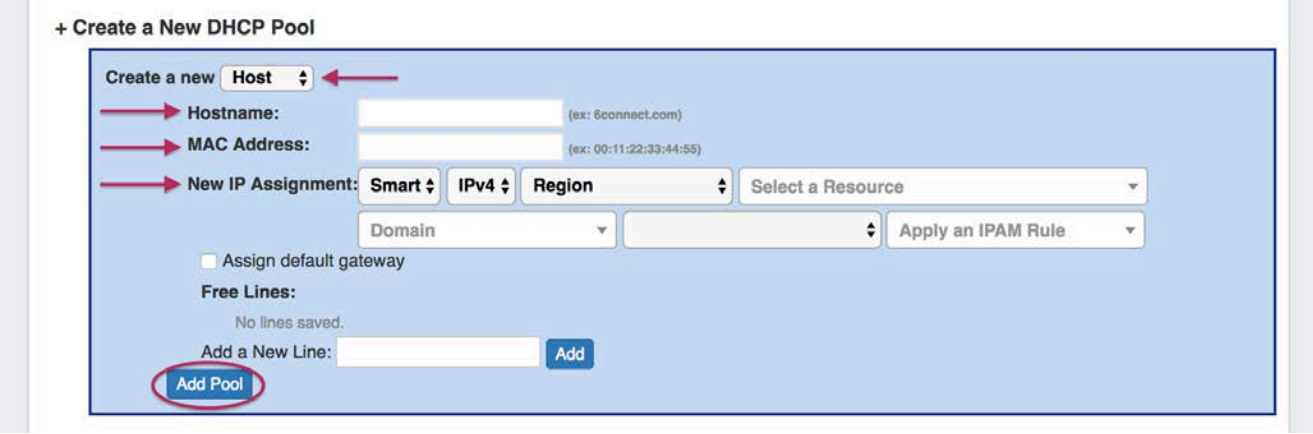


Create a New DHCP Pool - Host

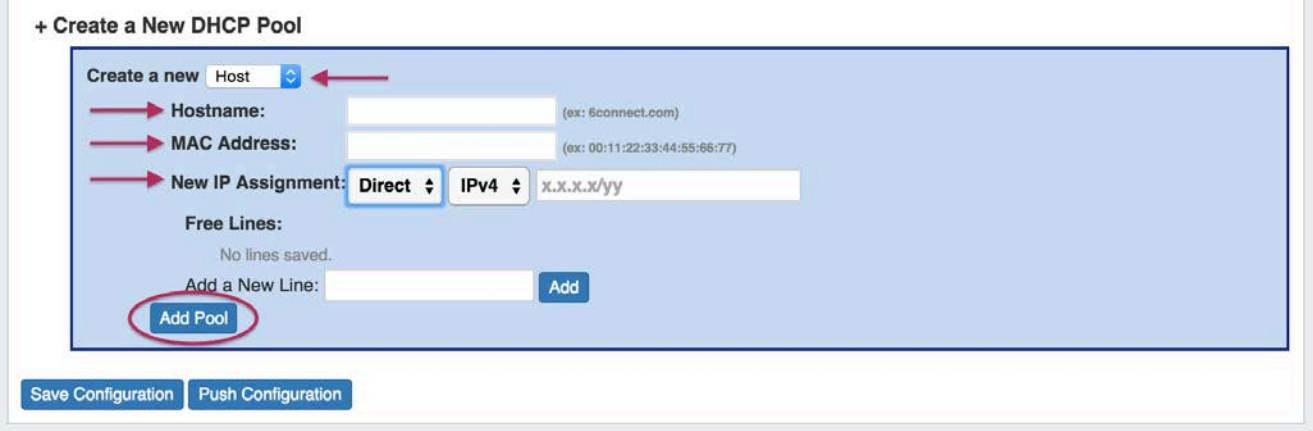
When reserving Hostname/MAC data, change the Dropdown to "Host". This will also give you an option to assign from an existing DHCP block (smart assign) or a specific IP address (direct assign).

Enter the name of the Hostname and MAC address, then select the assignment criteria:

For **Smart Assign**: select IPv4 / IPv6, then optional Region, Resource assignment, Domain - VLAN, and IP Rule criteria, add free lines if desired, then click "Add Pool".



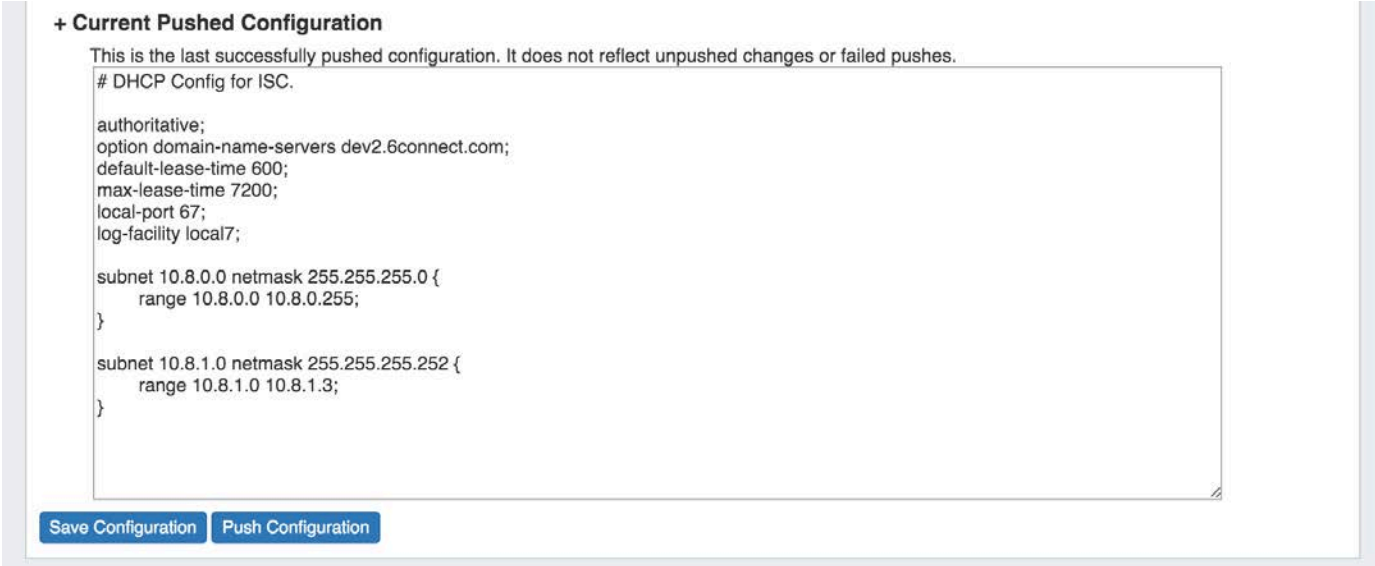
For **Direct Assign**: select IPv4 / IPv6 and enter the block in CIDR format, add free lines if desired, then click "Add Pool".



Current Pushed Configuration

Expanding "Current Pushed Configuration" allows you to view the detailed text (read-only) of the last sucessfully pushed configuration.

Current Pushed Configuration does not reflect any un-pushed changes or failed pushes.



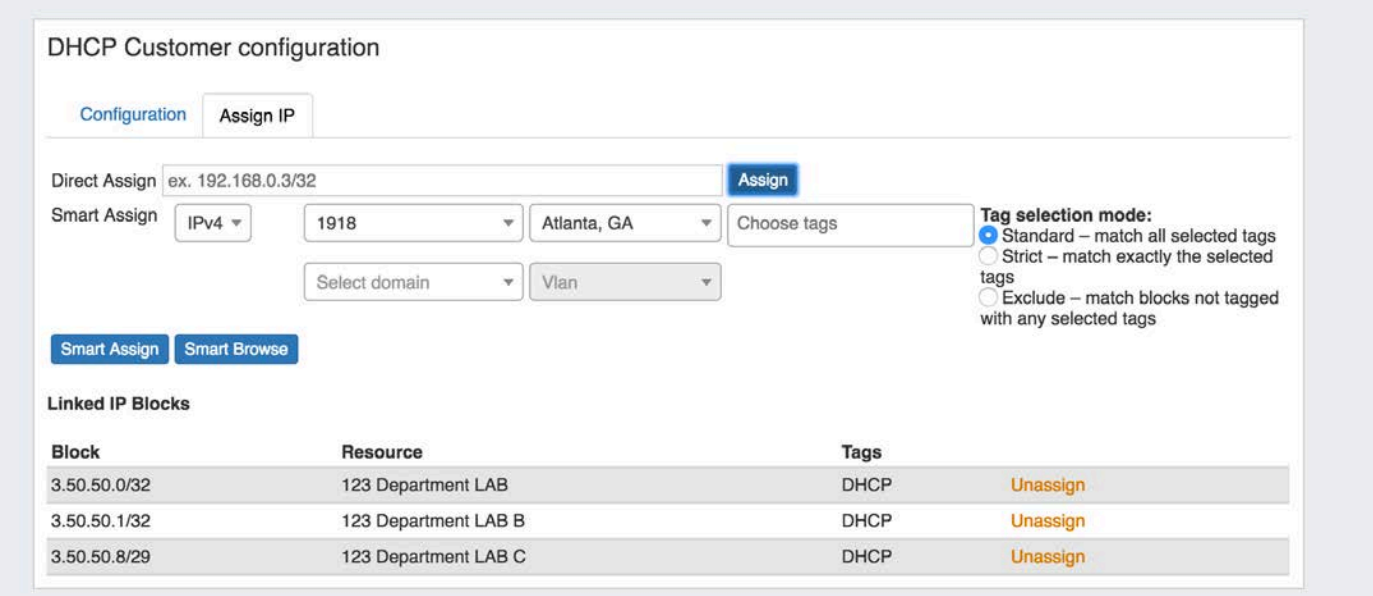
Saving/Pushing DHCP Server Configurations

It is recommended that you save your configuration after changes. Save your changes by clicking the "Save Configuration" button at the bottom of the DHCP Management Gadget.

When you Push a Configuration (by clicking "Push Configuration") the configuration is automatically saved.



Working with the DHCP Customer Configuration Gadget



The DHCP Customer Configuration Gadget allows users to assign IP aggregates to the DHCP server pools and generate DHCP Server Configuration changes.

These configurations are then sent to the associated DHCP server Management Gadget as "Unpushed Configurations", where they may be held until a manual or schedule push occurs.

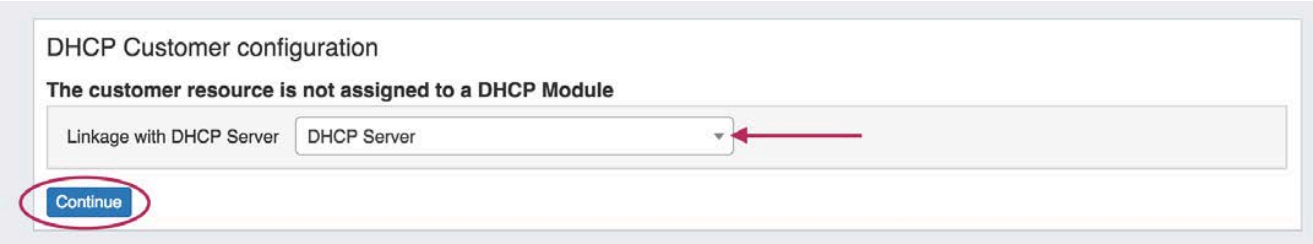
Setting up the DHCP Customer Configuration Gadget

Before using the DHCP Customer Configuration Gadget, the following should be set up in ProVision:

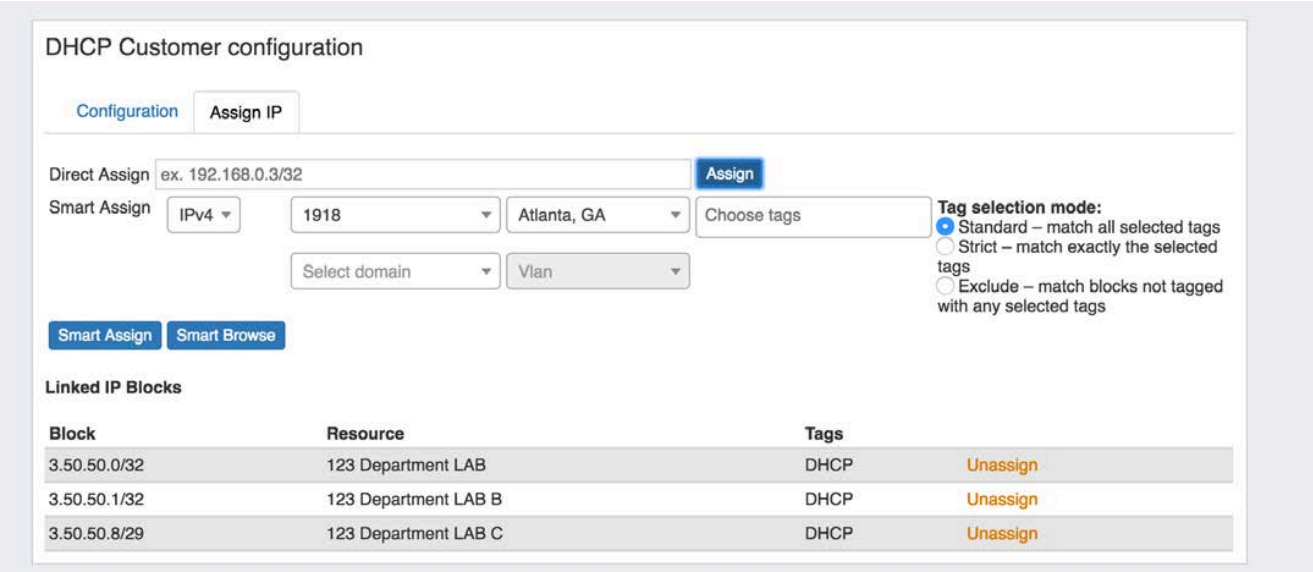
- The associated DHCP server should be created in ProVision and set up in the DHCP Management Gadget (See: DHCP Tab).
- Add the DHCP Customer Configuration Gadget to the desired Section. You may want to create a specific "DHCP Customer" Section for DHCP customer entries (See: Customizing Sections).
- Have, or set up IPAM Aggregates with subassignable blocks from the **IPAM** Tab with the desired IP space type, RIR, Region(s), Tags, and any desired VLAN criteria. Regions are a required field when assigning IPs from the DHCP Customer Configuration Gadget.

Using the DHCP Customer Configuration Gadget

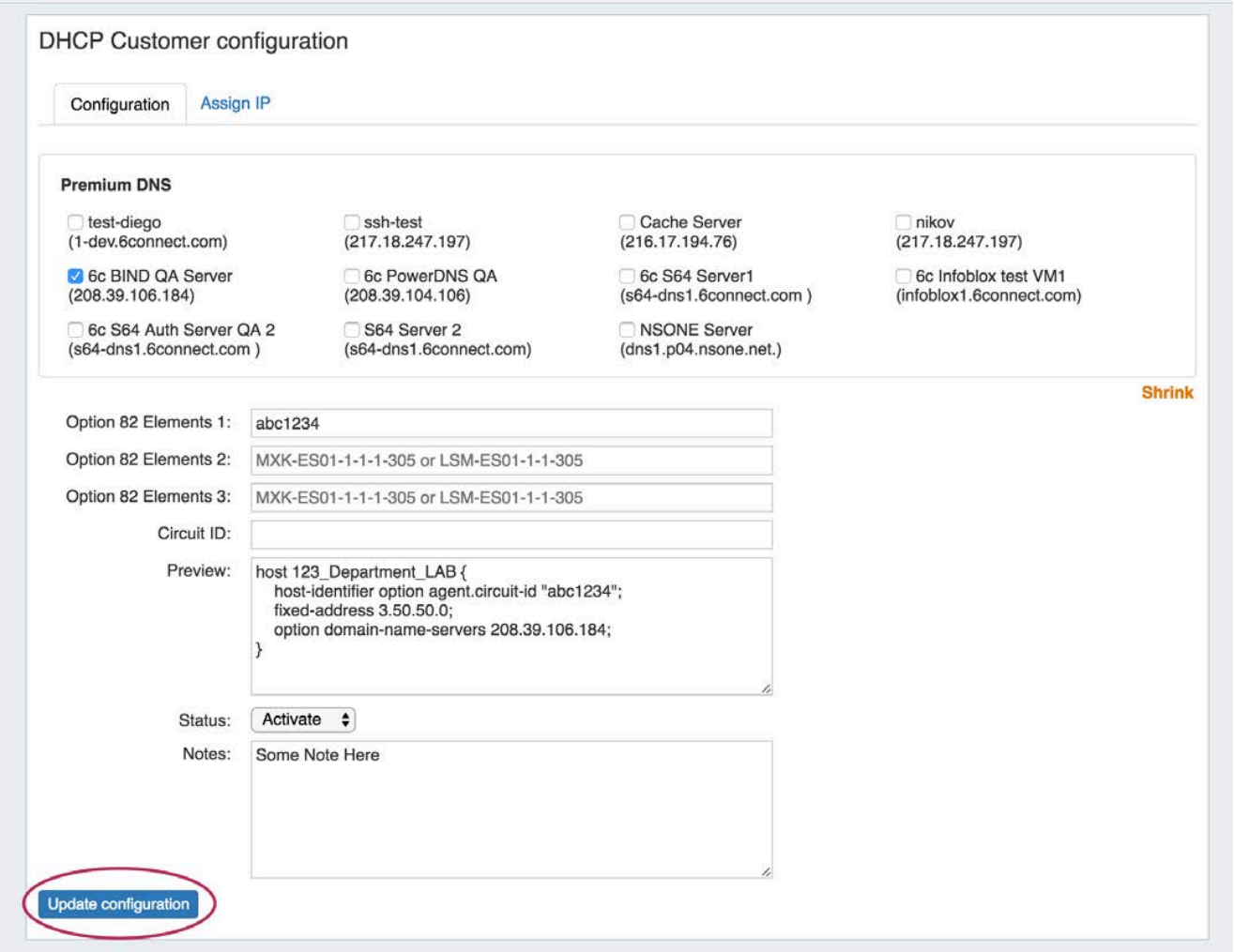
Step 1) Link the Gadget with the desired DHCP Server



Step 2) Assign IP's for Pools from DHCP Aggregates



Step 3) Set up configuration information - add Option 82 Elements, Circuit ID, and notes. Use the "Preview" field to confirm the accuracy of the data, and select the status as "Activate" or "Terminate".
Once saved, the updated configuration will be sent to the DHCP Management Gadget.



Step 4) an admin user may manually push the updated configuration from the DHCP Management Gadget, or use a scheduled DHCP push task to automate the pushes.

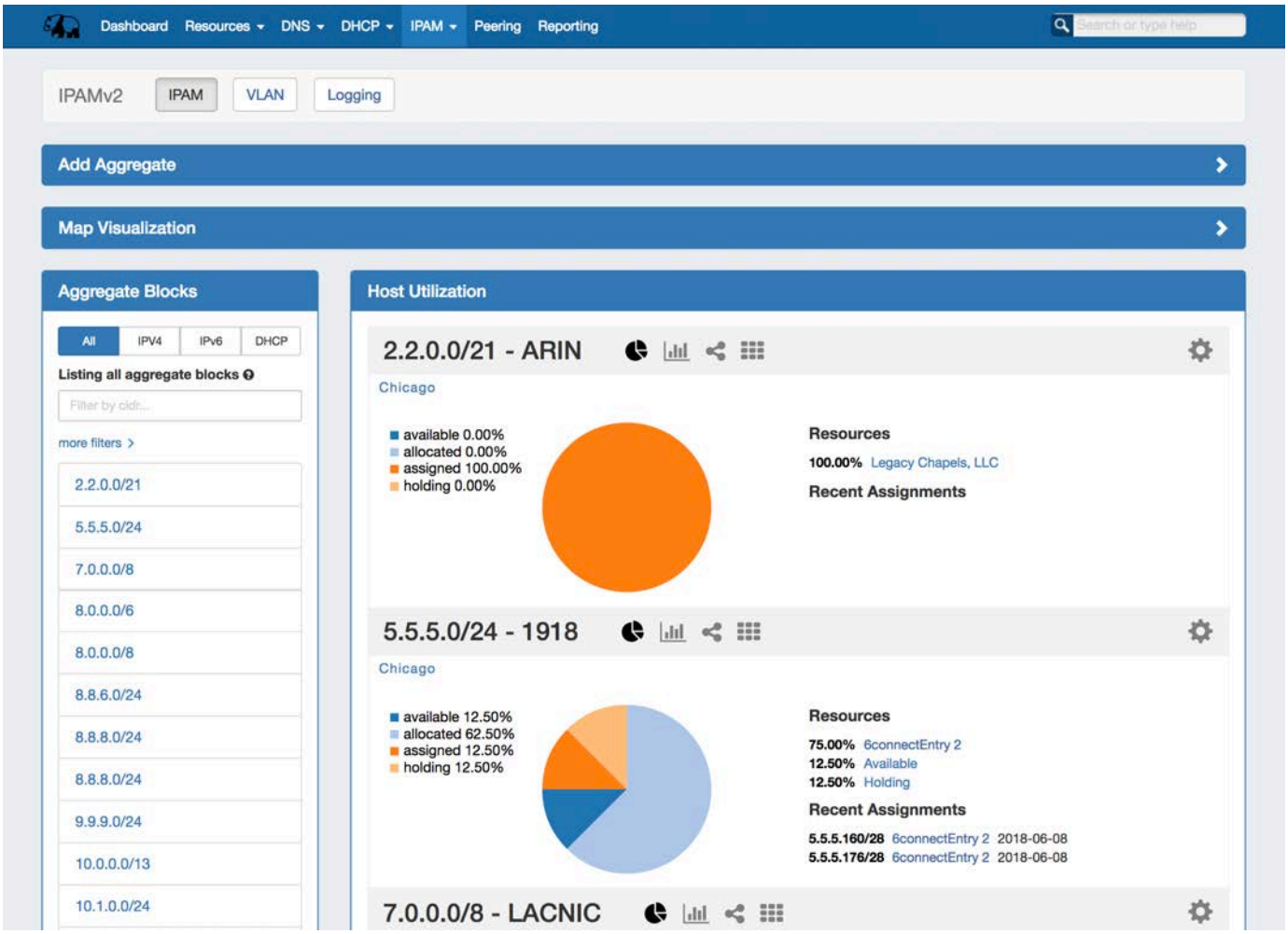
See Gadgets - DHCP Customer Configuration for more details.

Additional Information

For more detail on advanced DHCP topics and the DHCP API, see API Module - DHCP.

IPAM Tab

IPAM



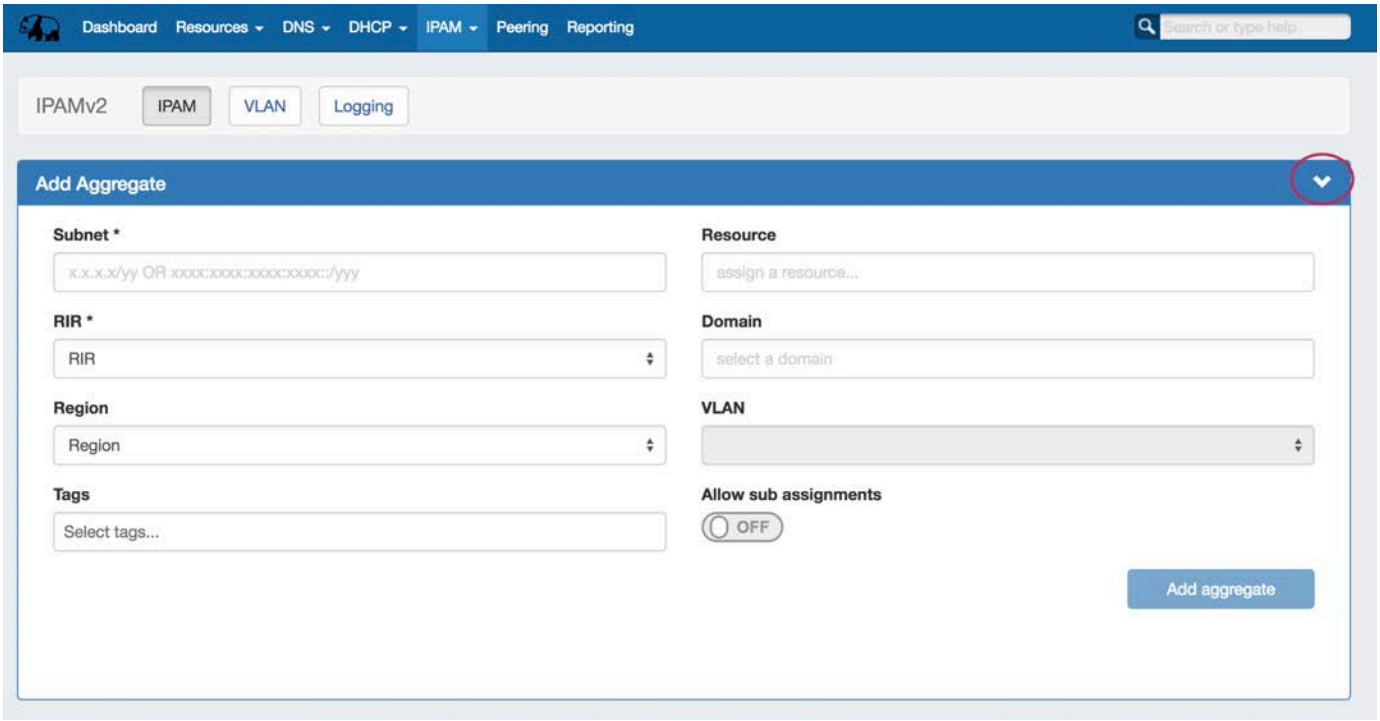
The **IPAM** tab provides a listing of aggregate blocks and tools to add and manage aggregates.

- IPAM
  - IPAM Tab Overview:
    - Add Aggregate:
    - Map View:
    - Aggregate Blocks List:
      - Search / Filter Aggregates
      - Merge Aggregates
    - Top Level Aggregate Host Utilization Box:
  - Working with IP Blocks
    - Additional Information

IPAM Tab Overview:

Add Aggregate:

Opens a menu to add an aggregate block with options for RIR, VLAN, Tags, Region, Resource, and enabling Sub-Assignments / DHCP Aggregate.

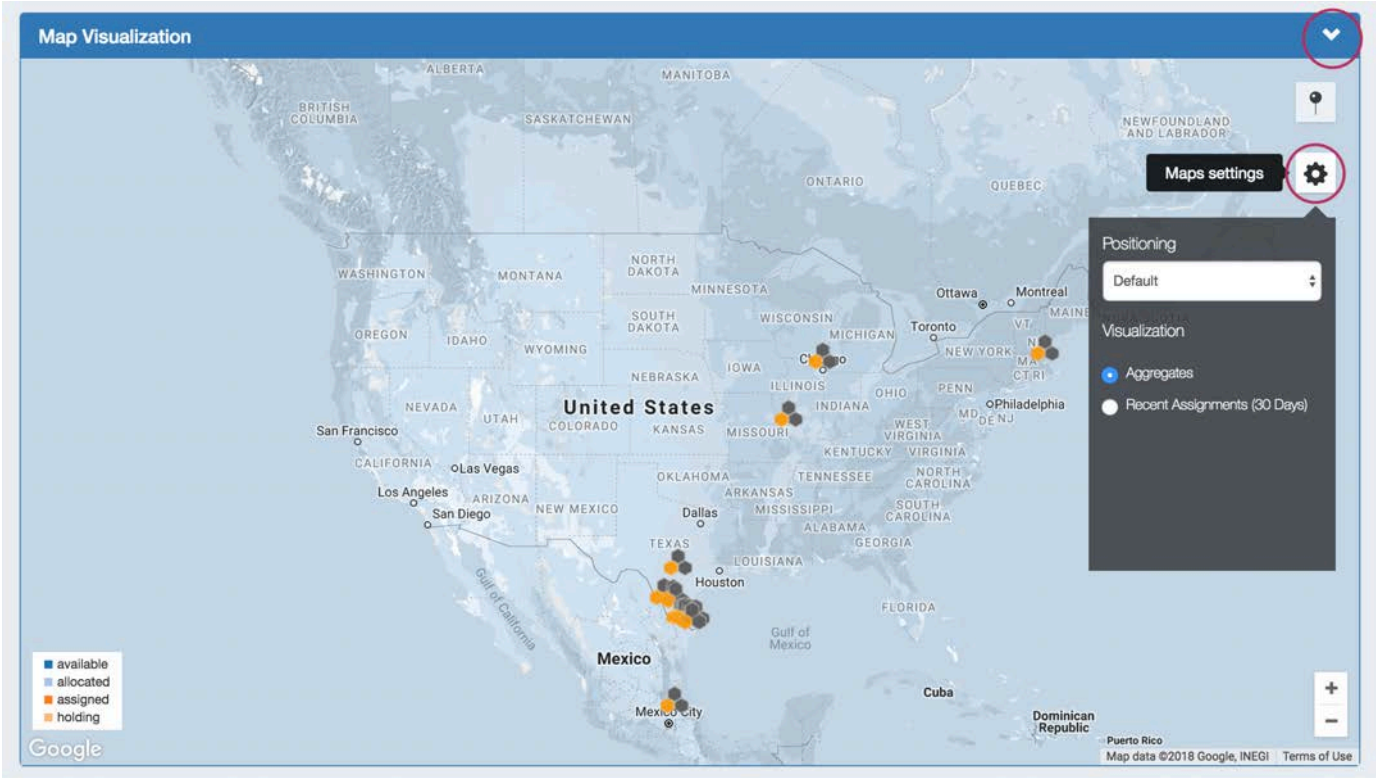


Map View:

The IPAM aggregate map view may be enabled / disabled from the IPAM Configuration page. Once enabled, a map module will be able to be expanded in the IPAM Tab.

Map View Overview:

Map View will show aggregate locations / recent assignments in geographical format, based on regions assigned to those aggregates. Clicking on a location icon will bring up a current utilization graph and a link to that aggregate's IPAM Manage page if in Aggregates view, or assignment details if in Recent Assignments view.



The map view may be set by either selecting a predefined map area from the "Maps Settings" "Positioning" dropdown (US, US & Europe, South America, All), or chose a custom area set by zooming / navigating to the desired map area.

To save the currently viewable area as the Default view, click the pin icon to "Save current position as Default".



Map View Settings:

Visualiation: Select "Aggregates" to view all available aggregates with region data, or "Recent Assignments" made within the past 30 days.

Positioning Dropdown: Select a predefined area for the map view range: US, US & Europe, South America, or All.

Save current Position as Default: Click the pin icon to save the current map view range as the default view.

Aggregate Detail Box: Provides utilization data and a shortcut link to the IPAM Manage screen for that aggregate if in "Aggregates" view, or assignment details if in "Recent Assignments" view.

Regions must be enabled, the region address field populated, and regions assigned to aggregates / blocks for the IPAM Map view to populate data. See Edit Regions for additional detail.

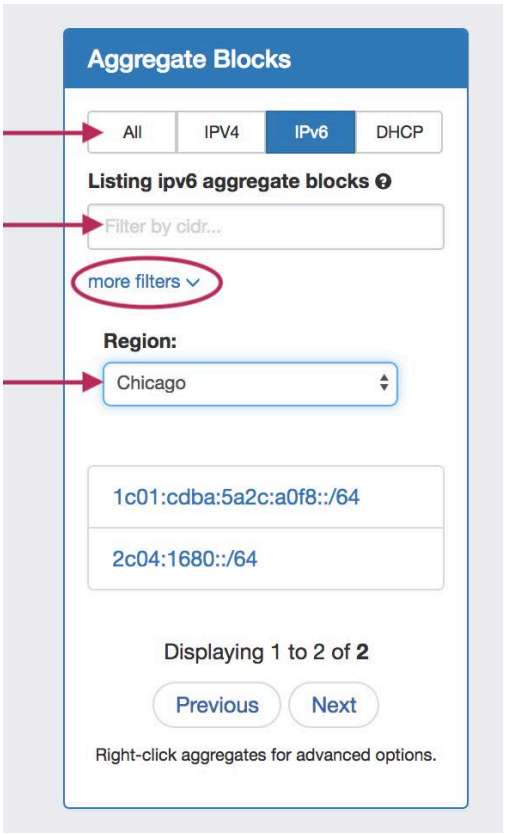
Aggregate Blocks List:

Provides a searchable listing of all aggregate blocks in the left sidebar. Selecting "All / IPv4 / IPv6 / DHCP" will filter the visible aggregates in the center of the page.

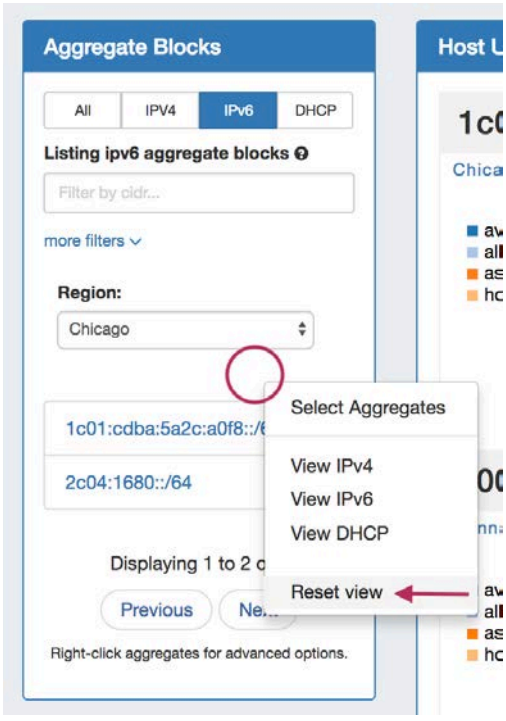
Search / Filter Aggregates

Search for and filter IPAM Aggregates from the Aggregate Blocks List:

- Select "All", "IPv4", IPv6", or "DHCP" to sort the list by type (You may also access the type filters by right clicking anywhere in the Aggregate Blocks box)
- User the filter / search box to search by CIDR, Region, or RIR
- Search by Region by clicking the "more filters" link
- A shortcut to IPAM Manage for an aggregate is provided by clicking the "open" link that appears when hovering over an aggregate block in the list.

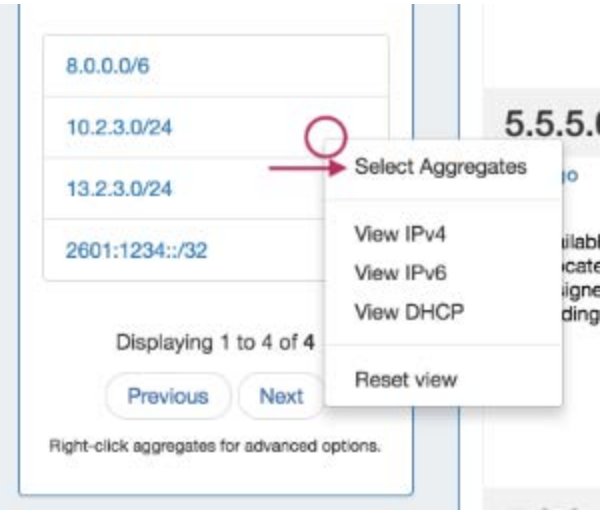


To reset the list, right click anywhere in the Aggregates Blocks list and select "Reset View".

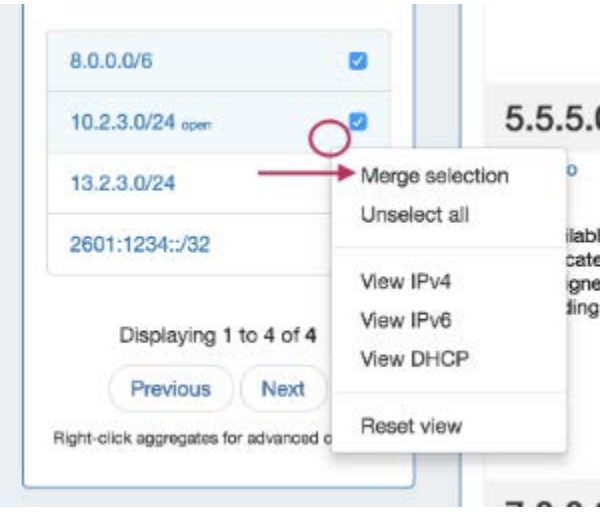


Merge Aggregates

Top Level Aggregates may be merged from the Aggregate Blocks List by right clicking on the list, and selecting "Select Aggregates" from the menu.



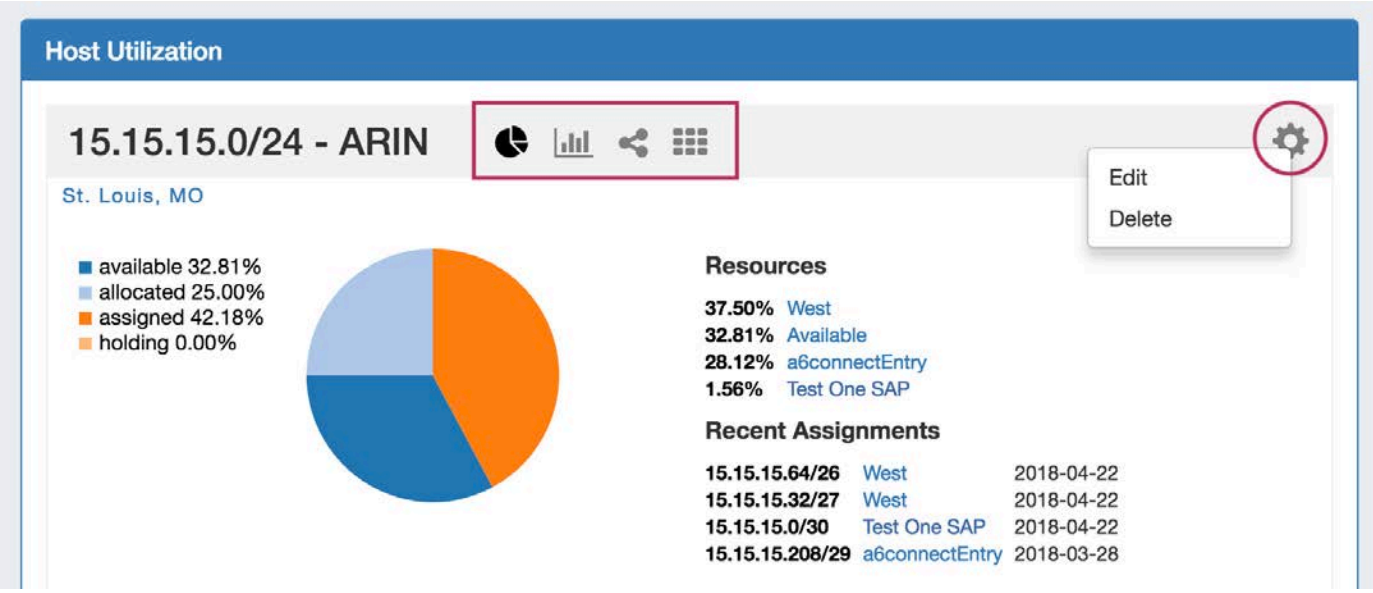
Then, click the check boxes for the two aggregates you wish to merge, and then right click and select "Merge Selection".



ProVision will pre-test the merge, and if valid, give you the option to proceed. To Exit out of the selection prior to merging, select "Unselect All".

Top Level Aggregate Host Utilization Box:

Provides detailed information on that aggregate, including percentage breakdowns, the top five Resources assigned under that aggregate, recent assignments, and additional charts.



Top Level Aggregate Action Menu (Gear icon):

Edit: Opens IPAM Manage for the aggregate. See Working with IP Blocks for information on working in IPAM Manage. Clicking on the Block CIDR in the header also opens the IPAM Manage page.

Delete: Deletes the Aggregate (provides a warning).

Aggregate Charts: Four Aggregate chart views are available - Pie, Bar, Tree, and Grid view. Switch between each chart by clicking on the respective icons in the Aggregate header.

Working with IP Blocks

For additional information on performing IPAM tasks and working with blocks, see the following sections:

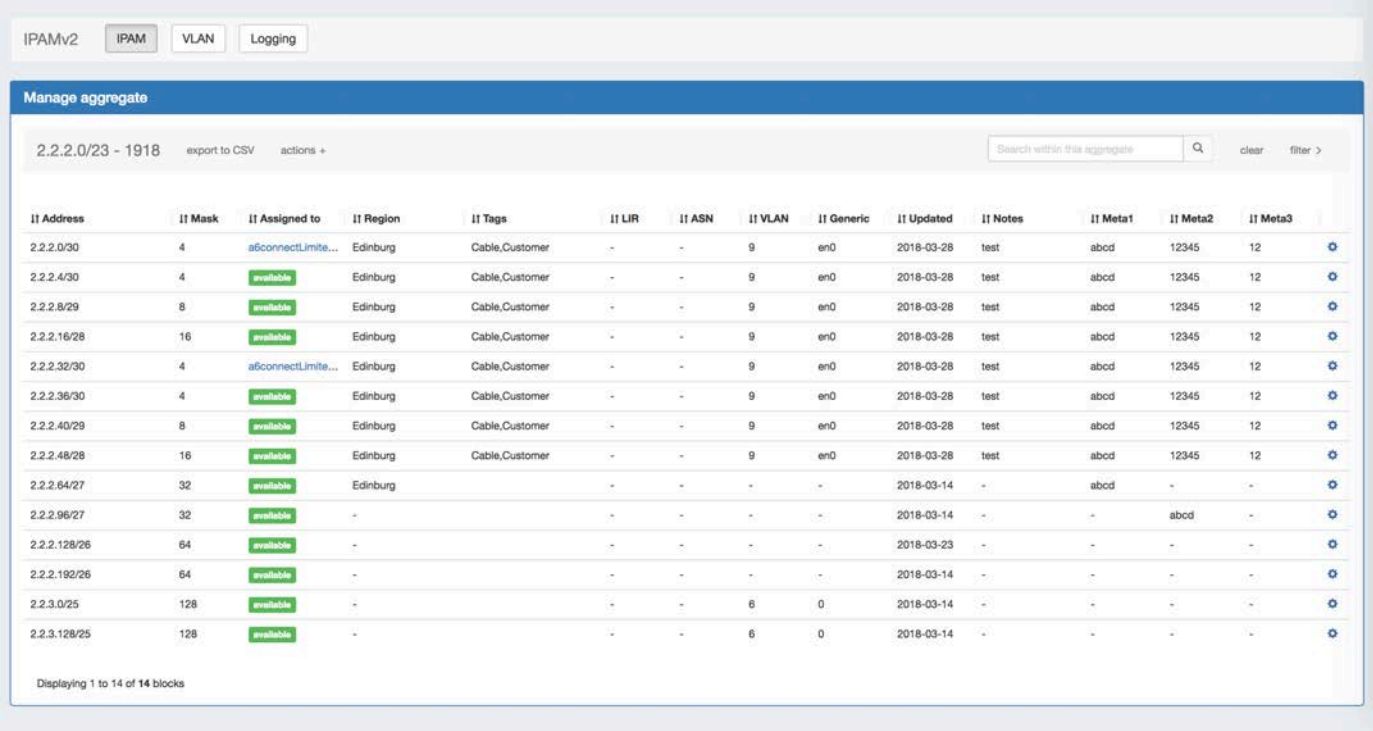
Additional Information

- Working with IP Blocks
- Working with IP Rules

Working with IP Blocks

Working with IP Blocks

IP Blocks are edited and managed in either the IPAM Manage page or IPAM Gadget. Both IPAM Manage and IPAM Gadget have customizable column names, order, and visibility, as well as options for user-defined meta columns. For details on setting up customization of the IPAM Manage / IPAM Gadget block list, see IPAM Parameters.

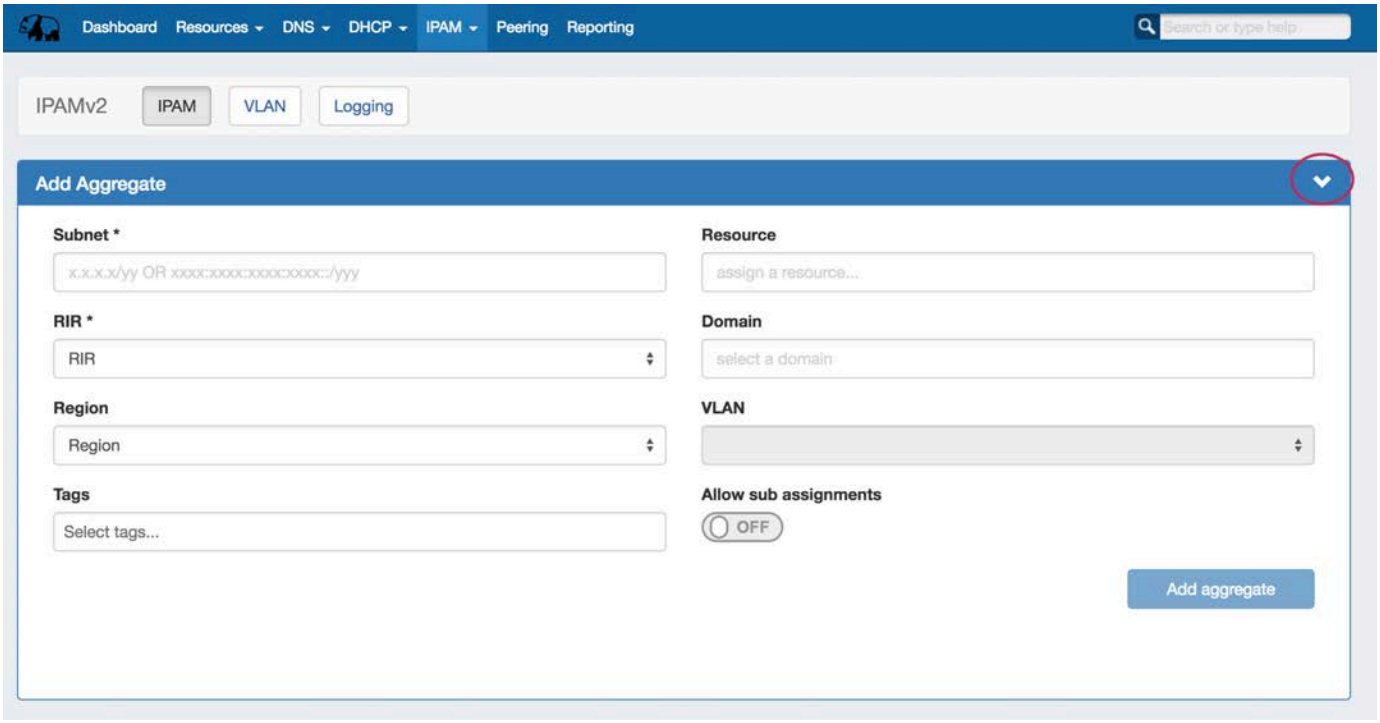


Common tasks in IPAM include creating IP Aggregates, Auto Split / Clean up an aggregate, setting alerts, and managing individual blocks.

- Working with IP Blocks
  - Adding/Deleting IP Aggregates
  - Open IPAM Manage for an Aggregate
  - Working in IPAM Manage
    - Aggregate Actions
      - Auto Split Aggregate
      - Clean Up Aggregate
      - Set Aggregate Alerts
  - Architecting IP Address Blocks
    - Splitting/Merging blocks manually
    - Splitting/Aggregating blocks with Templates
    - IP Block parameters and Editing Attributes
      - Edit Attributes Overview:
  - Assigning IP Space
    - Assigning Space from the IPAMv2 Gadget
      - Direct Assign:
      - Smart Assign / Smart Browse:
    - Manually Assigning Space from the IPAM Manager
  - Sub Assigning IP Space
  - Unassigning IP Space
  - Sort, Filter, and View Block Hierarchy in IPAM Manage
- Additional Information

Adding/Deleting IP Aggregates

On the main **IPAM** Tab, expand the "Add Aggregate" module.



Once clicked, you get a more detailed screen to add an aggregate block. Enter in the CIDR and RIR, at minimum, and Region, Tags, Resource, Domain, VLAN, and subassignment status if desired.

When done, click "Add Aggregate". The Aggregate will be listed under the Host Utilization and Aggregate Blocks list of the IPAM Tab.

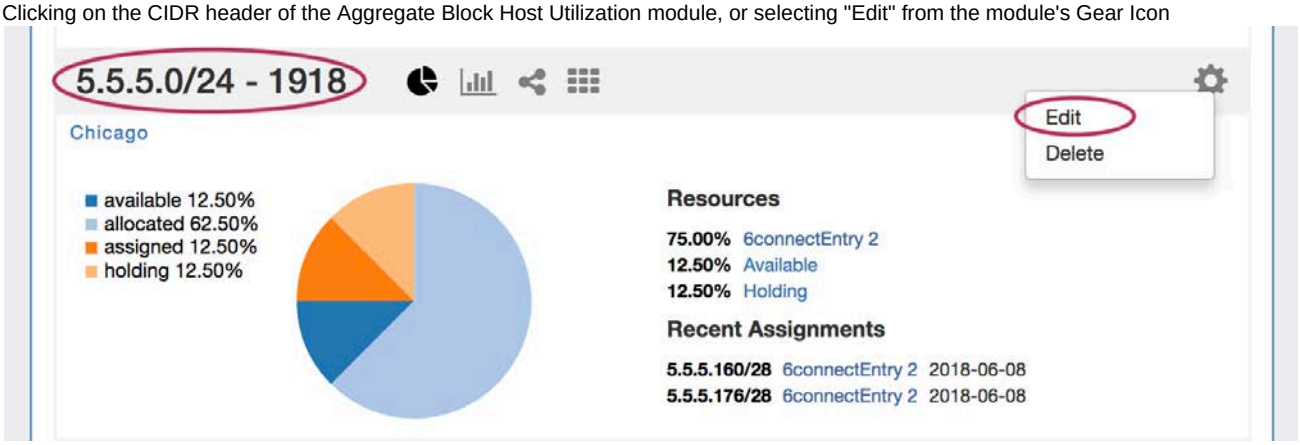
Open IPAM Manage for an Aggregate

Open IPAM Manage for a block by:

- Searching for the block in the Global search, and clicking on the resulting CIDR
- Clicking the "Open" link in the aggregate blocks list



- Clicking on the CIDR header of the Aggregate Block Host Utilization module, or selecting "Edit" from the module's Gear Icon

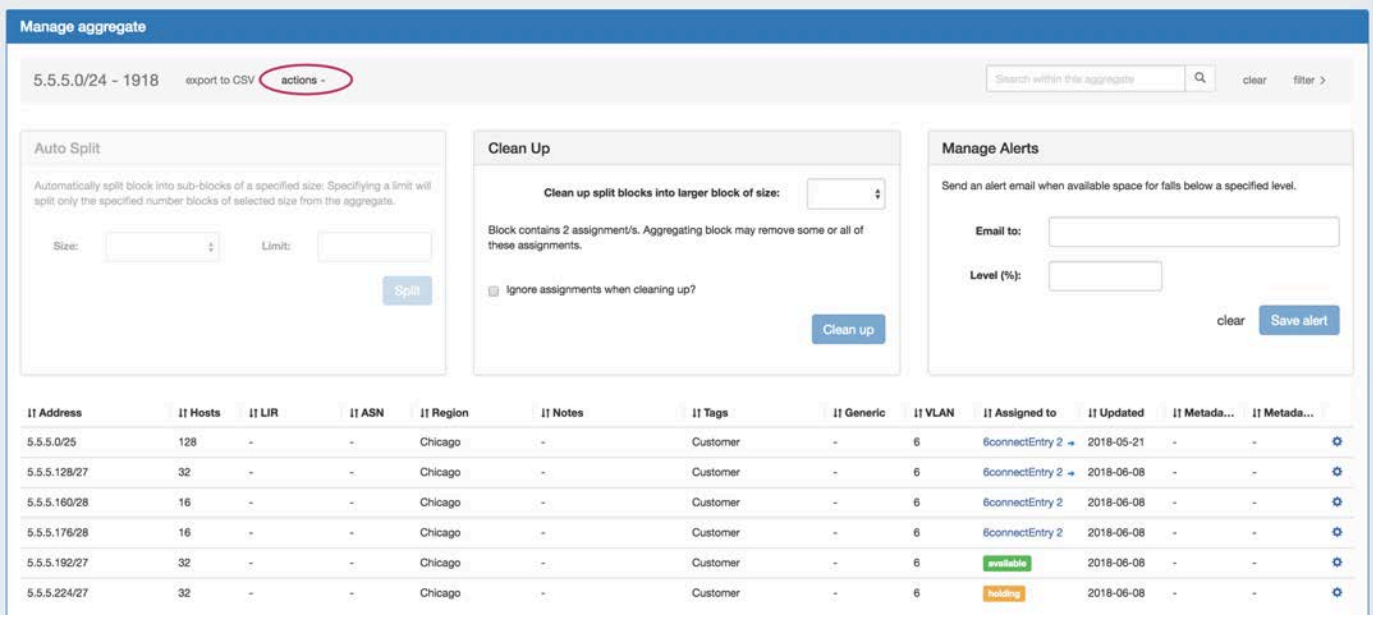


Working in IPAM Manage

In IPAM Manage, you can manage aggregate level actions, such as Split , Clean Up, Alerts, and Export, as well as viewing and managing individual blocks.

Aggregate Actions

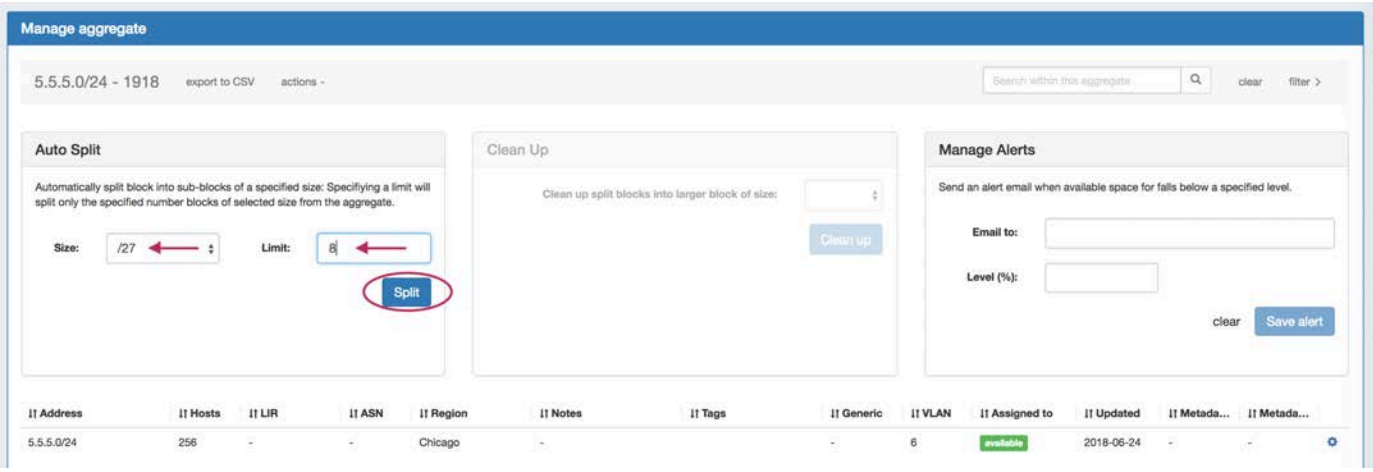
Open up aggregate level actions in IPAM Manage by clicking on the "Actions" link in the Manage Aggregate header. Three options are available - Auto Split, Clean Up, and Manage Alerts.



Auto Split Aggregate

Auto Split an available aggregate into blocks of a specific size, up to a specific quantity of blocks by selecting "Size" and "Limit" under "Auto Split".

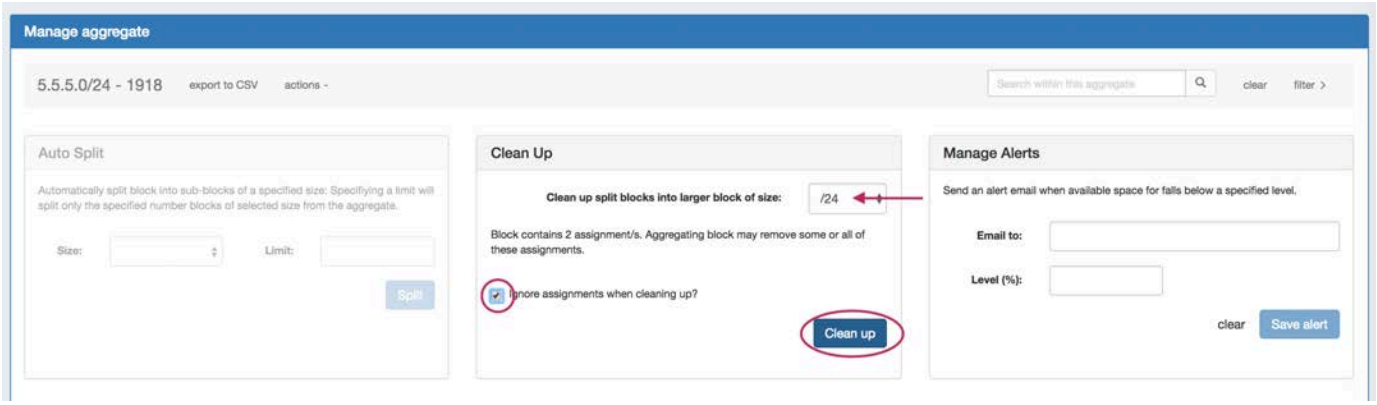
The block limit must be a power of two, or an error will appear. When done, click "Split".



Clean Up Aggregate

Clean up an Aggregate into fewer blocks of a specific size, by selecting your desired block size and clicking "ignore assignments" (if blocks have assignments that would prevent merging).

When done, click the "Clean Up" button. All blocks will be merged to the set size and reset to available.

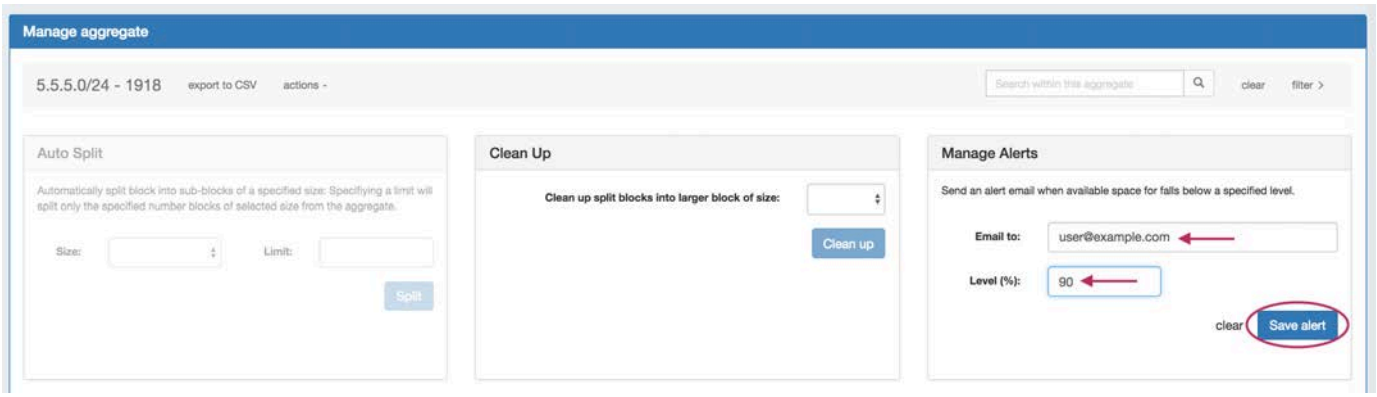


Set Aggregate Alerts

Enable or clear aggregate alert emails for the aggregate by setting an email address and utilization percentage level.

When the aggregate is utilized to the set percent, an email alert will be sent to the provided address.

Note: the "IPAM Alerts" task must be created and set to the desired notification frequency in the Admin Scheduler Tab in order for the emails to be sent.



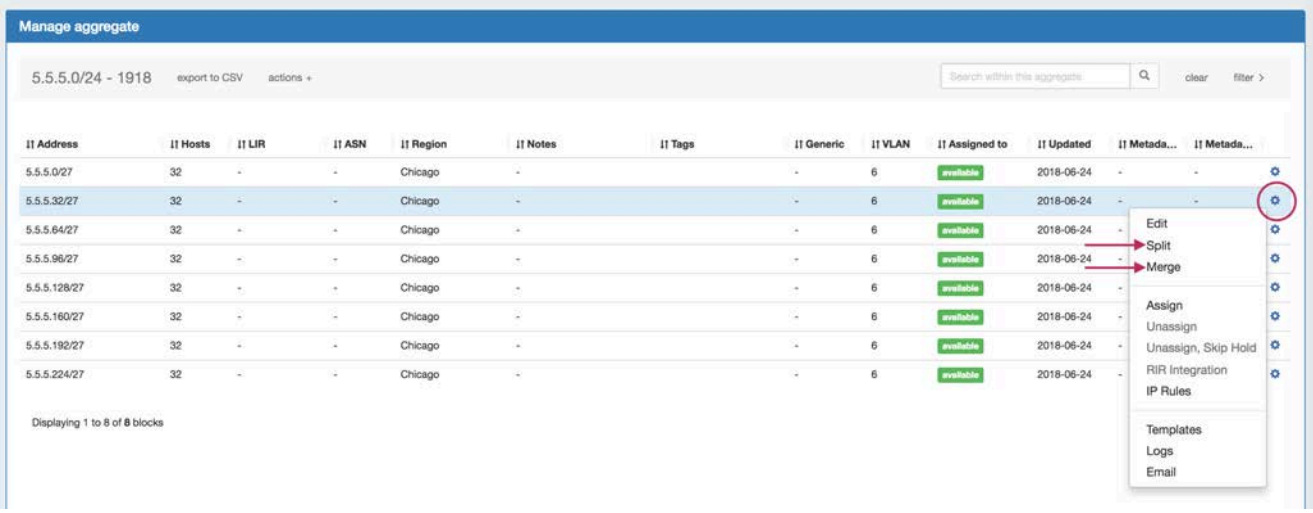
When done, click "Save Alert". To clear an existing alert, hit "Clear Alert", and the alert information will be removed and status saved.

Architecting IP Address Blocks

Splitting/Merging blocks manually

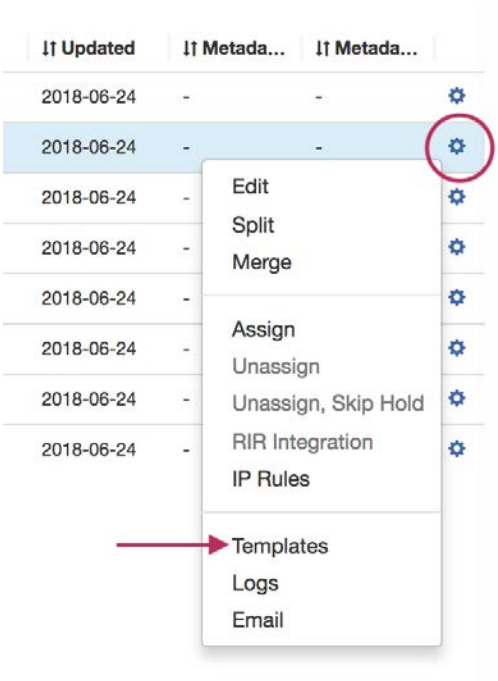
To split a block manually - While in the IPAM Mange screen, click on the Action Menu (Gear icon), or right-click anywhere in the row for the available block you wish to modify.

In the menu that appears, select the "Split" function. To aggregate blocks, select "Merge" from the same menu.



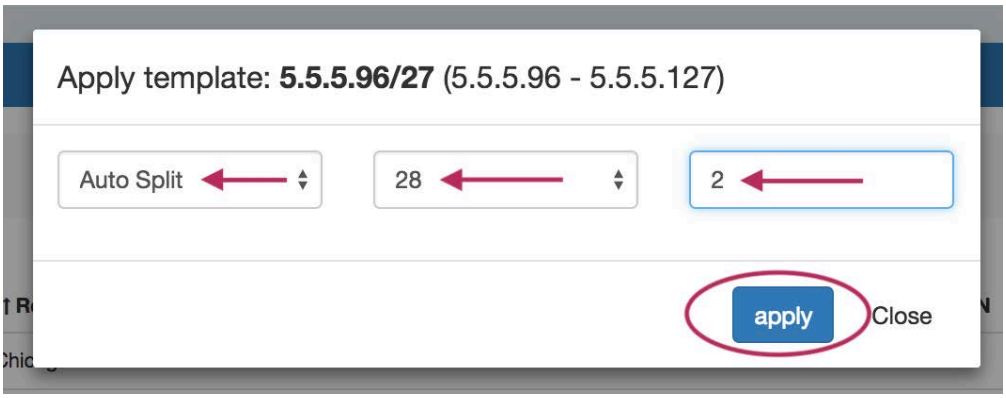
Splitting/Aggregating blocks with Templates

Templates for a block are available under the Action Menu (wrench icon or right-click) for that block.



There are two templates available: Cleanup (Only available for parent blocks) which auto-aggregates the block, and Auto-split.

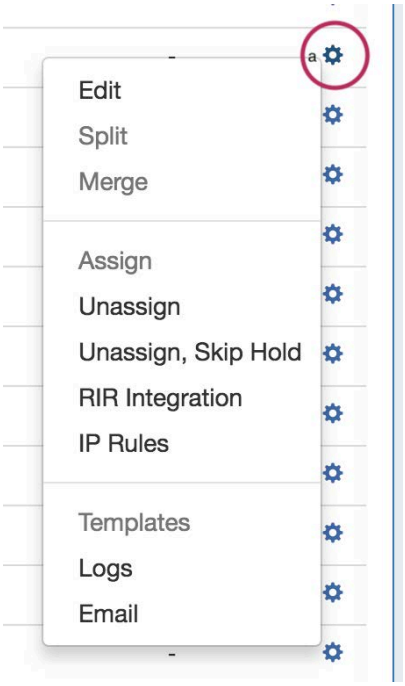
Select the available template action, the mask, and the limit. When done, click the "Apply" button.



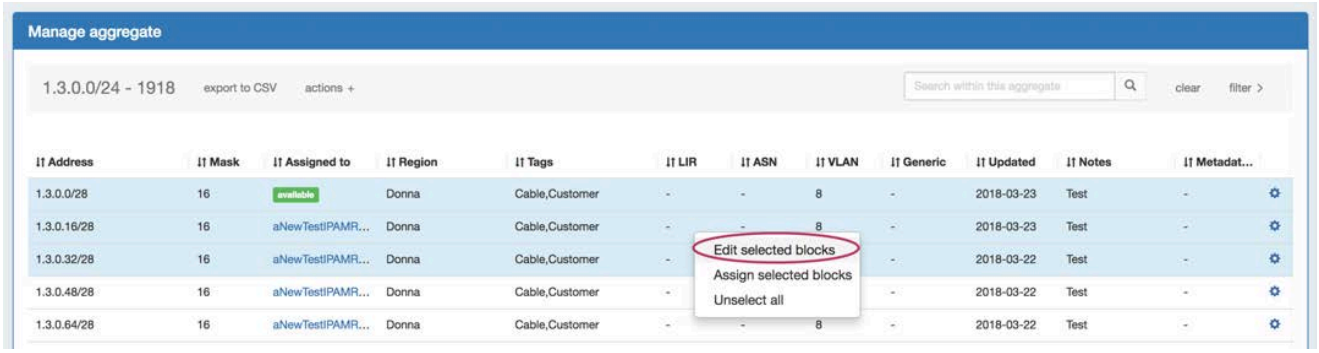
You can also use the "Templates" option from the Action Menu on the IPAM Manage screen for the specific block.

IP Block parameters and Editing Attributes

Select the "Edit" option from the Action Menu (or double click on the block row) for a given block to get the Edit Attributes menu.



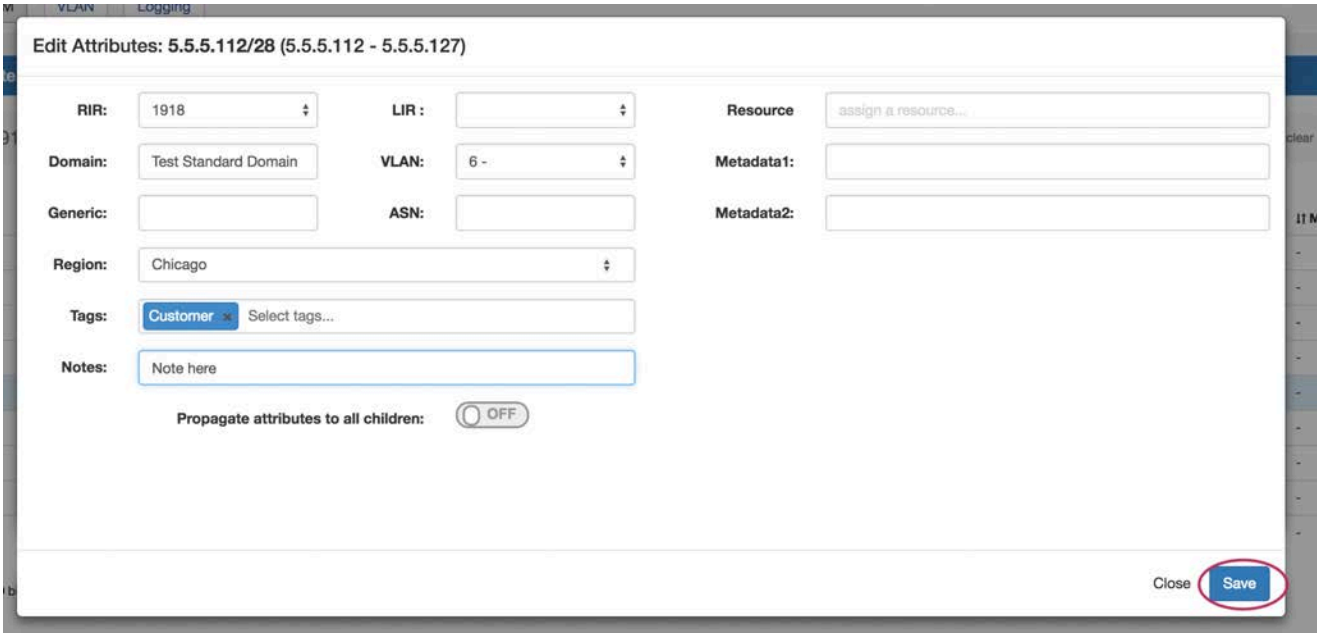
To edit multiple blocks at a time, shift-click each block you wish to edit - they will highlight in blue. Once selected, right-click on any highlighted block to open the multi-edit attributes screen.



Once selected, right-click on any highlighted block to open the multi-edit attributes screen.

From here you can set a variety of attributes for a given block. These values are also customizable from the Admin screen - see IPAM Administration and IPAM Parameters to customize, enable, or disable the available parameters.

After editing the desired attributes for the block, simply hit "Save".



Edit Attributes Overview:

RIR: The RIR associated with the aggregate / block.

LIR: If LIRs are set up in IPAM Admin for the associated RIR, they will be selectable in this dropdown. See LIR Management and Use.

Domain: The VLAN Domain to associate with the block. To set up domains in ProVision, see VLAN Administration.

VLAN: VLAN information for the block, must have domain selected to view available VLANs.

Generic: This is a customizable text field that can be used to track information specific to your needs. It can be filtered in the IPAM Manage screen.

The header, display, and enable/disable settings for this field are set under IPAM Configuration in the IPAM Administration section.

ASN: The ASN to associate with the block.

Region: Select the region from the drop down menu. Regions can be added and customized in the IPAM Admin section of ProVision - see IPAM Administration and IPAM Parameters.

Tags: Tags can be set under Edit Tags in the IPAM Administration section.

Notes: Freeform text field for additional information you wish to capture.

Resource: The resource assigned to the block.

Metadata1 - Metadata 10: User-defined text fields, as set in IPAM Administration Edit Columns.

Allow Subassignments: When editing a block that has been assigned, toggling this setting to "On" allows for further subassignments, indicated by a blue arrow next to the assignment in the Manage screen. Note: Subassign status cannot be changed if a block has children.

Propagate Attributes to Children: Toggle this setting to "On" when editing a parent block to carry through attribute changes to all children of that block. To view parent blocks, simply ensure that top level or all masks are selected in the Filter menu in the IPAM Manage screen.

Note: The VLAN of a child cannot be different from that of its parent, so for mutil-level situations (Parent -> Child -> Grandchild), VLAN should be updated at the top tier parent level.

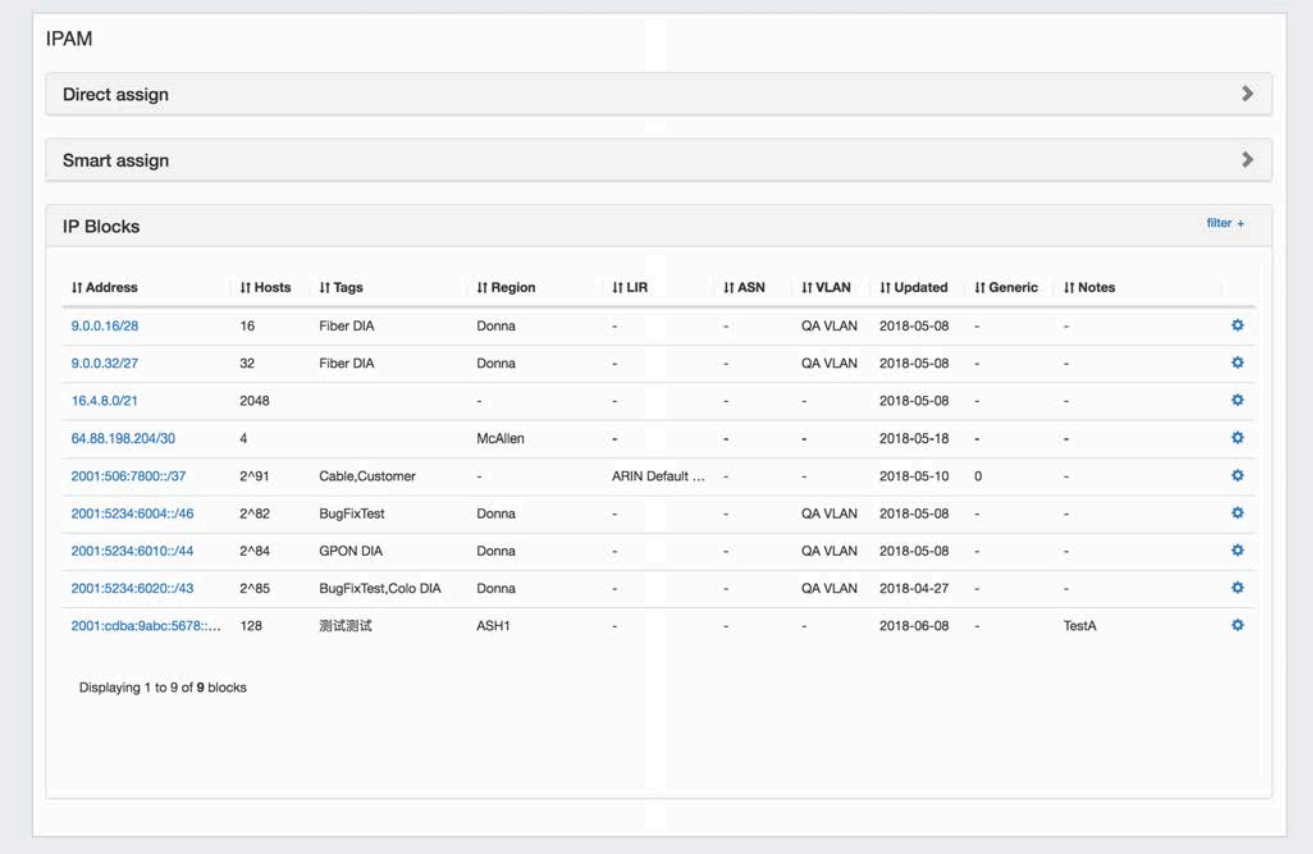
Assigning IP Space

There are two areas where you can assign IP Space: in the IPAM Gadget for the particular Resource, or through IPAM Manage for manually assigning a block to a resource. The IPAM Gadget allows for more detailed assignment options including Direct Assign, Smart Browse, and Smart Assign with advanced options, and is the primary tool for space assignment.

Assigning Space from the IPAMv2 Gadget

The IPAM Gadget is accessed from a Resource Entry page, once enabled for the Section (to add Gadgets, see Customizing Sections and Add Gadgets to your Section).

For additional details on the IPAMv2 Gadget, see Gadgets.



Address	Hosts	Tags	Region	LIR	ASN	VLAN	Updated	Generic	Notes
9.0.0.16/28	16	Fiber DIA	Donna	-	-	QA VLAN	2018-05-08	-	-
9.0.0.32/27	32	Fiber DIA	Donna	-	-	QA VLAN	2018-05-08	-	-
16.4.8.0/21	2048		-	-	-	-	2018-05-08	-	-
64.88.198.204/30	4		McAllen	-	-	-	2018-05-18	-	-
2001:506:7800::/37	2^91	Cable,Customer	-	ARIN Default ...	-	-	2018-05-10	0	-
2001:5234:6004::/46	2^82	BugFixTest	Donna	-	-	QA VLAN	2018-05-08	-	-
2001:5234:6010::/44	2^84	GPON DIA	Donna	-	-	QA VLAN	2018-05-08	-	-
2001:5234:6020::/43	2^85	BugFixTest,Colo DIA	Donna	-	-	QA VLAN	2018-04-27	-	-
2001:cdba:9abc:5678::...	128	测试测试	ASH1	-	-	-	2018-06-08	-	TestA

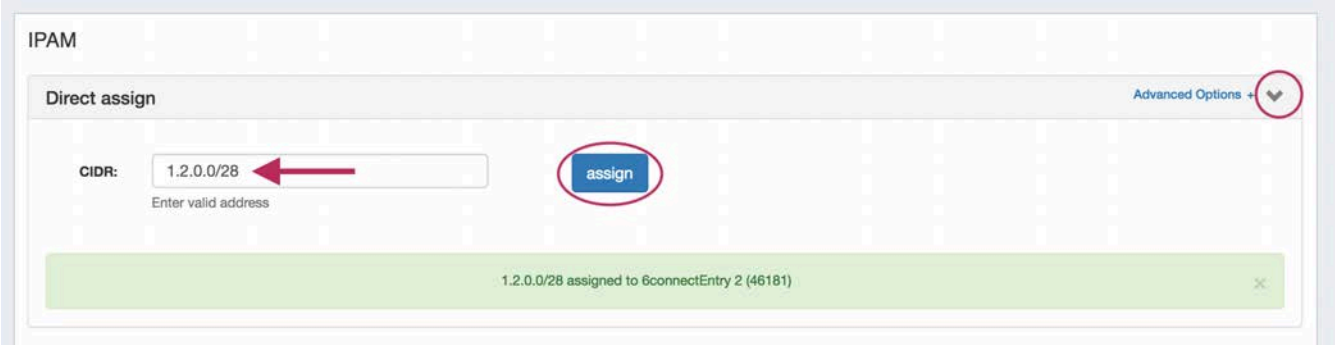
You have three options for assigning IP space using the IPAM Gadget: Direct Assign, Smart Assign, and Smart Browse.

Direct Assign:

Direct Assign

Simple Direct Assign:

Expand the "Direct Assign" module of the IPAMv2 Gadget by clicking the "Expand" arrow on the right side of the header.

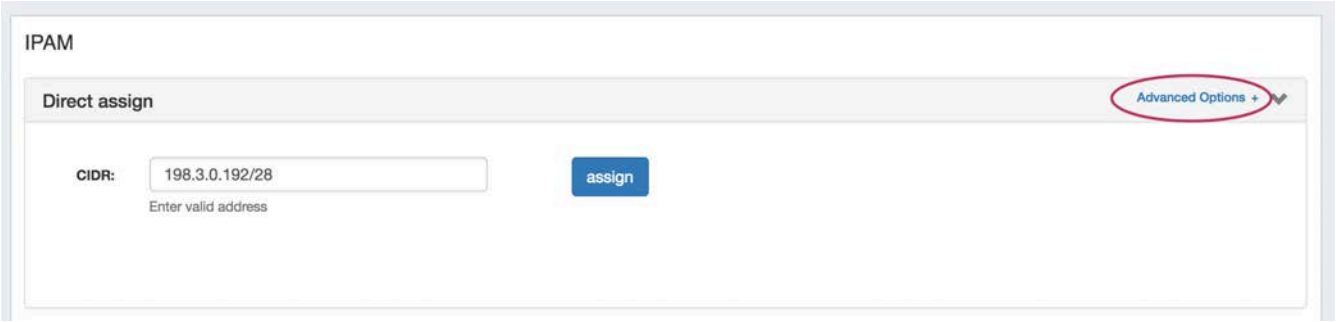


Type the desired CIDR to assign in the CIDR input box, then click "Assign". A message will appear to confirm success, or provide details on an error.

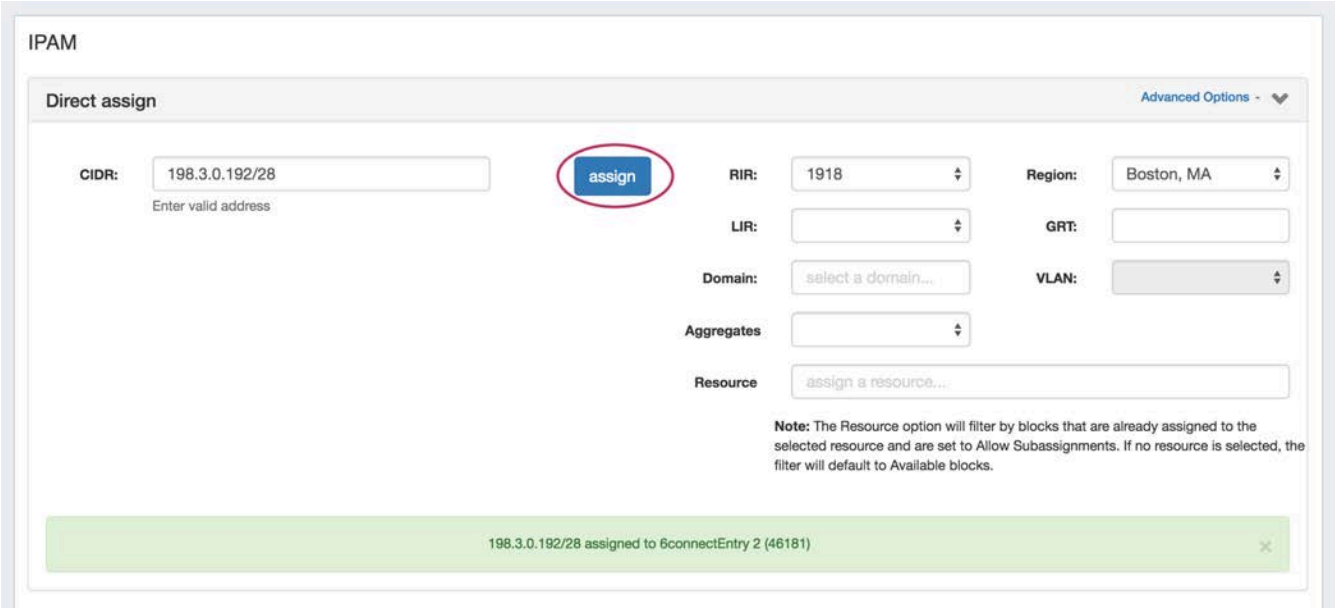
Advanced Direct Assign:

In some cases, you make need the ability to further specify a block to direct assign - such as if you have duplicate 1918 blocks. The "Advanced Options" filters allow for further detail to be provided to ensure ProVision assigns the specific block that is intended.

Expand the "Advanced Options + " link to the right of the "Direct Assign" module header.



Then, add in the desired CIDR, and set your additional filter criteria, such as RIR, region, LIR, VLAN, and so forth. When done, hit "Assign".

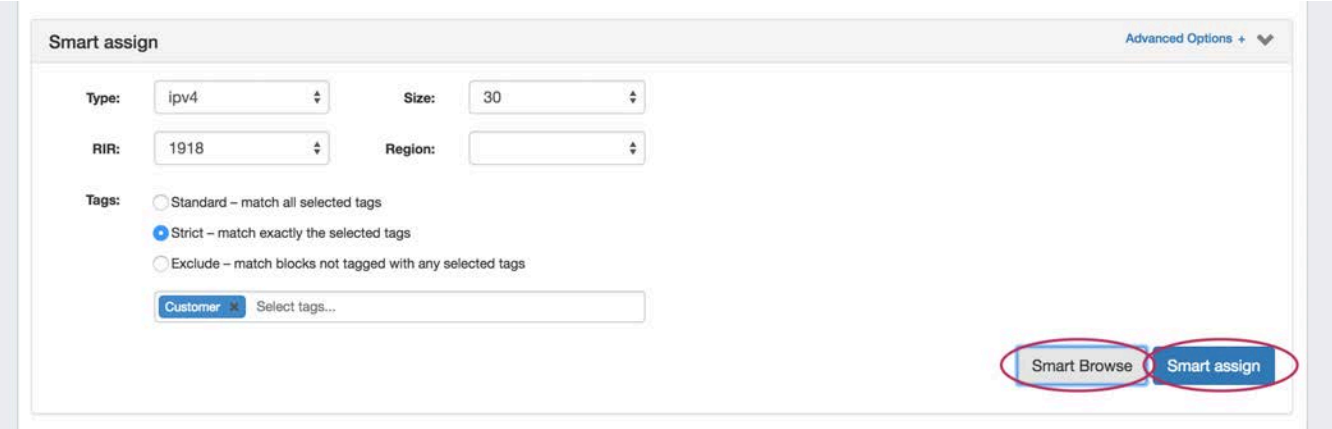


Smart Assign / Smart Browse:

Smart Assign / Smart Browse

Expand the "Smart Assign" module in the IPAMv2 by clicking the expansion arrow in the right side of the Smart Assign header.

For a basic assignment, enter at least the IP type, RIR, and block size.



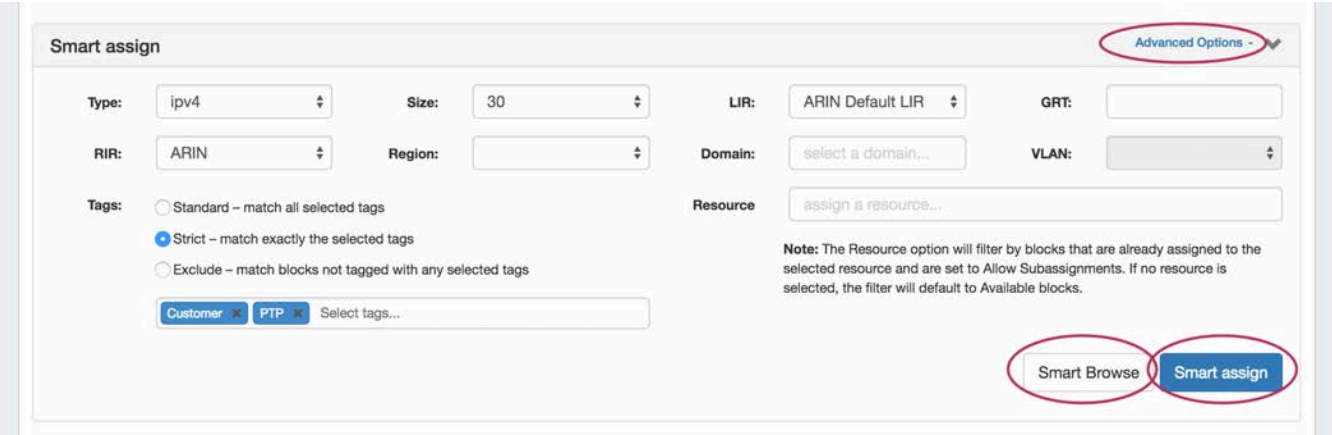
IP Tag filters are optional. IP Tag filters exist in three modes:

Standard - checks to see if the selected tag is associated with the block

Strict: limits results to blocks with exactly the selected tag(s)

Exclude - blocks without the selected tag(s)

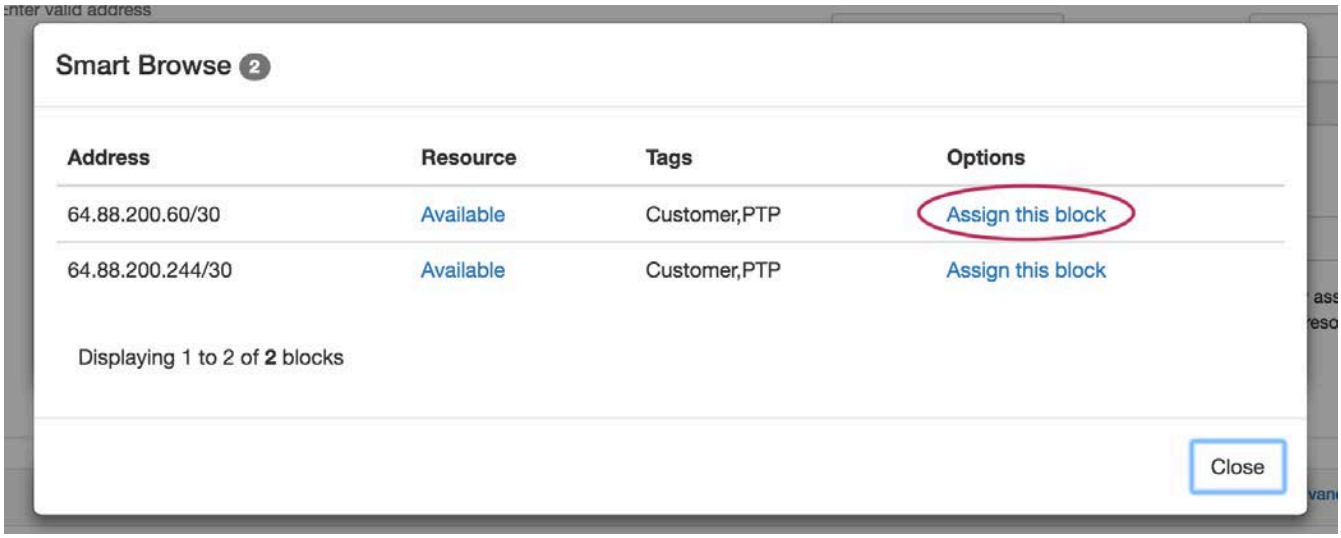
For Advanced Options, Click the "Advanced Options +" expansion link in the top right of the header. Additional filter criteria will be available to apply to both Smart Browse and Smart Assign.



Then click "Smart Browse" or "Smart Assign".

For Smart Assign, you will receive a message indicating a successful assignment, or information regarding an error. If no blocks are available meeting the criteria, the message will say so. In this case, try removing one or more of your filter criteria and try again.

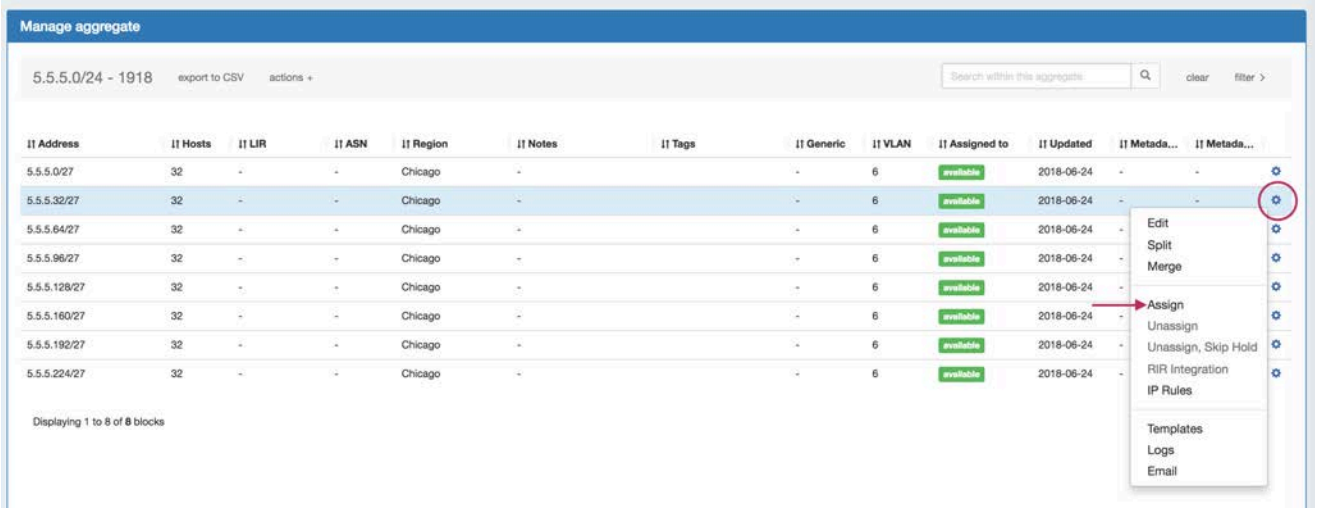
For Smart Browse, a screen will pop up showing blocks meeting your criteria. Browse to the block you want to assign, and click either "Assign this block", or "Assign from this block"(splits a larger block down).



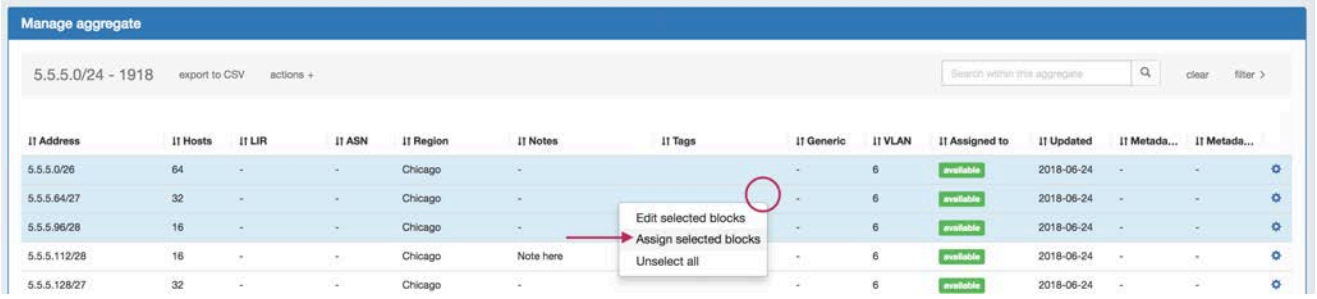
Successful assignments will be added to the "IP Blocks" list in the IPAMv2 Gadget, and are able to be managed in the IPAM Gadget, IPAMv2 Gadget, or IPAM Manage Screen.

Manually Assigning Space from the IPAM Manager

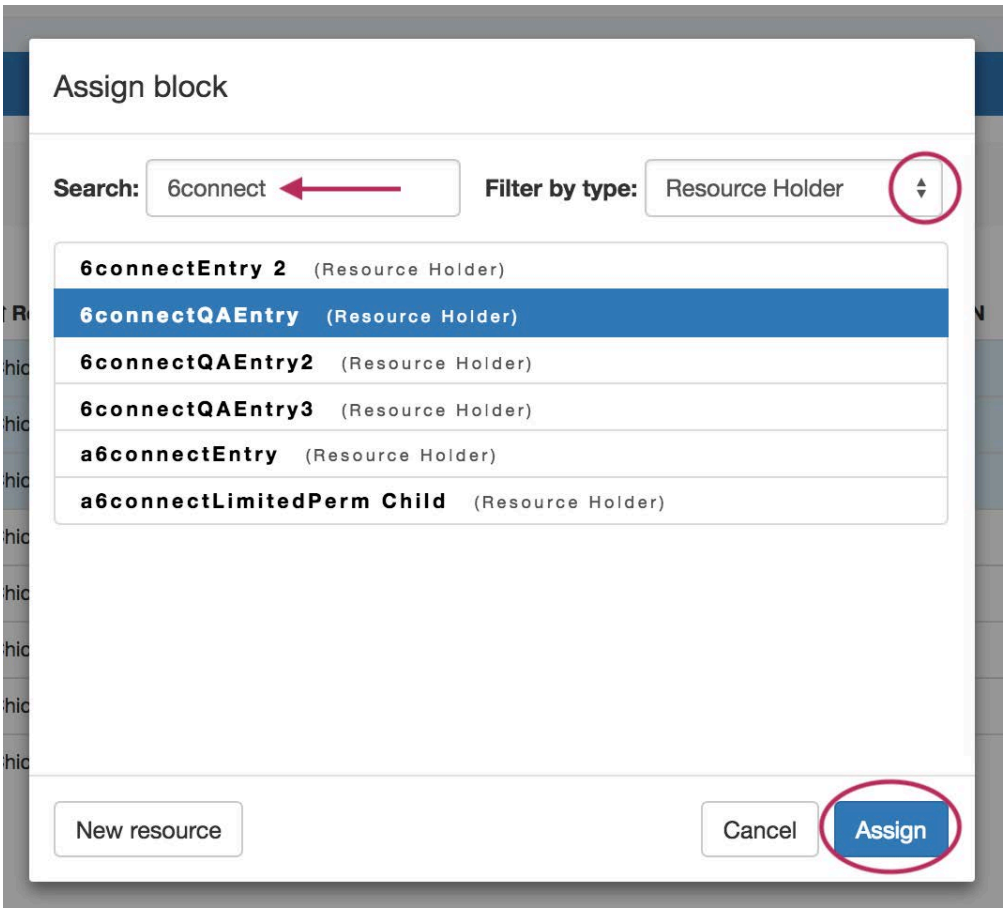
You can also assign blocks manually using the "Assign" function from the IPAM Manager screen (accessible from the IPAM Tab). Open the Action Menu for a block, then select "Assign".



Assignments may also be performed for multiple blocks at a time, by shift-clicking the desired blocks, and right-clicking on any highlighted row to open the multi-block menu (choose "Assign Selected Blocks") :



Then, search for / select the Resource to assign the block. A filter tool is provided to narrow the list to a particular Section type.



When the correct resource has been selected, click the "Assign" button. If needed, a new resource can be created by clicking the "New Resource" button.

After assigning, you can further edit the block attributes or subassign space.

'Reserving' IP Space

If you need to create a 'reserved' pool from which to assign blocks, you can achieve this by creating a "Reserved" Section. Create a Section called "Reserved", add the IPAM gadget to it, then create an Entry with that Section to be the address group. From there, use the IPAM gadget and the IPAM Manage page to assign and unassign IP space from that pool.

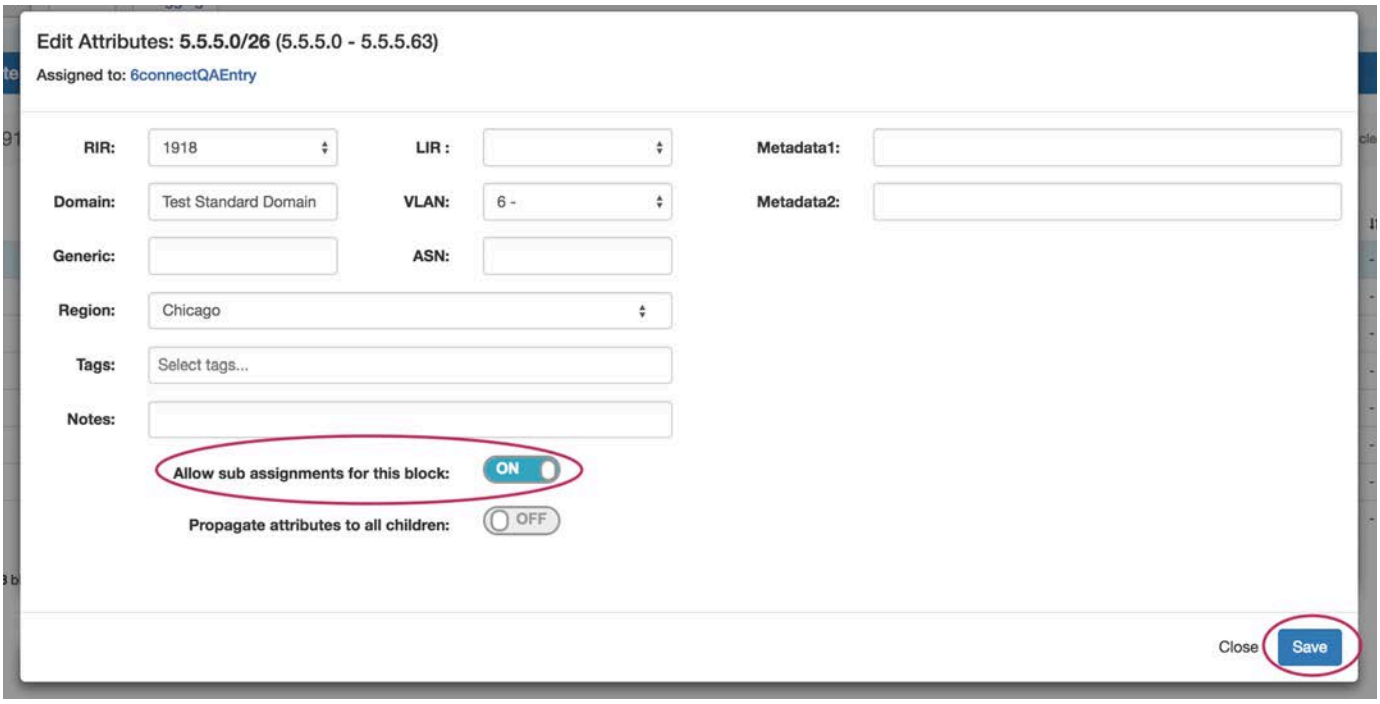
For more details, see the FAQ entry: "How do I 'reserve' IP Space?"

Sub Assigning IP Space

To allow sub assignments, just toggle the "Allow sub assignments" button to "On" under Edit block.

Once the allow sub assignments box is checked, the block may be further split and assigned to other resources. Split blocks may also be re-claimed to the originally assigned resource and re-aggregated. When allow sub assignments is checked, the block is counted as allocated, but not assigned - various statistics in IPAM, on the dashboard, and reporting will reflect this.

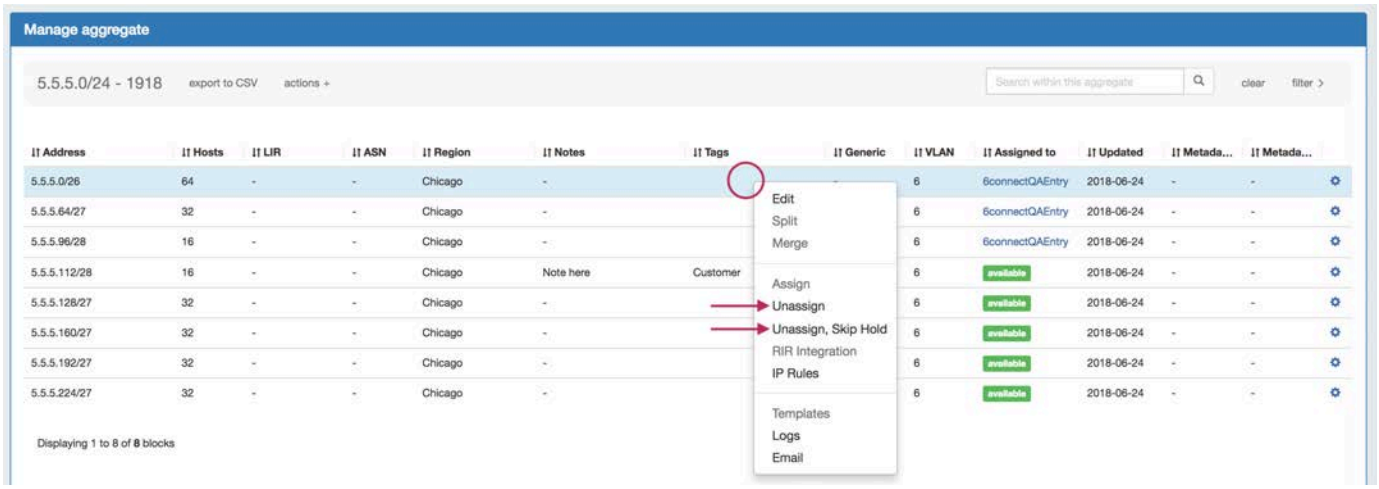
Sub assignments can be useful for tracking IPs assigned to a customer with multiple subsidaries, or locations.



Unassigning IP Space

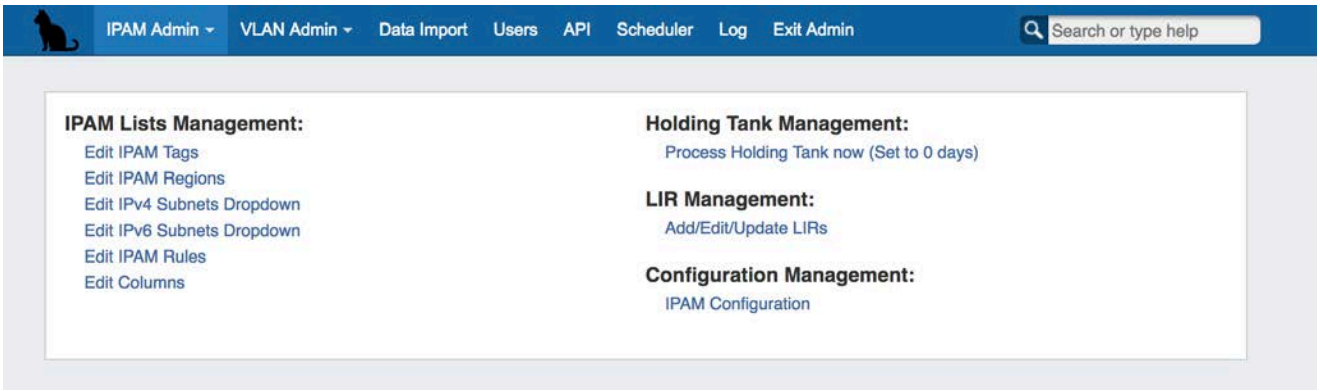
When a block is assigned, you will have the option of unassigning the block from the resource and returning it to the Holding Tank.

To unassign the block, simply open on the Action Menu for the block and select "Unassign". You may also chose "Unassign, Skip Hold", which unassigns the block and immediately returns it to available, bypassing the holding tank.



After unassigning blocks / skipping holding, newly available blocks will be merged upon next page refresh.

To return IP space in the Holding Tank to the Available Pool, process the Holding Tank via the Admin screen under **IPAM Admin** (this will only process blocks that were present for the specified number of days).



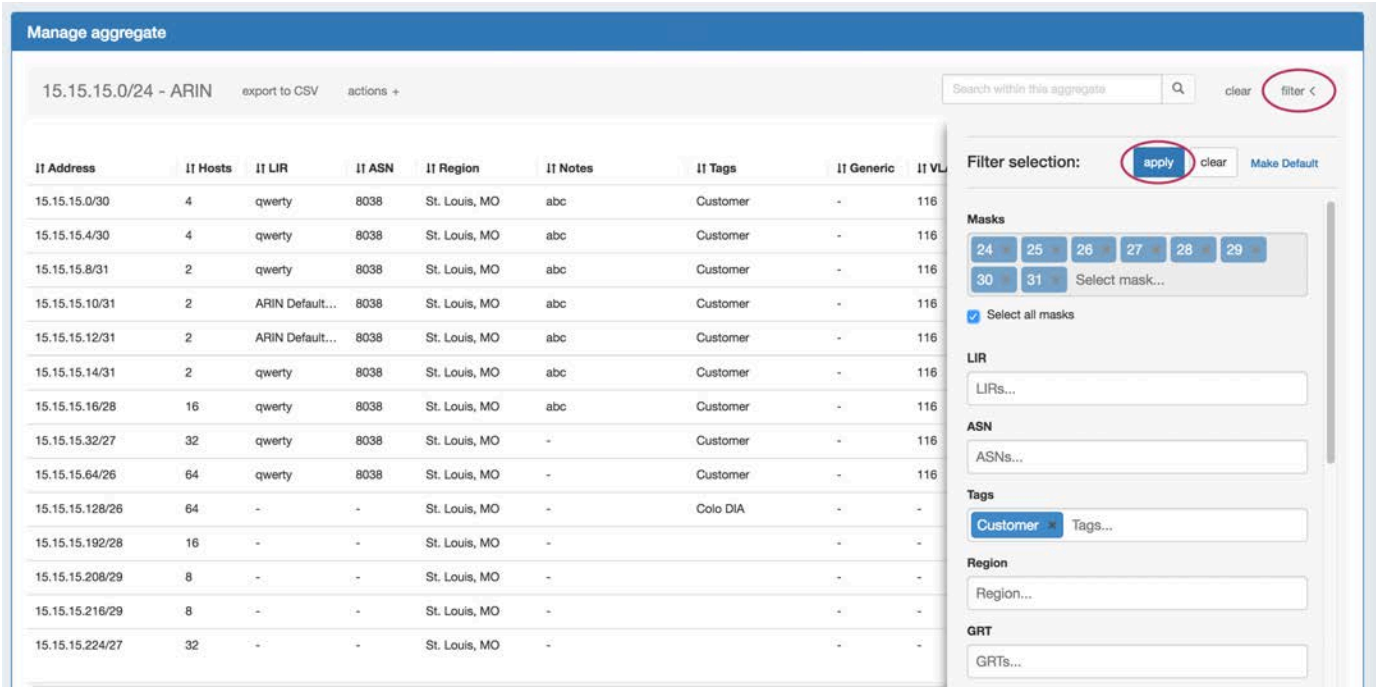
For more information on the Holding Tank, see Holding Tank Management.

Sort, Filter, and View Block Hierarchy in IPAM Manage

Sort the IPAM Manange blocks list by clicking on any header the in list - the items will toggle between ascending and descending order.

Filter the block list in IPAM Manage by toggling the "Filter +" link in the top right corner of the aggregate header.

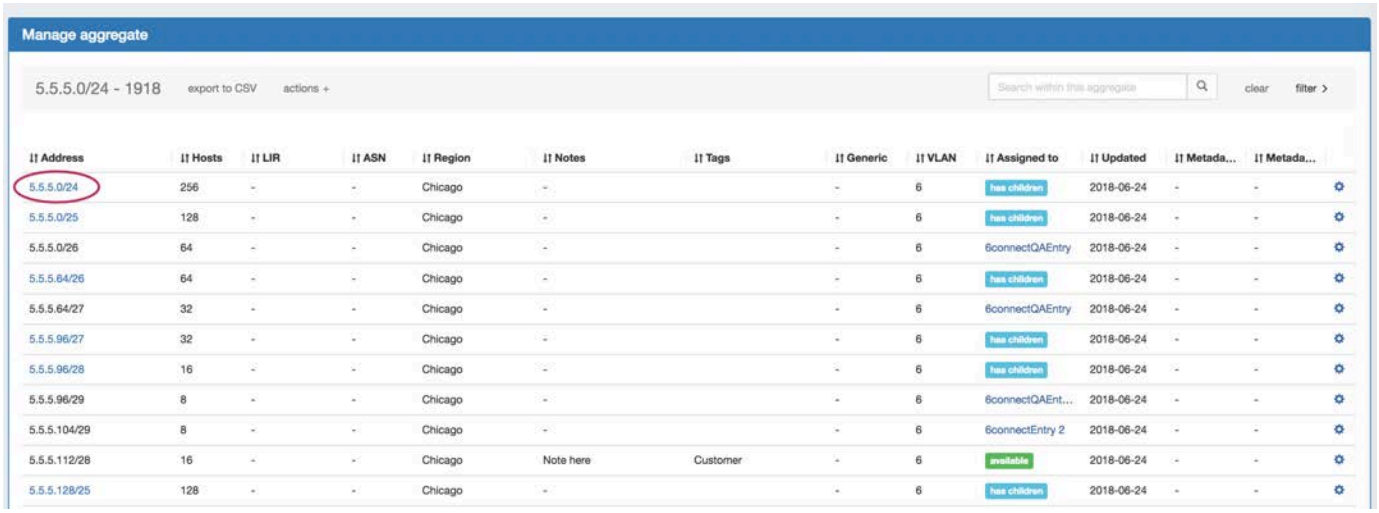
A flyout will appear with filter criteria options.Type in or search for the term under your desired field, and hit "Apply". When done, click "clear". To hide the filter flyout, just click the "filter -" link again.



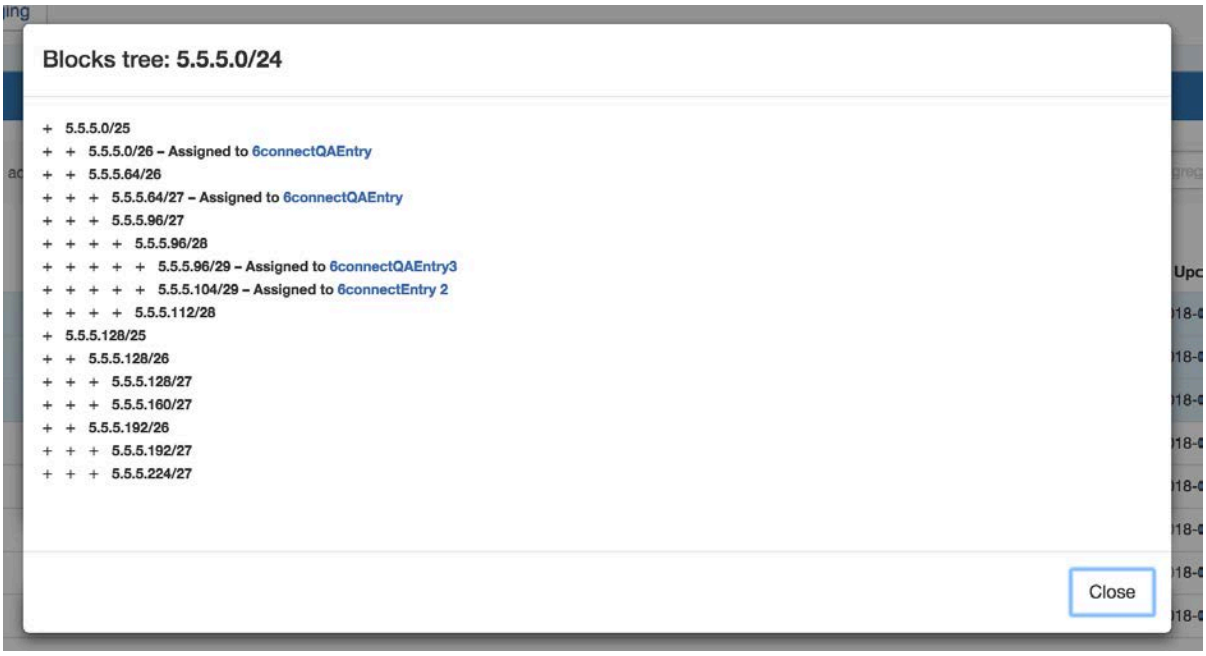
To view parent blocks / IP block hierarchy, in the "Filter" menu, check the "Select All Masks" checkbox and click "Apply".

Once all masks are enabled to view, the block list changes to include the "Parent" blocks, showing all masks throughout the assignment tree.

A parent block with children will show "Has Children" in the column where assignment status is shown, and the block CIDR will show as a clickable link.



Clicking on the CIDR for a block that has children will open "Block Tree" view, showing the hierarchy of blocks and their assignments.



Additional Information

For additional information on working with the IPAM system in ProVision, see the following areas:

- Gadgets (IPAM and IPAMv2 Gadgets)
- Working with IP Rules
- IPAM Administration

Working with IP Rules

Working with IP Rules

- Working with IP Rules
  - IP Rules
    - Creating a new rule
      - IP Rule Positions
    - Apply an existing rule
    - Remove a rule from a block

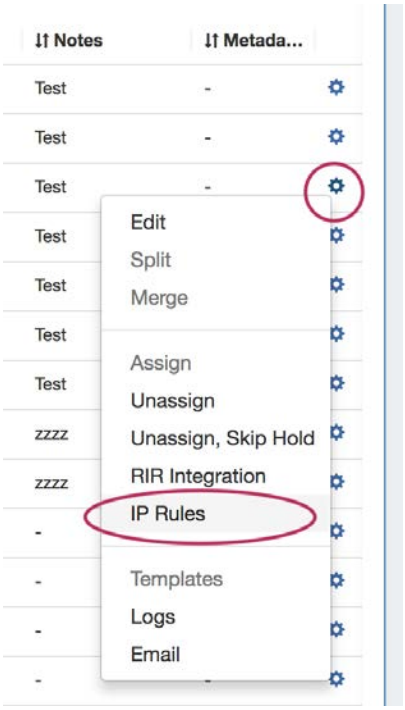
IP Rules

IP Rules allow users to exclude IP addresses from being assigned (via Smart Assign or Direct Assign) based on the address position in the block (i.e., first address, last address, nth from last).

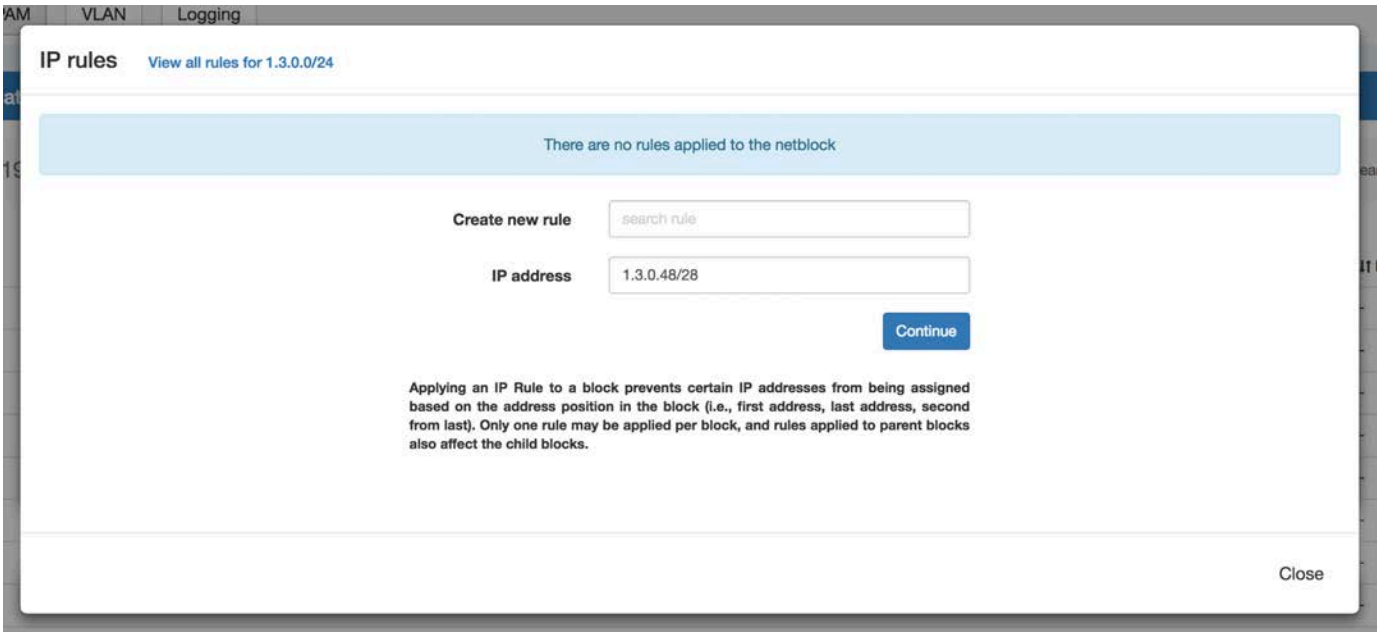
IP Rules can be accessed from IPAM Manage, in the block's Action Menu (wrench icon). IPAM users may view rules applied to a block, apply an existing rule to a block, or create a new rule for a block. Only one rule may be applied per block, and rules applied to a parent block affect the child blocks below it in the tree. Admin users may view all existing IPAM Rules, the blocks affected by each rule, and delete rules through the IPAM Admin - IP Rules page.

Creating a new rule

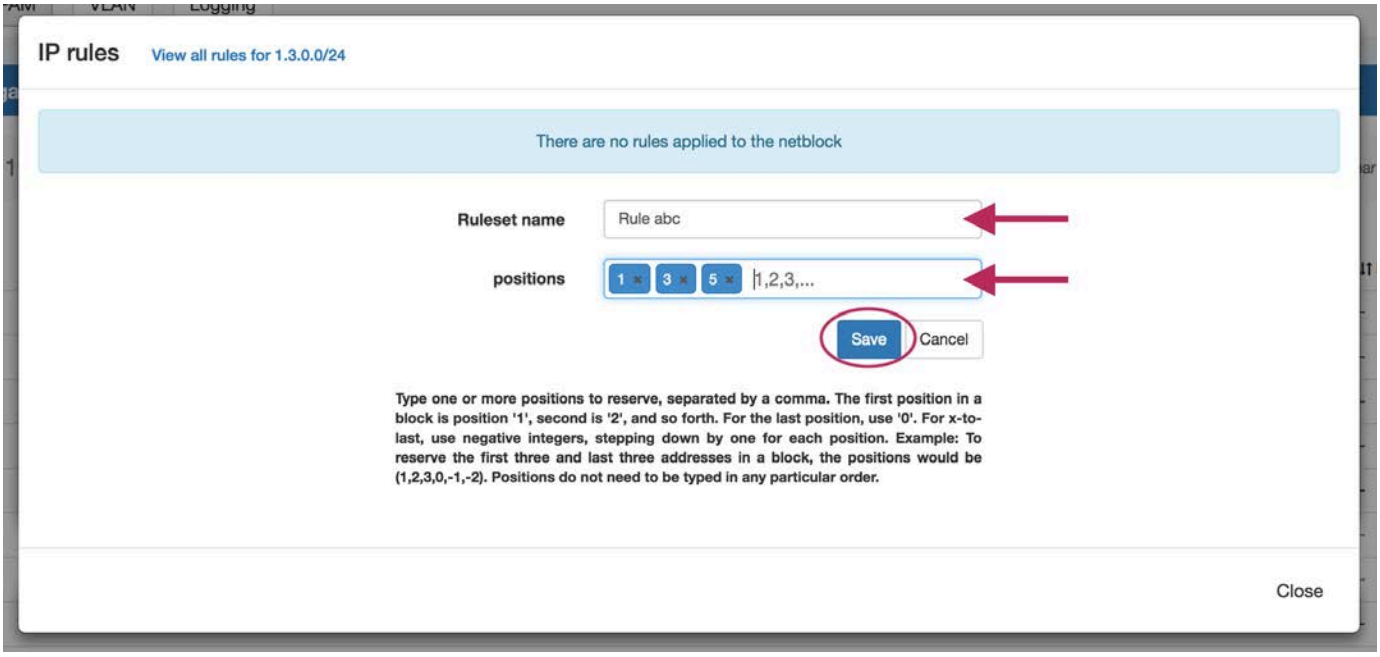
From the IPAM Manage screen, click on the Action Menu (wrench icon) for the block you wish to apply a rule. Select "IP Rules".



The IP Rules screen will pop up. From here, you can view existing rules applied to the block (or the parent block), search for and apply an existing rule, or create a new IP Rule.



To create a new rule, leave the "Search Rule" input empty, and hit the "Continue" button.



The Manage Ruleset screen will appear, prompting you to enter a Ruleset name, and select positions to reserve.

Type in the desired name for the new rule, and one or multiple positions (denoted by integers separated by commas) into the box below the name.

When complete, click the "Save" button, or hit "Cancel" to exit without saving. The Rule preview screen will appear.

IP Rule Positions

Reserved addresses are set by their position in the block. Positions start at '1', for the first address in a block, and step up by one for each subsequent IP address. The last position in a block is '0', representing the last address, and steps down a negative integer for each position from last - so the last three addresses in a block would be represented by (0, -1, -2). When reserving multiple positions, the typed order of the positions does not matter.

Position examples:

(1) - Reserves the first IP

(0) - Reserves the last IP

(1,2,3) - Reserves the first three IPs

(0, -1, -2) - Reserves the last three IPs.

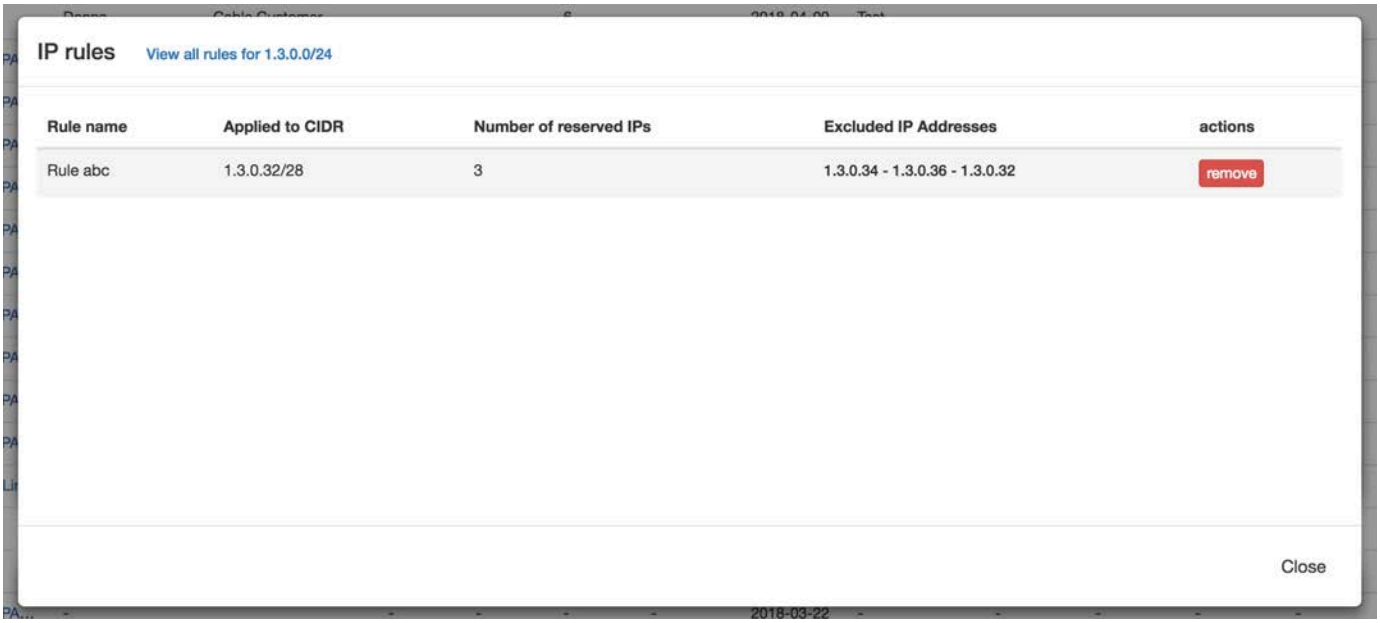
(1,2,3,0,-1,-2) - Reserves the first three and last three IPs

Review the list of excluded IP addresses affected by the rule, then click "Apply" to finalize the selection.

If desired, you can exit out of applying the rule by clicking "Discard", and closing the screen.

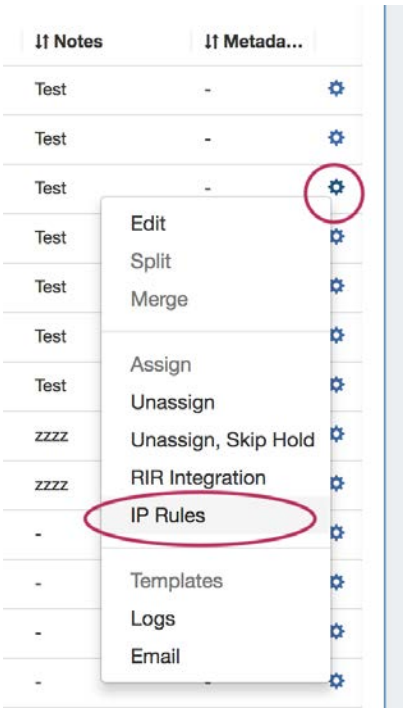


Once changes are applied, the rule will show the name, affected CIDR(s), affected IPs, and an option to remove the rule.

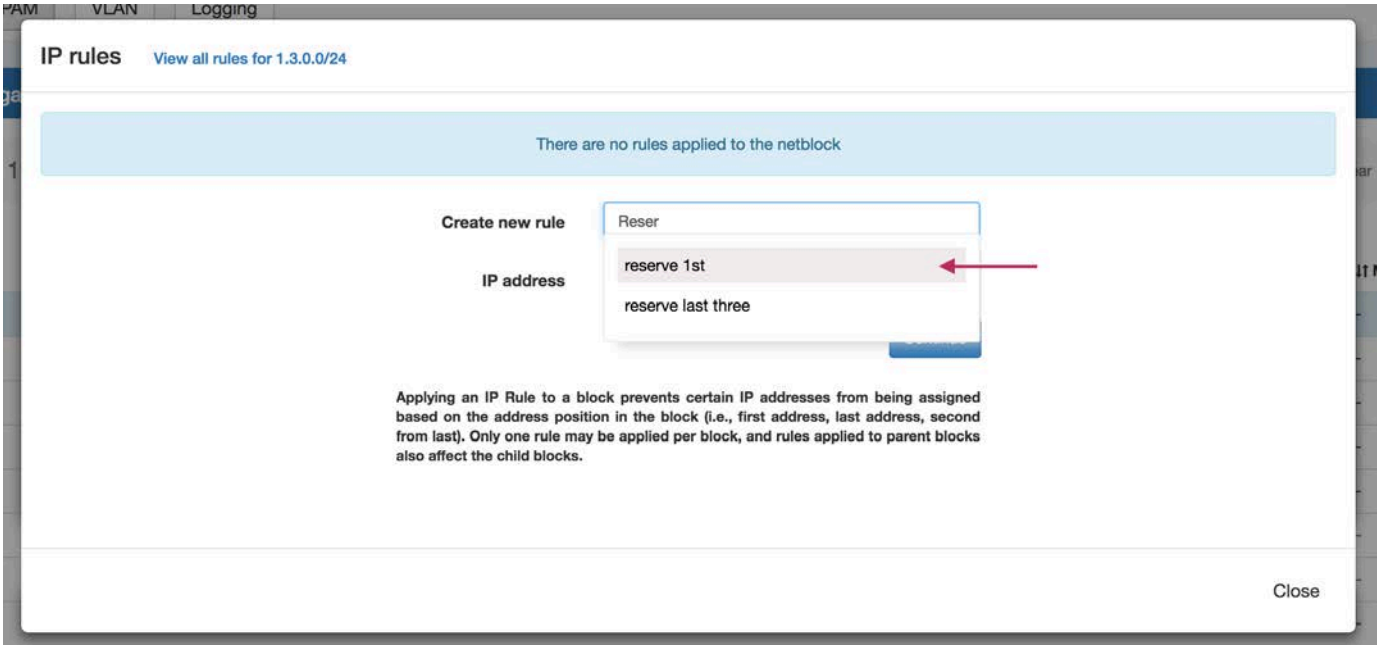


Apply an existing rule

From the IPAM Manage screen, click on the Action Menu (wrench icon) for the block you wish to apply a rule. Select "IP Rules".

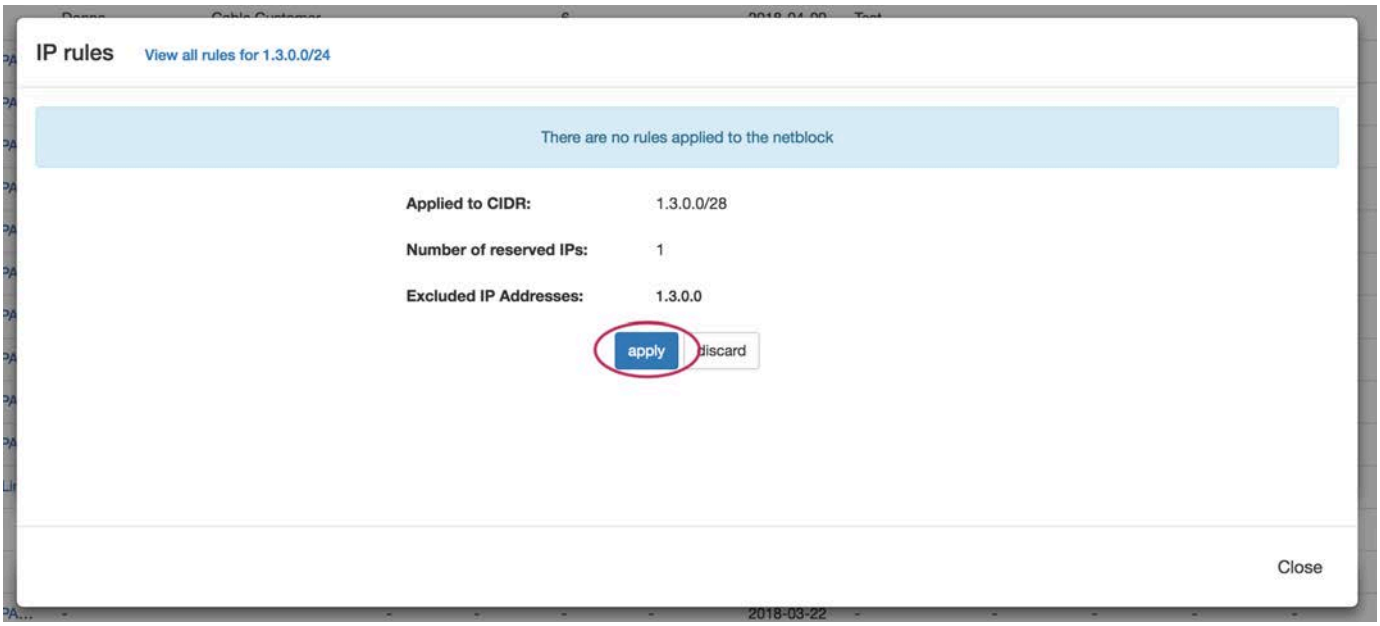


The IP Rules screen will pop up. From here, you can search for existing rules by typing a few letters of the Rule name in the search box, then selecting the desired rule.



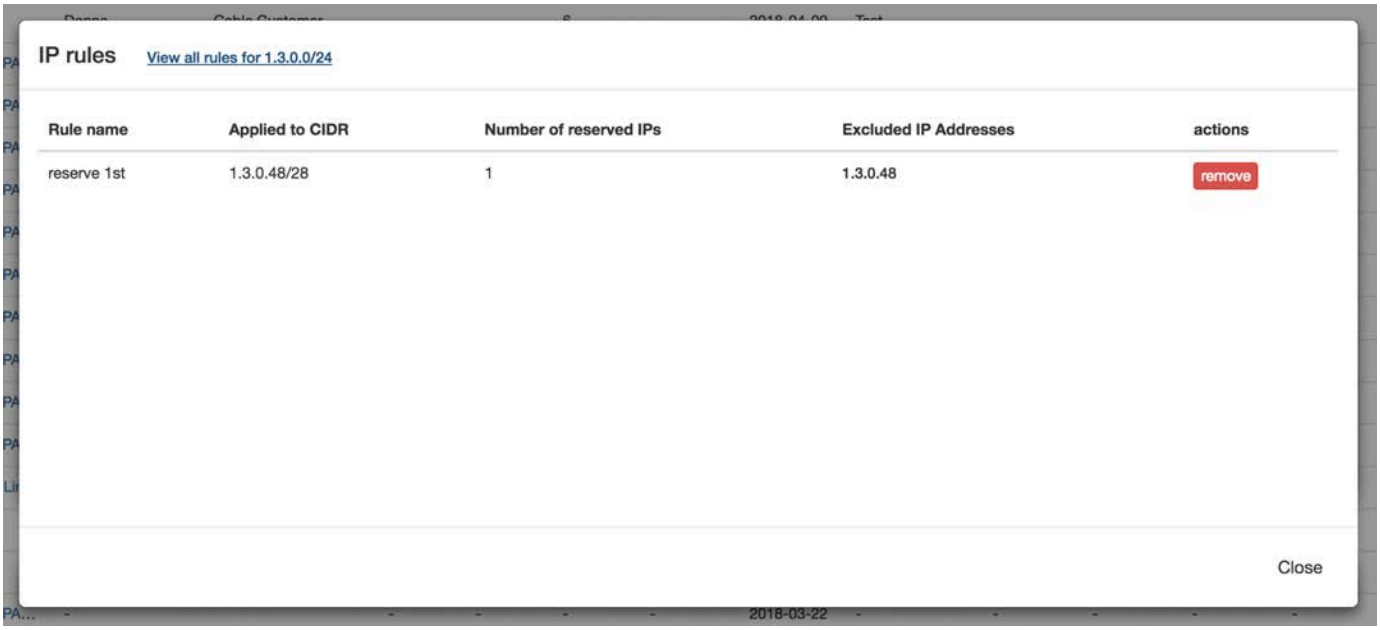
Once selected, hit the "Continue" Button.

Review the list of excluded IP addresses affected by the rule, then click "Apply" to finalize the selection.



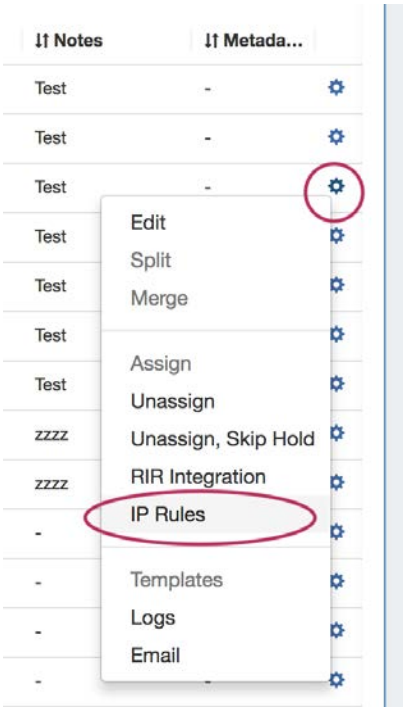
If desired, you can exit out of applying the rule by clicking "Discard", and closing the screen.

Once changes are applied, the rule will show the name, affected CIDR, affected IPs, and present with an option to remove the rule.



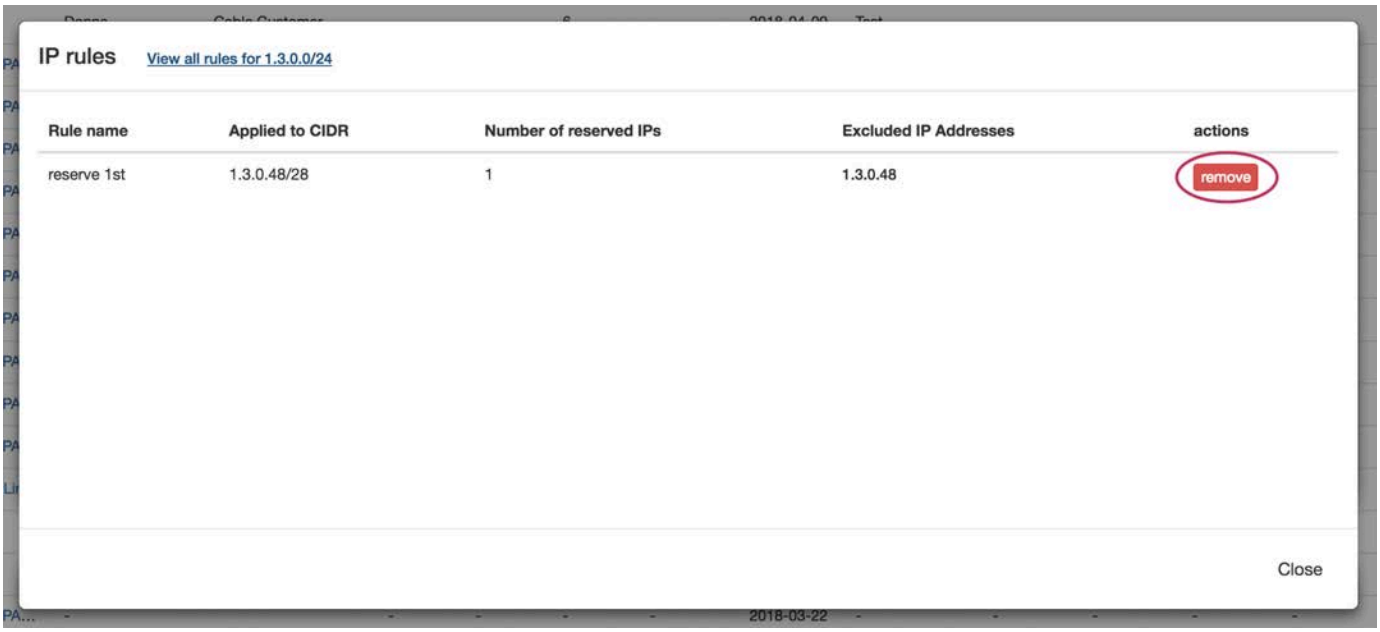
Remove a rule from a block

From the IPAM Manage screen, click on the Action Menu (wrench icon) for the desired block. Select "IP Rules".



The details for the existing rule applied to the block will be shown. To remove the rule from the block, click the "Remove" button and confirm.

The rule will still be able to be re-applied and usable by other blocks, simply no longer applied to the current block.

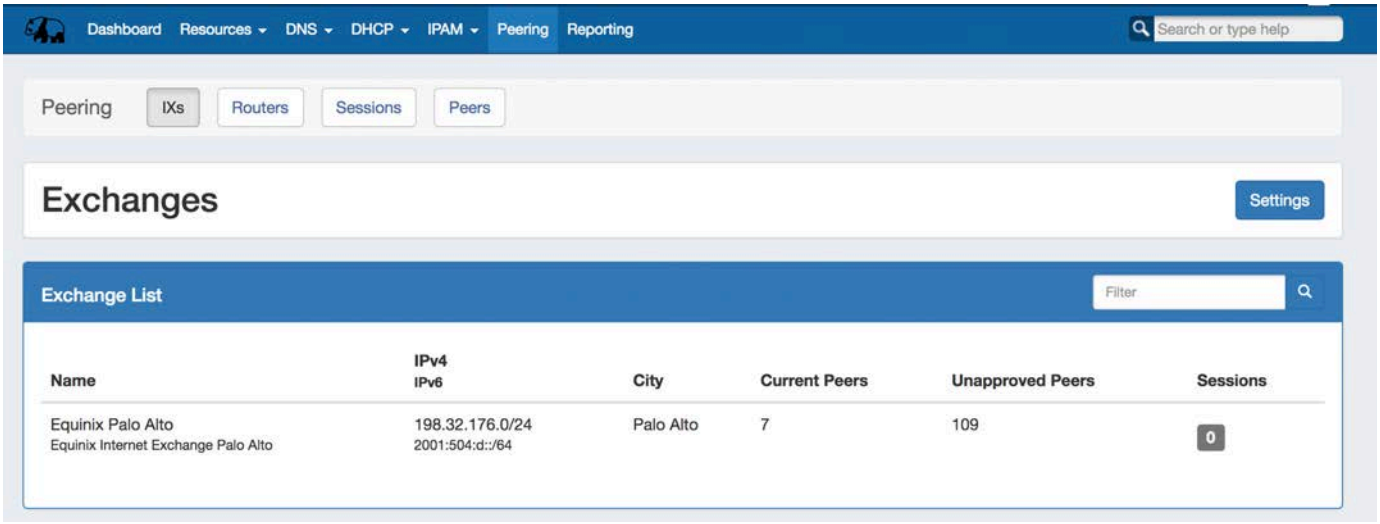


Peering v2

6connect Peering

The **Peering** tab displays allows you to add and manage exchanges, routers, sessions, and peers.

ProVision's Peering area interfaces with Peering DB's new API to ensure up to date information. Users may chose to use PeeringDBs "main" or "beta" systems, by editing PeeringDB_URL in ProVision's global configs.



You can navigate via four subtabs at the top of each peering page:

IXs: Shows the exchange list. From here, you can edit exchange settings and view, manage, and communicate with peers via the exchange details page.

Routers - Shows the list of routers in ProVision. From here, you can add new routers or manage existing routers.

Sessions: Shows the current list of sessions in ProVision. Clicking on a session name brings up session details, from which you may configure and manage each session. You may also add new sessions.

Peers - View and manage session peers.

Peering Setup - Local Installations

If you are using a locally hosted instance of ProVision, verify that you have followed the instructions on the Local Installations - Peering Setup page to set up Peering for your instance.

Before working in Peering, it is essential to set up the ASN and PeeringDB credentials in either the Exchanges Settings page, or from Admin - Peering Settings.

Table of contents

- Peering - Common Tasks
- Peering Exchanges
- Peering Routers
- Peering Sessions
- Managing Peers

Peering - Common Tasks

Peering Workflow

The ProVision peering workflow encompasses working with routers, peer groups, sessions, and managing communications.

Click here to see the graphical workflow...

PEERING WORKFLOW





Below is an outline of the steps needed for a basic workflow process.

- Peering Workflow
  - 1) Add Router(s)
  - 2) Add Peer Groups
    - With a New Router:
    - With an Existing Router:
  - 3) Add Session(s)
  - 4) Configure Sessions
    - Configure Sessions Individually by Exchange:
    - Configure Sessions for Existing / Approved Peers:
  - 5) Manage Communications
    - New Peer:
    - Outgoing Peer Request:
    - Incoming Peer Request:
    - Configure Sessions for Existing / Approved Peers:
  - 6) Repeat for new Routers, Sessions, and Peers.

1) Add Router(s)

Adding a router is the fundamental base and first step of working with peering in ProVision. Routers may be added through the "Add Router" button in the **Peering** Tab.

For a detailed breakdown of this task, see the Router Page Add Routers section.

2) Add Peer Groups

Associating the router with a peer group is necessary to link the router to a particular exchange. You may add the Peer Group information either in the "Add Router" dialog or in the Peer Group Gadget prior to adding sessions.

With a New Router:

To add a peer group through the "Add Router" dialog: After inputting the Router information, under Peer Groups, select the exchange, type in the name of the desired Peer Group name, select whether it is IPv4 / IPv6, and click "Add Group". Lastly, click "Add Router".

With an Existing Router:

If you need to add a Peer Group to an existing router, you may use the Peer Group gadget.

The Peer Group Gadget allows you to add peer groups for IPv4 and IPv6 for a selected exchange from a router's Resource Entry page.

To do this, simply select the exchange, type in a Peer Group name in the text box, select IPv4 or IPv6, the click "Add Group".

Peer Groups added from this gadget will be then be available to select in the "Add Session" dialog box in the **Peering** tab.

3) Add Session(s)

After Routers and Peer Groups have been set up, the next step is Adding Sessions to ProVision.

You may add a new session through the "Add Session" button in the **Peering** tab, or you may Import Sessions from an existing router. Importing sessions requires Admin level permissions, and is accessed through the **Data Import** Tab in the Admin section of ProVision.

4) Configure Sessions

Configure Sessions Individually by Exchange:

Once a session has been added, you will see it in the session list. For a newly added session, the status will show as "Not Configured".

You may configure the session through 1 click config or the Config Manager, accessed by clicking session and opening the Session Details page. 1 click config uses the default config settings, while the Config Manager allows you to manually create one time use unique settings.

Sessions may also be configured through the Communications list Action Menu, for Peers that have been "Approved" or marked "Existing". Configuring through the Communications menu allows you to configure all sessions for that Peer at once, for any or all exchanges.

Configure Sessions for Existing / Approved Peers:

If a Peer is marked "Existing" or "Approved", the option will appear to "Configure Sessions".

With this option, you can view and configure all sessions for that Peer from one menu (as opposed to the Sessions list configure option, which configures individual sessions by Exchange).

Clicking on "Configure Sessions" opens a modal showing the available exchanges with routers, sessions for that Peer under those exchanges, and options to change the router, peer group, and Peer type.

Select the checkmarks for the desired sessions to configure. As a shortcut, you may also check the "Select all exchanges" option to select all sessions under all exchanges, or, check "Select all networks" next to the exchange header to select all sessions in that exchange. From there, deselect items as needed.

When your selections have been made, click "Configure Now" to immediately configure all selected sessions, or you may select "Schedule" to automatically create sessions and schedule a configuration to occur in one hour (by default). The scheduled configuration may be edited through the **Scheduler** Tab.

5) Manage Communications

Once a session has been added with a peer, you will see the peer listed in the communications list under the Exchange Details page.

You will see the list of Peers, their ASNs, Name, Request Status, Notes (log), and the Action Menu (wrench icon).

From here, you may manage peer requests, mark peer status, and configure existing sessions for that Peer through the Action Menu (wrench icon).

From here the flow depends on the Peer Status and whether a request is Outgoing or Incoming.

New Peer:

For an existing peer that was just added, you can mark the peer as:

a) Mark Existing Peer, or,

b) Mark Not Qualified

You may also choose to send an outgoing peer request.

Outgoing Peer Request:

To send out a peer request, select "Send Request" from the Action Menu (wrench icon) . This sends an initial peering request email to the peering coordinator for that peer. The email template pre-populates data based on peeringdb data (To address, Subject line and Peering exchange information). You have the chance to edit the email prior to sending.

Once a Request has been sent out, it can proceed two ways:

a) The request is accepted and Peering is established (becomes current peer)

b) The request is not accepted / responded to, and the "Resend Request" option becomes available, allowing you to repeat the request.

Incoming Peer Request:

If a request is received, you have a few options from which you may mark the peer from the Action Menu:

a) If you accept the request, you may "Mark Approved" and Peering is established (becomes current peer).

b) You may "Mark Rejected", in which case a peer status is marked rejected.

c) "Reset Status" is available at various points in the communications process, if you want to reset the peer back to the beginning state and re-establish a different status condition. This reopens the initial options from which you may select a different peer mark.

d) "Resend Request" may be available to resend a request

Configure Sessions for Existing / Approved Peers:

If a Peer is marked "Existing" or "Approved", the option will appear to "Configure Sessions".

With this option, you can view and configure all sessions for that Peer from one menu (as opposed to the Sessions list configure option, which configures individual sessions by Exchange).

Clicking on "Configure Sessions" opens a modal showing the available exchanges with routers, sessions for that Peer under those exchanges, and options to change the router, peer group, and Peer type.

Select the checkmarks for the desired sessions to configure. As a shortcut, you may also check the "Select all exchanges" option to select all sessions under all exchanges, or, check "Select all networks" next to the exchange header to select all sessions in that exchange. From there, deselect items as needed.

6) Repeat for new Routers, Sessions, and Peers.

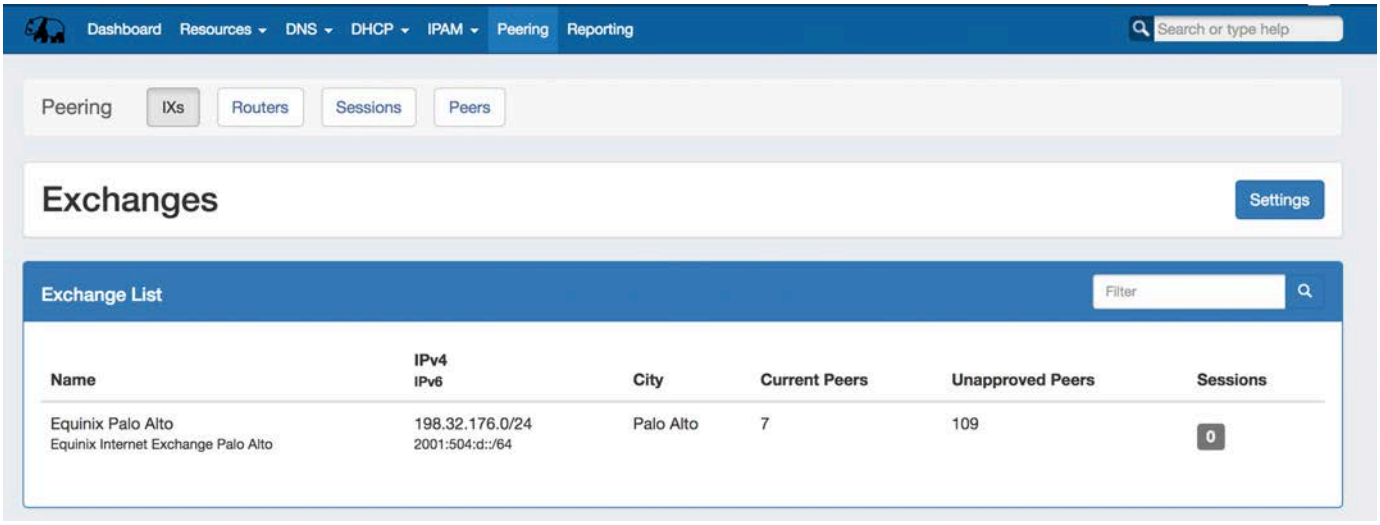
To add additional Routers, Sessions, and Peers repeat steps 1, 2, or 3, and manage your newly added sessions and peer communications similarly with steps 4 and 5.

Peering Exchanges

Working with Peering Exchanges

IXs Subtab

The Exchanges subtab shows the list of exchanges current set in ProVision, with IP type, city, peers, and session counts.

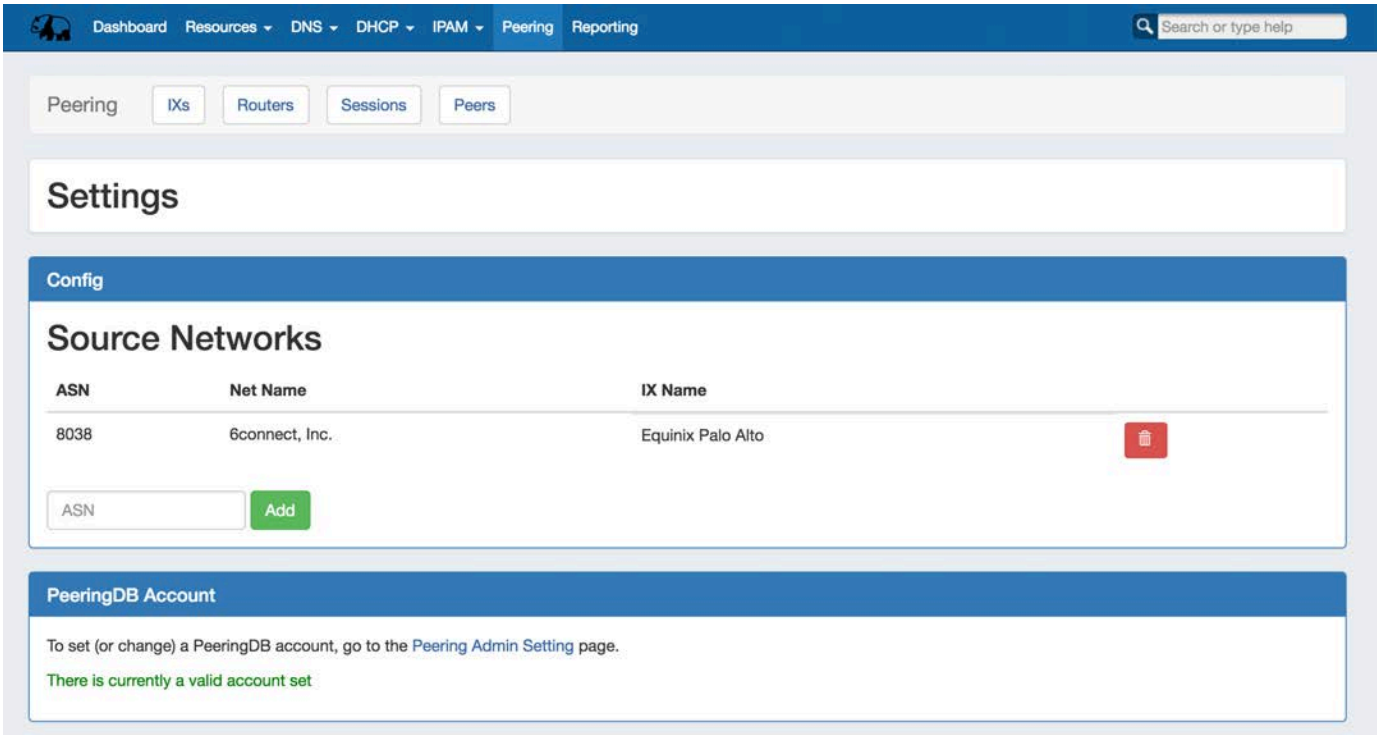


Exchange Settings

ASNs and PeeringDB account settings may be edited by clicking "Settings" in the IXs or Exchanges pages, or by going to ProVision's Admin settings page.

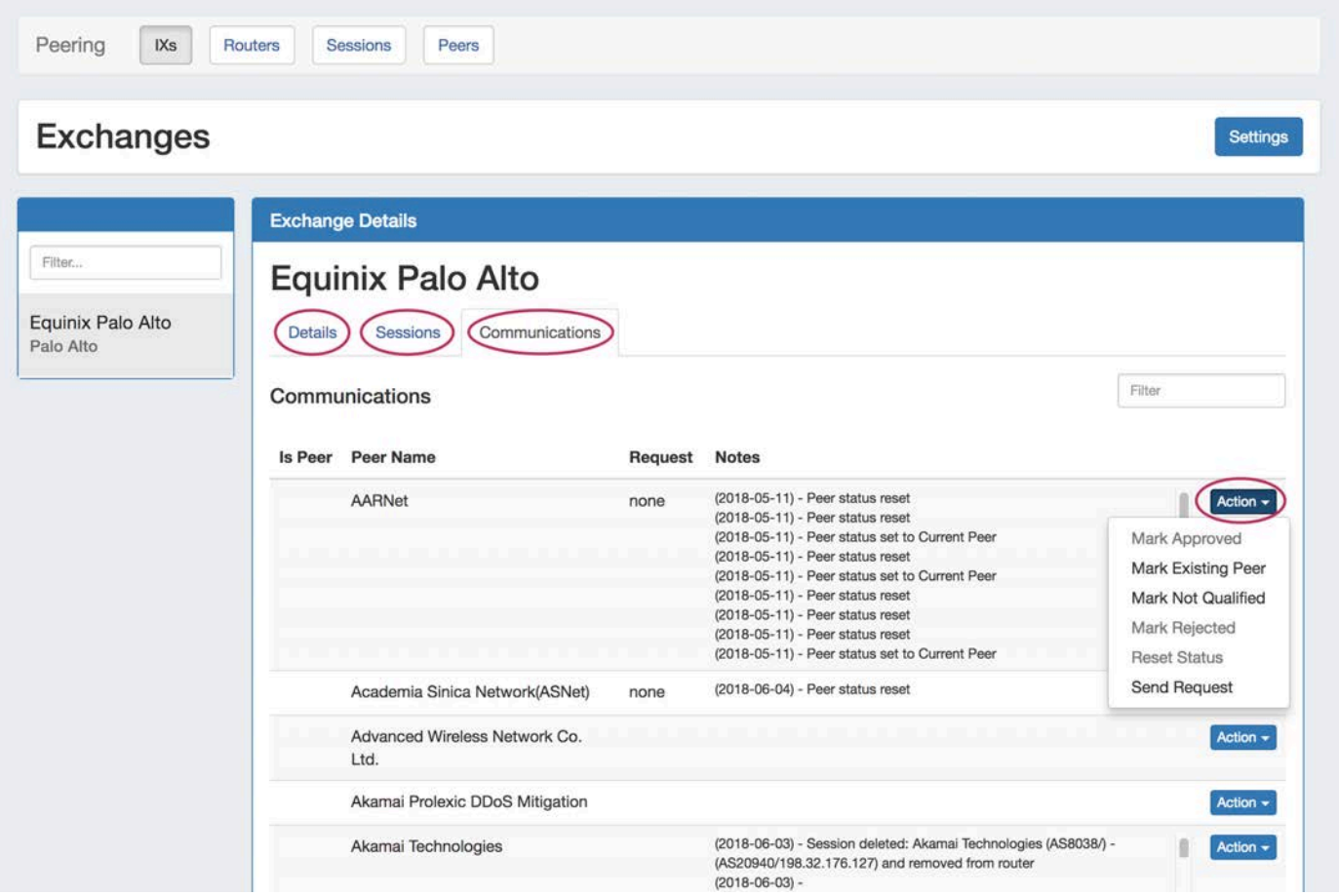
Add an ASN by typing a valid ASN into the Config - ASN box, and then click "Add".

Delete an ASN by clicking the red "Delete" icon for the row under Source Networks.



Exchange Details

Clicking on an exchange in the exchange list will bring up an exchange details page, where Exchange Details, Session Details, and Exchange Communications may be viewed and managed.



Exchange Details Tab:

The exchange details tab provides general information about the exchange, obtained from PeeringDB. It includes organization information, contact information, LAN, and a list of Peers at the exchange point.

Exchange Sessions Tab:

The sessions tab lists ProVision sessions associated with the exchange. From here, you may add sessions, or click on a session to go to the session's details page.

Communications Tab:

The Communications tab is where you can manage communications actions and mark status of peers.

Menu Actions

Mark Approved: Marks the peer as approved. Available after receiving a request response.

Mark Existing Peer: Marks a peer as an existing one and removes the email request options.

Mark Not Qualified Peer: Marks a peer as "not qualified" and removes the email request options.

Mark Rejected: Marks the peer as rejected. Available after receiving a request response.

Reset Status: Resets the status of the peer, opening up the options to mark peer as existing, not qualified, or to send email requests.

Resend Request: Resends the peering request.

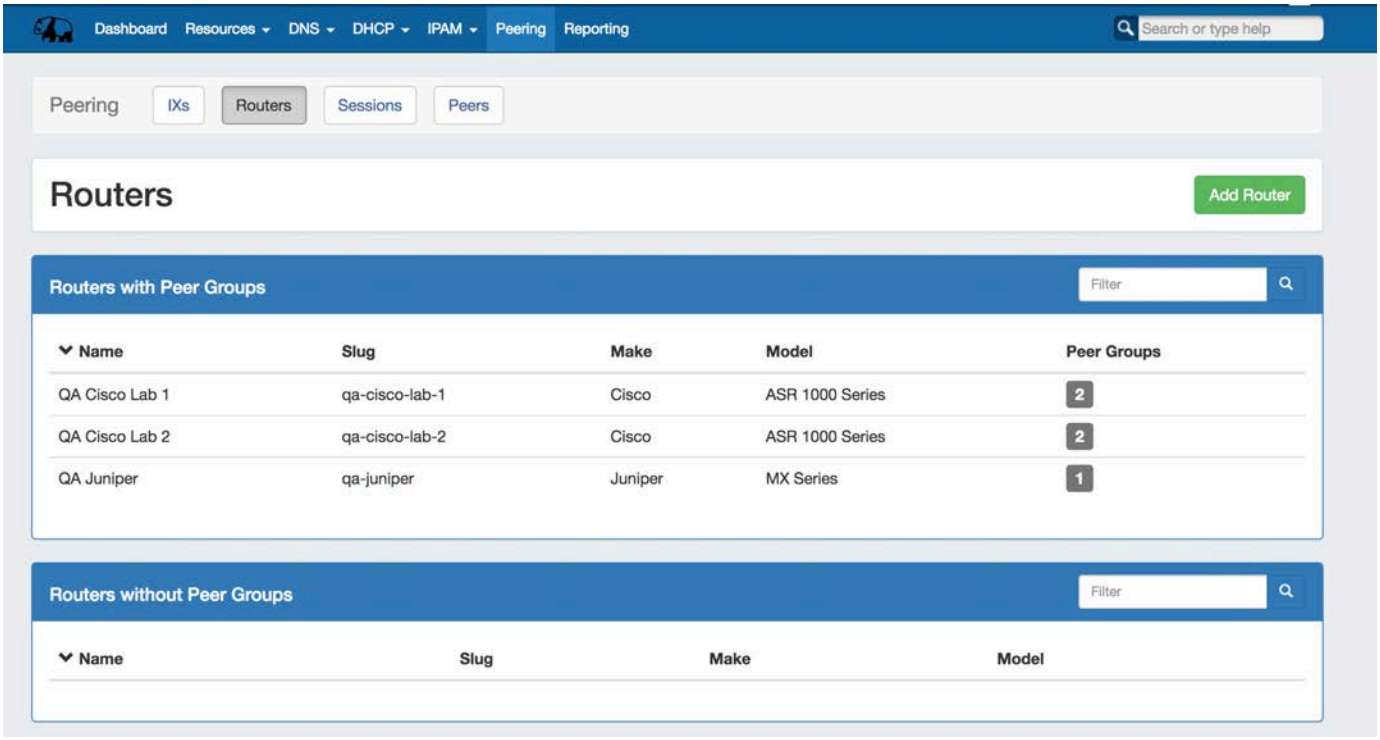
Send Request: Sends an initial peering request email to the peering coordinator. The email template pre-populates data based on peeringdb data (To address, Subject line and Peering exchange information). You have the chance to edit the email prior to sending.

Peering Routers

Working with Peering Routers

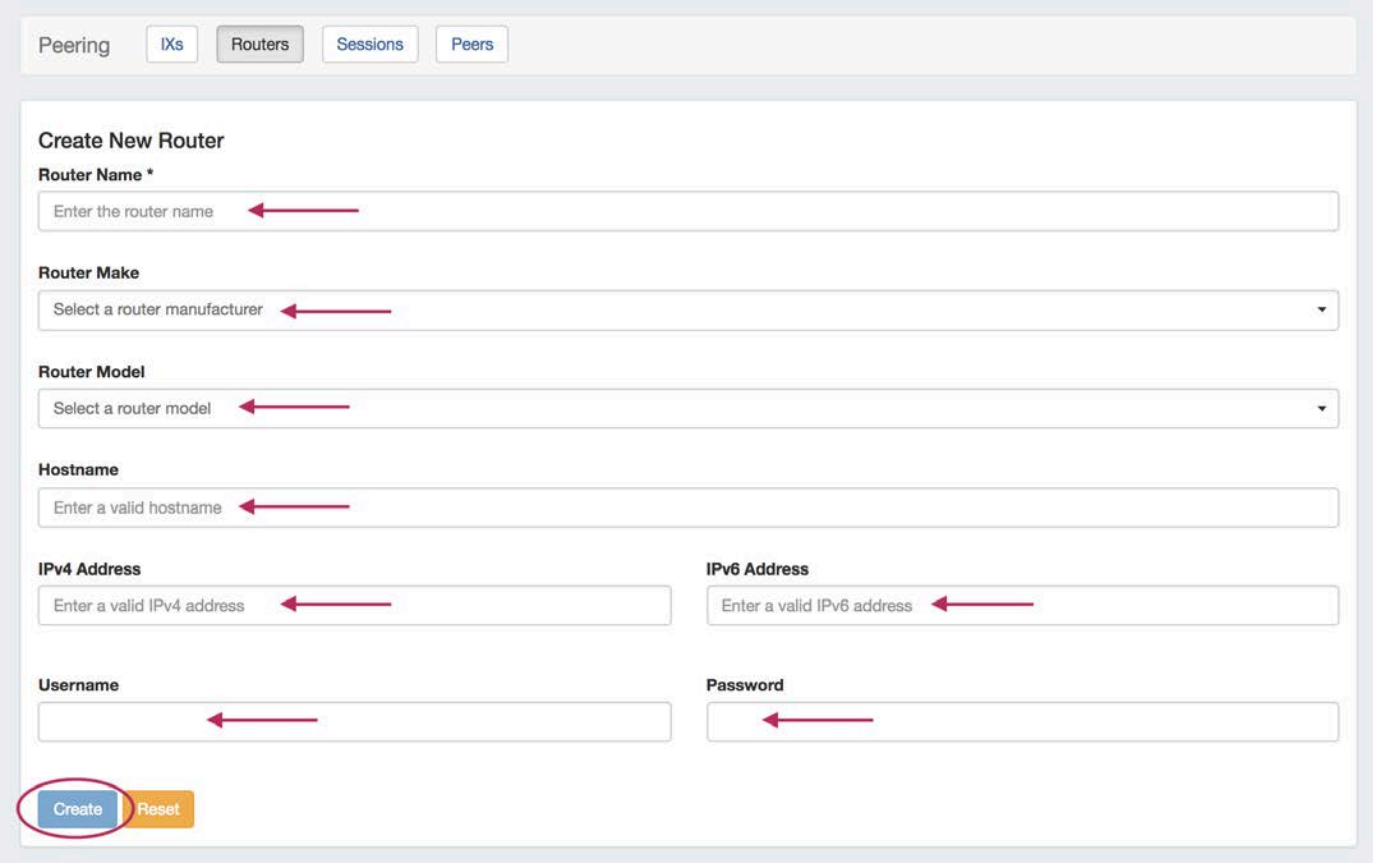
Routers Subtab

The Routers subtab is where routers may be added and managed. Routers are divided into two lists - those with Peer Groups, and those without.



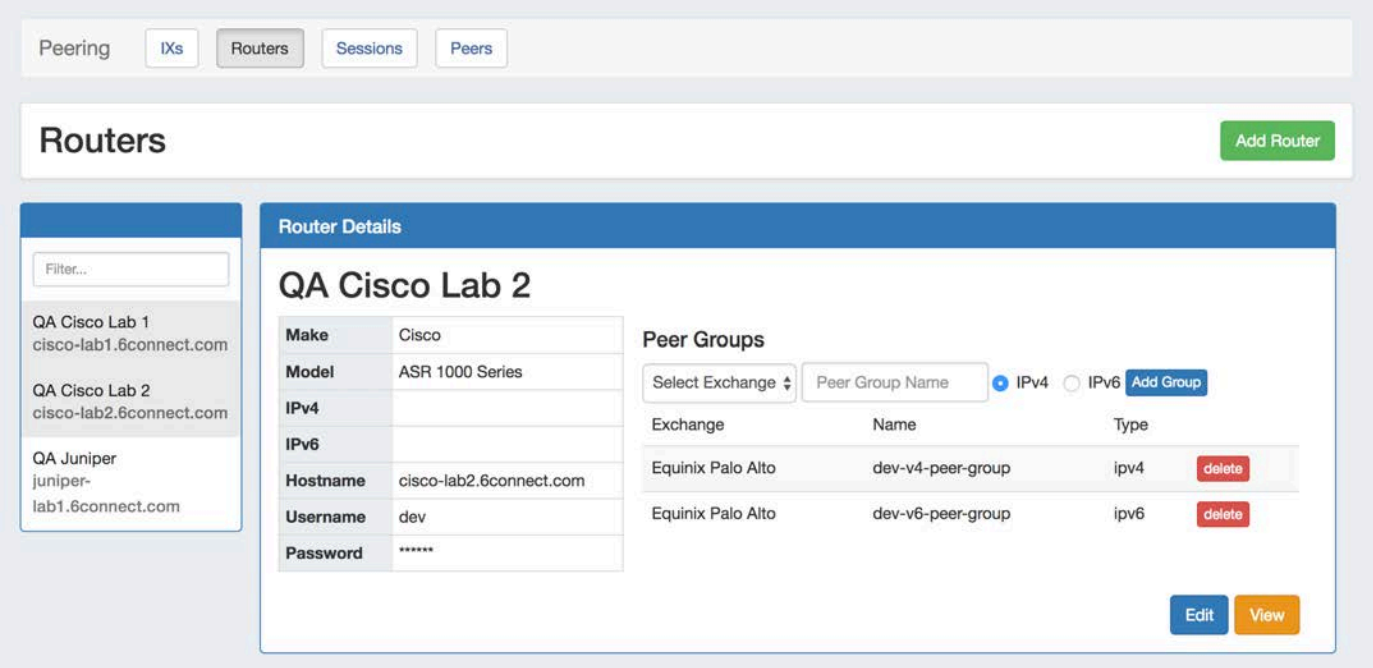
Add a Router

Add a new router by clicking the "Add Router" button, which will direct to a page to input the router details.



Session Details Page

Clicking on a router entry in the router list will open the router details page.



From here, routers may be added as well, or you can use the filterable list in the left sidebar to select a router to view.

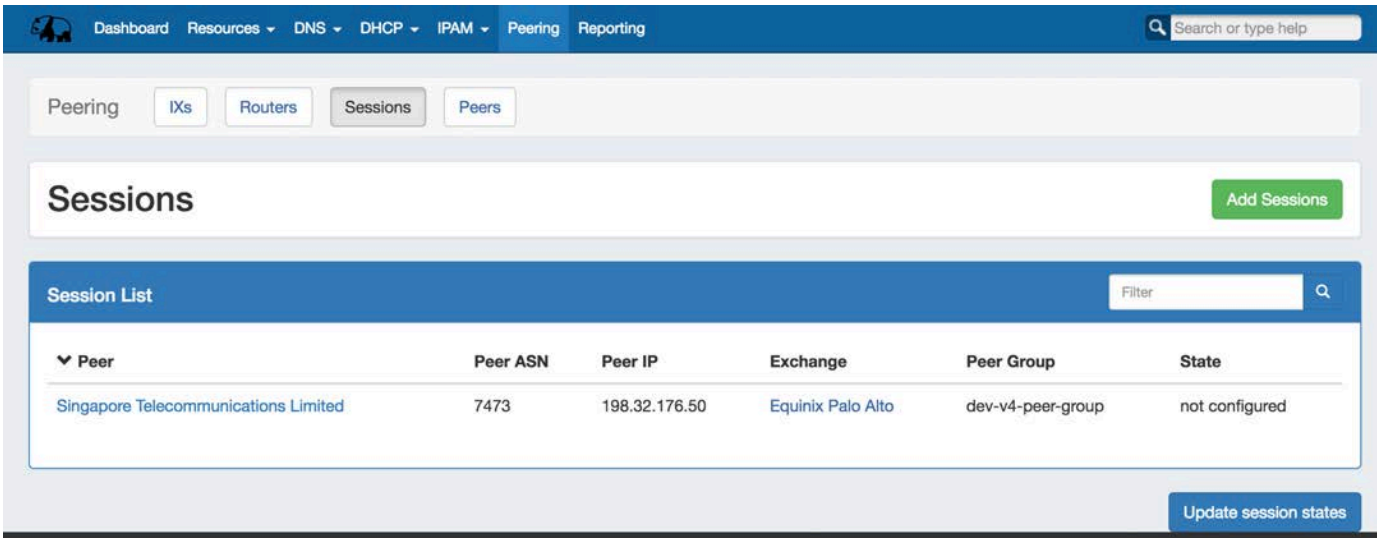
The router details area gives basic router information, which you may edit via the "Edit" button. You may also add and delete and Peer Groups associated with the router.

Peering Sessions

Working with Peering Sessions

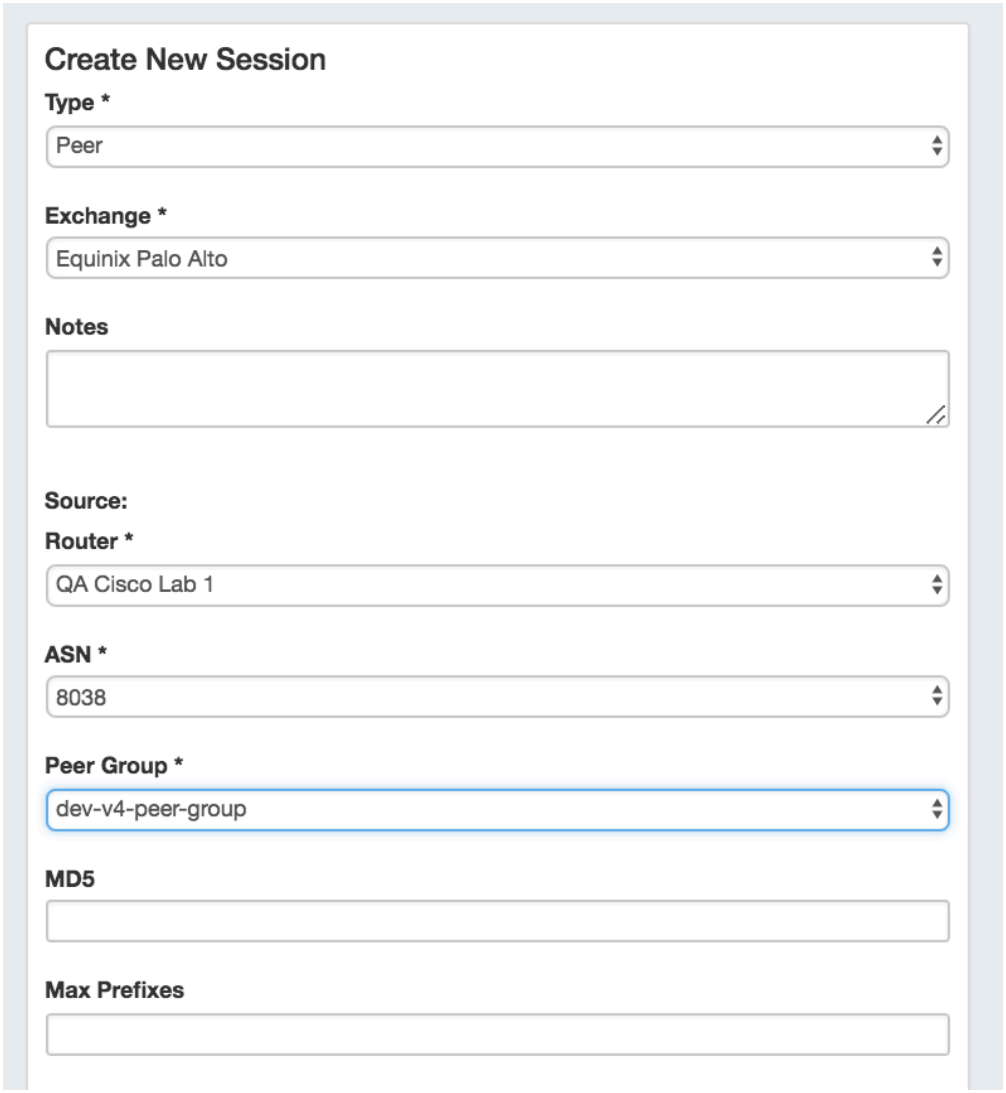
Sessions Subtab

The Sessions subtab is where current sessions may be added, managed, and updated. View details for a session by clicking on the session name. Update the session states for all sessions by clicking the "Update Session States" button.

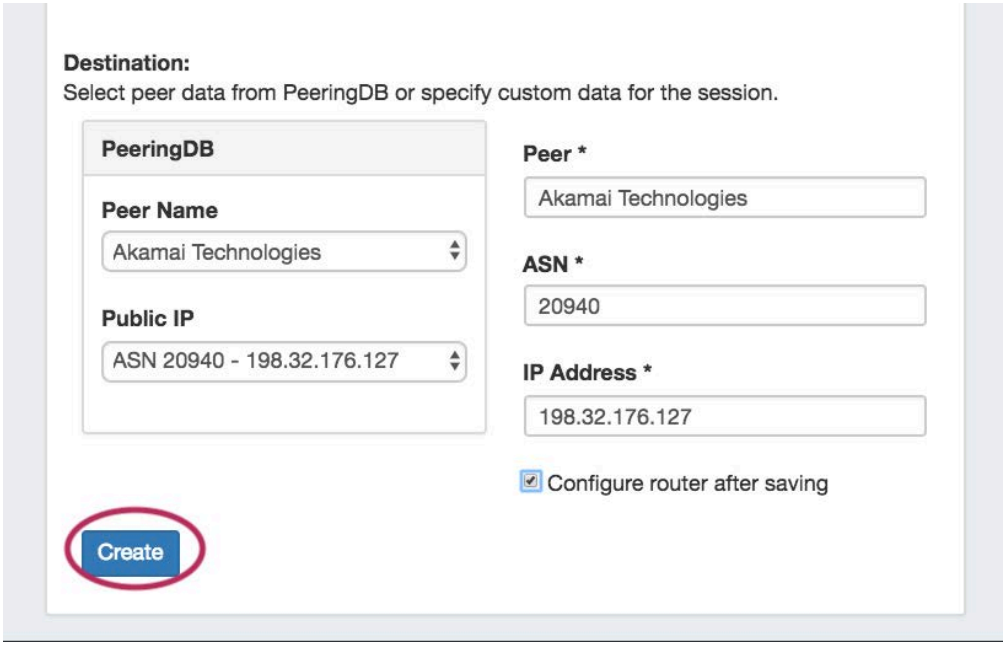


Add a Session

Add a new session by clicking the "Add Session" button, which will direct to a page to input the session details.



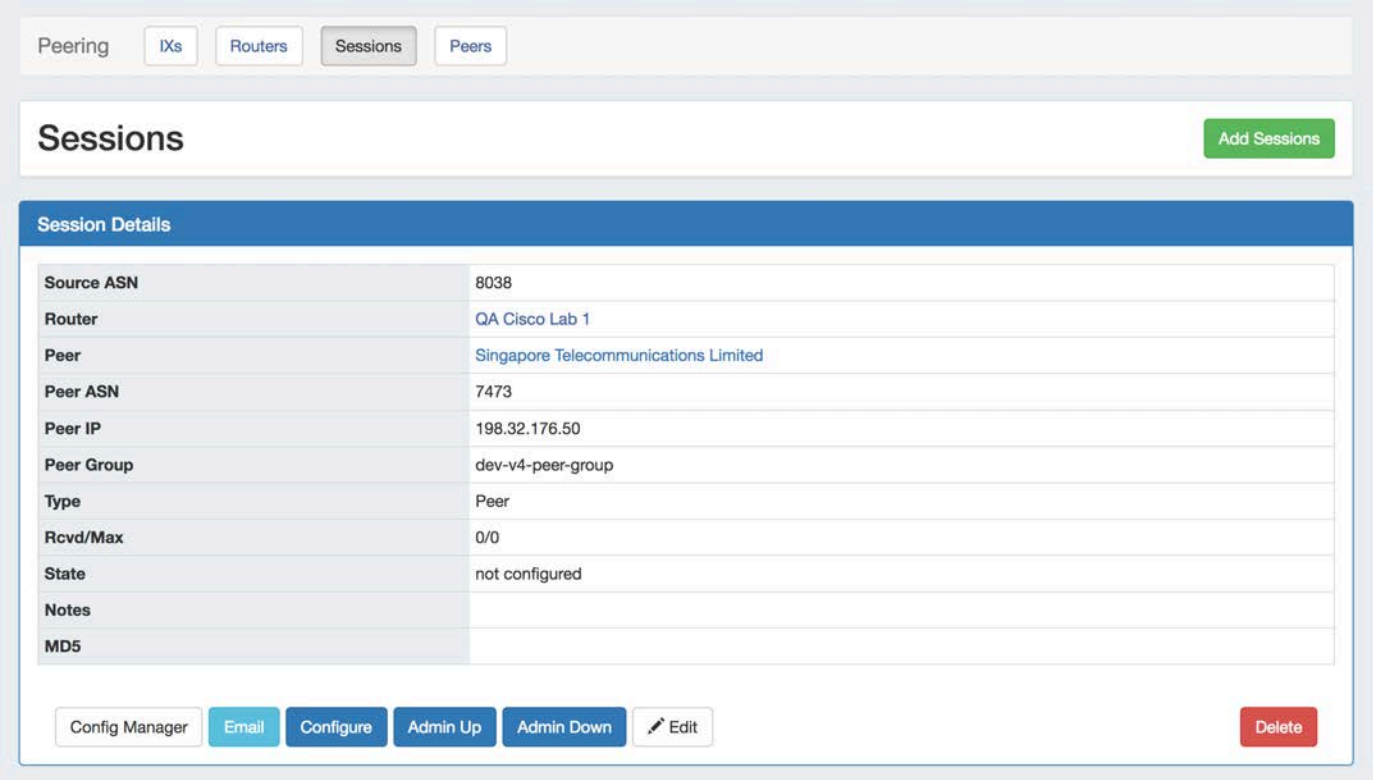
Enter the session details and router, then select the Peer Destination.



When done, click the "Save" button. If "Configure router after saving" was selected, the router will configure and automatically load the sessions page when done.

Session Details Page

Clicking on a Session entry in the list will open the Session details page.



From here, new session may be created, but it is primarily for session management. The Session may be manually configured via "Config Manager", or automatically via "Configure".

"Email" opens an Email communications form, with data automatically generated from PeeringDB. Other available tasks include Admin Up/Down, Edit, and Delete session.

Session Details Actions

Edit: Edit session information such as Type, Prefixes, Notes or MD5.

Configure: 1-click configure which uses default router configuration, username, and password settings.

Config Manager: The Config Manager allows for custom configuration commands and user-level username/ password to be entered prior to pushing the config. This is a one time use configuration.

Email: Brings up the NOC (Network Operations Center), Policy, Technical, or other preset email template. The email template pre-populates data based on peeringdb data (To address, Subject line and Peering exchange information). You have the chance to edit the email prior to sending.

Admin Up: Ups a bgp session without removing it or adding it to the config.

Admin Down: Downs a bgp session without removing it or adding it to the config. On Cisco, Admin Down moves the session to Idle (Admin) state, on Juniper it deactivates the session.

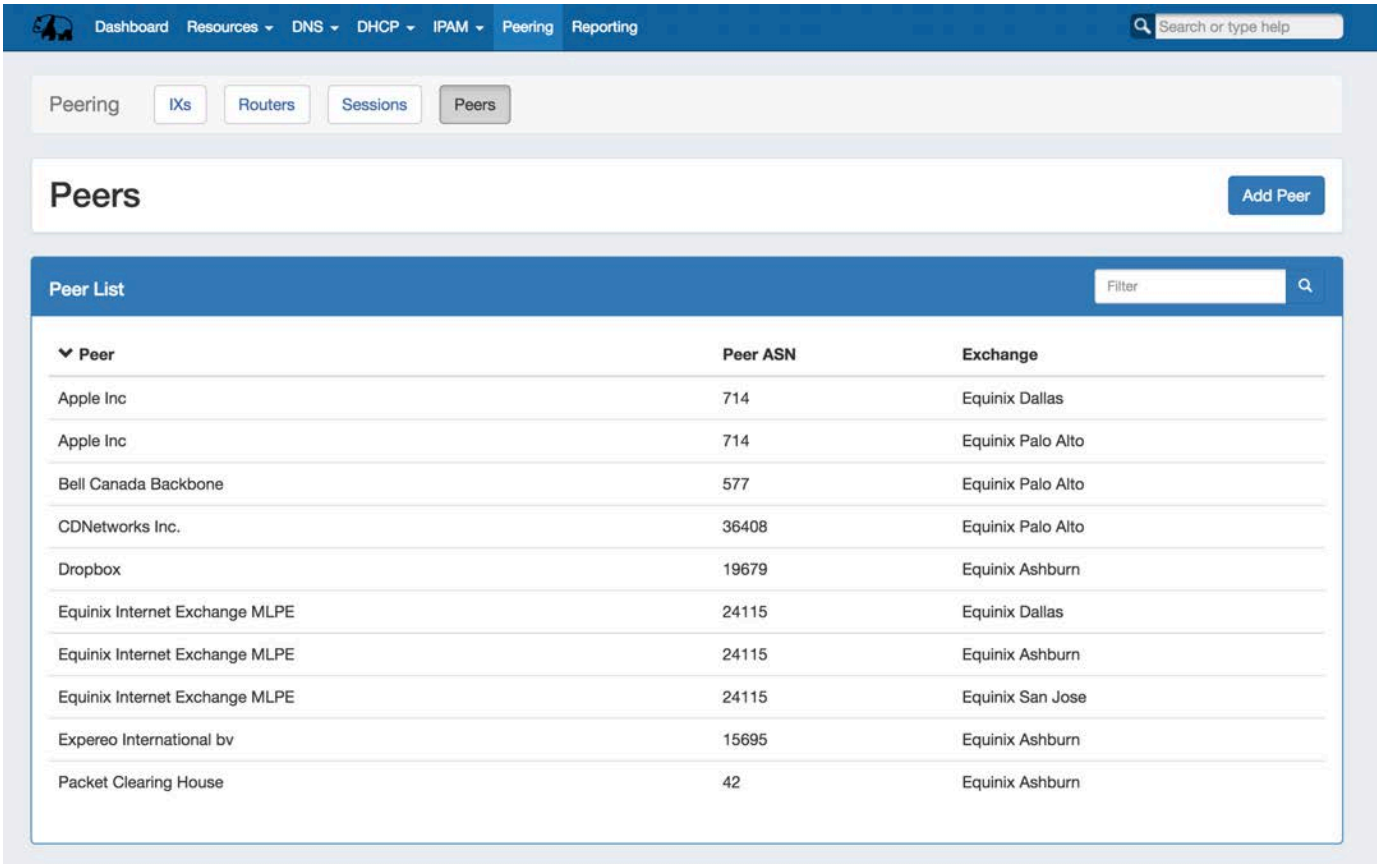
Delete: Sessions of type "Peer" are removed from the router when deleted in ProVision. Other sessions will only be removed from the sessions list in ProVision.

Managing Peers

Working with Peers

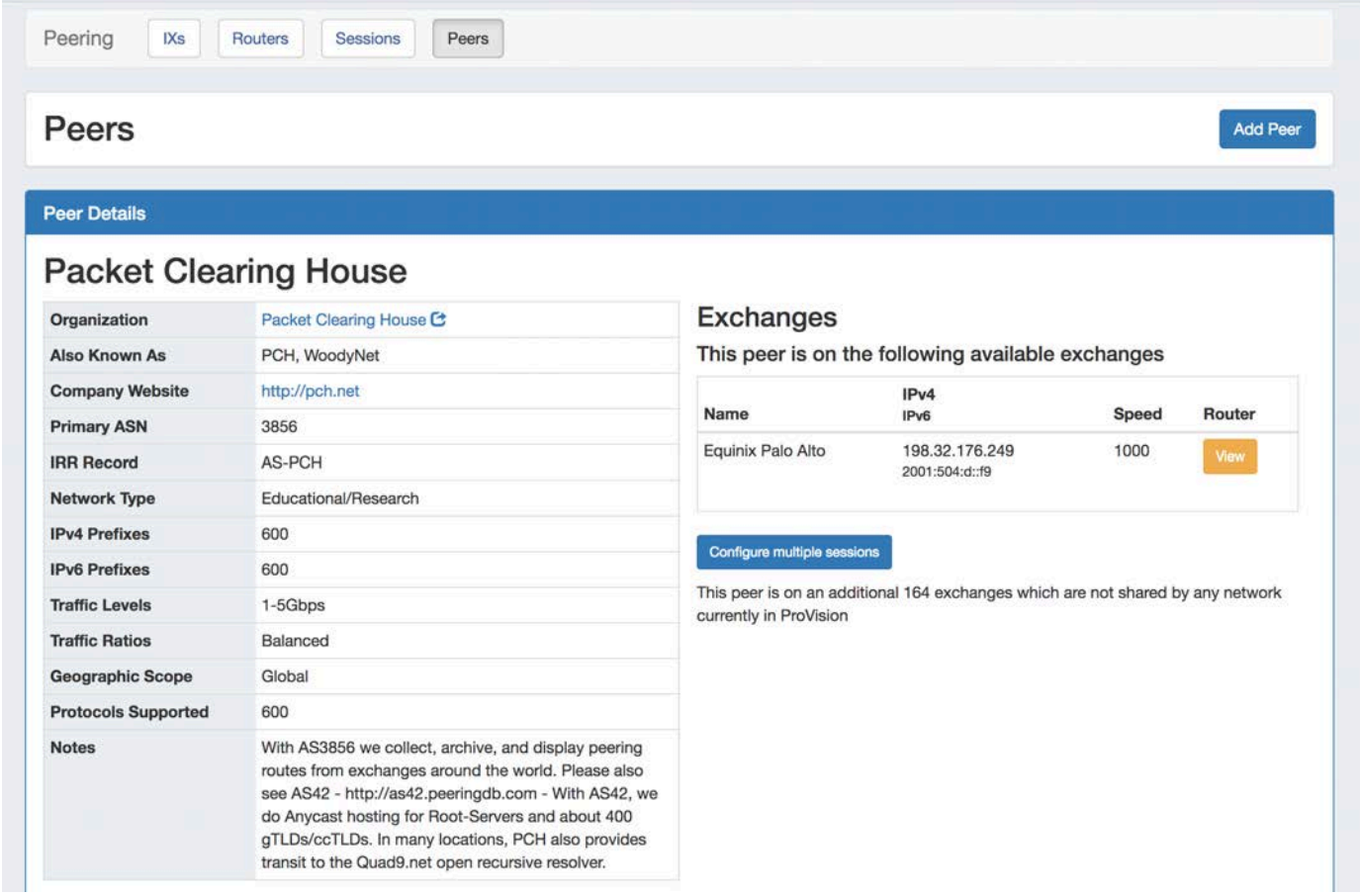
Peers Subtab

The Peers subtab lists existing peers, and allows for you to search for new peers to add via the "Add Peer" button.



Peer Details Page

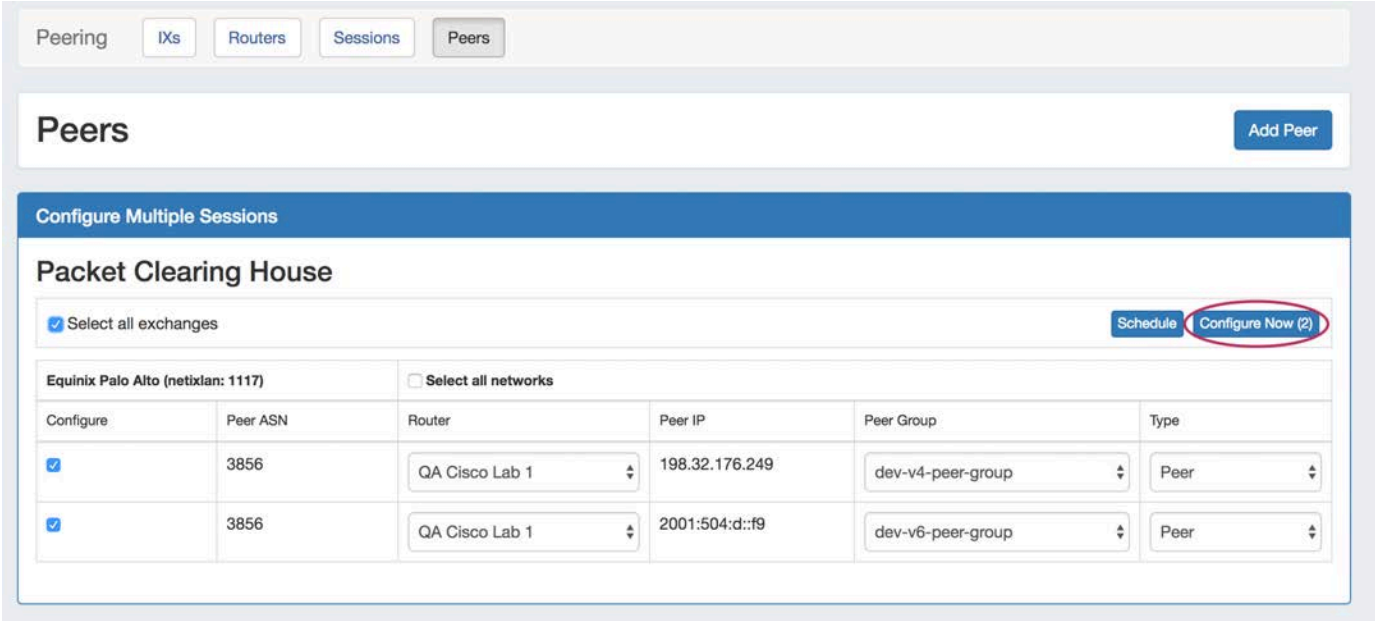
Clicking on a peer in the list will open the Peer Details page.



From here, additional Peers may be added, or you can view details for the existing peer, including available exchanges and what routers you have available on that exchange.

Configure Multiple Sessions

You may also choose to "Configure Multiple Sessions" - clicking the configure multiple buttons opens a page where you may select and set up multiple sessions for that peer, and then choose to either "Configure Now" or Schedule the config to run later via the **Scheduler** Tab.



Reporting

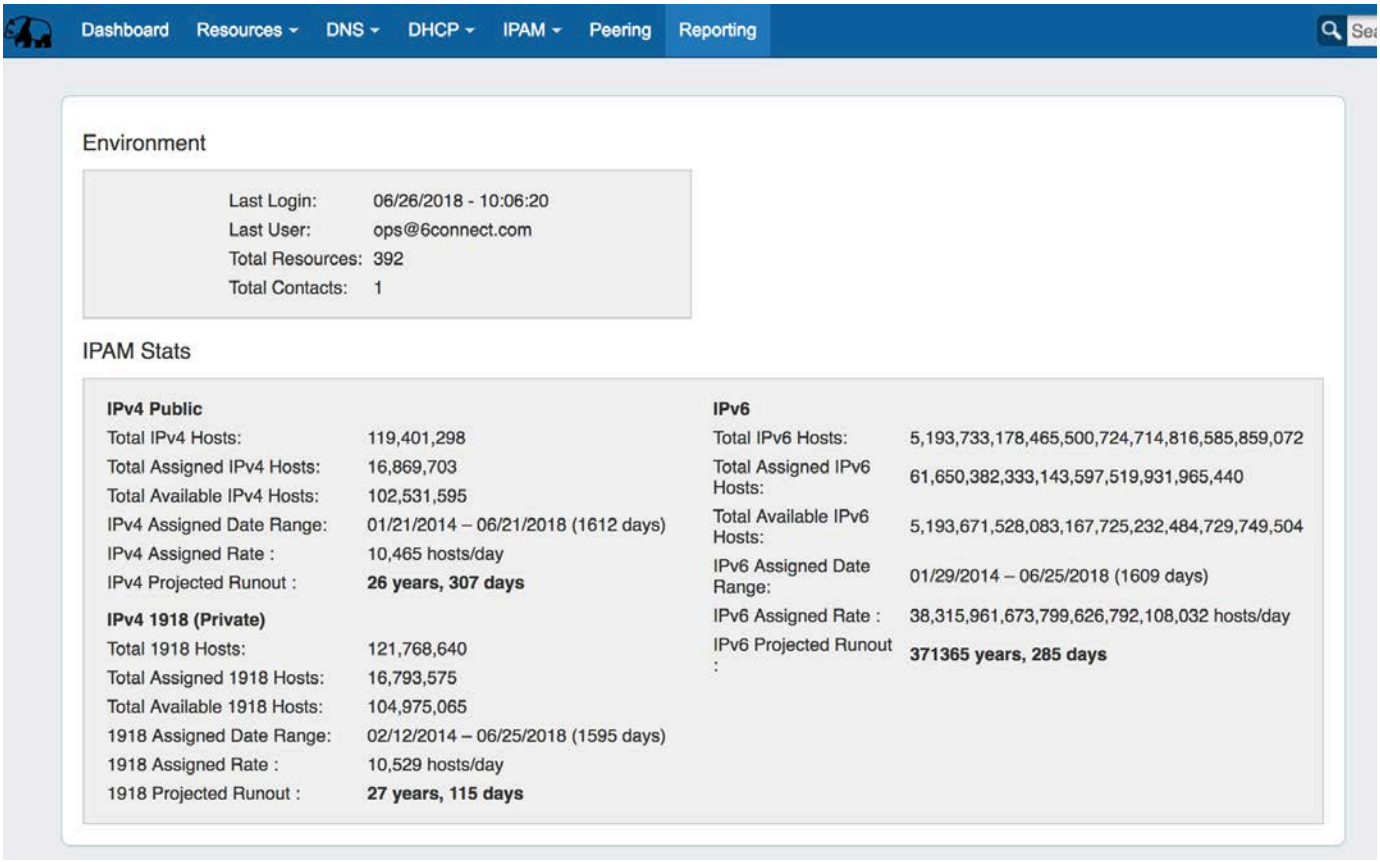
Reporting

The ProVision **Reporting** tab provides an overview of program statistics, as well as a way to view and download activity information.

- Reporting
  - Stats
  - Reports
    - User Activity
    - Customer List
    - IPAM

Stats

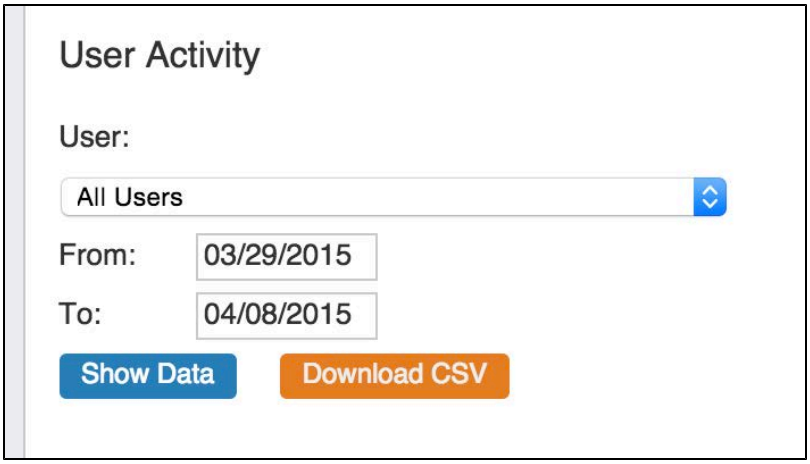
Items of interest provided under stats include most recent login, number of Resources, DNS zone breakdowns, IPAM hosts, and estimated IP runout time.



Reports

User Activity

To run a User Activity report, simply select the user from the drop down menu and a desired date range for the report. Clicking on "Show Data" will show the User, IP, Timestamp, and Action in a table at the bottom of the page. To export the data to .csv, simply select "Download CSV".



Customer List

The Customer List report reflects all Resources created under the Category of "Customer". Clicking on "Show Data" will show information collected from the Contact Info and Tech Info gadgets, parent information, and IP / zone assignment counts. To export the data to .csv, simply select "Download CSV".

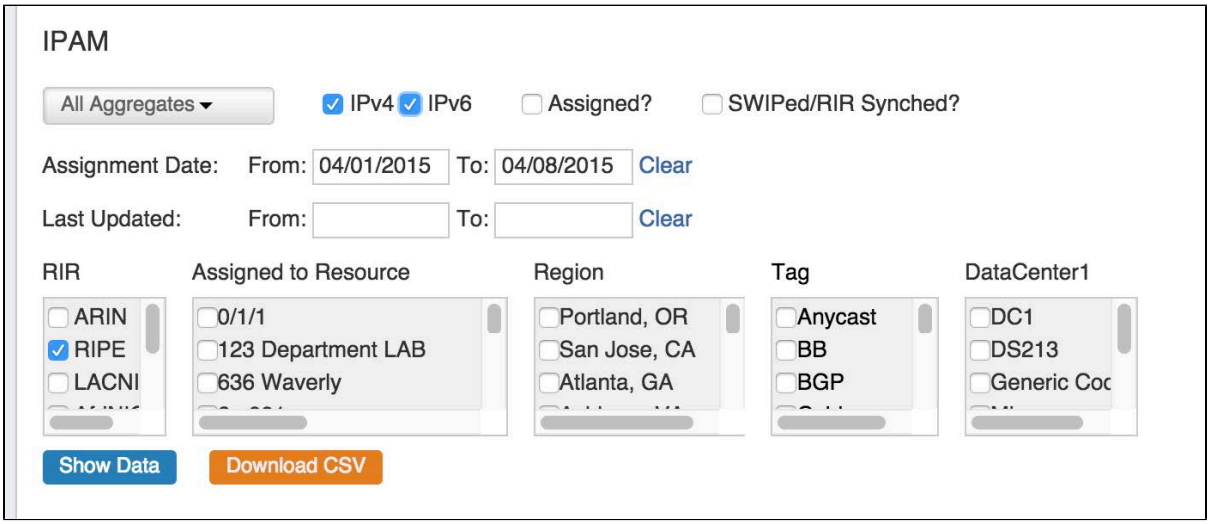


IPAM

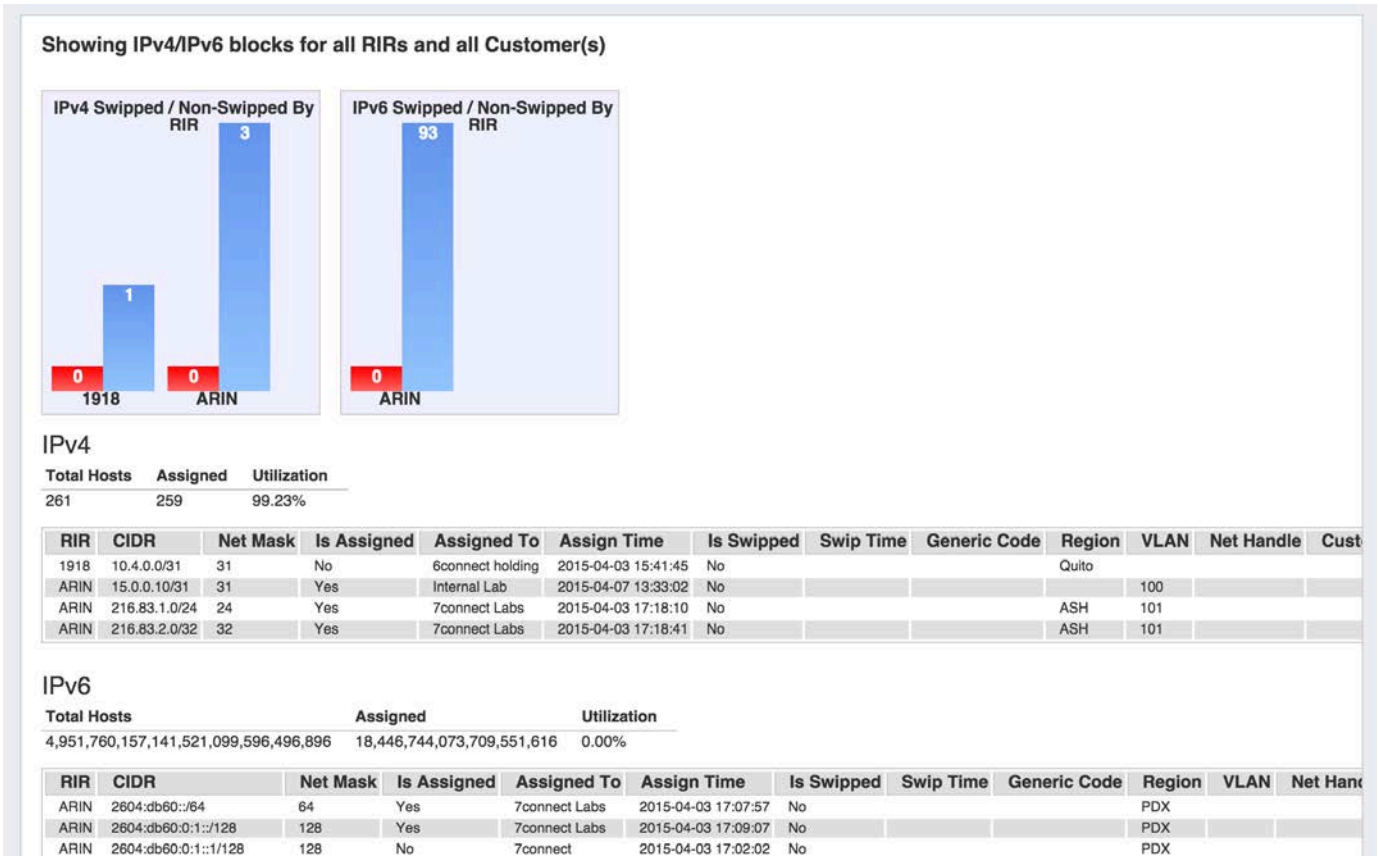
The IPAM report is highly customizable, allowing you to view information for all aggregates or selected blocks.

Required Fields: IPv4 and/or IPv6 must be selected for the report.

Optional Fields: Assigned, SWIP status, Assignment / Update dates, RIR, Assigned to Resource, Region, Tag, and Generic Code (in this case, "Datacenter1") are all optional parameters to narrow your results.



Clicking on "Show Data" will show bar charts for Swipped/ Non-Swipped by RIR, host and utilization stats, as well as detailed block information. To export the data to .csv, simply select "Download CSV".



Note: The default number of entries returned in the Reporting section for either html or .csv is limited to 5000 rows. If more rows are needed, the ProVision API may be used to retrieve larger datasets.

User Preferences

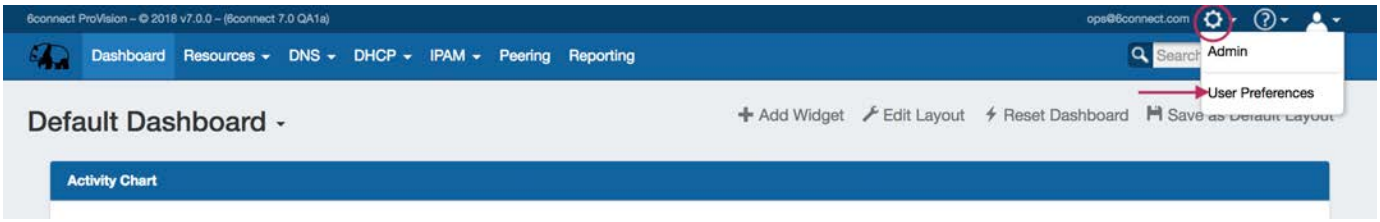
User Preferences

The User Preferences page allows the logged in user to reset their password as well as adjust per-user settings.

- User Preferences
  - Accessing User Preferences
  - Editing Preferences

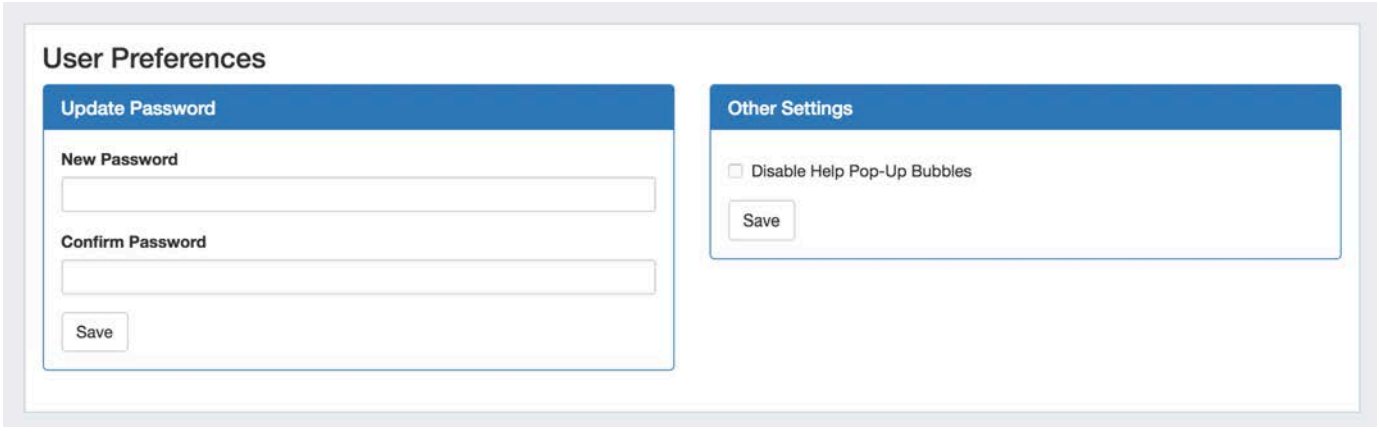
Accessing User Preferences

To access the User Preferences page, click the gear icon at the top right section of the header from any page. From there, select "User Preferences". You will be redirect to the User Preferences page.



Editing Preferences

The User Preferences page includes a section for updating the user's ProVision password, and Other Settings.



Update Password:

To update your password, type the new password in the "New Password" box, then retype under "Confirm Password" for verification. When complete, click the "Save" button.

Other Settings:

Under "Other Settings", help pop-up bubbles may be enabled or disabled. Click the check box next to "Disable Help Pop-Up Bubbles" to disable, or leave unchecked to keep enabled. When complete, click the "Save" button.

To leave the User Preferences page, click on any ProVision tab in the navigation bar.

ProVision Admin Guide

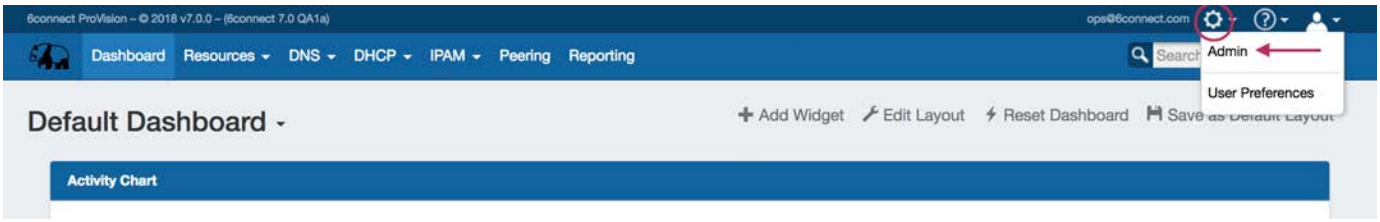
Admin Guide

The ProVision Admin Guide provides information on features accessible with Admin level permissions within ProVision. For more detailed information on features accessible in the standard user tabs , see the ProVision User Guide.

Accessing the ProVision Admin section

To access the Admin area, click the gear icon at the top right section of the header from any page.

From there, select "Admin". You will then have access the Admin section tabs of ProVision, and will see the Admin Preferences page.



Leaving the ProVision Admin section

To leave the Admin area, simply click the "Exit Admin" link in the navigation bar. You will be redirected to the Dashboard.

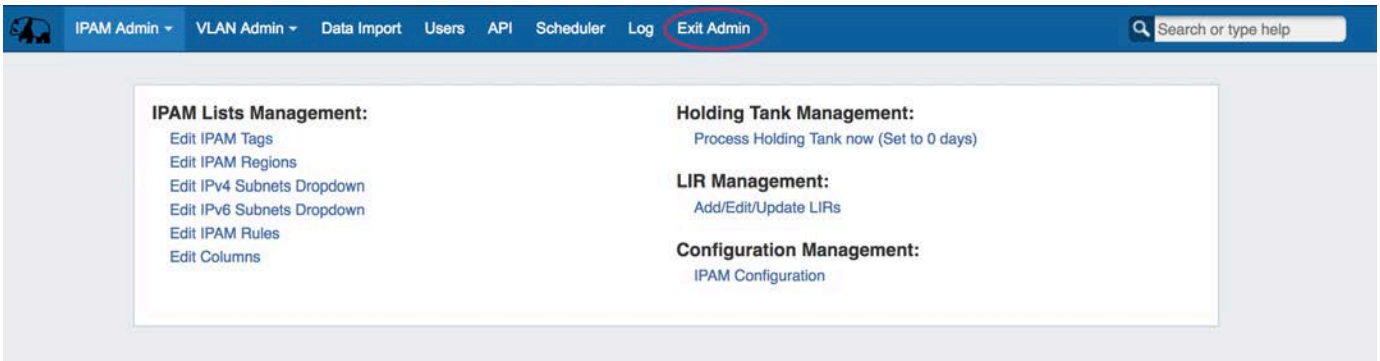
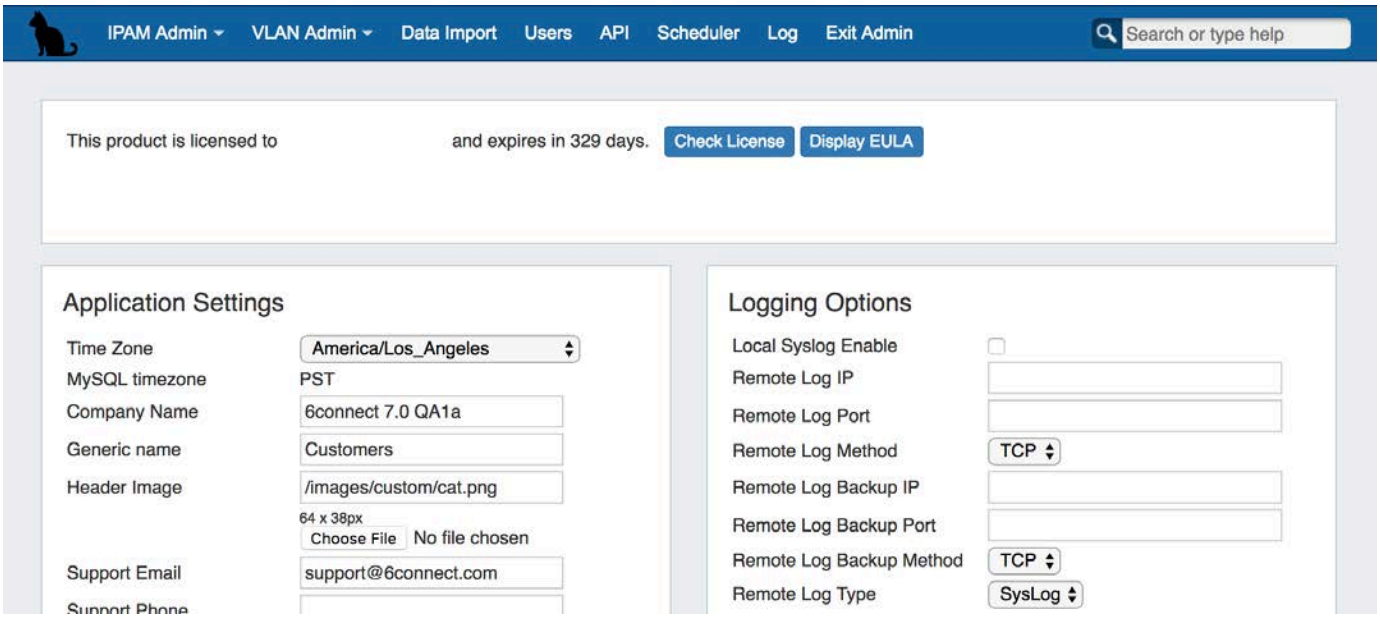


Table of Contents

- Admin Preferences
- Authentication Options
- IPAM Administration
- VLAN Administration
- DNS Administration
- DHCP Administration
- Importing Your Data
- Users & Permissions
- API Tab
- Scheduler
- Log
- RIR Integration: SWIP/RPSL

Admin Preferences

Overview



The Admin Preferences page is the home page of the Admin section of ProVision where general platform preferences may be set.

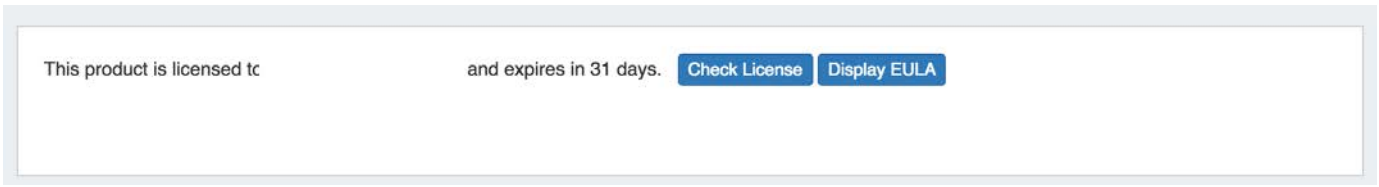
To access it, click the gear icon at the top right section of the header. From there, select "Admin". You will then have access the Admin section tabs of ProVision, and will see the Admin Preferences page.

- Overview
  - Video Walkthrough
  - License Info
  - Application Settings
    - Application Settings - Local Installation
  - DNS Settings
    - DNS Global Settings
    - DNSSEC Tools
  - Peering Settings
  - Backup Settings
    - Manual Backup - 6connect Cloud:
    - Manual Backup - Alternate Server:
    - Backup Settings - Local Installation
  - Logging Options
  - Authentication Options
    - RADIUS authentication options (local install only)
    - LDAP authentication
  - Remote Authentication Tester
  - Templates

Video Walkthrough

Note: Video at applies to versions 6.0.0 to 7.0.0. For earlier versions, see Admin Preferences Page Overview (v5.1.0+) .

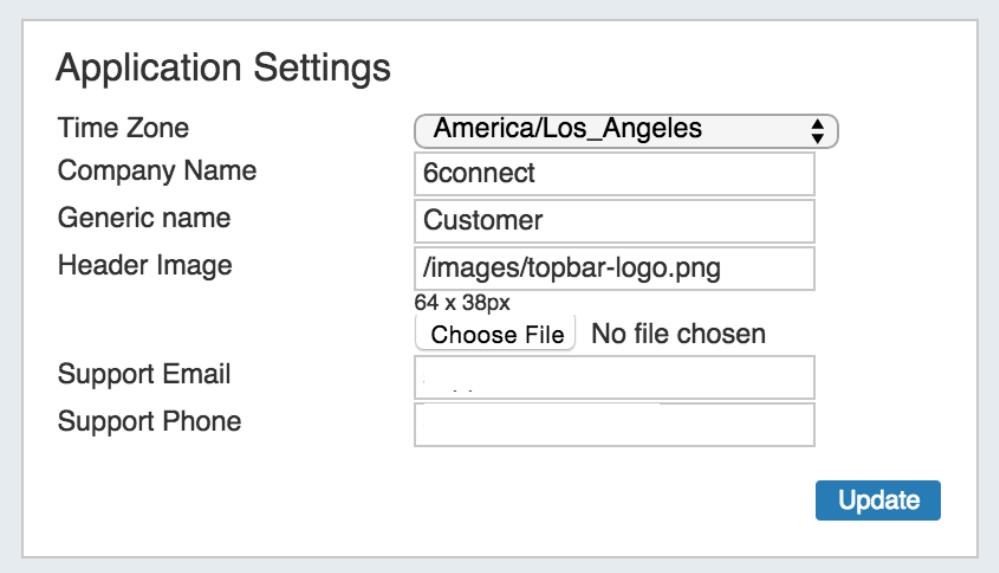
License Info



This section provides basic information on your 6connect license including the option to view the *EULA* and check your license status.

Application Settings

Application Settings is where you provide company specific info that appears in the header and Dashboard.



Time Zone: Supported Time zones are listed here: {EXT} http://www.php.net/manual/en/timezones.php. Default value is ('America /Los_Angeles') and can be modified at any time via the drop down menu

Company Name: Enter the preferred name for your company to be used.

Generic Name: This "short" name is used in abbreviated location for the "Customer" tab label, "Customer" and "Site" are common entries.

Header Image: Select an image file for the header

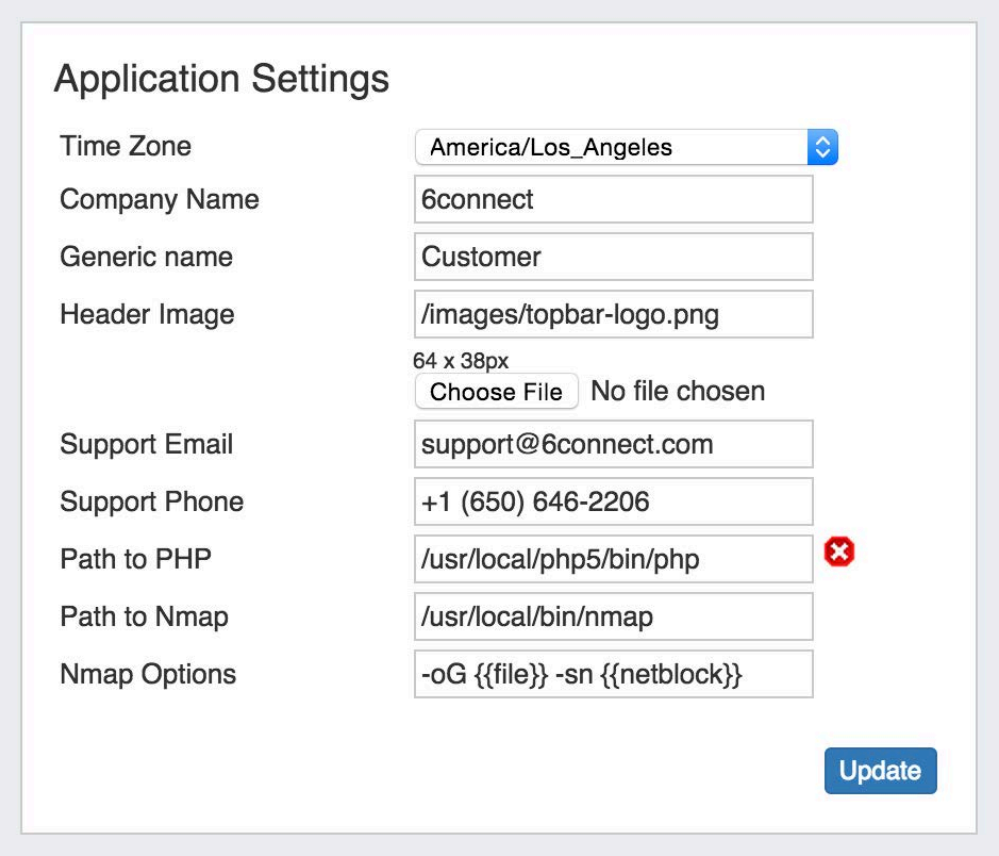
Support Email: Support Email address that displays on the Dashboard

Support Phone: Support phone number that displays on the Dashboard

Application Settings - Local Installation

Additional settings are available for local installations:

Local installations: Click here for additional settings...

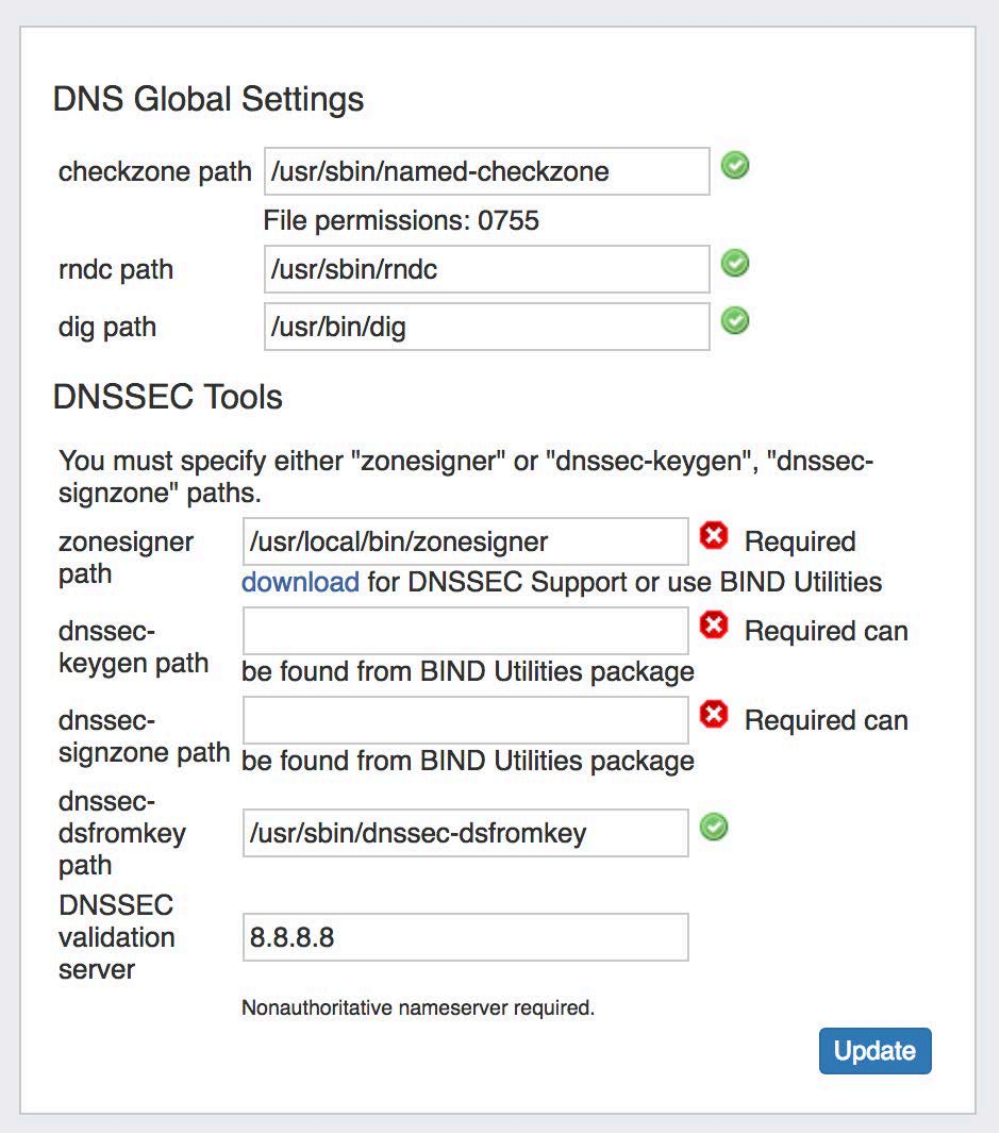


Path to PHP (Local Installation): The directory path to php location

Path to Nmap (Local Installation): The directory path to nmap location

Nmap Options (Local Installation): The directory path to nmap location

DNS Settings



DNS Global Settings

Checkzone path : Enter the checkzone path that will be used for DNS.

rndc path : Enter the rndc path that will be used for DNS.

dig path : Enter the dig path that will be used for DNS.

DNSSEC Tools

If using DNSSEC, enter the following information:

zonesigner path : Enter the zonesigner path that will be used for DNS. Zonesigner is required if dnssec-keygen and dnssec-signzone are not set.

dnssec-keygen path: Enter the keygen path. Required if zonesigner is not set.

dnssec-signzone path: Enter the signzone path. Required if zonesigner is not set.

dnssec-dsfromkey path : Enter the dnssec-dsfromkey path that will be used for DNS. Required in all cases.

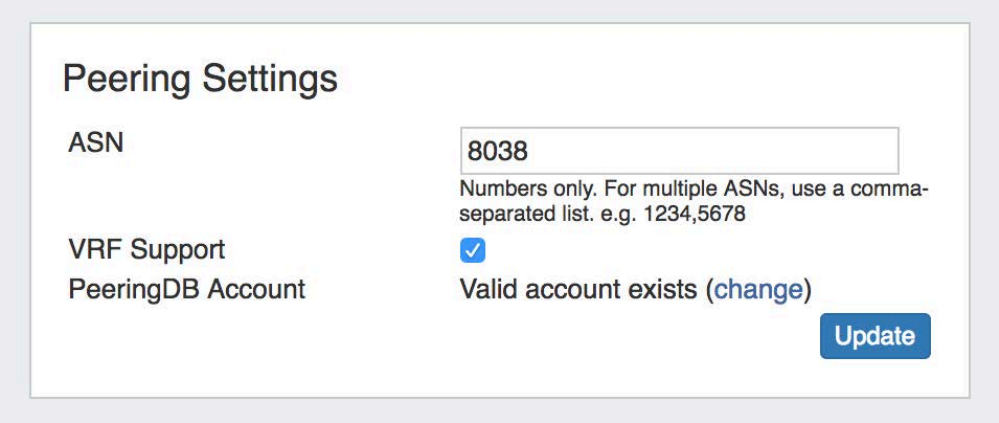
DNSSEC validation server: Enter the IP of the nonauthoritative DNSSEC validation nameserver. Optional.

Additional Entropy

If delays occur due to lack of available entropy on servers, see the following article on how to set up additional entropy using haveged here:

https://www.digitalocean.com/community/tutorials/how-to-setup-additional-entropy-for-cloud-servers-using-haveged

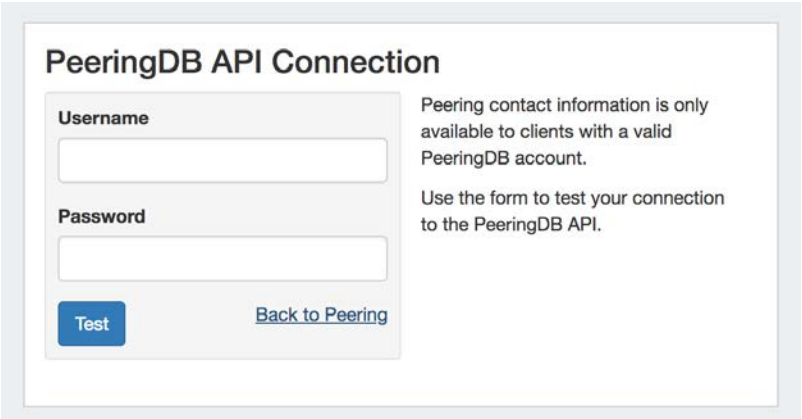
Peering Settings



ASN : Enter the ASN(s) that will be used for Peering, or may leave blank. Separate multiple ASNs with a comma.

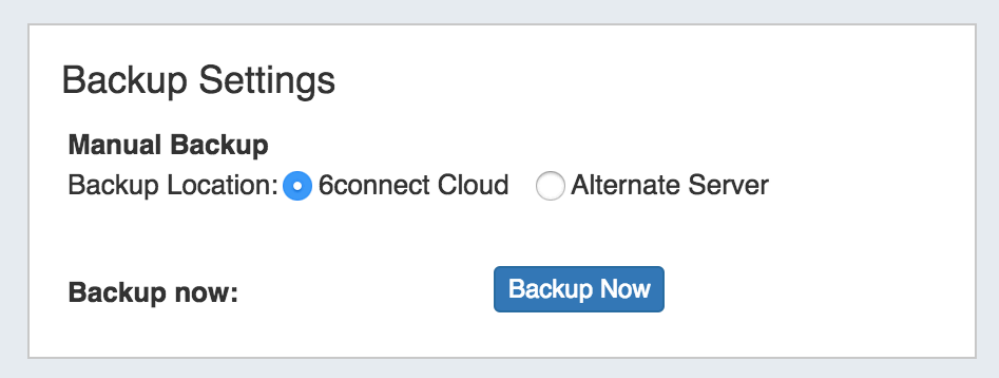
VRF Support: Check to enable adding the VRF gadget to the router Section. Currently, only supports Cisco routers.

PeeringDB Account: Click the "change" link to input PeeringDB account credentials. PeeringDB account information is required to retrieve and update peering information.



Enter a valid PeeringDB account Username and Password. When done, click the "Test" button to verify the account, and save.

Backup Settings

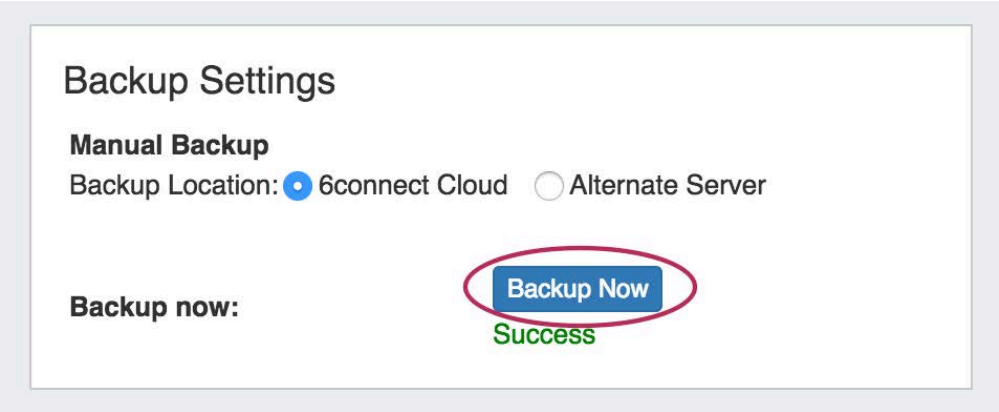


For cloud users, regular backups can be set up through the Scheduler. However, prior to imports or other large changes, you may wish to manually perform a backup.

Backup Location: The backups may be sent to the 6connect cloud, or to a specific server in the Resource system. Select the radio button for the desired location.

Manual Backup - 6connect Cloud:

Select "6connect Cloud" as your backup location, then click on the "Backup Now" button. You will see a success message below the button if successful.



Manual Backup - Alternate Server:

Manual Backup: Alternate Server saves a backup msql dump to a Server Resource already set up in ProVision.

This Resource needs to have the following information provided in the server section fields in order to establish a connection: Port, Hostname, Username, and Password.

Before selecting Alternate Server Backup:

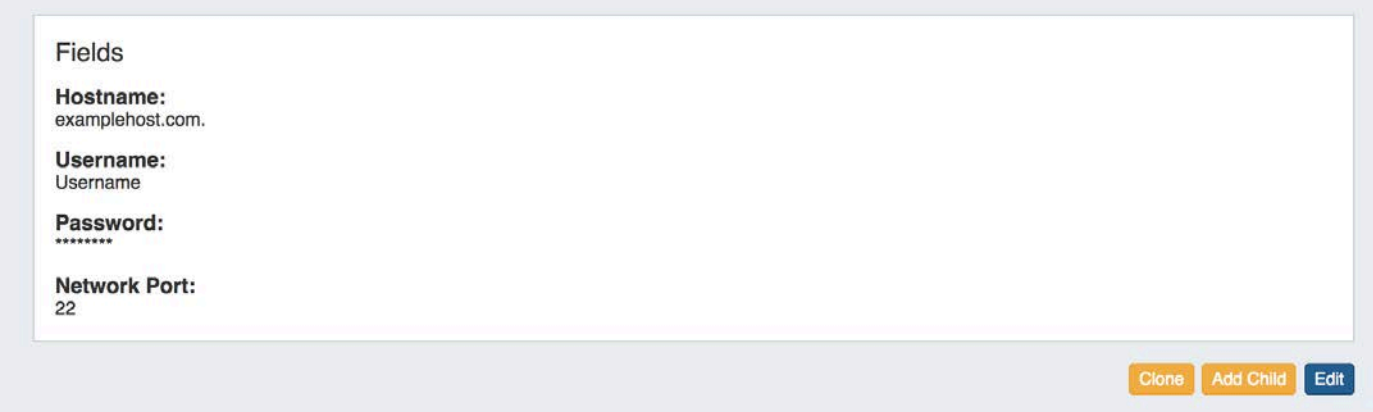
1) Ensure that the Server Resource exists in ProVision for the backup (Section = "Server"). If needed, create a new entry for the server.

2) Check that the following fields are added to the Server Section:

- Network Port (22 is typical)
- Hostname (this can also be an IP address)
- Username
- Password

See Customizing Sections and Customizing Fields for information on adding fields to sections. Hostname, Username, Password, and Network Port will all be contained under the "Existing Fields" selector when adding fields to the Section.

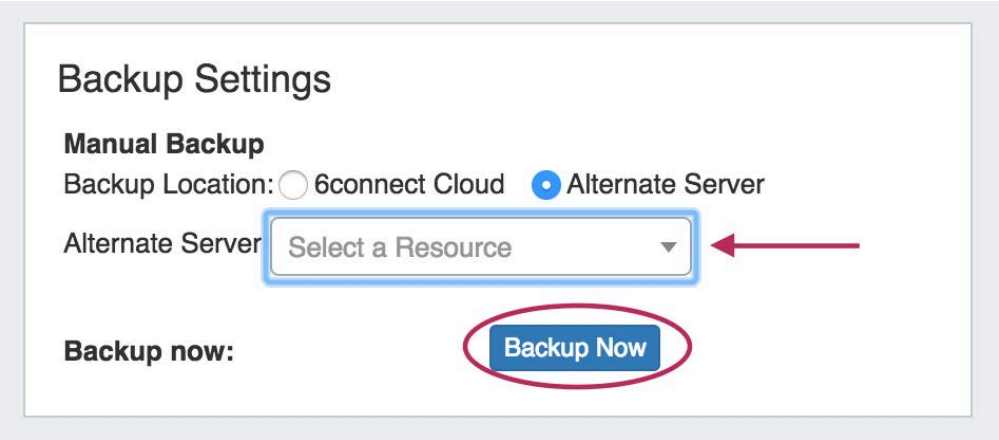
3) On the Backup Server's Entry page, verify that the server fields are filled in and correct for the ProVision server entry:



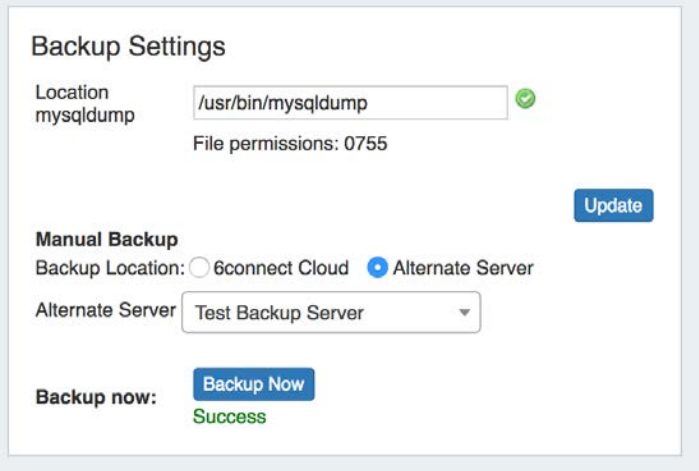
If needed, click "Edit" and add or update the information. See Working with Entries for additional information.

Once a Backup server has been set up in ProVision, it will be available to select under Backup Settings: Alternate Server.

Select "Alternate Server" as your backup location, then select the desired ProVision server Resource. After selecting your server, click on the "Backup Now" Button.



If successful, a mysql dump file will be sent to the selected server and a "Success!" message will appear.

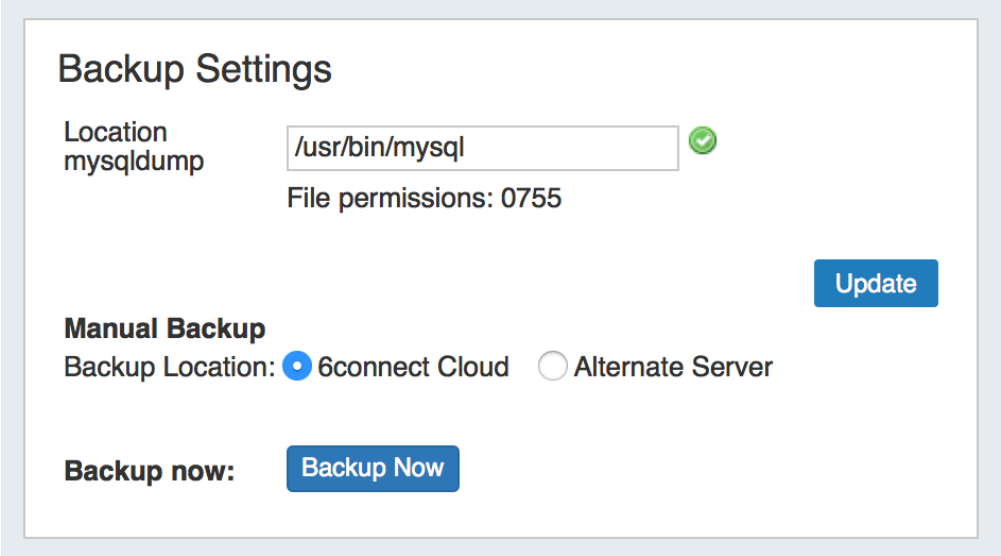


Once the connection is confirmed successful, Backups to alternate servers may be scheduled on a recurring basis through the Scheduler, or continued to be performed manually under Backup Settings.

Backup Settings - Local Installation

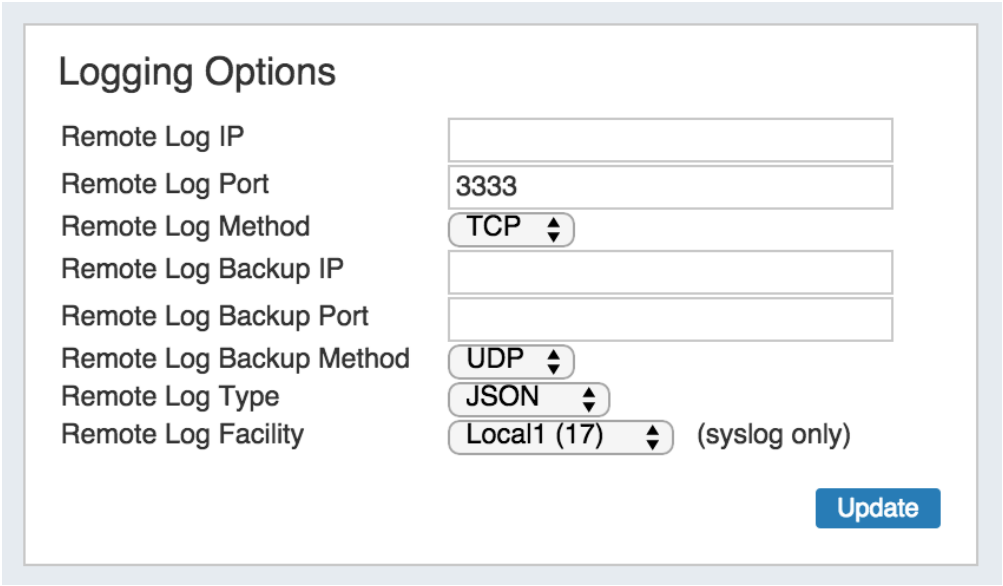
Additional settings are available for local installations:

Local installations: Click here for additional settings...



Location of mysqldump (Local Installation): This is the location of the mysqldump directory.

Logging Options



Remote Log IP: Target IP address that we will send log information to

Remote Log Port: Port number for the syslog server you will send log information to

Remote Log Method: Select TCP, UDP, SSL from the dropdown for the log delivery method

Remote Log Backup IP: Target IP address for the Backup syslog server you will send log information to

Remote Log Backup Port: Port number for the Backup syslog server you will send log information to

Remote Log Backup Method: Select TCP, UDP, SSL from the dropdown for the log delivery method

Remote Log Type: Select SysLog format or JSON output

Remote Log Facility: Select the Facility - applies to syslog only

Authentication Options

Authentication Options

Maximum Session Idle: 60 (minutes) This value controls how long a session can stay idle before being forced to log in again.

RADIUS functions are available.

Radius Enable: ☑

Radius Server Address: 208.39.104.106

Radius Authentication Port: 1812

Radius Accounting Port: 1813

Radius Key: 6connect

The Radius key is the Radius Server Secret.

Radius servers must be configured with the 6connect dictionary, located here.

LDAP functions are available.

LDAP Enable: ☑

LDAP Server Address: dev1-dfw.6connect.com

LDAP Port: 389

LDAP Security: None ⇕ Test Server

LDAP Auth DN List: cn=%LOGIN%,ou=people,dc=6con

LDAP Fetch DN: cn=%LOGIN%

LDAP Group Attribute: memberOf

The DN strings used to first authenticate the 6connect user and then to retrieve their permissions.

The string '%LOGIN%' should be inserted in place of the user's common name in both strings.

ex: cn=%LOGIN%,ou=people,dc=6connect,dc=com

Each Auth DN string will be tried in order until a user successfully authenticates.

LDAP servers must either be configured with the 6connect schema, located here, or have an internal list of user groups defined by the LDAP Group Attribute above. If a Group Attribute is set it will be used. If no Group Attribute is present the 6connect schema will be used. If both fail then users will not be able to log in to ProVision.

Update

Maximum Session Idle: This setting (minutes) controls how long a session can stay idle before being forced to log in again.

RADIUS authentication options (local install only)

Note: For implementation details, go here.

Radius Enable: Check this box to enable RADIUS functionality.

Radius Server Address: Set to the IP address of your radius server. If this is specified, it will force authentication over radius.

Radius Authentication Port: Set to the port for authentication. Default port is 1812

Radius Accounting Port: Set to the port for radius accounting. Default port is 1813

Radius Key: Set to the shared key of your radius server

LDAP authentication

Note: For implementation details, go here.

LDAP Enable: check the box to enable LDAP functionality.

LDAP Server Address: Set the IP address of your LDAP server.

LDAP Port: Set the port for your LDAP server

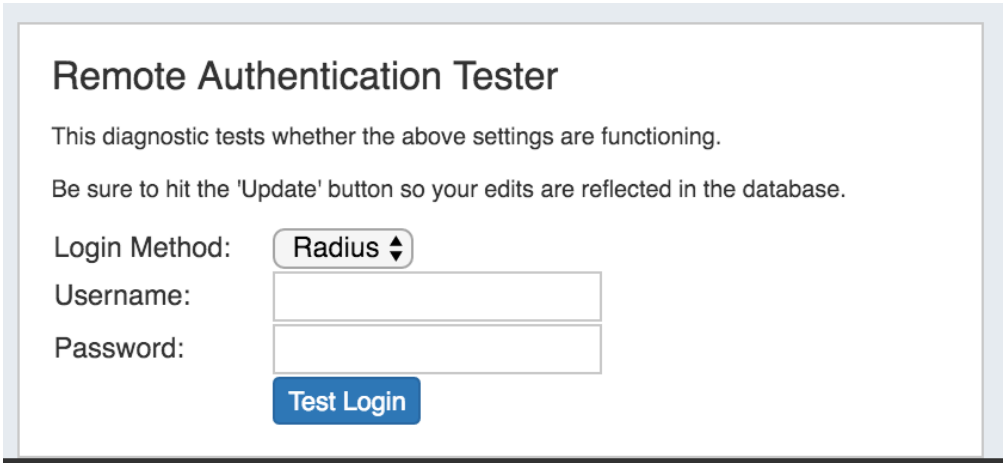
LDAP Security: Select the security method of your LDAP server - SSL, TLS or None

LDAP Auth DN/Fetch DN: These strings are used to first authentication the 6connect user and then to retrieve their permissions. The string '%LOGIN%' should be inserted in place of the user's common name both strings. (ex: cn=%LOGIN%,ou=people,dc=6connect,dc=com)

LDAP Group Attribute: If using an internal list of user groups instead of 6connect groups, enter the attribute name for the LDAP groups here. If a Group Attribute is set, it will be used first, otherwise the 6connect schema will be used.

Mapping Permissions to 6connect schema: To integrate 6connect permissions with your existing directory structure then you will need the 6connect schema. It should snap in with any existing LDAP structure and allow you to assign 6connect permissions to your existing users. You can download a copy of the schema from this section.

Remote Authentication Tester



The Remote Authentication Tester checks Radius / LDAP settings for a user.

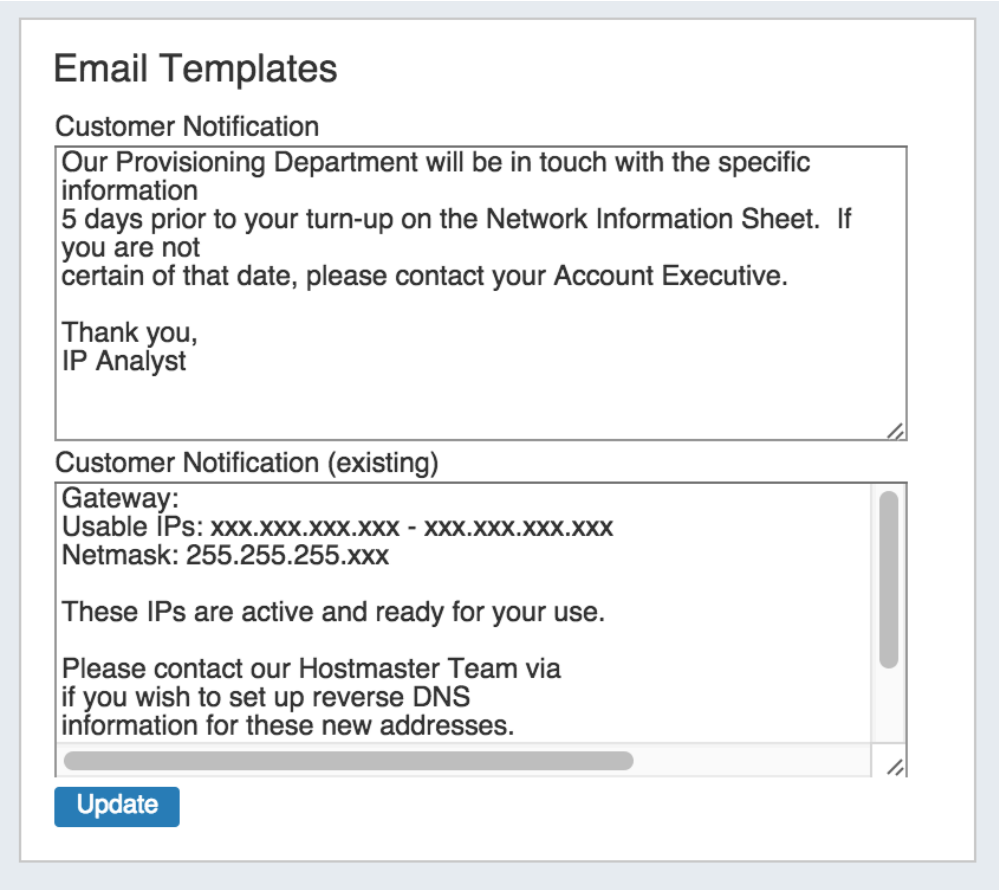
Select the Login Method (Radius or LDAP), enter the Username and Password for the user, and then click "Test Login".

Login Method: Select Radius or LDAP, according to your authentication settings.

Username: The username for the user you are testing.

Password: Password for the user you are testing.

Templates



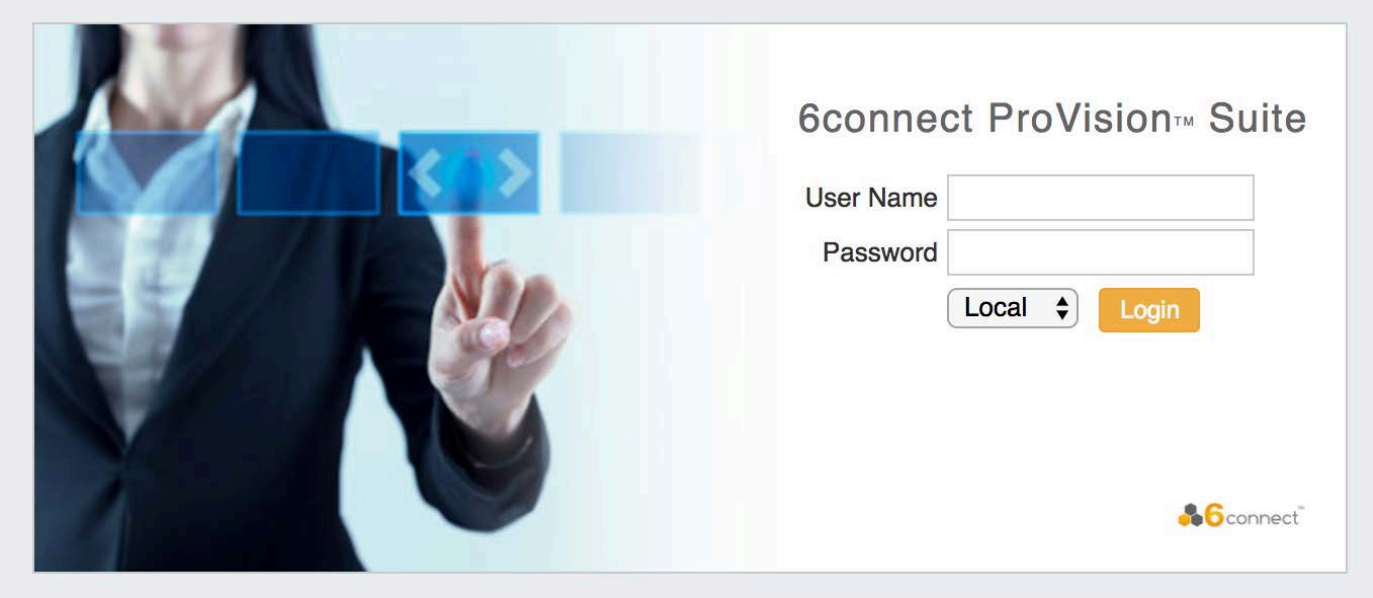
This is where you can edit outgoing email templates for IP block assignments.

To edit, click inside the text area box, make the desired changes, then click the "Update" button.

After making changes to settings, clicking any "Update" button will save your changes for all sections of the page, not just the section the update button is contained in.

Authentication Options

Authentication



Depending on the authentication method chosen by your organization, there may be a separate authentication to login or logout of the application via the drop down menu.

Change Order of Login Menu Dropdown

The drop down menu defaults to "local" - if you are using another authentication method, you can use the following to change the default ordering and improve usability.

In the file data/globals.php, add a line:

```
define('DEFAULT_LOGIN_TYPE', 'ldap');
```

Acceptable values instead of 'ldap' are 'local', 'radius' and 'ldap'.

By default, credentials are managed via the local authentication mechanism provided by 6connect. See the following sections for more detail authentication configurations:

- LDAP Authentication
- LDAP Authentication on Windows Server
- RADIUS Authentication

LDAP Authentication

LDAP Authentication

To setup an LDAP server for authentication, you must perform the following three procedures:

- LDAP Authentication
  - Configure the LDAP Server:
    - LDAP Schema - Example
  - Creating a LDAP User:
    - Test the LDAP Server
  - Configure ProVision for LDAP Authentication

Configure the LDAP Server:

Configuring the LDAP server involves ((adding the schema)) and adding LDAP users / groups to the server:

LDAP Schema - Example

```
attributetype (1.3.6.1.4.1.5023215.2.3.21 NAME 'sixConnGroup' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
objectclass ( 1.3.6.1.4.1.5023215.2.4.2 NAME 'sixConnectPermissionsV2' DESC '6Connect Permissions
Object v2' SUP top AUXILIARY MUST ( sixConnGroup ) )
```

Creating a LDAP User:

SSH into your openLDAP server and create a new 'ldif' file. Example:

```
dn: cn=JoeSmith,ou=people,dc=6connect,dc=com
cn: JoeSmith
sn: JoeSmith
objectclass: top
objectclass: person
objectclass: sixConnectPermissionsV2
sixConnGroup: "Global Admins"
sixConnGroup: "IT Engineering"
sixConnGroup: "Sales"
sixConnGroup: "Customer Admin"
userPassword: testpass
```

To create a new user, make a new ldif file and change all instances of "JoeSmith" to whatever username you wish to create and update the password. Keep all of the object class definitions as listed above. Add a sixConnGroup declaration for each ProVision user group a user is in.

After the file is created, run the following command to add the new user to LDAP server:

```
ldapadd -h [SERVER] -x -f [LDIF FILE] -D [ROOTDN] -w [ROOT PW] -v
```

Example:

```
ldapadd -h localhost -x -f 6connect.ldif -D "cn=Manager,dc=6connect,dc=com" -w secret -v
```

The user will now be active in openLDAP and can be used to login to ProVision.

Test the LDAP Server

To query the LDAP server, run the following command on any server which has openLDAP enabled:

```
ldapsearch -b [BASE] -h [IPADDRESS] -D [DOMAIN] -w [PASSWORD] [USER]
```

Note: We have not been able to use a v6 address at with this tool, even though multiple sources say it should work.

At the end of the command where [USER] is specified, user or groups can be used (in LDAP format) to query.

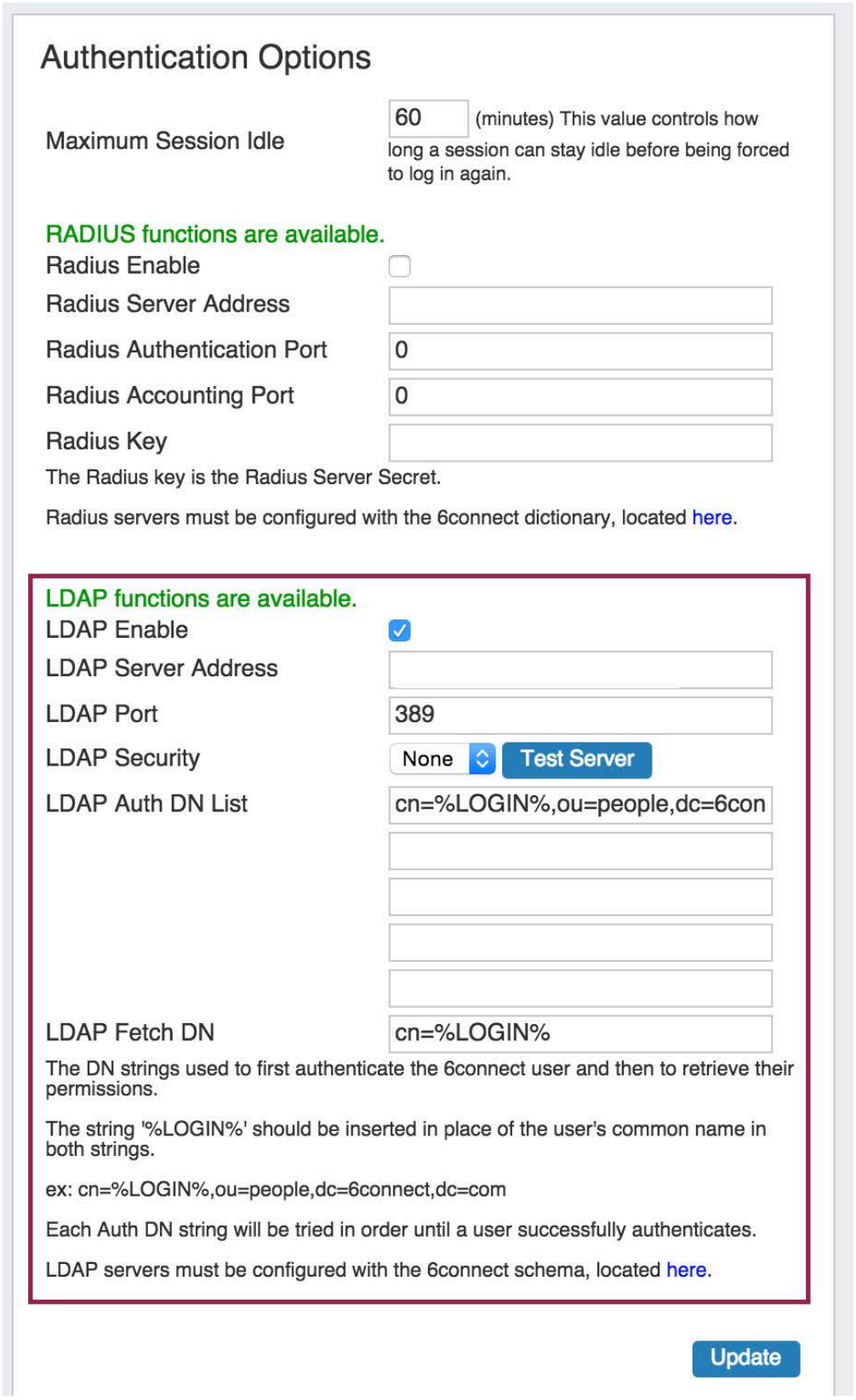
Example:

```
ldapsearch -b "dc=6connect,dc=com" -h 50.240.195.129 -D "cn=Mayor,ou=people,dc=6connect,dc=com" -w
testpass "cn=MajorMiner"
```

Configure ProVision for LDAP Authentication

To configure the use of LDAP authentication with ProVision, follow the steps below.

- Log into 6connect ProVision
- Go to Admin -> General Settings -> Authentication
- Click the LDAP Enable checkbox.
- Fill in the hostname or ip address, authentication port, LDAP Security, Auth DN, and Fetch DN.



Example values in this case would be:

LDAP Enable: (Checked)

LDAP Server Address: 52.240.195.12

LDAP Port: 389 (or SSL/TLS port is 636)

LDAP Security: None

LDAP Auth DN: cn=%LOGIN%,ou=people,dc=6connect,dc=com

LDAP Fetch DN: cn=%LOGIN%

Setting default login authentication options

In the login screen, you would select the authentication method from the dropdown. If you like, you can set the default login option in the following way:

Go to the /data/globals.php and open in vi (or other editor). Add in the following text as the last line of the file (before the closing ?>)

define('DEFAULT_LOGIN_TYPE', 'radius');

Acceptable values are "local", "radius" and "ldap". If this line is not present in globals.php, the default option is "local".
Using SSL encryption

To use SSL encryption with LDAP, the ldap.conf file must be correctly configured on the ProVision server.

Typically, the LDAP configuration file is kept at "/etc/ldap/ldap.conf". Make sure the following line is present:

TLS_REQCERT allow

and restart the webserver.

LDAP Authentication on Windows Server

LDAP Authentication on Windows Server

Starting in 3.6, ProVision supports LDAP authentication (including Windows Server!). To setup an LDAP server for authentication, you must perform the following procedures:

- LDAP Authentication on Windows Server
  - Configuring the LDAP functions on your Windows Server
    - LDAP Schema - Example
    - LDAP User Example
  - Test the LDAP Server
  - Configure ProVision for LDAP Authentication

Configuring the LDAP functions on your Windows Server

You should confirm these steps with your LDAP admin - the purpose of this walkthrough is to provide some level of detail on how to extend LDAP functionality to support integration with an application like ProVision.

Step 1: Prepare to extend the Schema (http://technet.microsoft.com/en-us/library/cc961754.aspx)

This is not a minor operation and requires interaction with various control modification areas of Windows Server:

- If you have not modified the schema before, you will need to use the Active Directory Schema console on a DC (Domain Controller) to permit write access to the DC schema.
- Since the schema object has dedicated permissions, admins must be a member of the Schema Administrator group (Schema Admins).
- Note that the DC that is holding the Schema Master Role is the only one allowed to write to it.

Step 2: Decide on method for Installing/executing Schema Extensions (http://technet.microsoft.com/en-us/library/cc961742.aspx)

If you have already used other AD integrations, this should be straightforward. We recommend using the LDIF script method

Step 3: Add and Modify a Schema Object (http://technet.microsoft.com/en-us/library/cc961575.aspx)

To add a new attribute to the schema, you first have to create a attribute object. The you will need to complete the following steps:

- Select a name for the attribute (ProVision assumes that the name will be '**sixConnGroup**')
- Get a valid Object Identifier (OID) from an issuing authority (http://msdn.microsoft.com/en-us/library/ms677620.aspx)
  - Generate an Object Identifier

    Microsoft has released a script that can generate an Object Identifier (OID): https://gallery.technet.microsoft.com/scriptcenter/56b78004-40d0-41cf-b95e-6e795b2e8a06
- Document the attribute syntax
- Confirm that the attribute should be single-value
- Confirm the attribute indexing behavior
- Decide if the attribute needs to be distributed to the Global Catalog

LDAP Schema - Example

```
attributetype (1.3.6.1.4.1.5023215.2.3.21 NAME 'sixConnGroup' SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
objectclass ( 1.3.6.1.4.1.5023215.2.4.2 NAME 'sixConnectPermissionsV2' DESC '6Connect Permissions
Object v2' SUP top AUXILIARY MUST ( sixConnGroup ) )
```

LDAP User Example

SSH into your openLDAP server and create a new 'ldif' file. Example:

```
dn: cn=JoeSmith,ou=people,dc=6connect,dc=com
cn: JoeSmith
sn: JoeSmith
objectclass: top
objectclass: person
objectclass: sixConnectPermissionsV2
sixConnGroup: "Global Admins"
sixConnGroup: "IT Engineering"
sixConnGroup: "Sales"
sixConnGroup: "Customer Admin"
userPassword: testpass
```

To create a new user, make a new ldif file and change all instances of "JoeSmith" to whatever username you wish to create and update the password. Keep all of the object class definitions as listed above. Add a sixConnGroup declaration for each ProVision user group a user is in.

After the file is created, run the following command to add the new user to LDAP server:

```
ldapadd -h [SERVER] -x -f [LDIF FILE] -D [ROOTDN] -w [ROOT PW] -v
```

Example:

```
ldapadd -h localhost -x -f 6connect.ldif -D "cn=Manager,dc=6connect,dc=com" -w secret -v
```

The user will now be active in openLDAP and can be used to login to ProVision.

Test the LDAP Server

To query the LDAP server, run the following command on any server which has openLDAP enabled:

```
ldapsearch -b [BASE] -h [IPADDRESS] -D [DOMAIN] -w [PASSWORD] [USER]
```

Note: We have not been able to use a v6 address at with this tool, even though multiple sources say it should work.

At the end of the command where [USER] is specified, user or groups can be used (in LDAP format) to query.

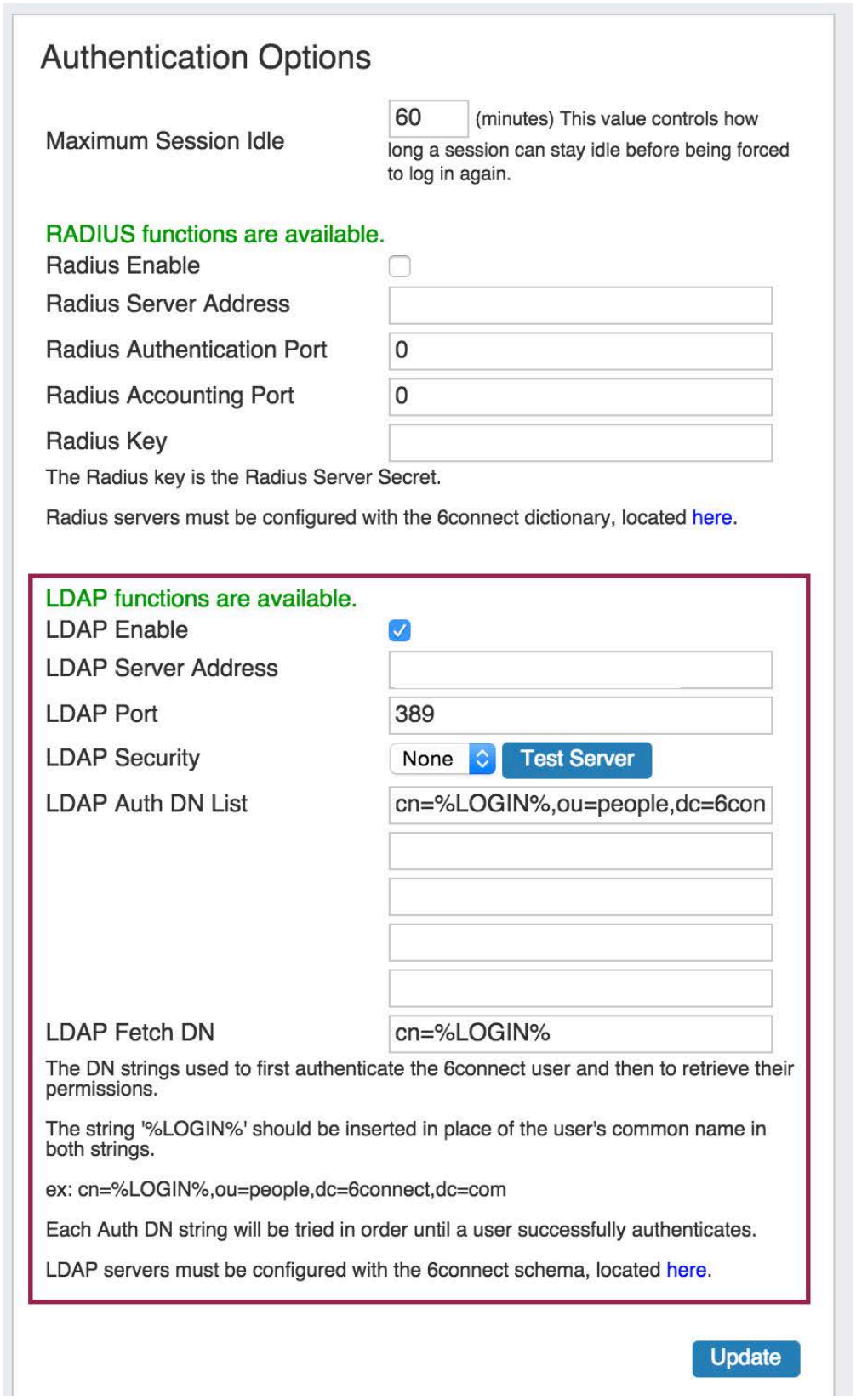
Example:

```
ldapsearch -b "dc=6connect,dc=com" -h 50.240.195.129 -D "cn=Mayor,ou=people,dc=6connect,dc=com" -w
testpass "cn=MajorMiner"
```

Configure ProVision for LDAP Authentication

To configure the use of LDAP authentication with ProVision, follow the steps below.

- Log into 6connect ProVision
- Go to Admin -> General Settings -> Authentication
- Click the LDAP Enable checkbox.
- Fill in the hostname or ip address, authentication port, LDAP Security, Auth DN, and Fetch DN.



Example values in this case would be:

LDAP Enable: (Checked)

LDAP Server Address: 52.240.195.12

LDAP Port: 389 (or SSL/TLS port is 636)

LDAP Security: None

LDAP Auth DN: cn=%LOGIN%,ou=people,dc=6connect,dc=com

LDAP Fetch DN: cn=%LOGIN%

Setting default login authentication options

In the login screen, you would select the authentication method from the dropdown. If you like, you can set the default login option in the following way:

Go to the /data/globals.php and open in vi (or other editor). Add in the following text as the last line of the file (before the closing ?>)

define('DEFAULT_LOGIN_TYPE', 'radius');

Acceptable values are "local", "radius" and "ldap". If this line is not present in globals.php, the default option is "local".
Using SSL encryption

To use SSL encryption with LDAP, the ldap.conf file must be correctly configured on the ProVision server.

Typically, the LDAP configuration file is kept at "/etc/ldap/ldap.conf". Make sure the following line is present:

TLS_REQCERT allow

and restart the webserver.

RADIUS Authentication

RADIUS Authentication

ProVision supports 6connect vendor-specific attributes (VSAs) for use with RADIUS authentication. To use these attributes, you must perform the following procedures:

- RADIUS Authentication
  - Add the 6connect VSA to the Radius Installation
  - Configure Radius Accounts
  - Test Radius Accounts
  - Configure ProVision for Radius Authentication

Add the 6connect VSA to the Radius Installation

To use the 6connect VSA, the attributes must be defined on the RADIUS server. Add the following RADIUS dictionary file to your RADIUS server and name it dictionary.6connect:

ProVision 4.0 and greater:

```
VENDOR 6connect 36009

BEGIN-VENDOR 6connect

ATTRIBUTE 6connect_user_group 10 string
#A 6connect User Group to which this user belongs.

END-VENDOR 6connect
```

Make sure to add the following to the primary dictionary file: $INCLUDE dictionary.6connect

Configure Radius Accounts

On the Radius server, configure the user accounts that will have access to the ProVision system.

An example of a ProVision account configuration for the user file on a Freeradius system for ProVision 4.0 and greater:

Example: To add a new radius user, edit the 'users' file found at /etc/raddb/users and add a block like:

Setting up a RADIUS account

```
bobber Cleartext-Password := "hello"
6connect_user_group = "Global Admins,Group 2,Group 1,Group Nonexistant"
```

The Radius server must be restarted every time you add, remove, or modify users. To restart the Radius server, use this command:

```
/etc/init.d/radiusd restart
```

Note on RADIUS attributes

There are many Radius attributes, but '6connect_user_group' is the one used by 6connect ProVision. It is a comma-separated list of all the group names that the user belongs to.

Test Radius Accounts

For ProVision 4.0 and higher, test and response should look like the following:

To query a radius server, use the following command format:

```
radtest [USERNAME] [USERPASSWORD] [SERVER] 0 [SECRET]
```

Example:

```
radtest bobber hello 208.39.140.106 0 6connect
```

A successful response will look like this:

```
Sending Access-Request of id 198 to 208.39.140.106 port 1812
User-Name = "bobber"
User-Password = "hello"
NAS-IP-Address = 67.221.240.229
NAS-Port = 0
Message-Authenticator = 0x00000000000000000000000000000000
rad_recv: Access-Accept packet from host 208.39.104.106 port 1812, id=198, length=69
Attr-10 =
0x476c6f62616c2041646d696e732c47726f757020322c47726f757020312c47726f7570204e6f6e6578697374616e74
```

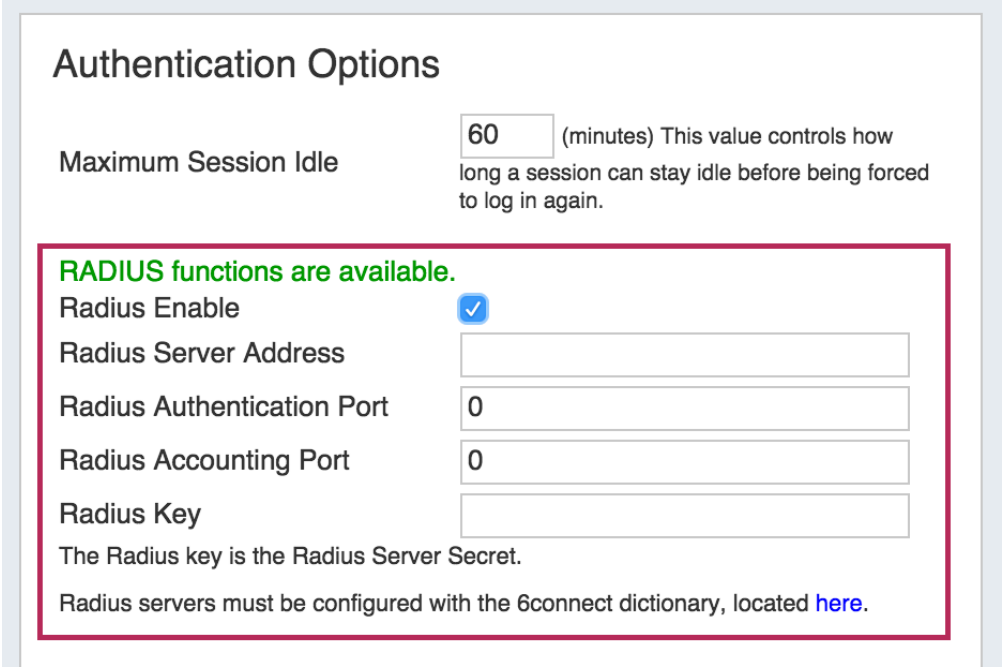
A rejected response may look like this:

```
Sending Access-Request of id 68 to 208.39.140.106 port 1812
User-Name = "bobberbro"
User-Password = "hello"
NAS-IP-Address = 67.221.240.229
NAS-Port = 0
Message-Authenticator = 0x00000000000000000000000000000000
rad_recv: Access-Reject packet from host 208.39.104.106 port 1812, id=68, length=20
```

Configure ProVision for Radius Authentication

To configure the use of Radius authentication with ProVision, follow the steps below.

- Log into 6connect ProVision
- Go to Admin -> Authentication
- Ensure that Radius functions are marked as available. Radius functions are always available on 6connect cloud instances. Radius functions are available on VM Images and Local Installations only if the relevant PHP Pear Radius Libraries have been installed.



- Click the Radius Enable checkbox.
- Fill in the hostname or ip address, authentication ports, accounting port, and shared Radius key as specified.

Setting default login options

In the login screen, you would select the authentication method from the dropdown. If you like, you can set the default login option in the following way:

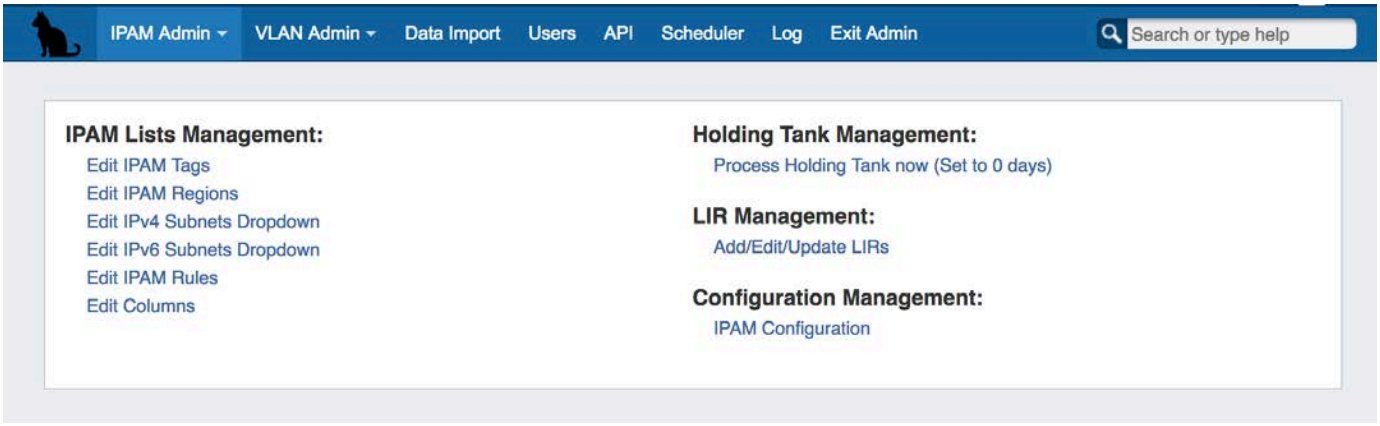
Go to the /data/globals.php and open in vi (or other editor). Add in the following text as the last line of the file (before the closing ?>)

define('DEFAULT_LOGIN_TYPE', 'radius');

Acceptable values are "local", "radius" and "ldap". If this line is not present in globals.php, the default option is "local".

IPAM Administration

Overview



IPAM Administration is accessed through the Admin area of ProVision. It includes sections to manage IPAM Lists, the Holding Tank, LIR, and IPAM Configuration.

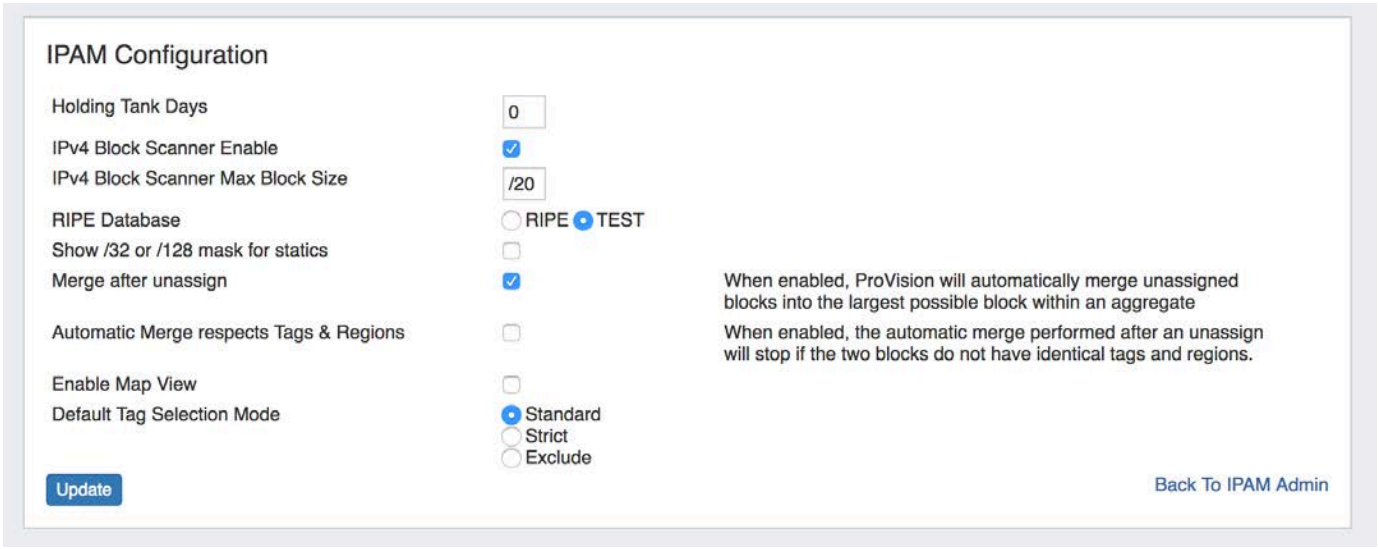
- Overview
  - IPAM Lists Management
  - IPAM Configuration
  - Holding Tank Management
  - LIR Management and Use
  - Additional Information:

IPAM Lists Management

These links are to the respective IPAM Parameters that are available for customization - tags, regions, subnets, and IPAM Rules.

See the IPAM Parameters page for additional details and examples for updating IPAM Parameters, IPAM customization, and IPAM Rules for additional information on managing IPAM Rules.

IPAM Configuration



Holding Tank Days: This is the number of days that a block will be held in "Holding" status before being available to be moved to the Available pool, and thus ready to be assigned. By default this is initially set to 30 days.

IPv4 Block Scanner Enable: This is a beta feature that allows a user to scan a block of IPv4 space and show host counts of responding addresses.

IPv4 Block Scanner Max Block Size: The max size (mask) the IPv4 Block Scanner considers.

RIPE Database: Select the desired database - RIPE or TEST.

Show /32 or /128 mask for statics: Enable to show /32 or /128 masks.

Merge after unassign: Select to automatically merge adjacent blocks when they become available after unassign.

Automatic Merge respects Tags & Regions: When "Merge after unassign" is selected, the automatic merge will stop if the two blocks do not have matching tags and regions.

Enable Map View: Select to enable a map view of assigned aggregates on the IPAM tab, determined by assigned region.

Default Tag Selection Mode: Set which radio button will be selected by default when working with tags in the IPAM Gadget

Holding Tank Management

When IPv4/IPv6 resources are reclaimed, they are placed into the "Holding Tank". This feature allows for a block to stay out of the available address pools until the administrator approves it.

See the Holding Tank Management page for more details.

LIR Management and Use

ProVision supports multiple LIRs from the UI. This allows users to select from various LIRs when they want to update SWIP/RPSL information for a subnet allocation.

See the LIR Management and Use page for more details.

Additional Information:

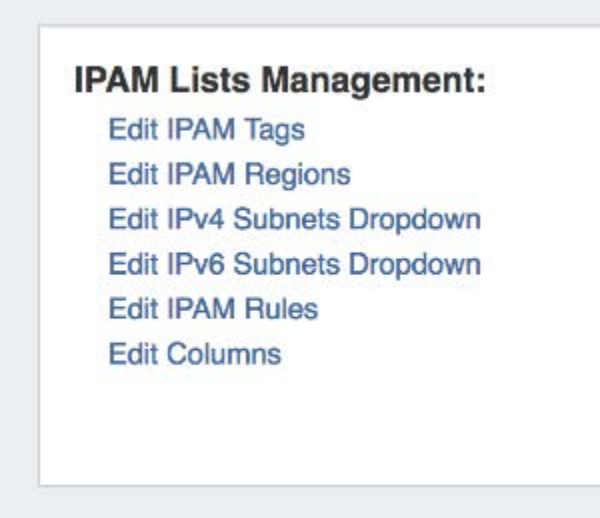
- IPAM Parameters
- IPAM Rules
- Holding Tank Management
- LIR Management and Use

IPAM Parameters

IPAM Parameters - Overview

IPAM Lists Management is access from the Admin section of ProVision, under the **IPAM Admin** tab.

Through this area, admin users can update IPAM tags, IPAM Regions, IPv4 / IPv6 subnets, IPAM Rules, and IPAM Columns.

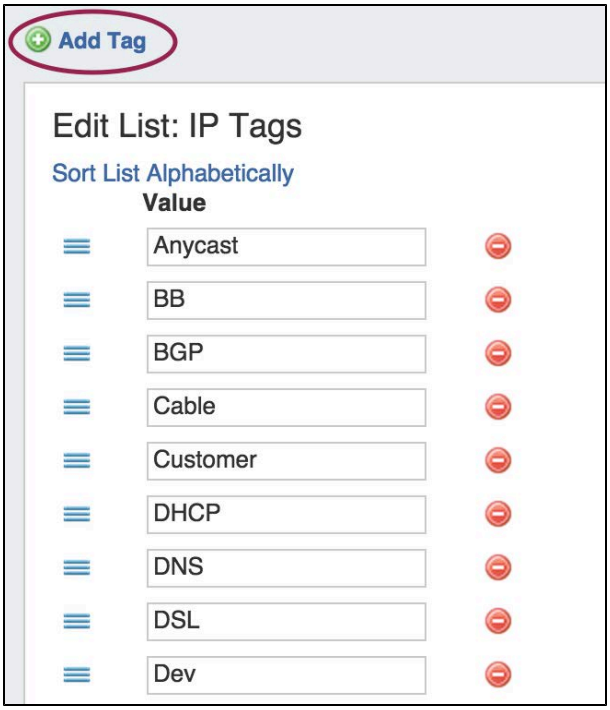


- IPAM Parameters - Overview
  - Add / Edit IPAM Tags
  - Add / Edit Regions
  - Add / Edit IPv4 / IPv6 Subnet Dropdowns
  - Edit IPAM Columns
  - Edit IPAM Rules
  - Additional Information

Add / Edit IPAM Tags

When you are applying properties to IP blocks, you have the option to edit tags. IPAM Tags are used in a number of areas in ProVision and can be added or edited from this screen.

To add a new IPAM tag, click on "Add Tag" at the top of the Edit IPAM Tags page.



Then, type in the desired name value for the new tag, and hit "Add Tag".



To **edit** a tag, simply type your changes in to the text box with the tag name. Tags with unsaved changes will be highlighted until saved.

To **reorder** the tag list, click and hold on the "list" icon (three lines) to the left of the tag name, then drag to the desired location in the list and release.

To **delete** a tag, click on the red "delete" symbol to the right of the tag name.



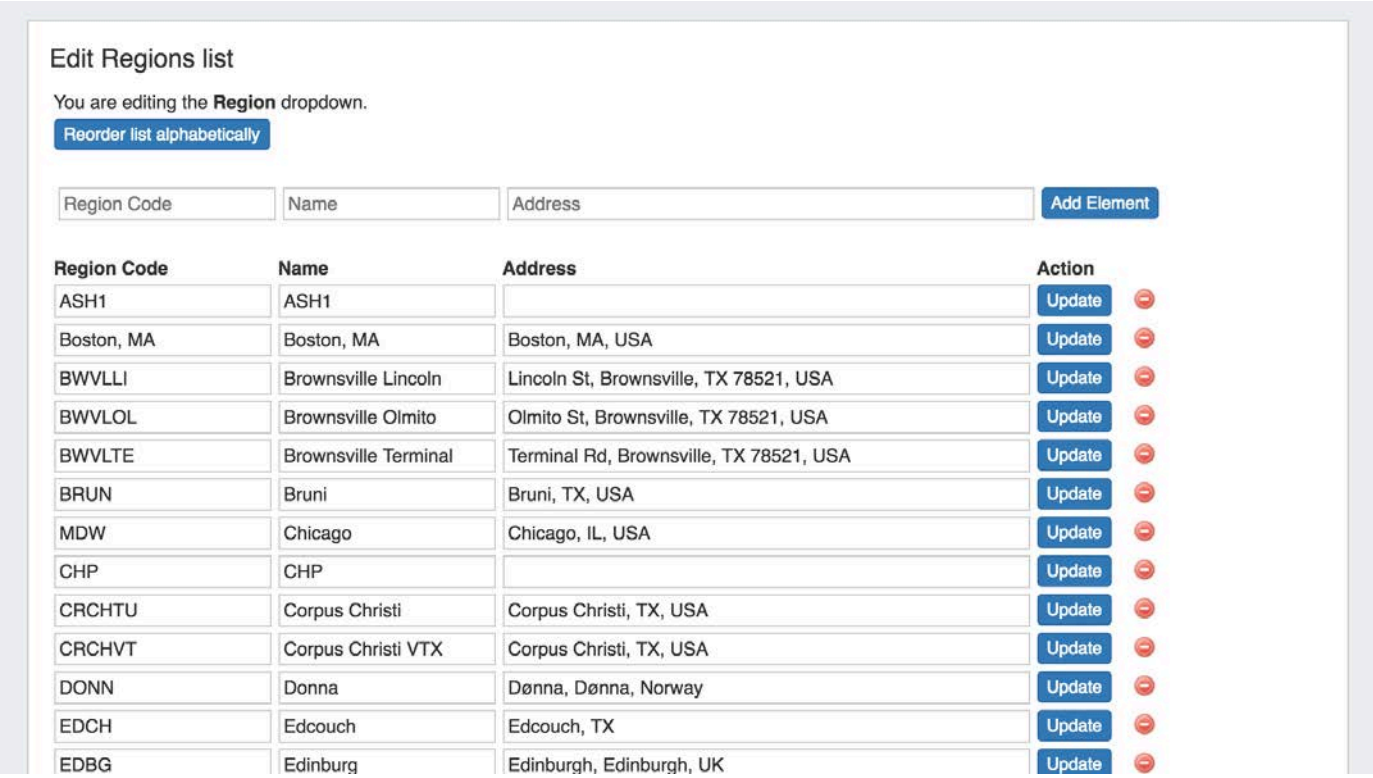
When complete, be sure to click on the "Save" button to save your changes.

Add / Edit Regions

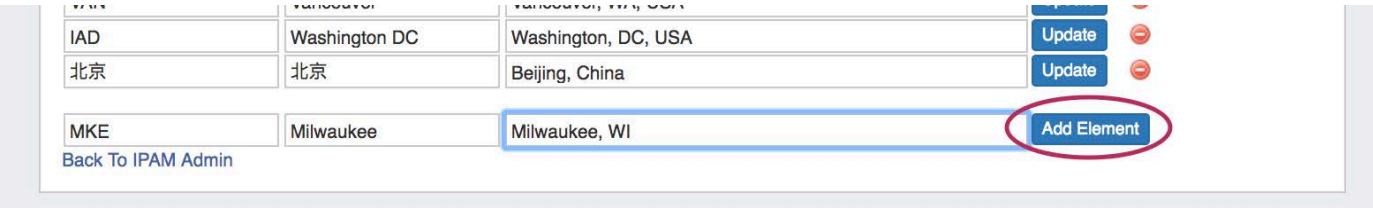
If enabled, Regions can function as a way to further define your network segments (regional tie-downs, etc.). This gives you flexibility for allocations and assignments beyond simply using Tags. Regions are used by the IPAM Gadget, the IPAM Manage UI, and IPAM Map View.

The Regions display includes fields for Value, Name, and Address. 'Name' is the region name that will show in the IPAM Gadget and IPAM Manage screens, whereas 'Value' is the value that will be written to the database, used for API calls, and is also used for some filter selectors. The Address field is used by IPAM map view to geolocate aggregates with that assigned region. Address may be in the form of a City / State / Country or a full street address. The more accurate your address information, the more accurate map view will be with your aggregate locations.

To add a new Region item, type in a new Value and Name into the empty fields at the bottom of the list.



Then, click "Add Element.



To **edit** a region, simply type your changes in to the text box with the region name. Then, click on the "Update" Button.

To **delete** a region, click on the red "delete" symbol to the right of the region name.

To **reorder** the regions list, click on the "Reorder list alphabetically" button.

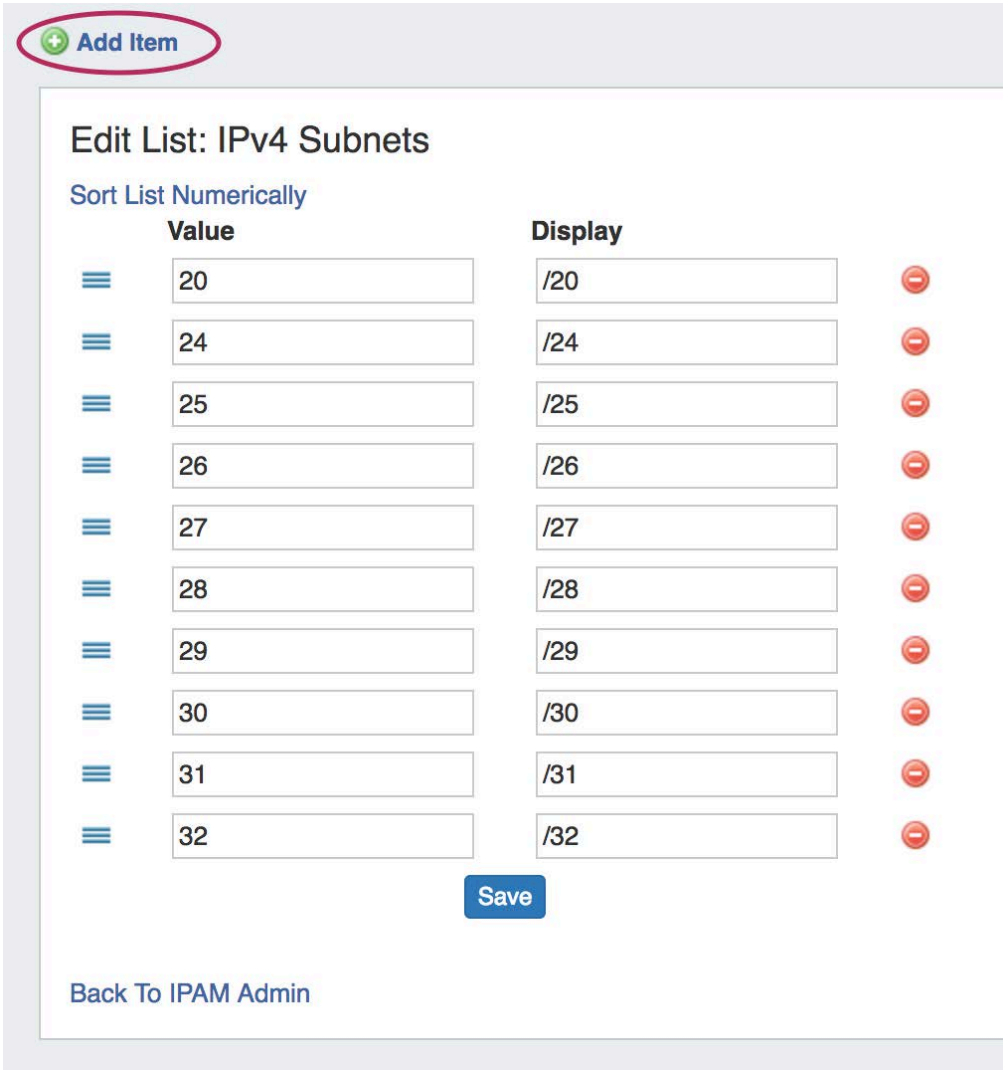
Add / Edit IPv4 / IPv6 Subnet Dropdowns

When assigning blocks using the "Smart Assign" function in the IPAM Gadget, the user has an option to assign an IP resource by allocation size. ProVision supports assignments down to a single host level (/32 for IPv4, /128 for IPv6).

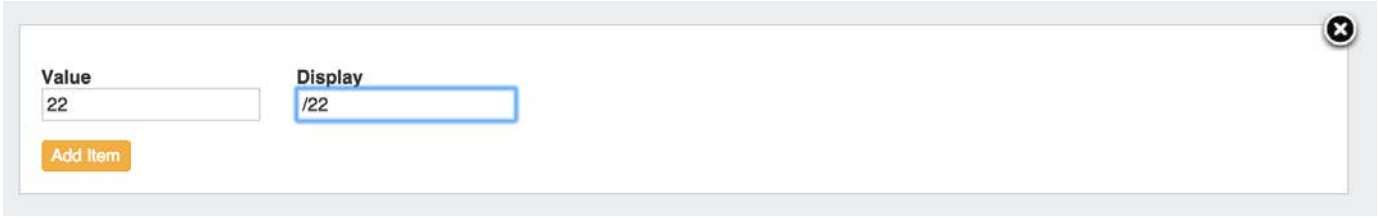
Note on Editing the Subnet Dropdown

Keep in mind that this is a global edit. If the values in the dropdown are changed, it will affect ALL users of the ProVision application.

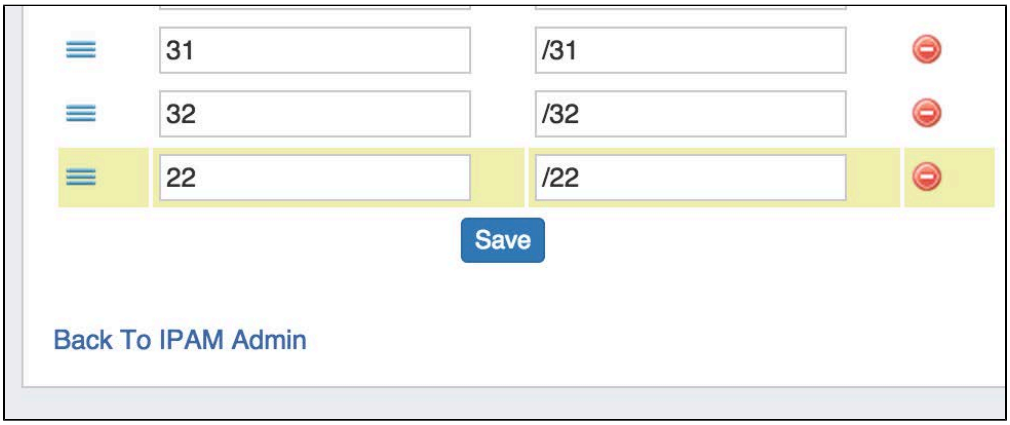
To add a new Subnet item, click on "Add Item" at the top of the Edit List: IPv4 or IPv6 Subnets page.



Then, type in the desired Value and Display value for the Subnet, and hit "Add Item".



After adding a new item, it will show at the bottom of the list highlighted in yellow. Hit "Save" to save your changes.



To **edit** a subnet, simply type your changes in to the text box. Entries with unsaved changes will be highlighted until saved.

To **reorder** the subnet list, click and hold on the "list" icon (three lines) to the left of the subet, then drag to the desired location in the list and release.

After adding or editing a subnet, you may wish to clean up the list order by hitting "Sort List Numerically" at the top of the page to reset the list order including the new entry.

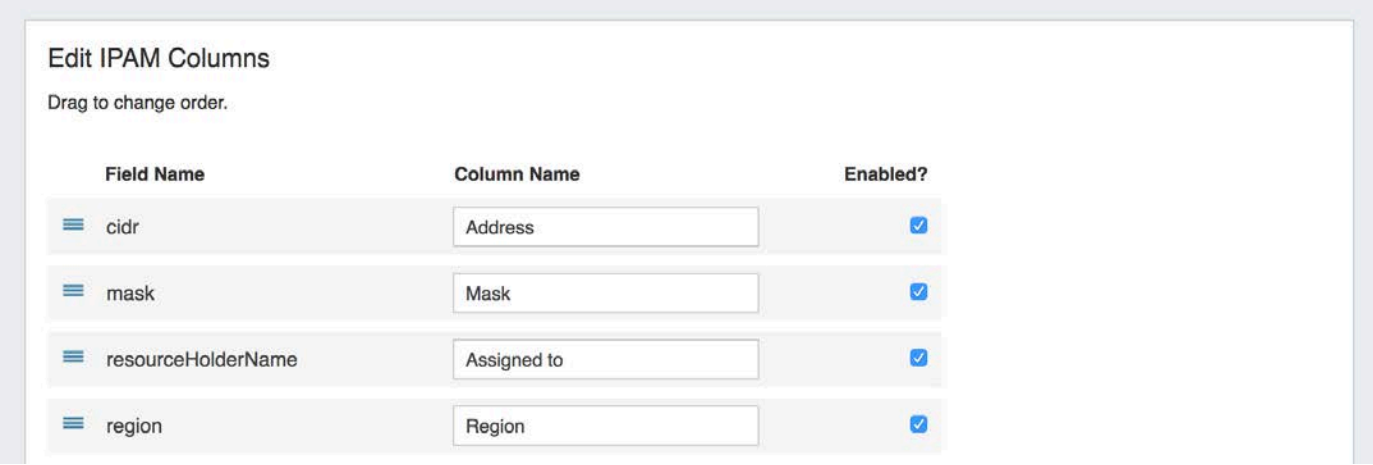
To **delete** a subnet, click on the red "delete" symbol to the right of the subnet entry.

When complete, be sure to click on the "Save" button to save your changes.

Edit IPAM Columns

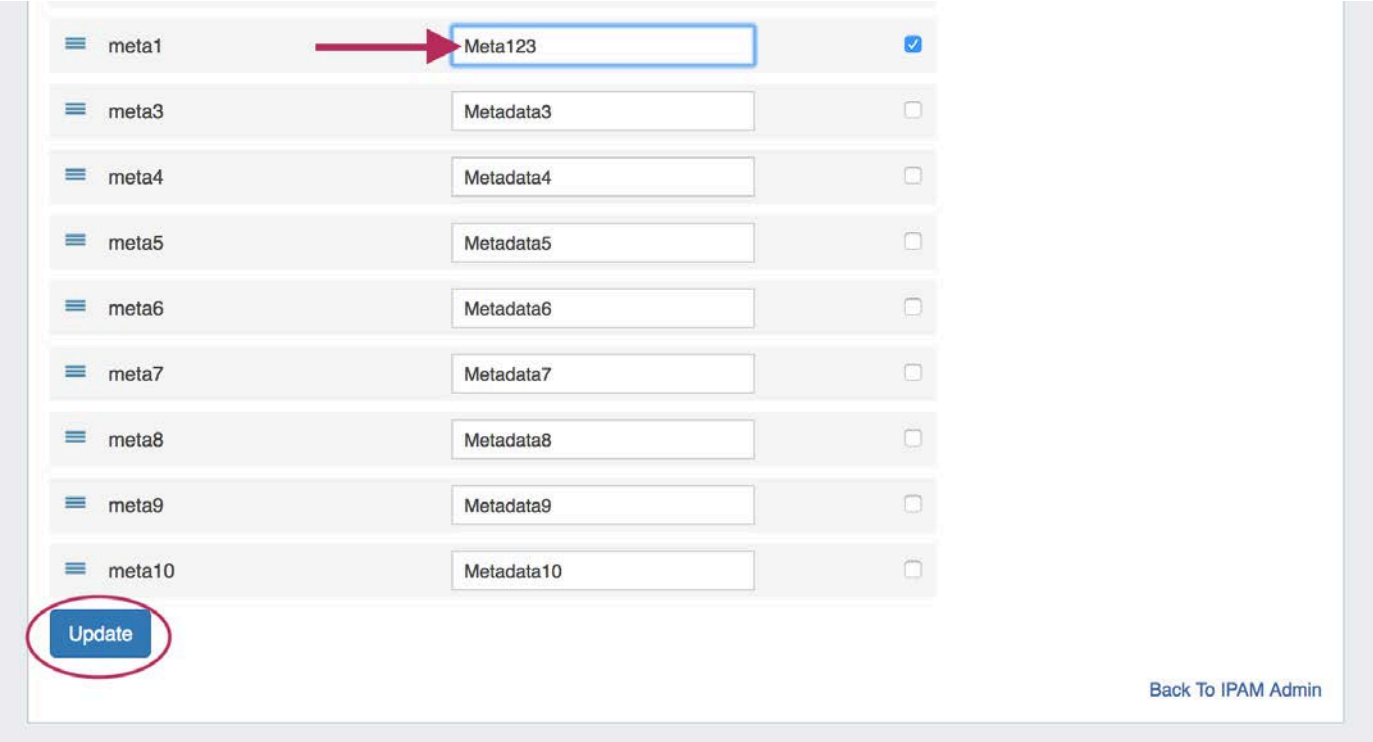
The Edit IPAM Columns page lets you change the order, column name, and visibility of IPAM columns globally for IPAM areas in ProVision.

Eleven default ProVision fields and ten user-defined "Metadata" fields are available for customization.



Keep in mind that "Edit Columns" is a global edit. If the values for order, header, or visibility are changed, it will affect ALL users of the ProVision application!

To **edit** a column name, simply type your changes in to the text box. and click the "Update" button.



To **reorder** the columns, click and hold on the "list" icon (three lines) to the left of the subet, then drag to the desired location in the list and release.



To **toggle visibility** for a column, click in the checkmark field to the right of the column name to enable / disable visibility.

This shows or hides the column in both IPAM Manage and the IPAM Gadget, and affects all users.

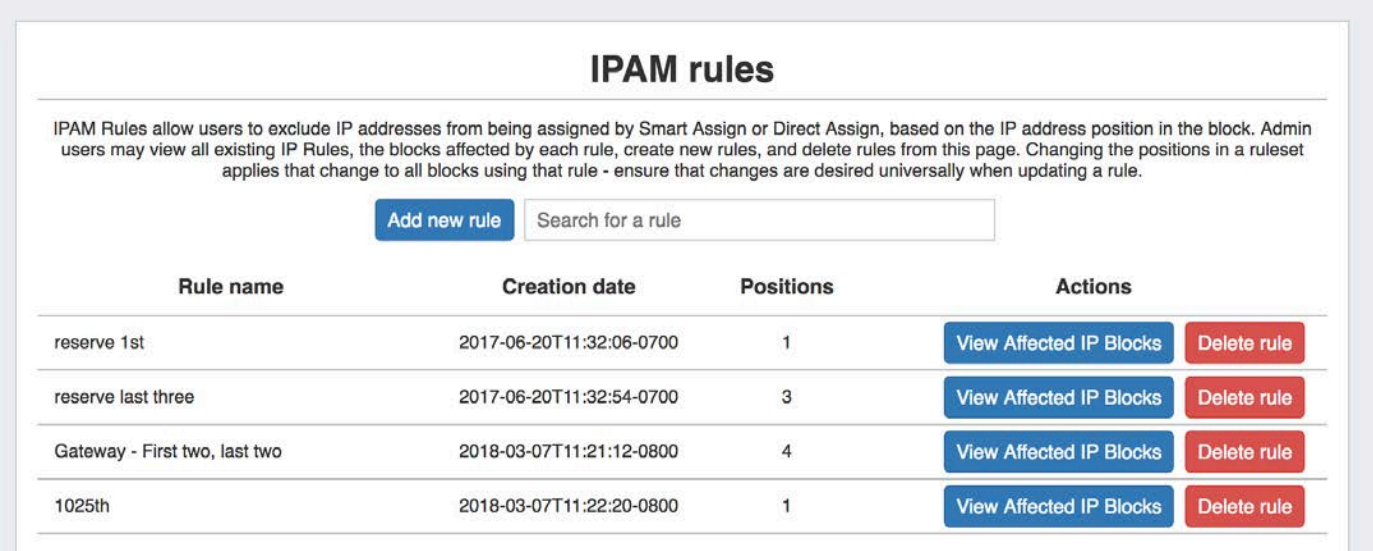


When complete, be sure to click on the "Update" button to save your changes.

Edit IPAM Rules

IPAM Rules allow for certain address positions within an IP Block to be reserved when utilizing Direct Assign and Smart Assign.

These rulesets may be managed in the "Edit IPAM Rules" section of IPAM Admin.



For additional details, see IPAM Rules.

Additional Information

Continue on to the following pages for additional information on IPAM Admin tasks in ProVision:

- IPAM Rules
- Holding Tank Management
- LIR Management and Use

For additional information on non-admin IPAM areas, see:

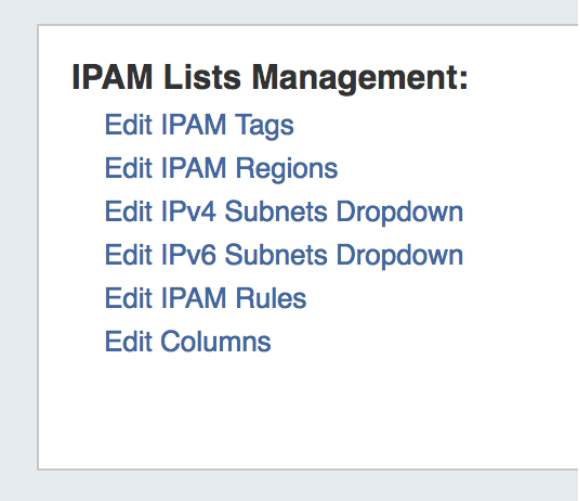
- IPAM Tab

IPAM Rules

IPAM Rules

IPAM Rules are accessed from the Admin section of ProVision, under the **IPAM Admin** tab by clicking on "Edit IPAM Rules".

In the IPAM Admin "Edit IPAM Rules" section, Admin users may view all existing IPAM Rules, the blocks affected by each rule, and delete rules.



- IPAM Rules
  - IPAM Rules Overview
  - IPAM Rules Behavior and Bounds:
  - Edit IPAM Rules Page
  - Add an IPAM Rule:
    - IP Rule Positions
  - Edit an IPAM Rule:
  - View Affected IP Blocks:
  - Delete an IP Rule:
  - Working with IP Rules:
  - Additional Information

IPAM Rules Overview

IPAM Rules allow users to exclude IP addresses from being assigned by Smart Assign or Direct Assign, based on the address position in the block (ie, first address, last address, n^{th} from last).

Reserved addresses are set by their position in the block, starting with position '1' to represent the first address, '2' the second, and so on (1,2,3). The last position in the block is represented as '0', with each step backwards set as the next lowest negative integer (0, -1, -2, 3). Rulesets may reserve one or many positions in a block, so you can choose to reserve only the first address (1), only the last address (0), or combinations as desired - reserving the first three and last three addresses on a block would look like (1,2,3,0,1,-2).

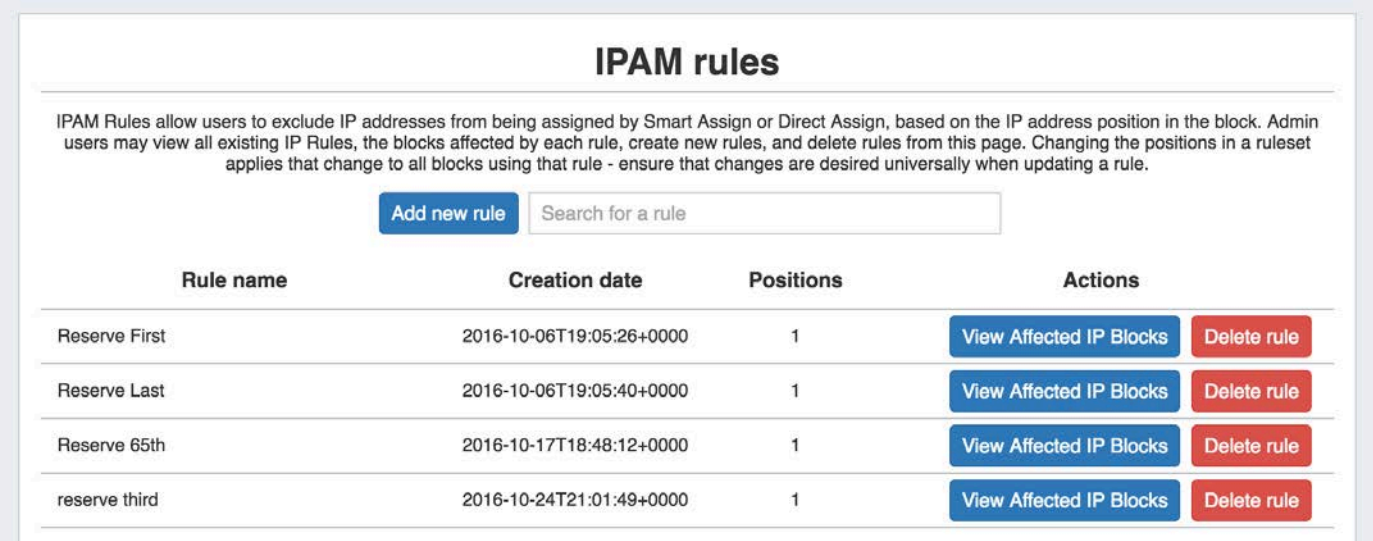
IPAM Rules may be created and applied to blocks under the IPAM Tab -> IPAM Manage screen, in the Action Menu for the desired block. Additional management tasks are available under the IPAM Admin -> "Edit IPAM Rules" page.

IPAM Rules Behavior and Bounds:

- Rules may be applied to IPv4 or IPv6 blocks
- Once a rule has been applied to a block, the addresses at the selected rule positions will by bypassed when Smart Assigning from that block, or return an error when Direct Assigning that position from the IPAM Gadget.
- Only one ruleset may be applied per block - so if a rule already exists for reserving the first address in a block, and a second rule exists for reserving the last address, and you would like to use both, a third, new ruleset must be created that combines reserving both first and last.
- Rules cascade down the IP tree. If a rule is applied to a /24, child blocks under that /24 (/30s, etc) will have the same ruleset applied.
- Changing the positions in a ruleset applies that change to all blocks using that rule - ensure that changes are desired universally when updating a rule.
- If a rule's position(s) is outside of the boundary of the selected block (for example, trying to reserve the 60th address of /30), an error message will notify the user and prompt for a different rule selection.

Edit IPAM Rules Page

In the IPAM Admin "Edit IPAM Rules" section, Admin users may view all existing IPAM Rules, the blocks affected by each rule, create new rules, and delete existing rules.



Add New Rule: Opens the "Manage Ruleset" interface where a new rule may be created. This module is also accessible from the "IP Rules" option on the IPAM Manage Action Menu.

Search: Filter the list of rules by searching for a rule name.

Rule Name: The user-created name of the rule. **Clicking** on the rule name will bring up the Manage Ruleset screen, showing an editable list of rule positions.

Creation Date: Date the rule was created.

Positions: How many positions are reserved by the rule.

View Applied Networks: Shows a list of blocks that have the rule applied.

Delete Rule: Deletes the selected rule.

Add an IPAM Rule:

Adding an IPAM Rule from the Admin area of ProVision functions the same way as if accessed from IPAM Manage.

Click on "Add new Rule", and the Manage Ruleset screen will appear, prompting you to enter a Ruleset name, and select positions to reserve.

IP Rule Positions

Reserved addresses are set by their position in the block. Positions start at '1', for the first address in a block, and step up by one for each subsequent IP address. The last position in a block is '0', representing the last address, and steps down a negative integer for each position from last - so the last three addresses in a block would be represented by (0, -1, -2). When reserving multiple positions, the typed order of the positions does not matter.

Position examples:

(1) - Reserves the first IP

(0) - Reserves the last IP

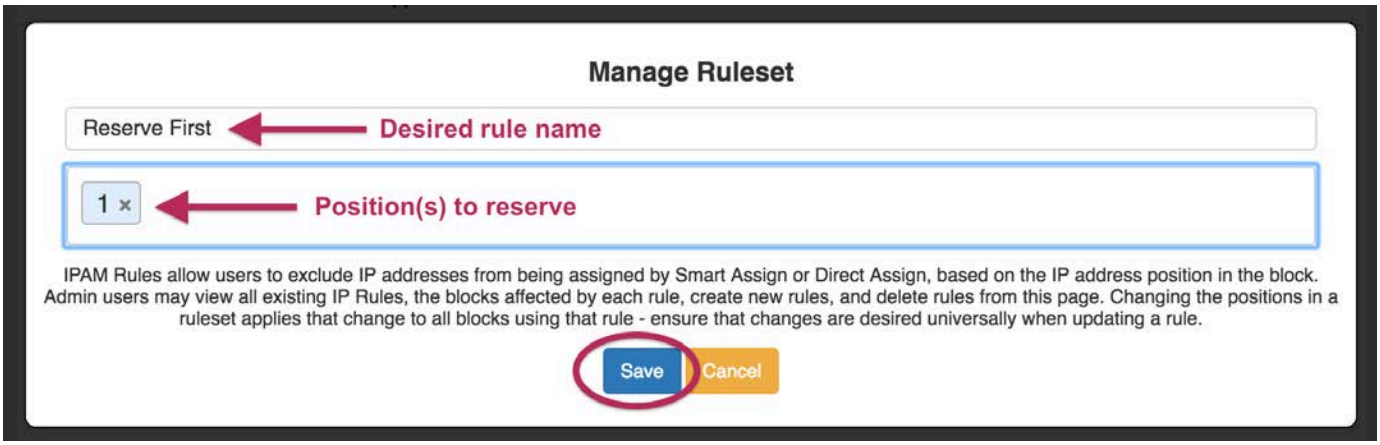
(1,2,3) - Reserves the first three IPs

(0, -1, -2) - Reserves the last three IPs.

(1,2,3,0,-1,-2) - Reserves the first three and last three IPs

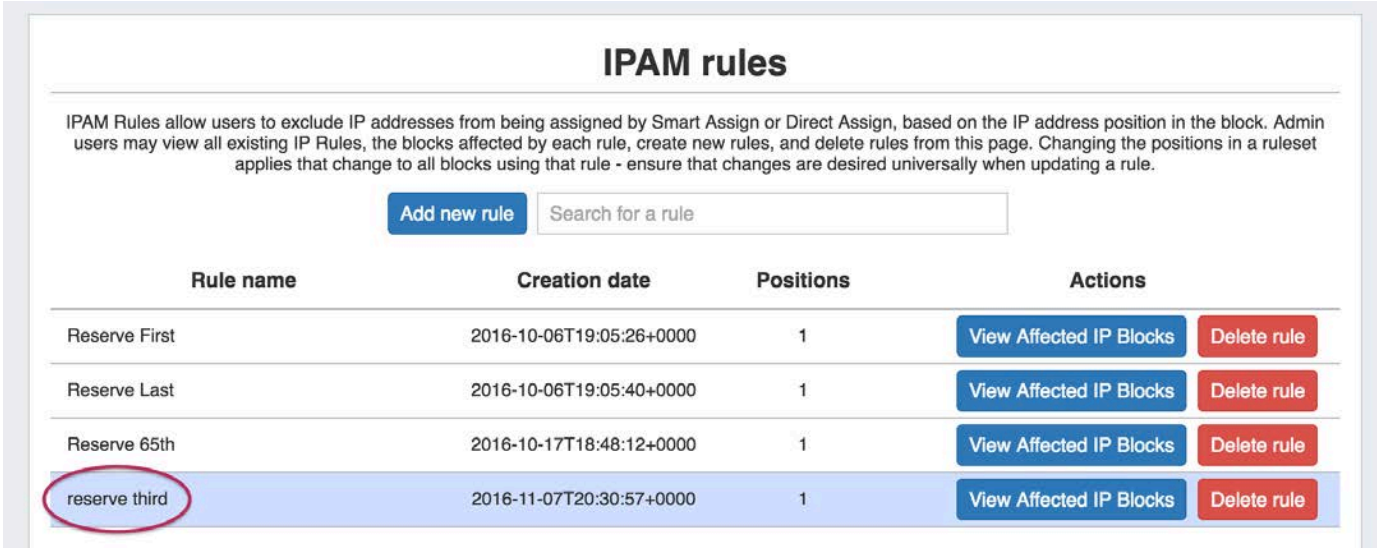
Type in the desired name for the new rule, and one or multiple positions (denoted by integers separated by commas) into the box below the name.

When complete, click the "Save" button, or hit "Cancel" to exit without saving. The Rule preview screen will appear.



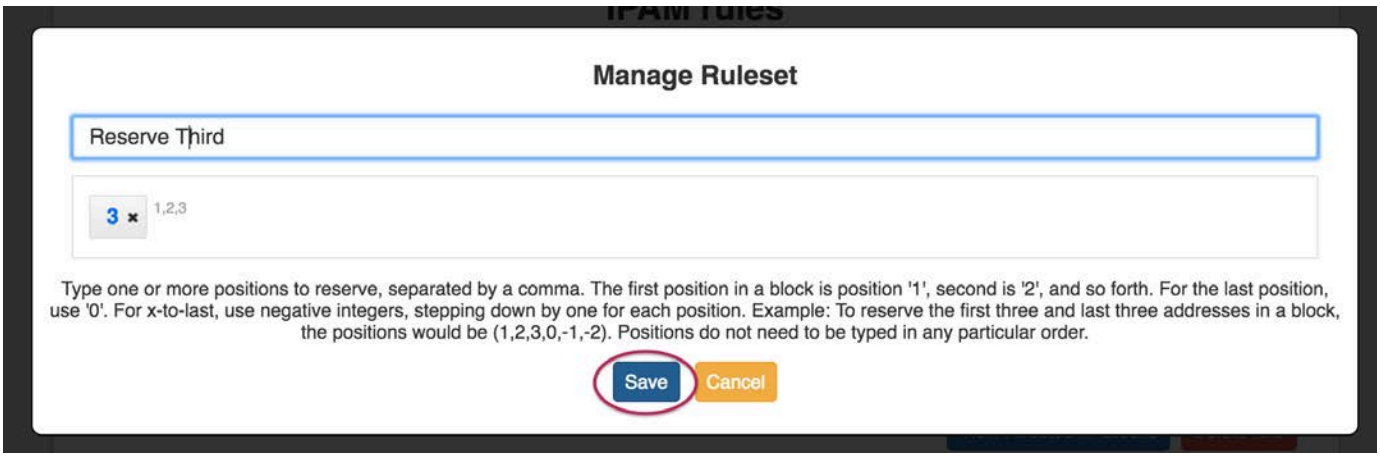
Edit an IPAM Rule:

From the Admin IPAM Rules page, click on the Rule entry in the list when highlighted.



This will bring up the "Manage Ruleset" module, showing the rule name and the currently reserved positions for the ruleset.

To change the rule name, simply type the desired changes into the Rule Name text box, and click "Save".



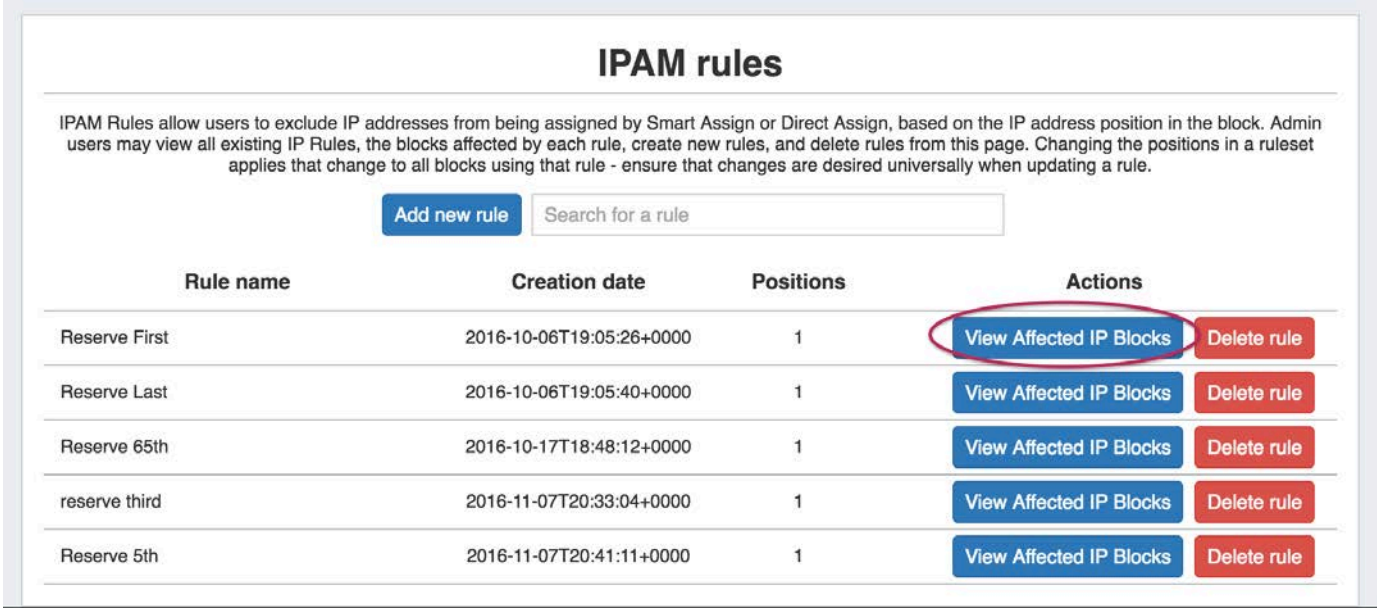
To edit the positions in the rule, either delete a position by clicking on the "x" for the position, or add a new position by typing the position number and then hit enter.

When you are done with your changes, click the "Save" button.

To discard changes, or exit the module, click "Cancel".

Note: When positions are changed, the change applies to all blocks using that rule! Ensure that the change is desired universally for all blocks using that rule prior to updating.

View Affected IP Blocks:



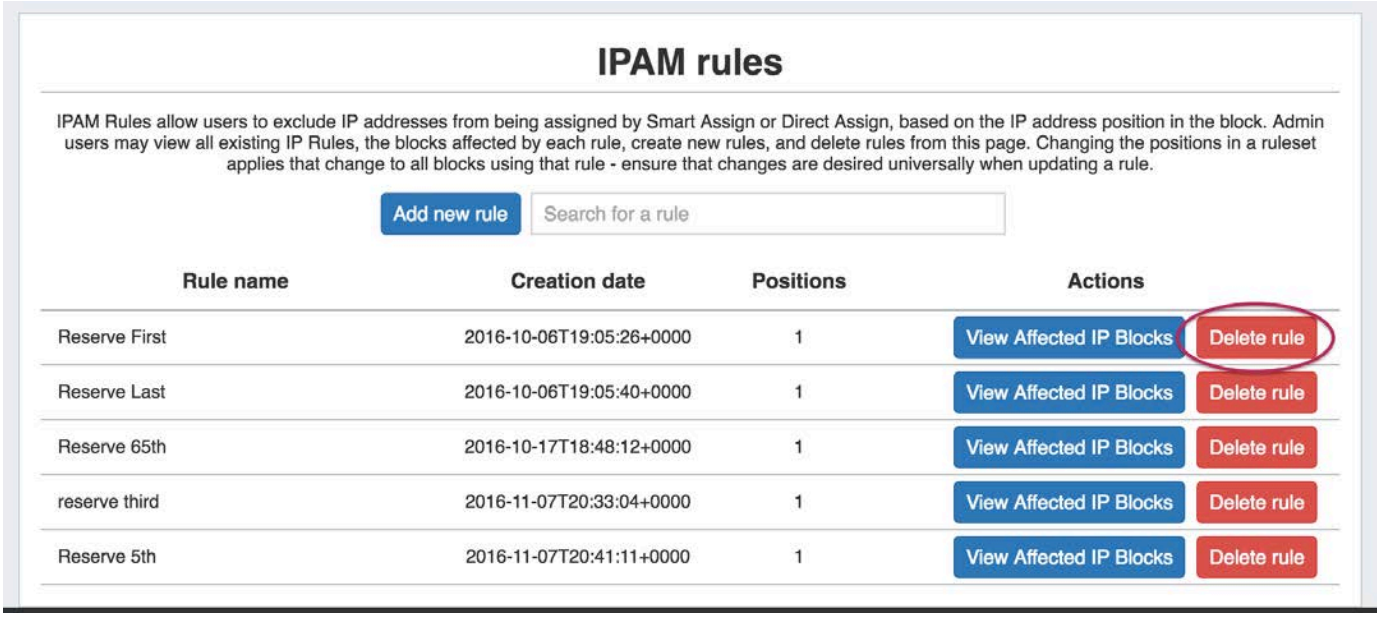
Clicking on "View Affected IP Blocks" in the IP Rules list for a rule will show a list of blocks that currently have the rule applied. It is recommended that you check the affected blocks list before editing or deleting an IP Rule to ensure that the changes are desired for the affected blocks.



Delete an IP Rule:

To delete an IP Rule, click on the "Delete Rule" button for the rule in the IPAM Rules list. This will permanently delete the rule and remove it from any blocks currently affected by the rule.

If you only wish to remove a rule from a block, but retain the rule for future use (or to retain use by other blocks), please see "Remove a rule from a block" detailed at Working with IP Rules.



Working with IP Rules:

For additional information on common IP Rules tasks that do not require Admin access, see Working with IP Rules in the ProVision User Guide.

Detailed walkthroughs of creating a new rule, applying an existing rule to a block, and removing a rule from a block are provided.

Additional Information

See the following pages for additional information on IPAM Admin tasks in ProVision:

- IPAM Parameters
- Holding Tank Management
- LIR Management and Use

For additional information on non-admin IPAM areas, see:

- IPAM Tab

Holding Tank Management

Holding Tank Management

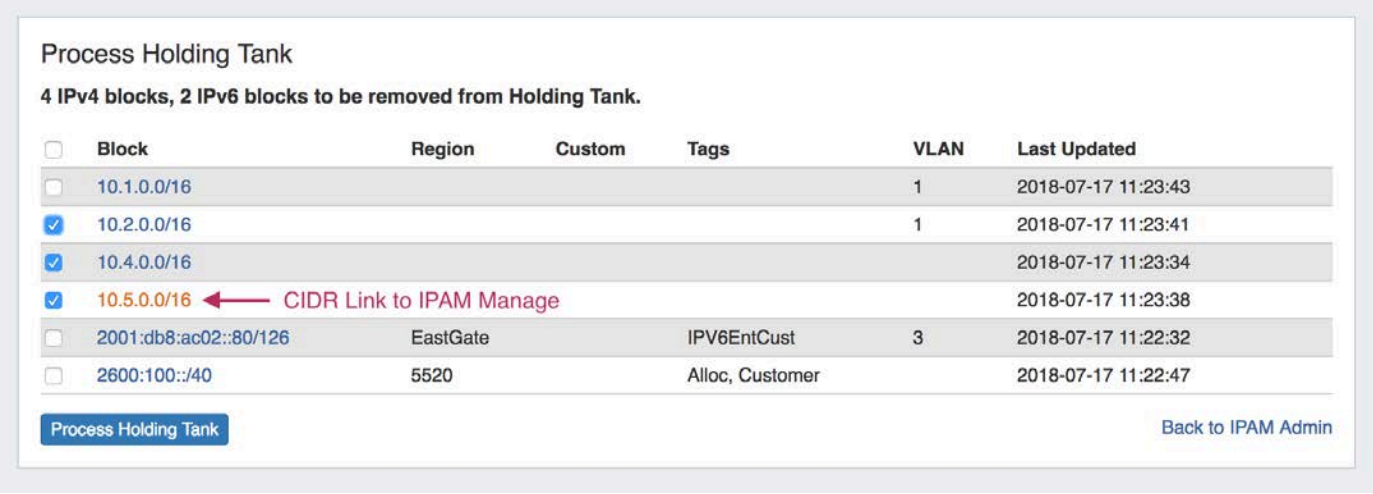
How it Works

The "**Process Holding Tank**" link will process all selected blocks with the "Holding" status older than the "Holding Tank Days" setting, and change them to the "Available" status.

The default time for blocks to be held prior to showing the in holding tank is 30 days, but this number is customizable from the IPAM Configuration page.

While in the Holding Tank, you may open the IPAM Manage page for a specific block by clicking on the block's CIDR - this allows you to review the block or manually manage it otherwise.

Regardless of the holding tank days setting, blocks may be manually reset to available at any time.



Select individual blocks to release from holding by clicking the checkbox for the desired block(s), then click "Process Holding Tank".

To release all blocks from the Holding Tank, click the "select all" checkbox at the top left header of the table, ensuring all blocks are selected, then click "Process Holding Tank".

When an administrator elects to process the Holding Tank, it will show the information above.

Pro-Tip!

If you need to do a bulk "empty" of the holding tank, set the "Holding Tank Days" setting to "0" days. This will allow you to process all blocks in holding immediately.

Holding Tank Permissions

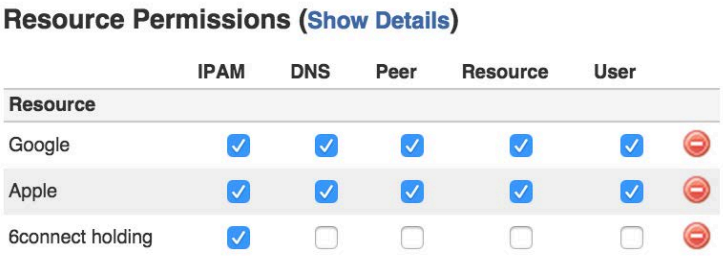
For blocks with subassignments, the Holding Tank can utilize ProVision's permissions structure in order to override holding. This allows the subbassigned block to be set as assigned to the parent resource.

Using this feature requires:

1) A block (block 1) assigned to a Resource (Resource A) that allows subassignments.

2) A block (block 2) subassigned to another Resource (Resource B) from the parent (block 1).

3) A User assigned to a Group with permissions to both Resource A and Resource B, and IPAM permissions for the 6connect Holding Resource. See Users and Groups for additional information on setting up Users and Group permissions.



Steps:

1) While logged in as the user with the above permissions, go to the IPAM manage screen for the subassigned block (block 2).

2) Select "Unassign" from the Action Menu. The 'Assigned To' column will change from showing "Resource B" to "Holding"

3) For the same block, select "Override Holding" from the Action Menu. This is the step that setting the 6connect Holding resource permissions allows, that would otherwise be unaccessible.

4) You will then see the 'Assigned To' field change from "Holding" to "Resource A", as the assignment is reverted to match the parent level.

Override Holding through the permissions structure is intended only for blocks that are subassigned. It is not intended for blocks that are not subassigned. Setting holding permissions for non-subassigned blocks simply allows viewing of the block(s) in holding, and permissions do -not- extend to allowing overrides by design.

Additional Information

See the following pages for additional information on IPAM Admin tasks in ProVision:

- IPAM Parameters
- IPAM Rules
- LIR Management and Use

For additional information on non-admin IPAM areas, see:

- IPAM Tab

LIR Management and Use

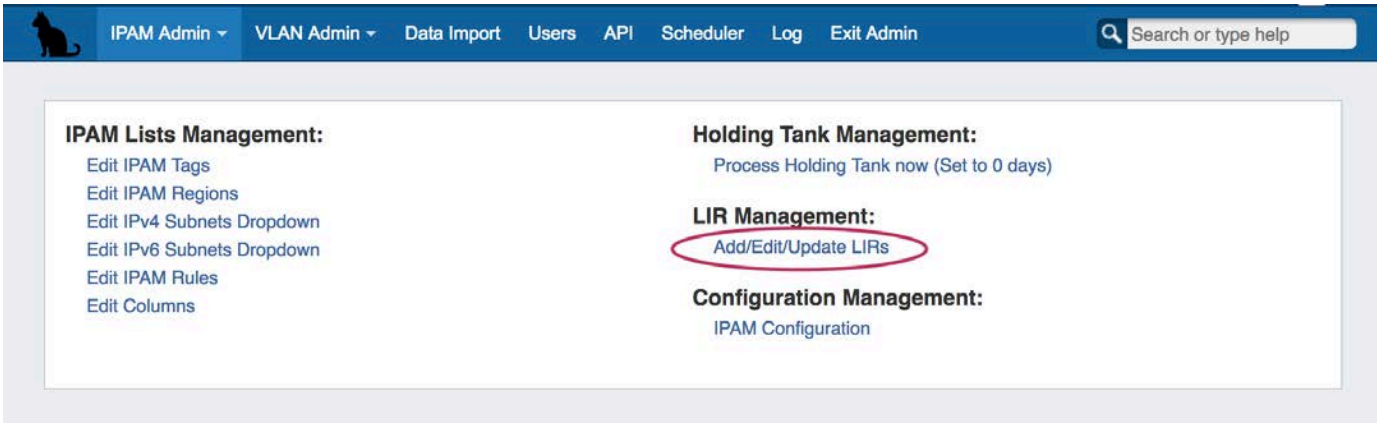
Overview

ProVision supports multiple LIRs (Local Internet Registries) in a single instance. This means that you have the ability to update SWIP/RPSL functions for a given allocation with the LIR information that you wish.

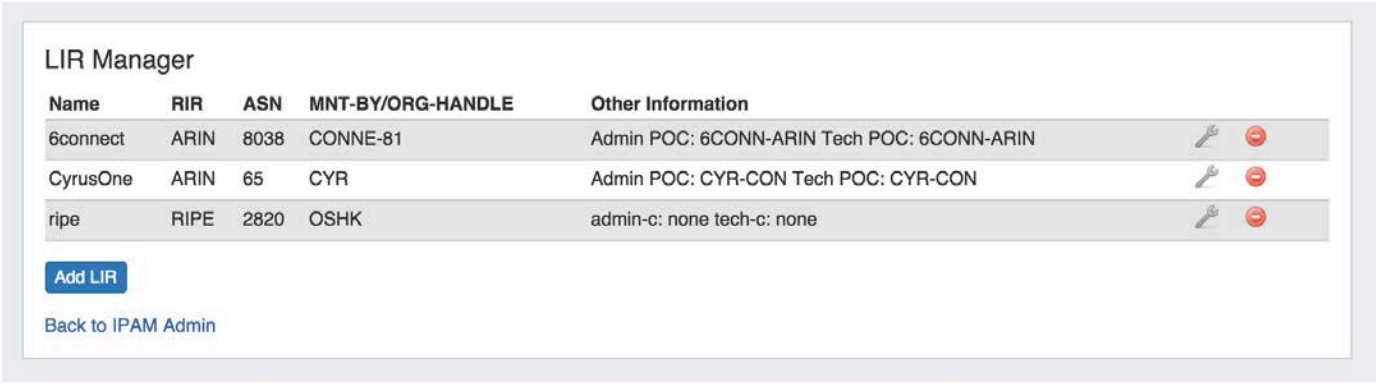
- Overview
  - LIR Setup and Use
  - Edit a LIR
  - Add a LIR
    - ARIN
    - RIPE
  - RIR Integration

LIR Setup and Use

There is an LIR Manager available from the IPAM Admin page. In the Admin section of ProVision, select the **IPAM Admin** tab, then "LIR Management: Add / Edit / Update LIRs".

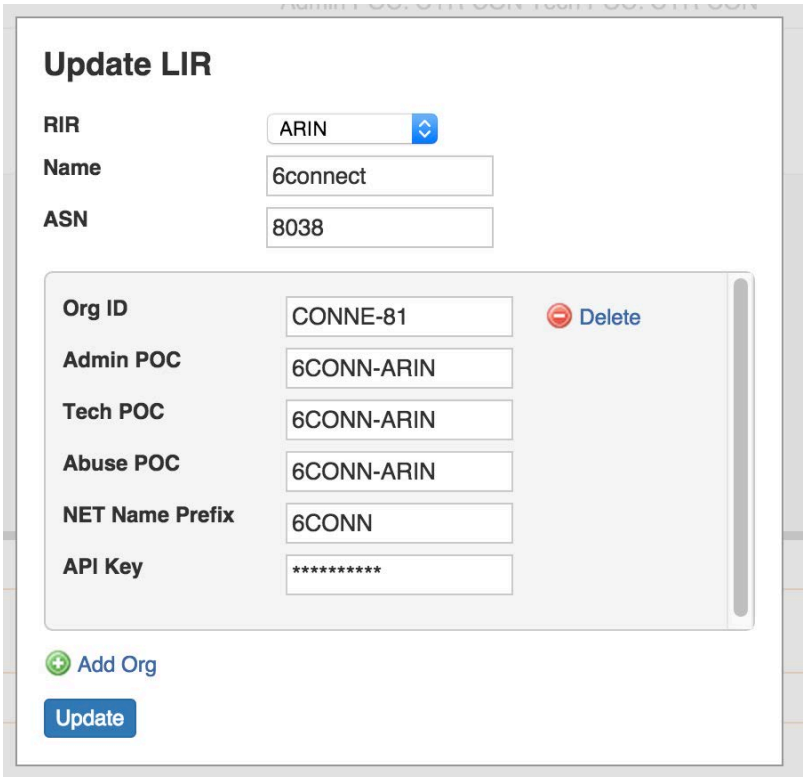


You will be taken to the LIR Manager page, showing your current LIRs. You may Add LIRs, edit the existing LIRS by hitting the Action Menu (wrench icon), or delete LIRs through the red delete icon.



Edit a LIR

Edit a LIR by clicking on the Action Menu (wrench icon) next to the entry in the LIR Manager. The Update LIR dialog will open. From here, you can edit RIR, Name, ASN, and Org information, as well as add / delete Orgs.

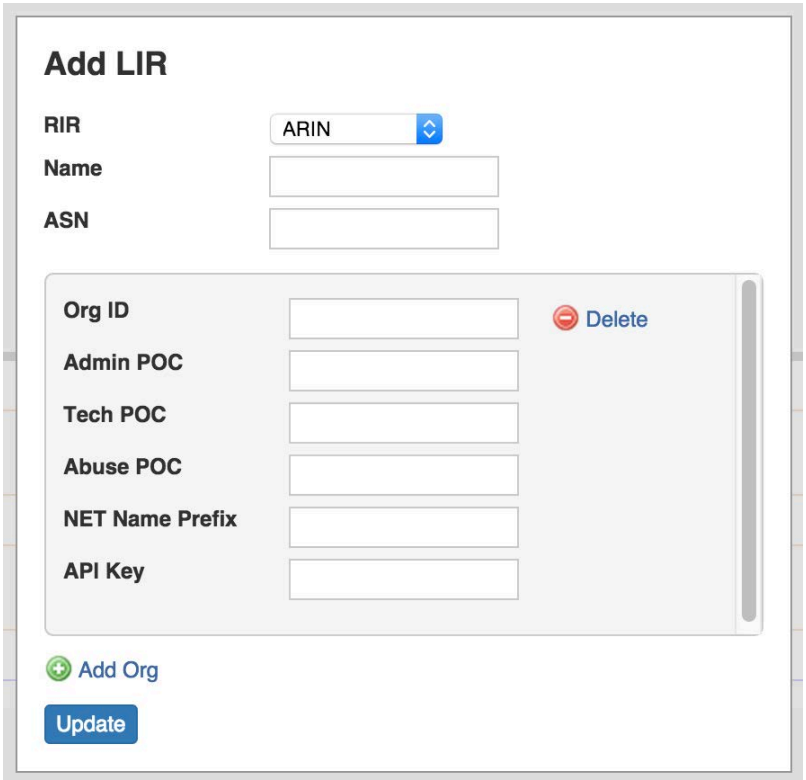


Add a LIR

After clicking on the **Add LIR** button, you can setup the required data for the specific RIR/LIR combination:

ARIN

Adding a new LIR with ARIN selected for RIR will bring up the following fields.



Enter the Name, ASN, ORG ID, POC information, NET Name Prefix, and API Key. Additional Orgs may be added by selecting the "Add Org" button.

Be sure to click the "Update" button when done to save your changes.

Press UPDATE to SAVE!

Make sure to press the Update button or else the LIR data will not save.

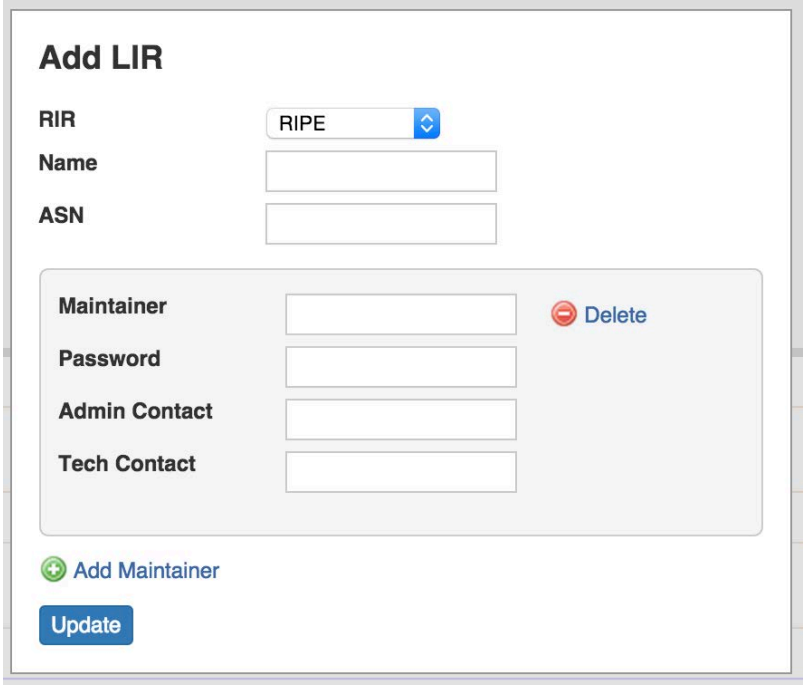
RIPE

Adding a new LIR with RIPE selected for RIR will bring up the following fields.

RIPE Integration

You must be using an MD5 password to update RIPE objects from ProVision! Please refer to RIPE's authentication page for additional details:

https://www.ripe.net/manage-ips-and-asns/db/support/security/protecting-data#solving-authentication-problems



Enter the Name, ASN, Maintainer, Password, and Contact information. Additional Maintainers may be added by selecting the "Add Maintainer" button.

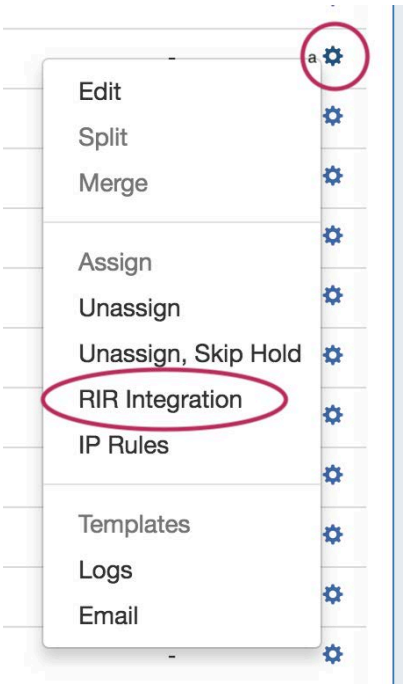
Be sure to click the "Update" button when done to save your changes.

Press UPDATE to SAVE!

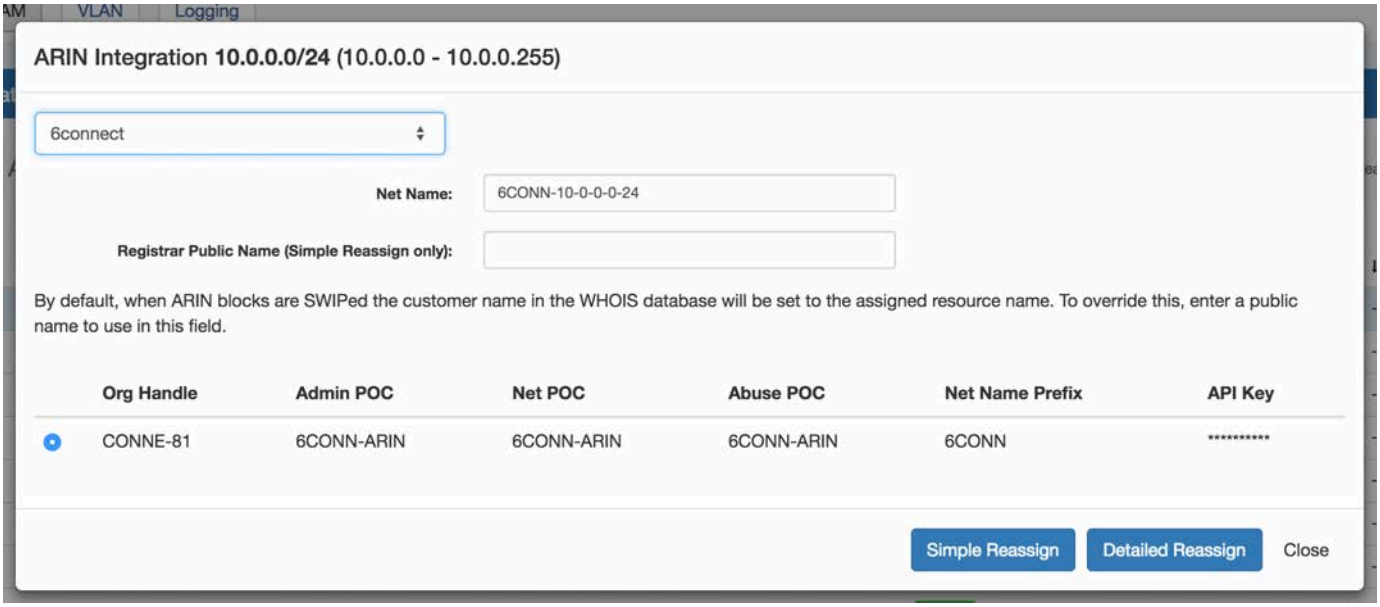
Make sure to press the Update button or else the LIR data will not save.

RIR Integration

Once these have been configured, you will be able to use the **RIR integration** feature from the Action Menu on the IPAM Manage screen or IPAM Gadget:



RIR specific options will pop up (see ARIN example below) and give the options for "Simple Reassign", "Detailed Reassign", or "Close" to exit.



For additional detail, continue on to:

- ARIN LIR Setup and Use
- RIPE LIR Setup and Use

ARIN LIR Setup and Use

LIR Setup

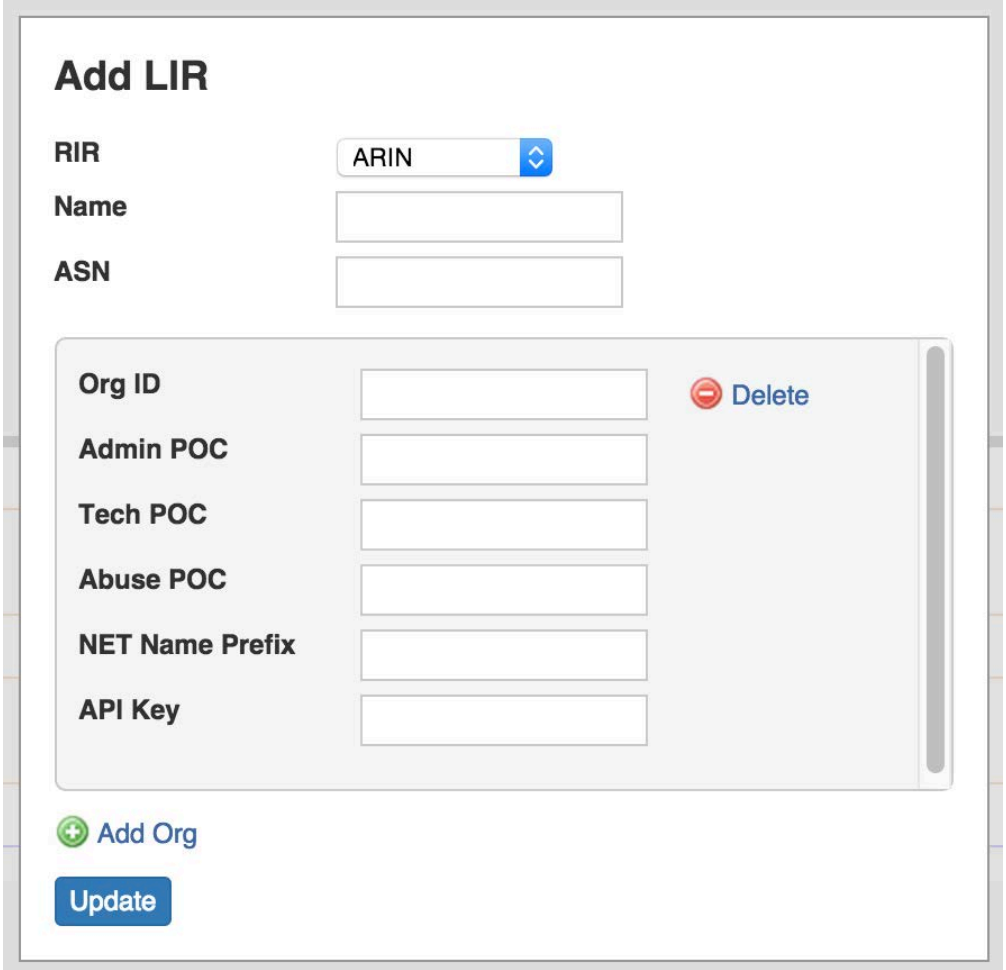
- LIR Setup
  - Step 1: Setup the LIR information via the LIR Manager
  - Step 2: Assign an IP block to a Resource using the IPAM Gadget or the Assign function from the IPAM Manage screen.
  - Step 3: Update SWIP information
    - Simple Re-assign
    - Detailed Re-assign

Step 1: Setup the LIR information via the LIR Manager

You will be prompted to the select the RIR



Add in the requisite Org and POC information

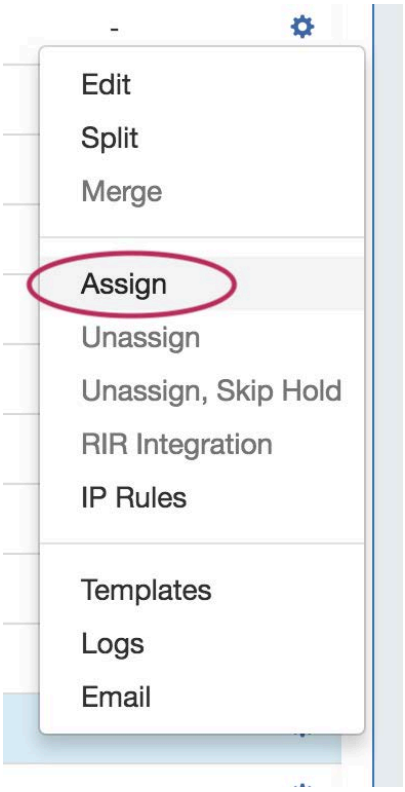


Multiple Org Support

Note that we support multiple Org Handles per ARIN entry. Simply click on the Add Org link at the bottom of the Add LIR dialog box.

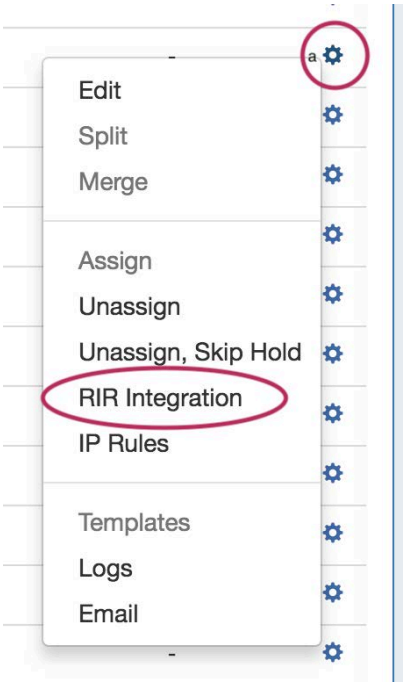
Step 2: Assign an IP block to a Resource using the **IPAM Gadget** or the Assign function from the IPAM Manage screen.

Go into IPAM Manage or the IPAM Gadget, and assign a block. See Working with IP Blocks for additional detail.

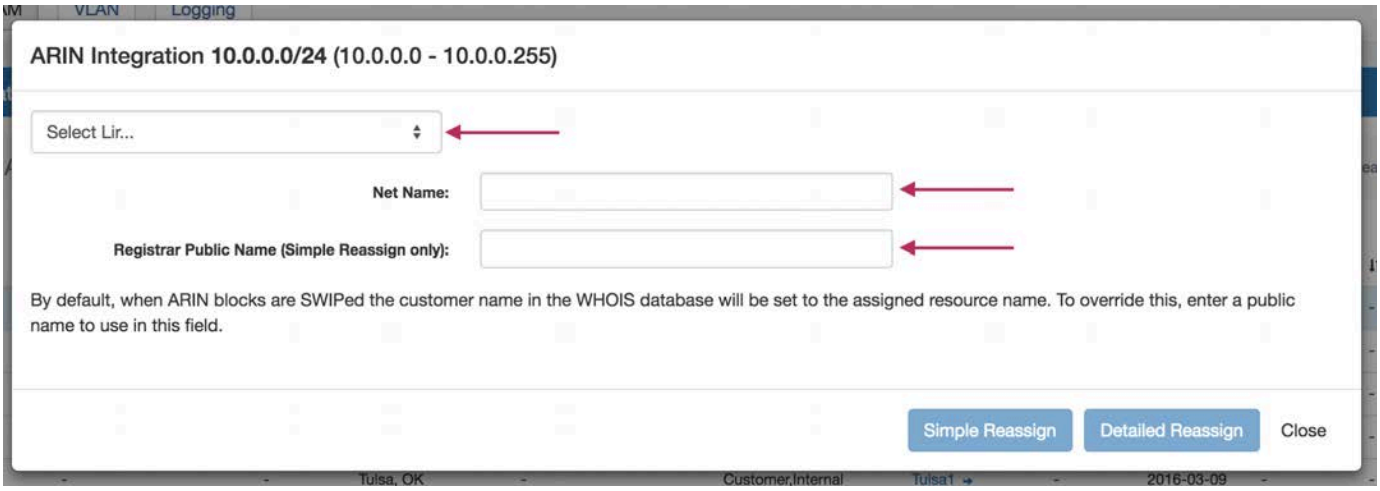


Step 3: Update SWIP information

Select "RIR Integration" from the Action Menu in IPAM Manage.

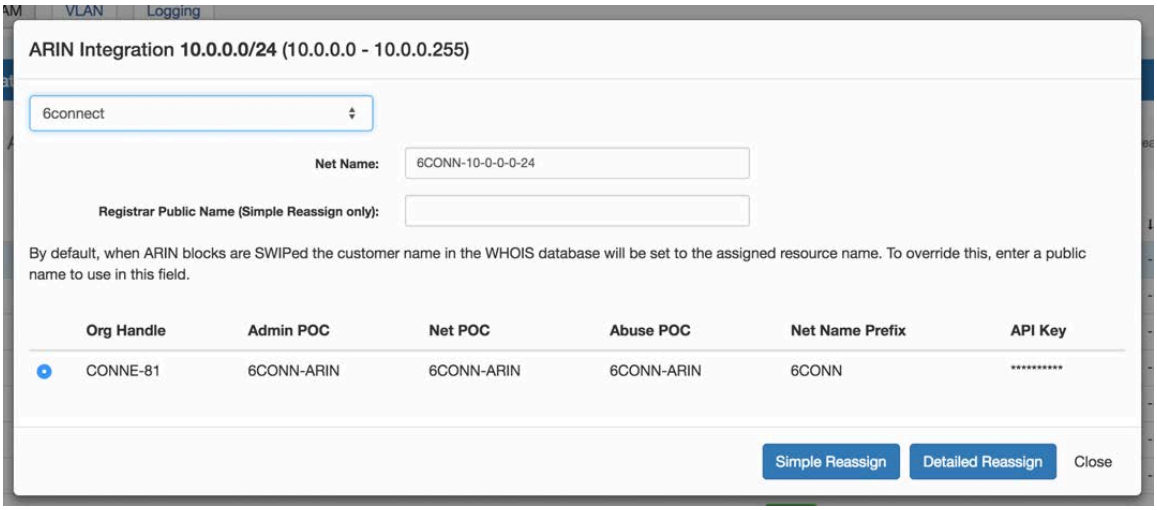


It will bring up an RIR Integration dialog, dependent on the selected RIR (ARIN example shown):



Select the ARIN LIR that has been created in ProVision, and add the Net Name / Registrar Public Name if desired.

Afterwards, you can select the Org Handle and chose "Simple Reassign", "Detailed Reassign", or "Close" to exit.



SWIP Update Functionality Details

In the case when a user already has SWIPped blocks to ARIN, 6connect checks prior to actually performing a SWIP. In the process, if the IP block is already SWIPped, it will check for existing ARIN customer data and update the 6connect data to reflect what ARIN has on file. Once that is complete, the user can then perform a de-SWIP function using ProVision.

Simple Re-assign

From ARIN.net:

Used to subdelegate IP addresses to a customer that does not need to:

- subdelegate the addresses to their own customers
- maintain their own in-addr.arpa delegation
- display their own point of contact (POC) information.

It can also be used to change the customer name and address information (but not the range) on an existing simple reassignment and to remove simple reassignments. It is submitted by an ARIN Online user account linked to the parent organization's Admin or Tech POC, or the Tech POC for the resource.

Detailed Re-assign

From ARIN.net:

Used to subdelegate IP addresses to a downstream organization that does not need to further subdelegate the IP addresses, but does need to maintain its own reverse name servers and/or display separate point of contact (POC) information. It is submitted by an ARIN Online user account linked to the parent organization's Admin or Tech POC, or the Tech POC for the resource.

RIPE LIR Setup and Use

LIR Setup - RIPE

- LIR Setup - RIPE
  - Before you begin:
  - Step 1: Setup the LIR information via the LIR Manager
  - Step 2: Assign an IP block to a Resource using the IPAM Gadget or the Assign function from the IPAM Manage screen.
  - Step 3: Update RPSL information

Before you begin:

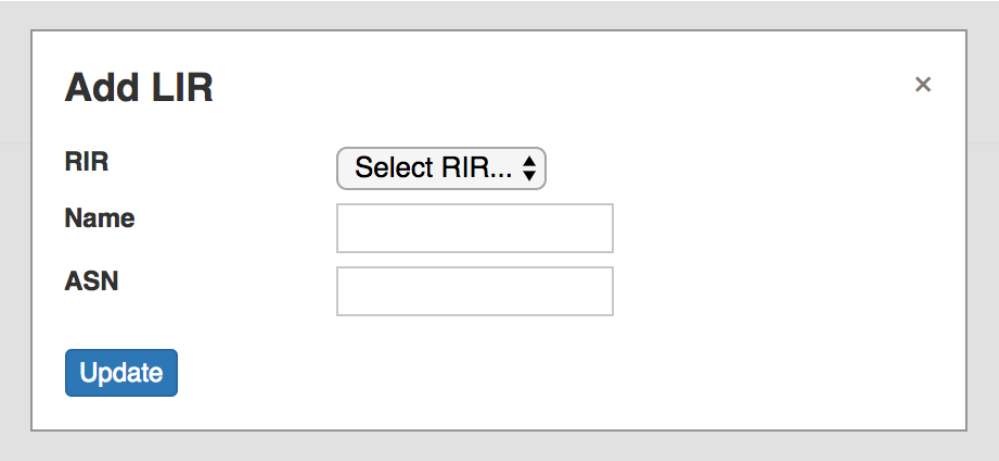
RIPE Integration

You must be using an MD5 password to update RIPE objects from ProVision! Please refer to RIPE's authentication page for additional details:

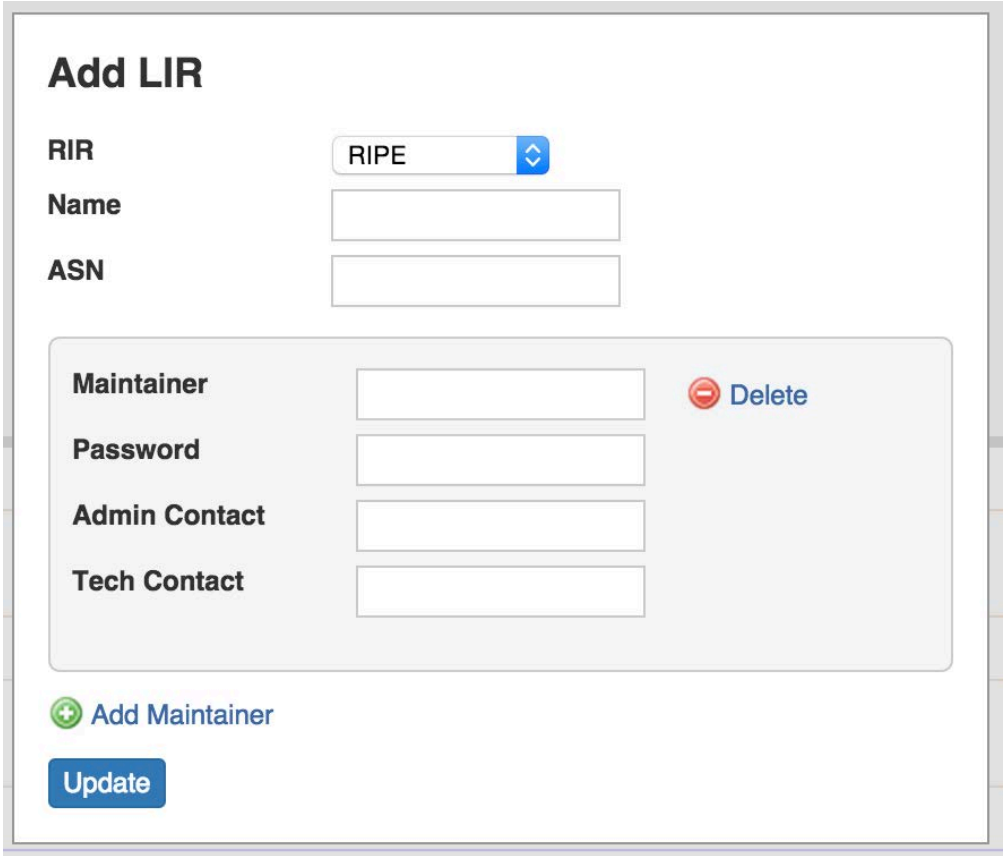
https://www.ripe.net/manage-ips-and-asns/db/support/security/protecting-data#solving-authentication-problems

Step 1: Setup the LIR information via the LIR Manager

You will be prompted to the select the RIR



Then add in the requisite Maintainer Object related information:



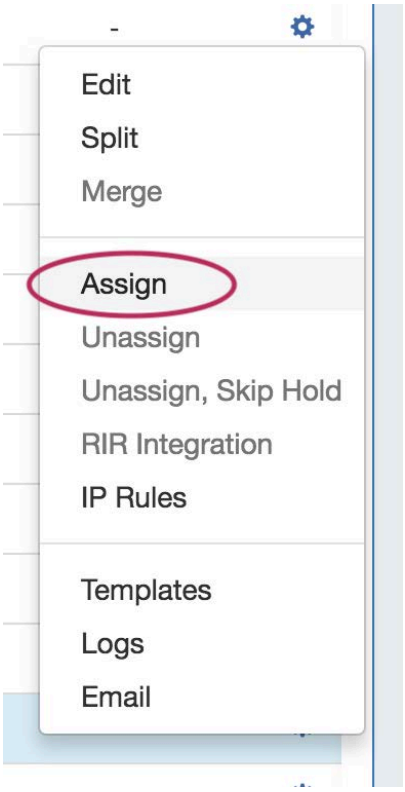
Be sure to hit "Update" when done to save your changes.

Multiple Maintainer Object Support

Note that we support multiple maintainer objects per LIR entry. Simply click on the Add Maintainer link at the bottom of the Add LIR dialog box.

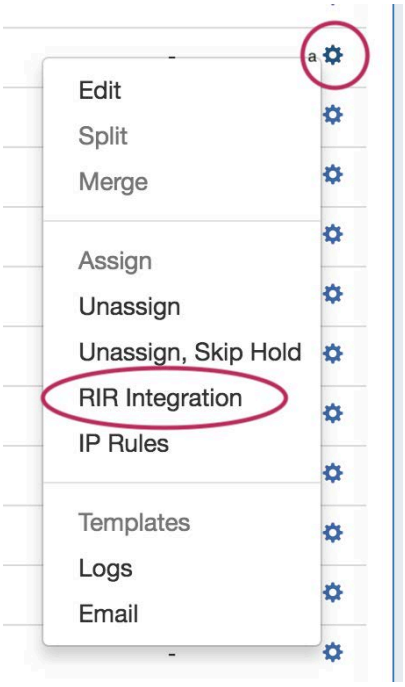
Step 2: Assign an IP block to a Resource using the **IPAM Gadget** or the Assign function from the IPAM Manage screen.

Go into IPAM Manage or the IPAM Gadget, and assign a block under the RIPE RIR. See Working with IP Blocks for additional detail.

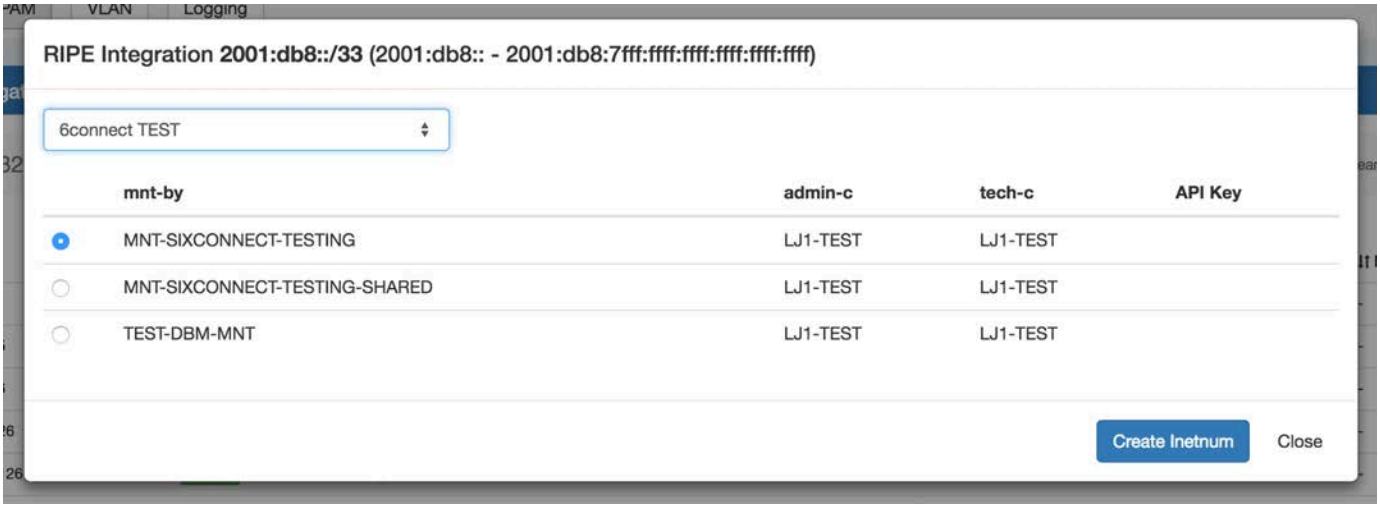


Step 3: Update RPSL information

Select "RIR Integration" from the Action Menu in IPAM Manage.

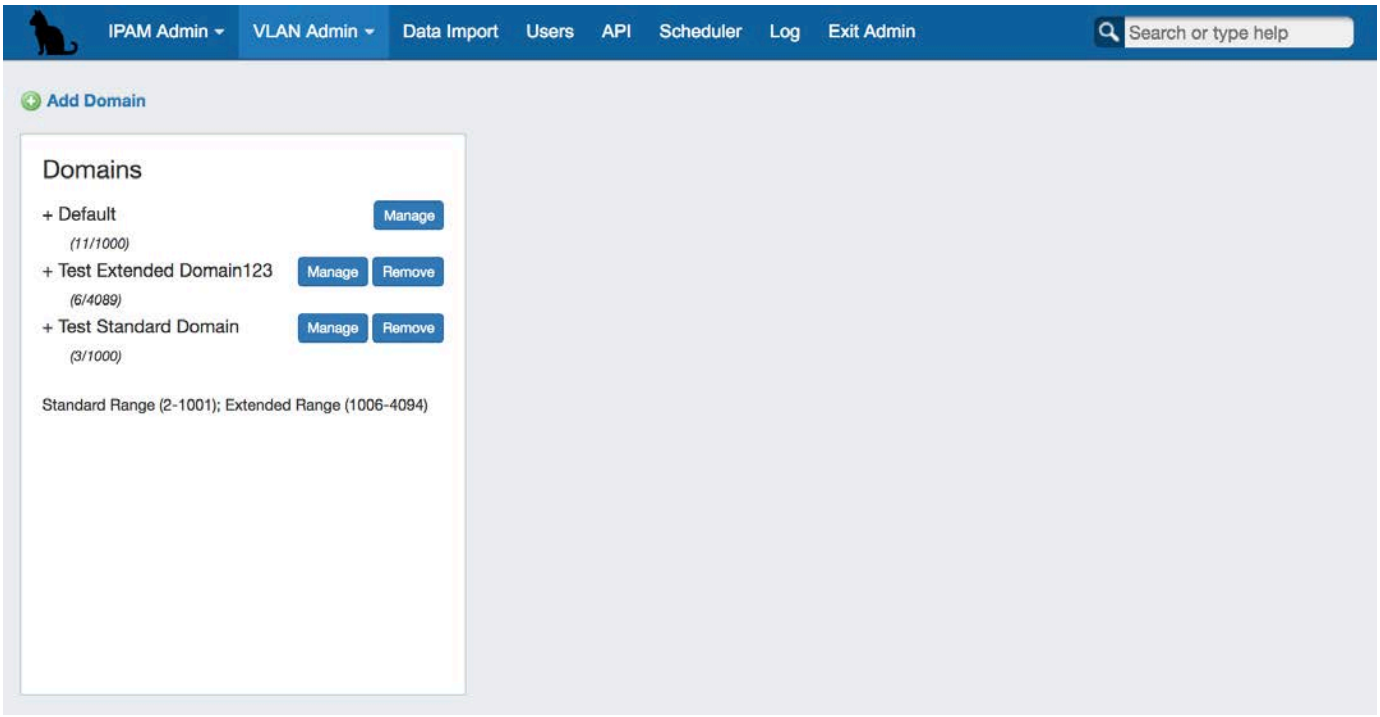


Identify which LIR data you want to use for the netnum update, and select either "Create Inetnum" or "Cancel" to exit.


RIPE Integration 2001:db8::/33 (2001:db8:: - 2001:db8:7fff:ffff:ffff:ffff:ffff:ffff)
6connect TEST
mnt-by
admin-c
tech-c
API Key
MNT-SIXCONNECT-TESTING
LJ1-TEST
LJ1-TEST
MNT-SIXCONNECT-TESTING-SHARED
LJ1-TEST
LJ1-TEST
TEST-DBM-MNT
LJ1-TEST
LJ1-TEST
Create Inetnum
Close

VLAN Administration

VLAN Manager Overview



The VLAN Manager allows Admin users to add domains and VLANs to their ProVision instances, and associate them with IP Blocks. Optionally, VLAN tags may be associated with ranges or individual VLANs under a domain, to help limit VLAN search results when enabling VLANs.

Although some VLAN functionality occurs under the "standard user" area of ProVision (IPAM Tab), only Admin-level users may access the IPAM -> VLAN page. Therefore, VLAN Manager processes are included under the ProVision Admin Guide.

VLAN Manager Workflow

Most of VLAN Manager workflow is split between the Admin area of ProVision (VLAN Admin Tab), and the IPAM Tab, under the VLAN submenu. There are two types of workflow. One is the "Standard" flow, where the VLAN tag system is not used - the only primary tasks are adding domains, VLANs, and associating the domain/VLAN pairs with IP Blocks. The second workflow is using optional VLAN tags, which requires creating the VLAN tags and associating them with domains / VLANs during the domain / VLAN add processes.

For detailed information on each of these workflows, click below:

Click here for detailed VLAN Manager Workflow information...

Standard flow (without VLAN tags):

The workflow starts with creating a domain in the VLAN Admin section of ProVision. During creation, domains may be selected as "standard" or "extended" domains, declaring the size of the VLAN pool from which VLANs are available to be enabled.

Next, VLANs must be enabled and added to the domain. This is done under the IPAM Tab -> VLAN section of ProVision. Add VLANs to the domains by clicking "Add", searching for the desired range of VLANs to enable from the standard / extended pool, and selecting the desired VLANs to enable.

Once enabled, VLANs may be edited or have IP blocks associated with that VLAN. Editing VLANs and Direct / Smart Browse / Search IPs functions for adding blocks to VLANs are available from the Domain/ VLAN list under under the IPAM Tab -> VLAN section of ProVision by expanding the desired domain and clicking on the VLAN link. IP blocks may also be edited individually through the IPAM gadget, IPAM Manage, and VLAN Manage areas to add domain and VLAN information to the block.

Optional flow (with VLAN tags):

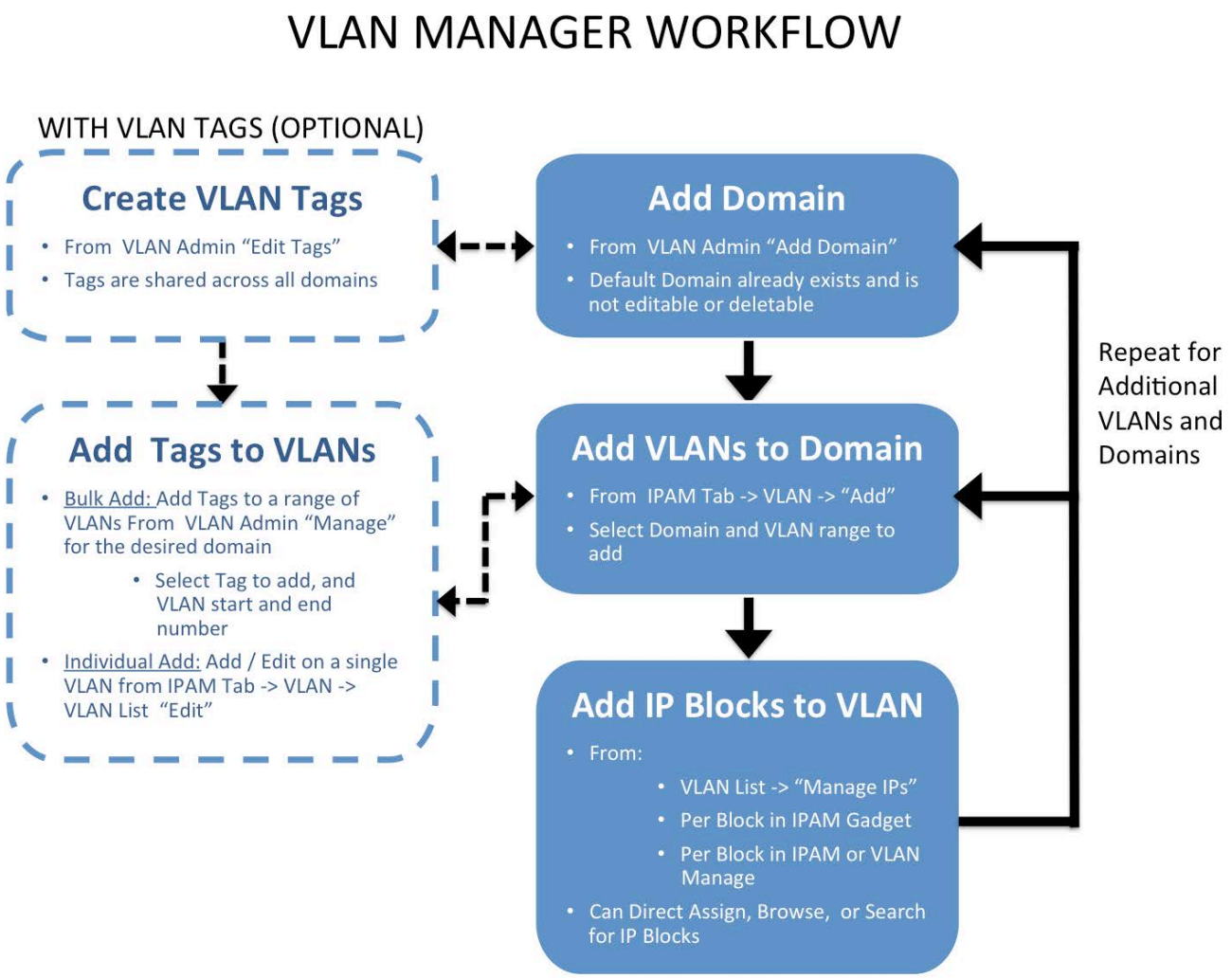
The workflow starts with creating a domain in the VLAN Admin section of ProVision. During creation, domains may be selected as "standard" or "extended" domains, declaring the size of the VLAN pool from which VLANs are available to be enabled. VLAN tags may be added (from the VLAN Admin Tab -> Edit Tags submenu) before or after domain creation. Tags created in this area are available to all domains and VLANs.

If the tags created need to be added to a large range of VLANs, the next step would be to add tags to the desired range(s) of VLANs from the VLAN Admin -> Domain "Manage" button. Domain Manage allows you to select an existing tag, type in the numeric VLAN start and end point, and assign that tag to that entire range of VLANs under the current domain.

Once a bulk range of tags is associated with VLANs, you can use the tag as a search criteria when adding VLANs to a domain. This is done under the IPAM Tab -> VLAN section of ProVision. Add VLANs to the domains by clicking "Add", using the tag name as a search field for the desired range of VLANs, and then selecting the desired VLANs to enable.

If tags are only desired for individual VLANs, they may be added when editing a VLAN after the VLAN has already been added/enabled for the domain. Editing VLANs may be accessed from the IPAM Tab -> VLAN section, then expanding the desired domain and clicking on the VLAN link, selecting "Edit".

Next, VLANs must be enabled and added to the domain. This is done under the IPAM Tab -> VLAN section of ProVision. Add VLANs to the domains by clicking "Add", searching for the desired range of VLANs to enable from the standard / extended pool, and selecting the desired VLANs to enable. Once enabled, VLANs may be edited or have IP blocks associated with that VLAN. Editing VLANs and Direct / Smart Browse / Search IPs functions for adding blocks to VLANs are available from the Domain/ VLAN list under under the IPAM Tab -> VLAN section of ProVision. IP blocks may also be edited individually through the IPAM gadget, IPAM Manage, and VLAN Manage areas to add domain and VLAN information to the block.



Additional Information:

For details on performing specific tasks with the VLAN Manager, proceed to Working with the VLAN Manager.

Working with the VLAN Manager

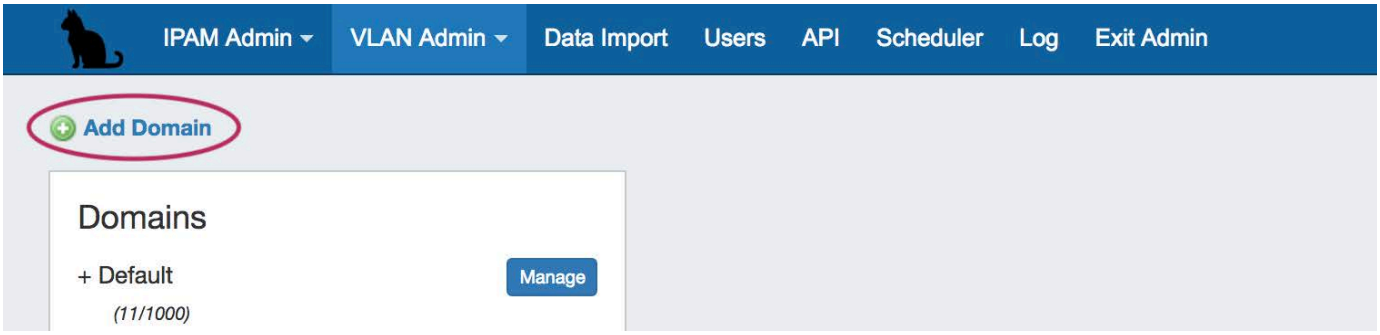
Working with the VLAN Manager (Standard Flow)

- Working with the VLAN Manager (Standard Flow)
  - Step 1) Add Domain
    - Viewing the Domain List
    - Edit a Domain
    - Remove a Domain
  - Step 2) Add VLANs to a Domain (Enable)
    - The IPAM VLAN Page
    - Add VLANs to a Domain (Enable)
    - Remove VLANs from a Domain (Disable) - API
    - Edit a VLAN
  - Step 3) Add Blocks to VLANs
    - Add / Update Blocks to VLAN from VLAN - Manage IPs:
      - Direct Add Block to VLAN:
      - Browse IP Blocks to add to VLAN:
      - Search IP Blocks to add to VLAN:
    - Add / Update VLAN to Blocks from IPAM Manage or VLAN Manage:
    - Add / Update VLAN to Blocks from IPAM Gadget:
  - Working with VLAN Tags - (Optional)
    - Creating VLAN Tags
    - Add VLAN Tags to VLANs
      - Add Tag to an Enabled VLAN
      - Add Tag to a Range of the VLAN Pool
  - Working with VLAN Metadata - (Optional)
    - Enable / Disable VLAn Metadata Fields
    - View / Edit Metadata Field Information

Step 1) Add Domain

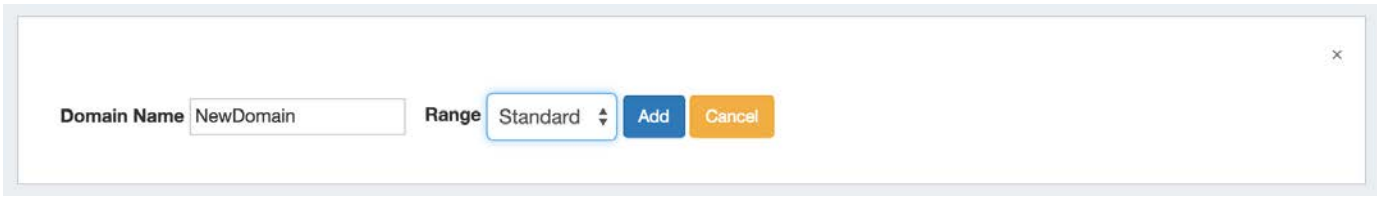
Users will already have a non-editable Default Domain created, which may house VLANs entered under the previous VLAN text field, however, you may choose to create new domains.

To create a new VLAN domain, navigate to the **VLAN Admin** Tab, in the Admin area of ProVision, and click on "Add Domain".



Enter the domain name, and select whether it is standard or extended - Standard has a range of VLAN IDs from 1-1005; Extended includes up to 4094.

NOTE: The ProVision UI and API has restricted usage of VLANs 1 and 1002-1005 in tagging and selecting or searching VLANs.



Click "Add", and your new domain will be added to the Domain list below.

Viewing the Domain List



If a domain already has VLANs enabled for it, clicking on the "+" sign in front of the domain name will expand the domain list to show enabled VLANs.

The numbers in parentheses under the domain name (shown as 21/1000 in the above image for a standard domain; an extended domain would show as 21/4089) indicate the quantity of enabled VLANs over the available pool of VLANs to select from. There are 5 VLANs (1, 1002-1005) which are reserved, and not included in the count.

Edit a Domain

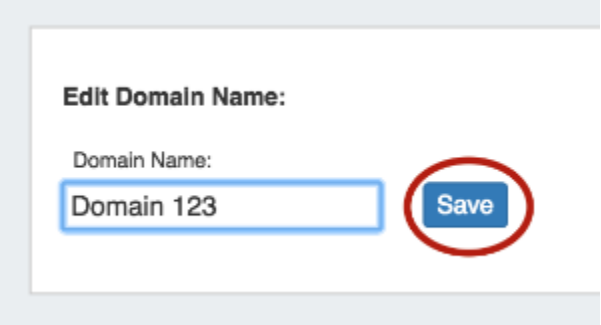
The Default Domain is not editable, however users may edit the name of user-created domains.

Once a domain has been created, users may not edit the range (standard / extended) of a domain - if a domain is created with a mistaken range, it must be deleted and re-created.

To edit a domain name, click on the "Manage" button next to the domain name.



Under the "Edit Domain Name" section of the page, edit the name as desired, then hit "Save".



Remove a Domain

To edit a domain, click on the "Manage" button next to the domain name.



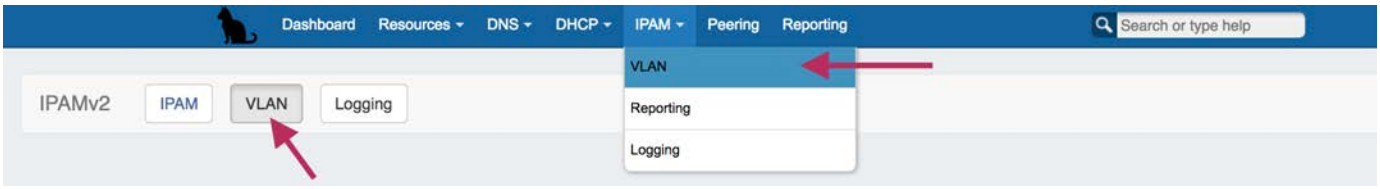
Then, edit the name as desired and hit "Save".

Step 2) Add VLANs to a Domain (Enable)

After creating domains, VLANs need to be added (enabled) for that domain out of the pool of available VLANs.

The IPAM VLAN Page

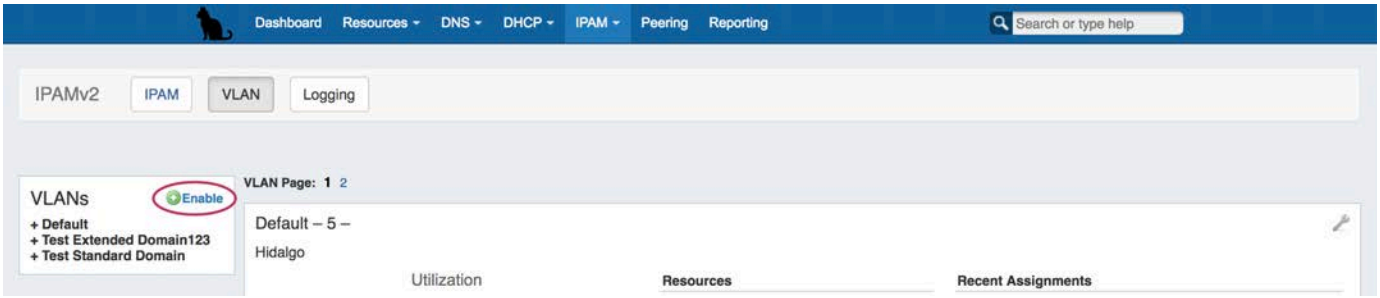
Go to the IPAM Tab, and then either select VLAN from the IPAM dropdown menu, or click "VLAN" from the subtab buttons that appear on the IPAM page.



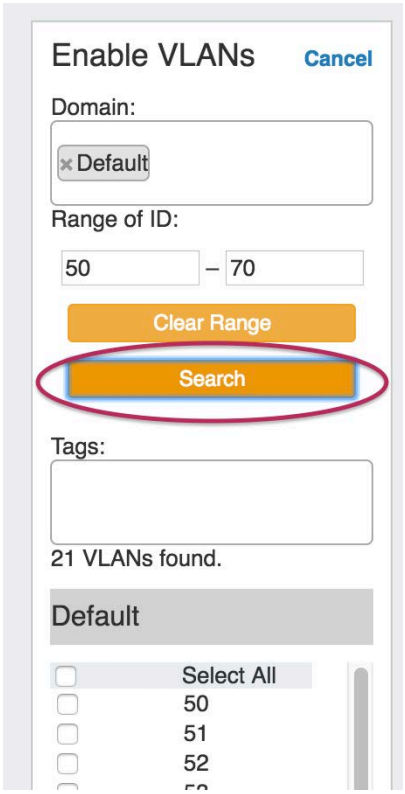
This takes you to the IPAM - VLAN page, which shows an overview of Domains and VLANs on the left sidebar (click the + next to the domain names to see the enabled VLANs under each domain), and similar to the IPAM page, shows IPAM Utilization data for each Domain / VLAN pair. Clicking on the wrench will open "VLAN Manage", which has the same type of functionality for IP blocks as IPAM Manage, but organized under the scope of that VLAN.

Add VLANs to a Domain (Enable)

In the IPAM - VLAN page VLAN sidebar, click "Enable".



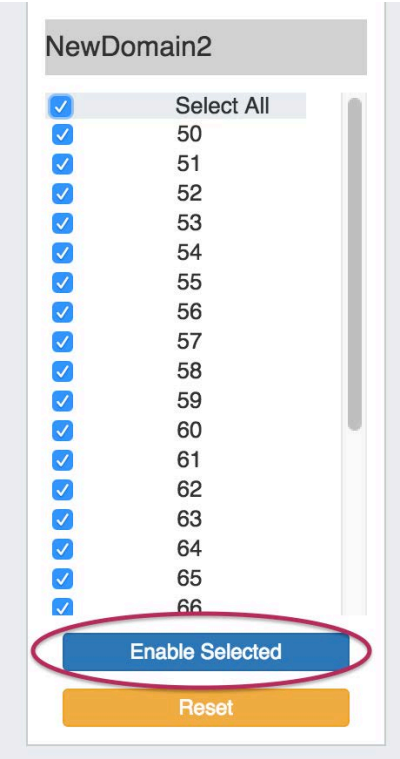
This changes the sidebar into a VLAN search module. Select the desired domain, range of VLANs to view, and click "Search".



Note: If VLAN Tags have already been applied to a range of VLANs, you may also include VLAN tag criteria in your search. If tags have not been created or applied to a range of VLANs, searching by tags will not return results.

Below the search criteria, the results list will show VLANs meeting your criteria.

Select the desired VLANs to enable for the domain - you may either select all VLANs in your results list, or check VLANs individually.



When done selecting the VLANs, click "Enable Selected". The selected VLANs will now be enabled for the domain, show in Domain / VLAN lists, and be available to edit or add blocks to. To add more VLANs to another domain, click "Reset" and repeat the process. If done, hit the "Cancel" button at the top of the Add VLAN module, and you will return to the main IPAM - VLAN page view.

Remove VLANs from a Domain (Disable) - API

Disabling a VLAN (removing the VLAN from a domain) may be done through the VLAN API "update" endpoint.

A VLAN may be enabled or disabled by setting its "enabled" parameter to true (enable) or false (disable), after providing the required id.

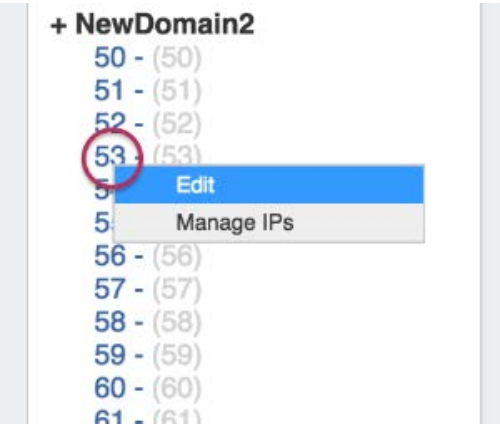
An example API call to disable a VLAN with id=41438 would look like:

```
(instance url)/api/v1/api.php?target=vlan&action=update&id=41438&enabled=false
```

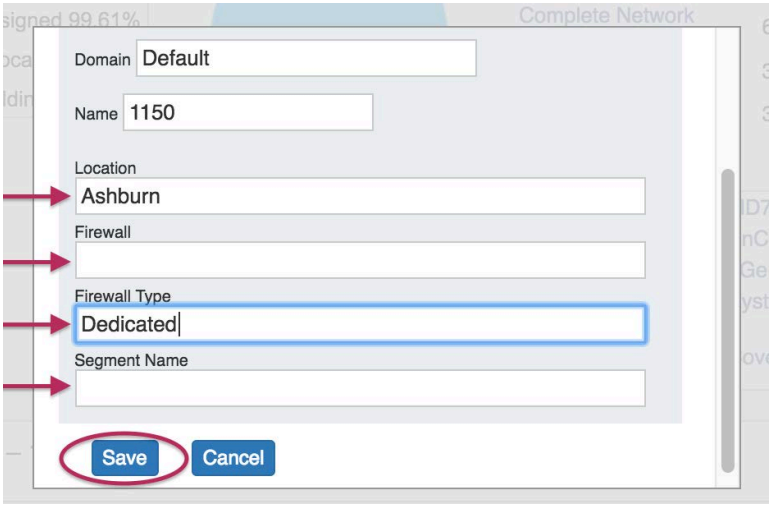
For details on the VLAN update endpoint, see API Module - VLAN - (update).

Edit a VLAN

You can edit a VLAN by clicking on the VLAN in the VLAN List sidebar, then selecting "Edit".



You cannot change the Domain of the VLAN, but you can add a descriptive text name, apply a VLAN tag (if using the tag system), or edit VLAN Metadata (if enabled).



After making your changes, click the "Save" button. The named VLAN will now show both VLAN number and name when viewing Domain / VLAN lists or when working with blocks.

Step 3) Add Blocks to VLANs

After VLANs have been enabled for domains, you can associate those VLANs with IP blocks.

Add / Update Blocks to VLAN from VLAN - Manage IPs:

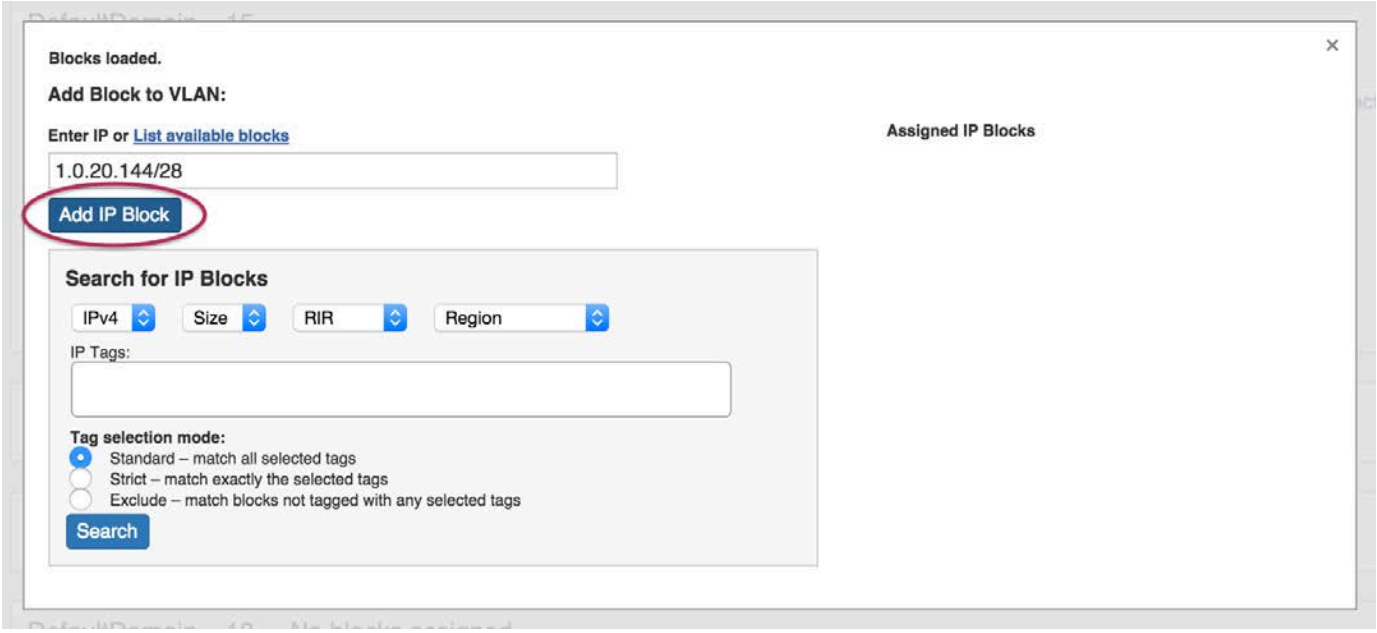
While in the IPAM Tab -> VLAN Page, expand the desired domain and click on a VLAN. Then select "Manage IPs".



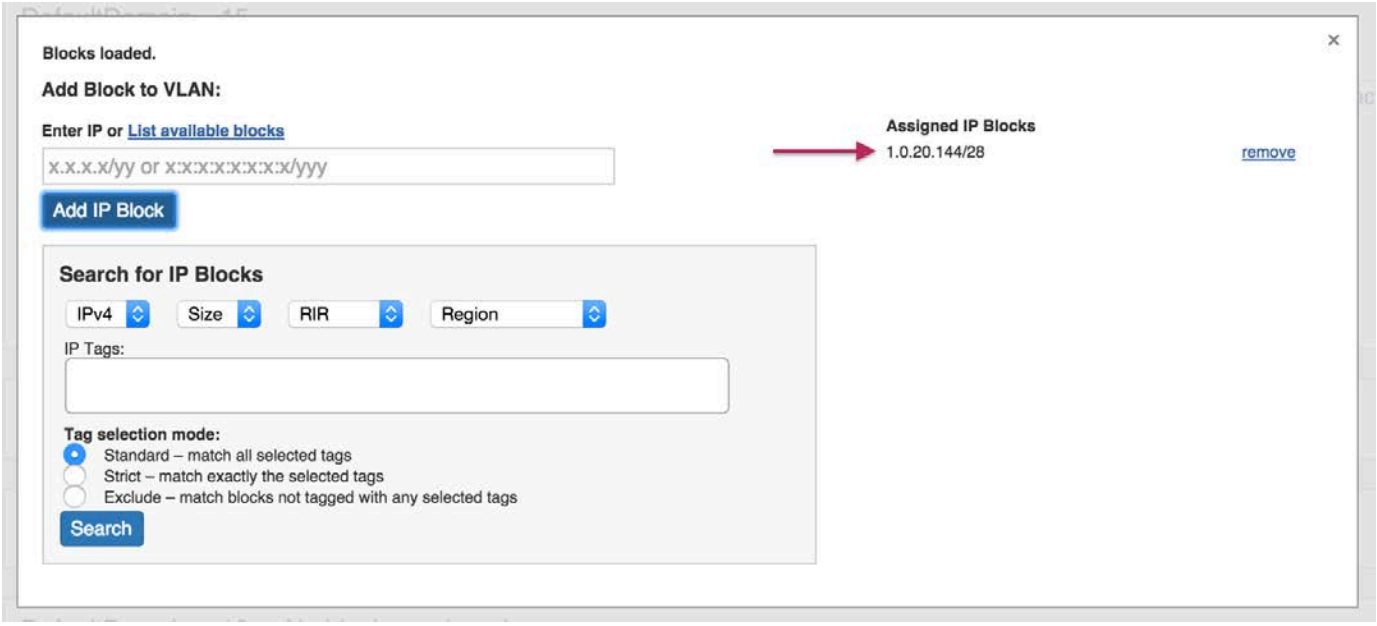
This opens up a window where IP Blocks may be associated to the VLAN. From here, you can "Direct Assign" a block to the VLAN, "Browse Assign" blocks, or Search for a block to add to the VLAN.

Direct Add Block to VLAN:

Directly add a known, available IP Block by typing in the CIDR, then hitting "Add IP Block".

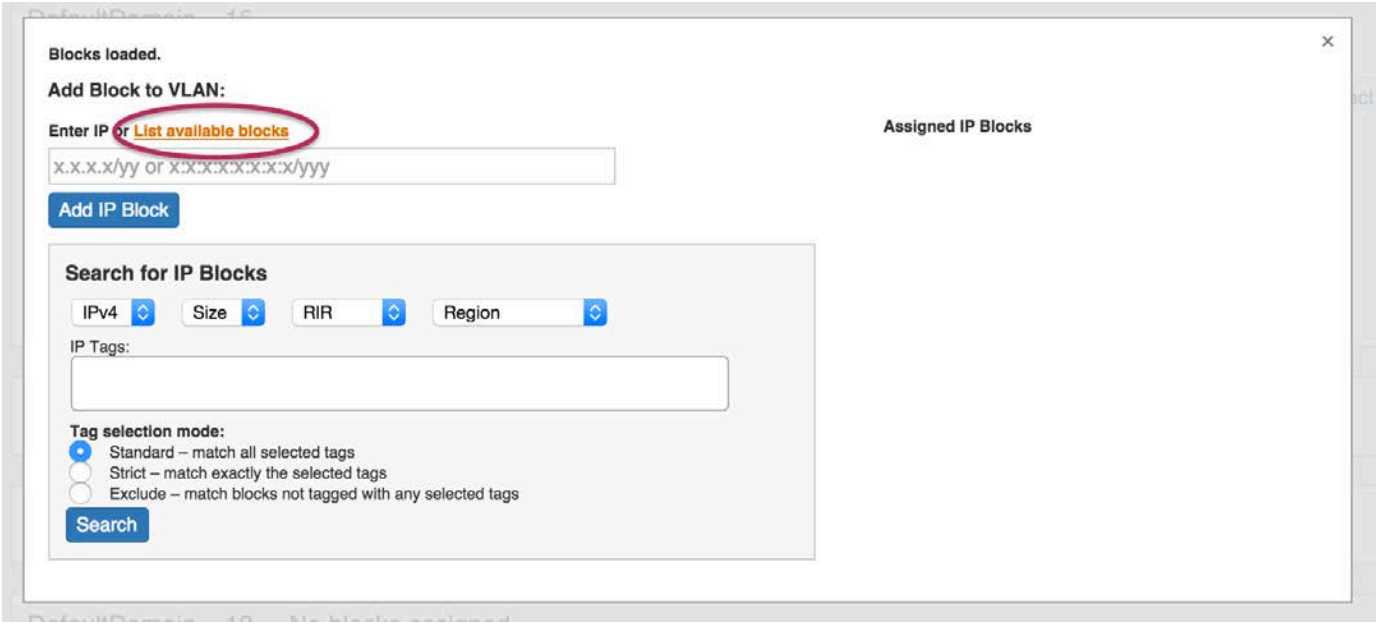


When complete, the block will be added to the "Assigned IP Blocks" list on the right of the window, with the option to remove if desired.



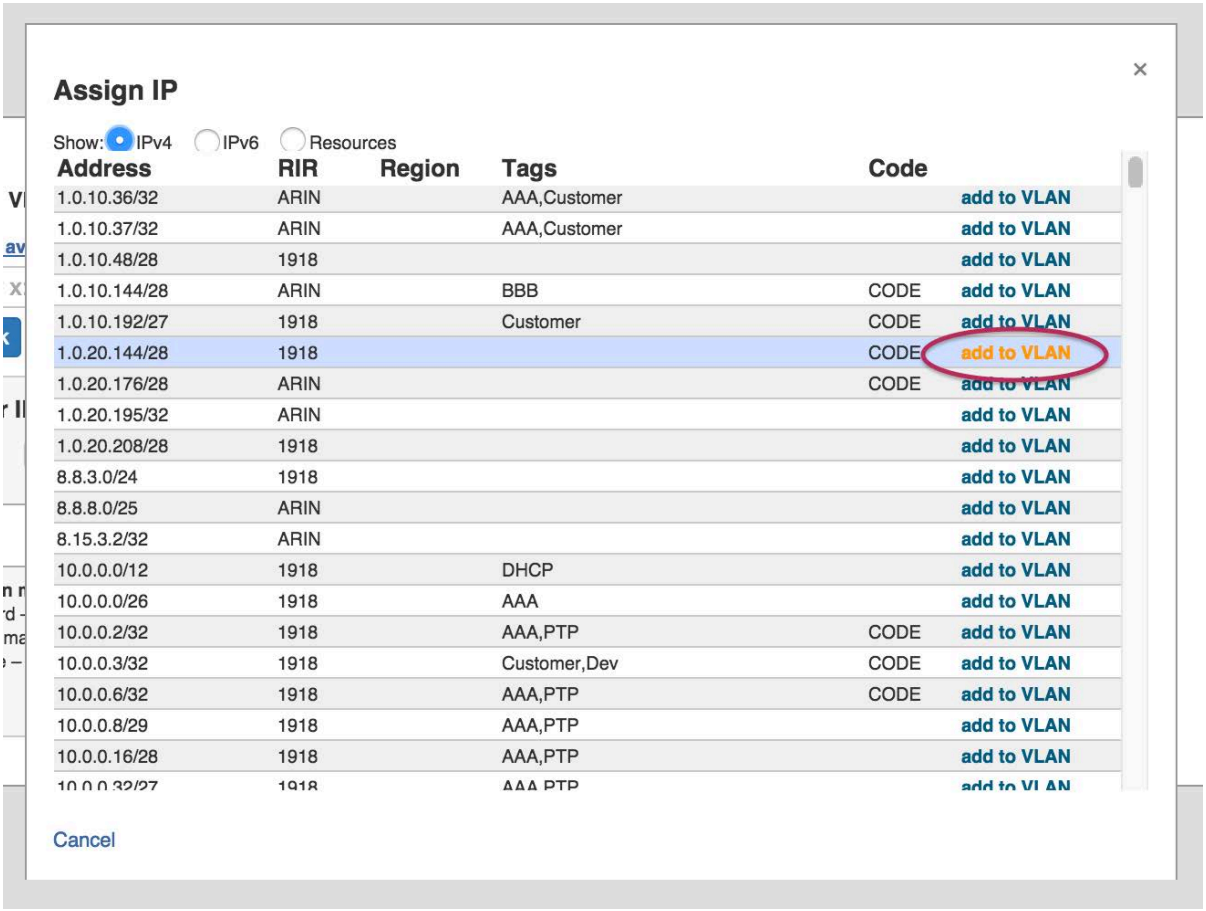
Browse IP Blocks to add to VLAN:

Browse blocks available to add to the VLAN by clicking on the "List available blocks" link above the IP block input box.

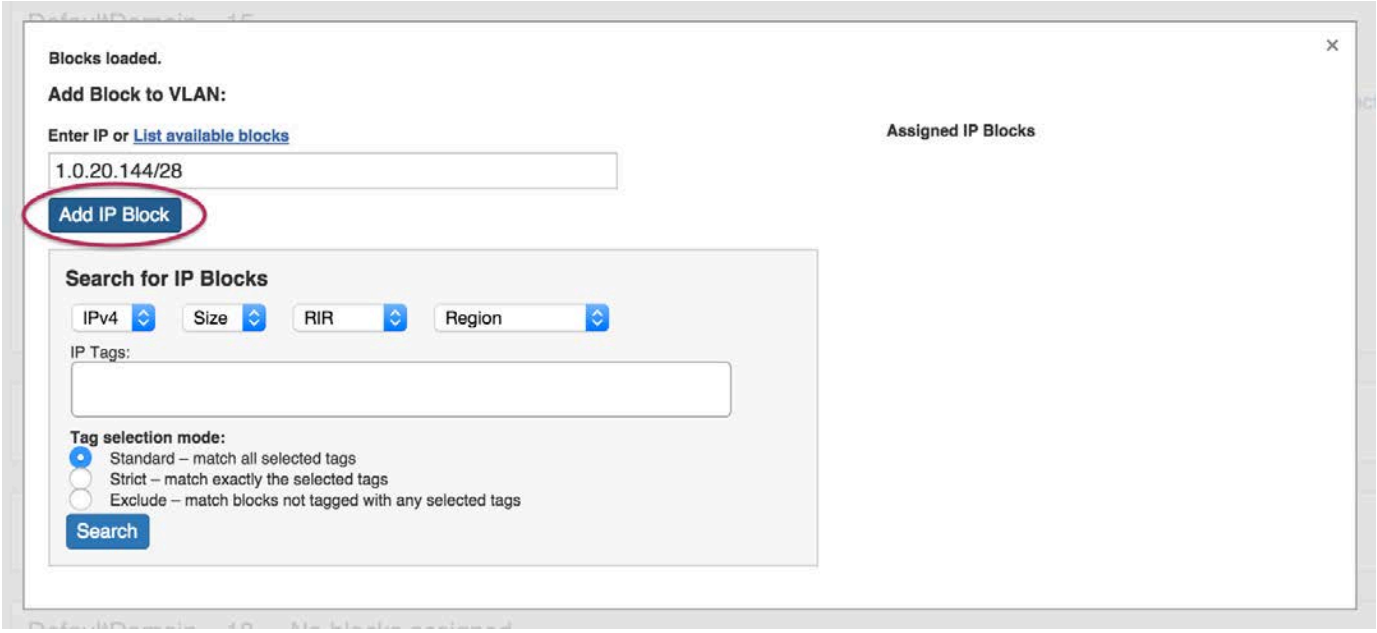


This opens a window similar to the "Smart Browse" used in assigning IPs to a resource - just select IPv4, IPv6 blocks, or Resources to browse from the radio buttons at the top of the list.

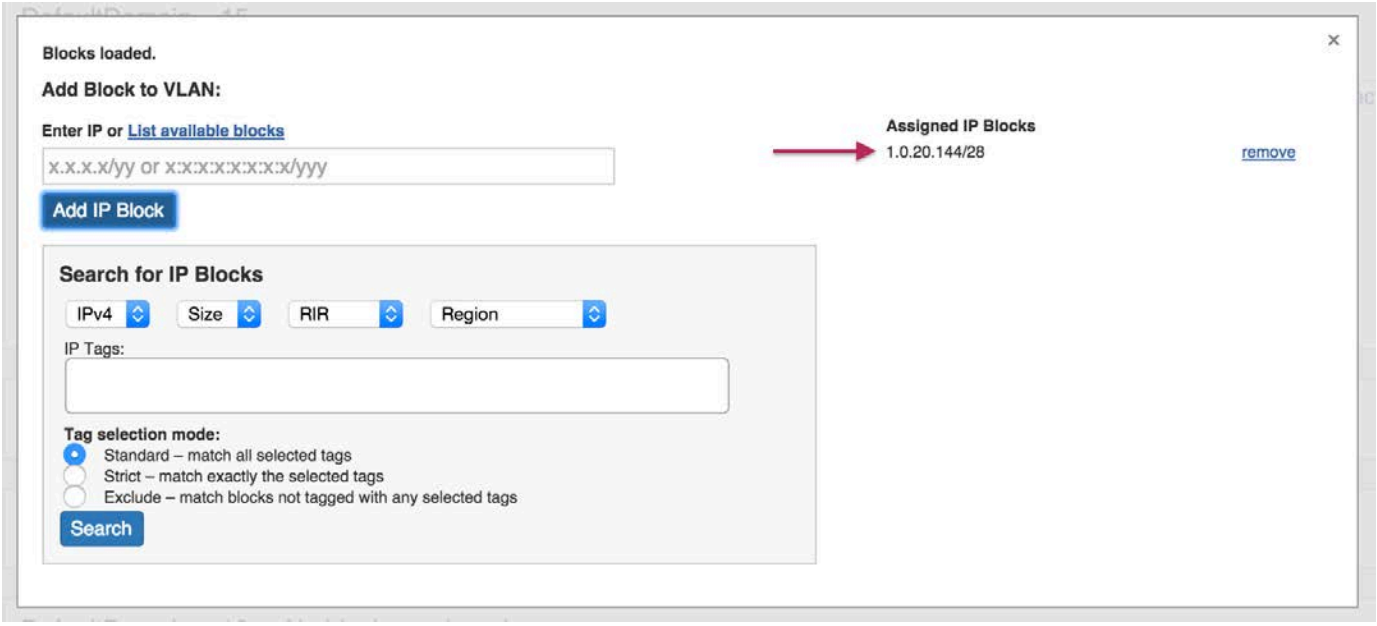
When you have selected the block to add, click "add to VLAN".



The selected block will automatically fill in the block input box - just hit "Add to IP Block" below the input to finalize the addition.

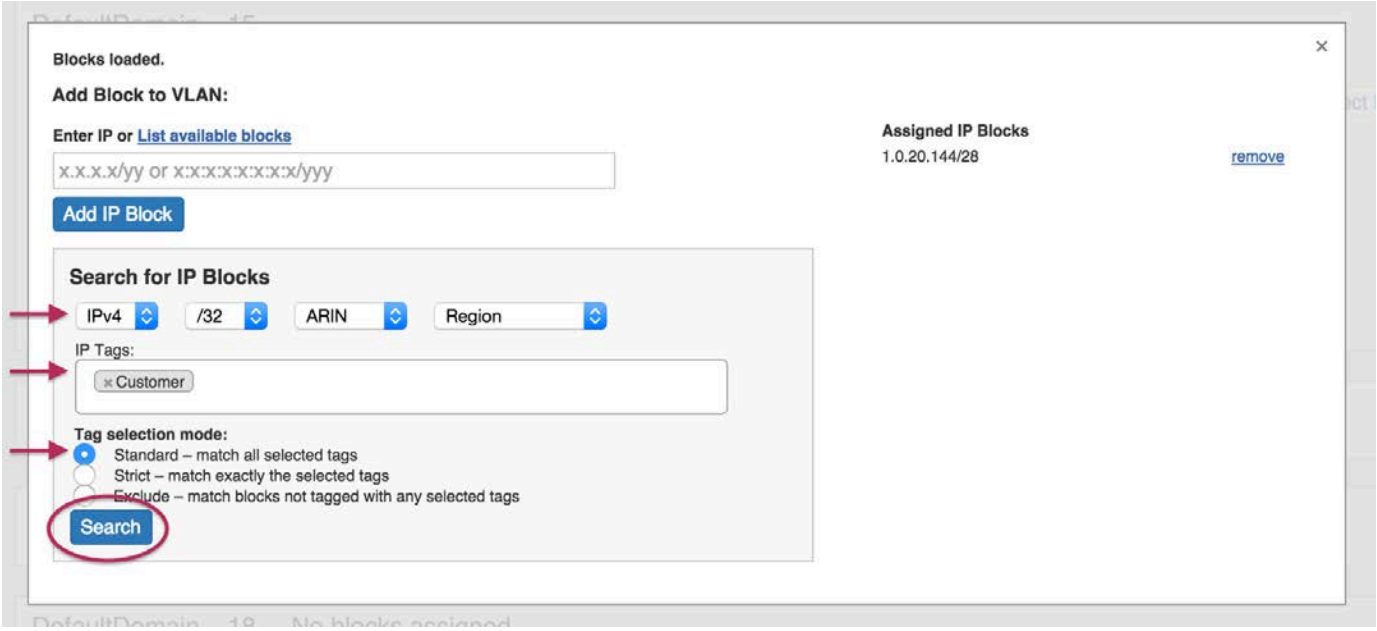


When complete, the block will be added to the "Assigned IP Blocks" list on the right of the window, with the option to remove if desired.



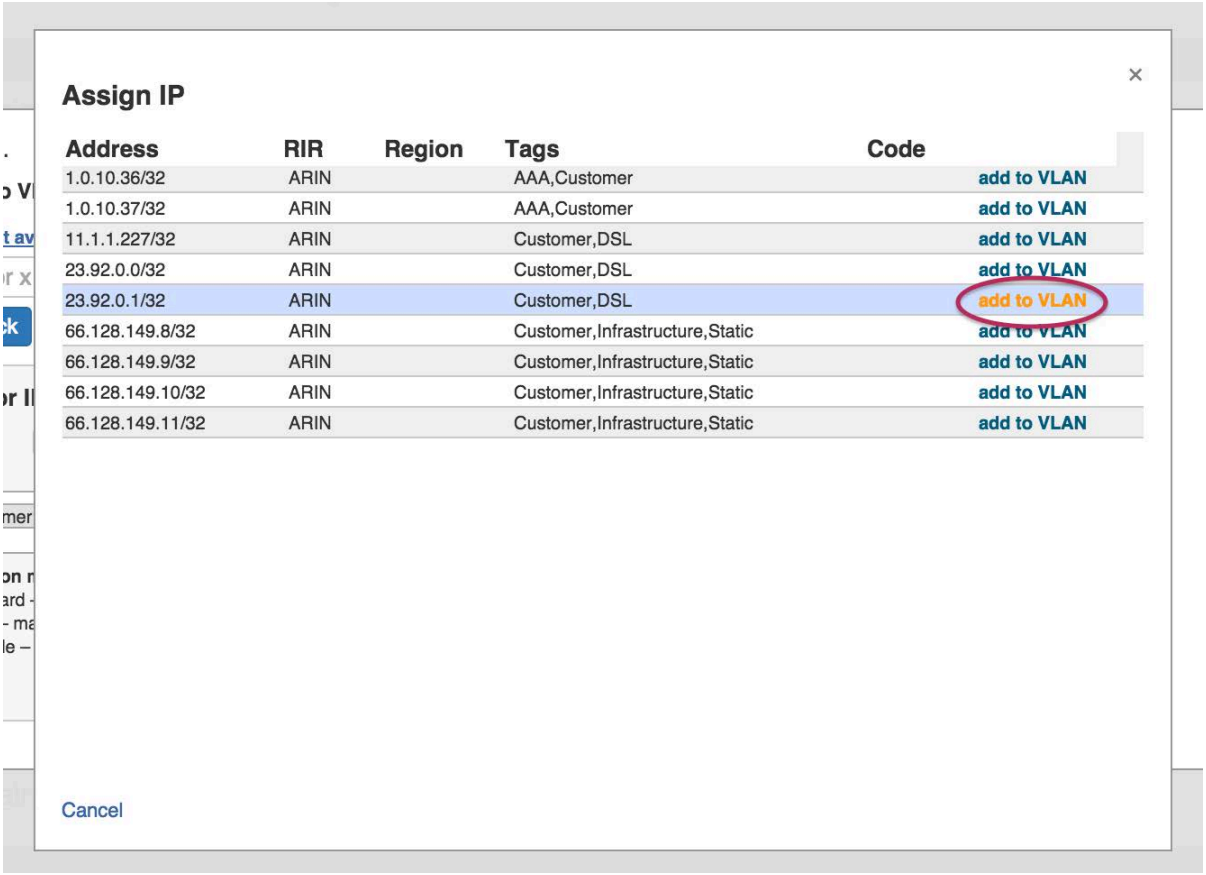
Search IP Blocks to add to VLAN:

Select search criteria of IPv4 / IPv6, size, RIR, Region, or Associated Tags / Tag selection mode to create a filtered list of IP blocks meeting that criteria. Then, click "Search".

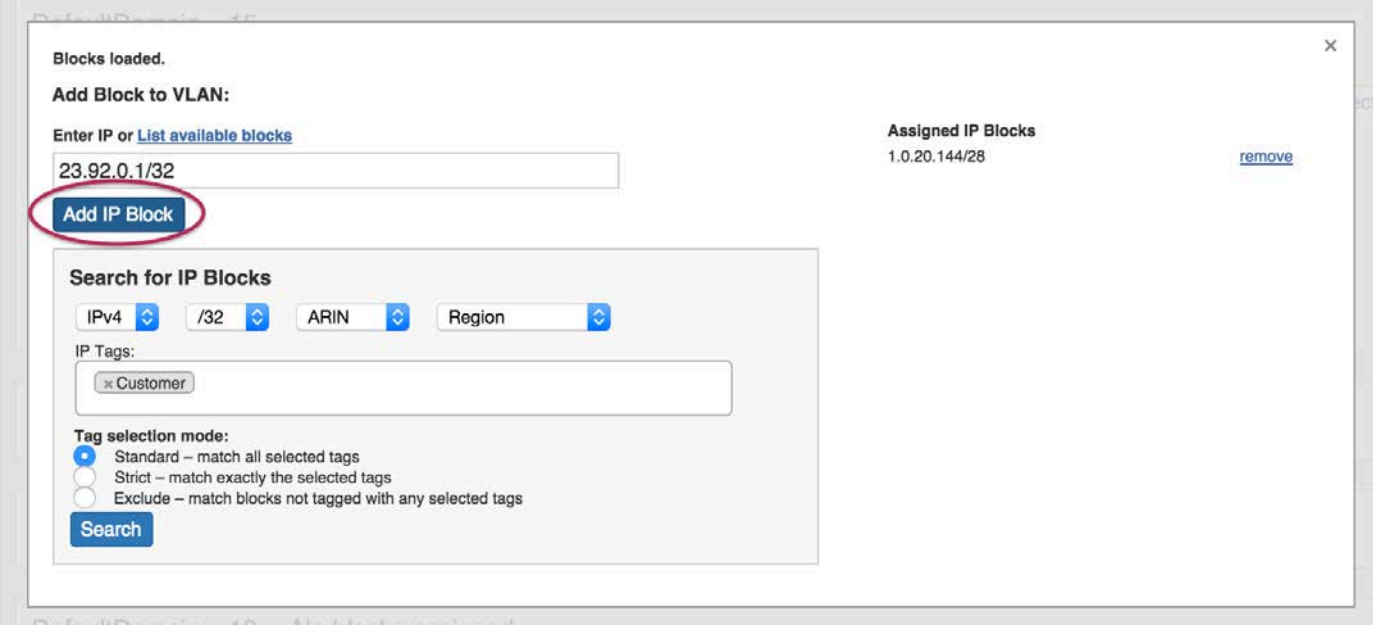


This opens a window with filtered results matching your criteria.

When you have selected the block to add, click "add to VLAN".



The selected block will automatically fill in the block input box - just hit "Add to IP Block" below the input to finalize the addition.



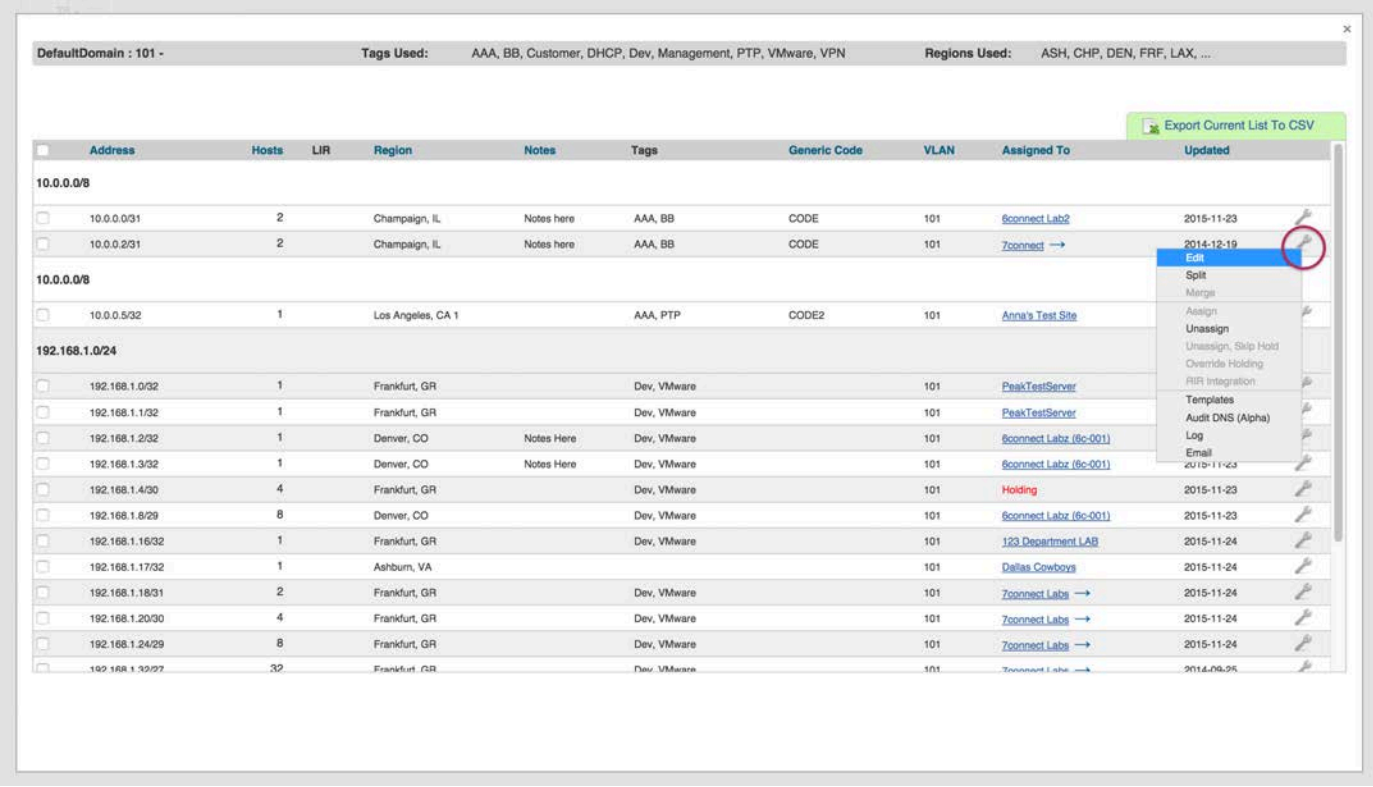
When complete, the block will be added to the "Assigned IP Blocks" list on the right of the window, with the option to remove if desired.

Add / Update VLAN to Blocks from IPAM Manage or VLAN Manage:

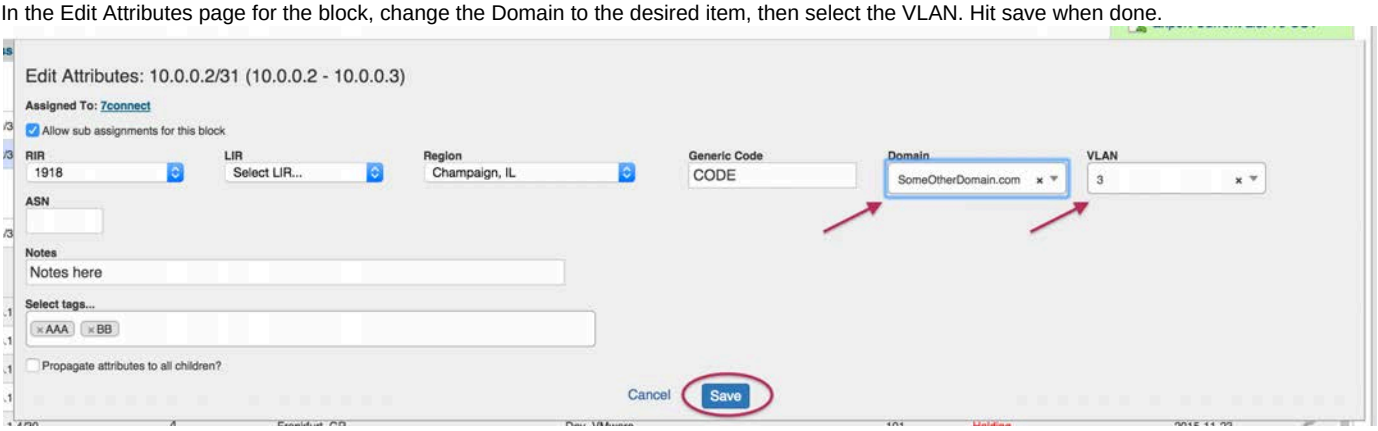
When editing a block in either IP Manage (IPAM Tab -> Action Menu (wrench) -> Manage), or VLAN Manage (IPAM Tab - VLAN -> Action Menu (wrench)) , you can select an existing domain and VLAN to associate to the block.

VLAN Manage:

From the VLAN Manage screen, click on the Action Menu (wrench icon) for the desired block, the select "Edit".

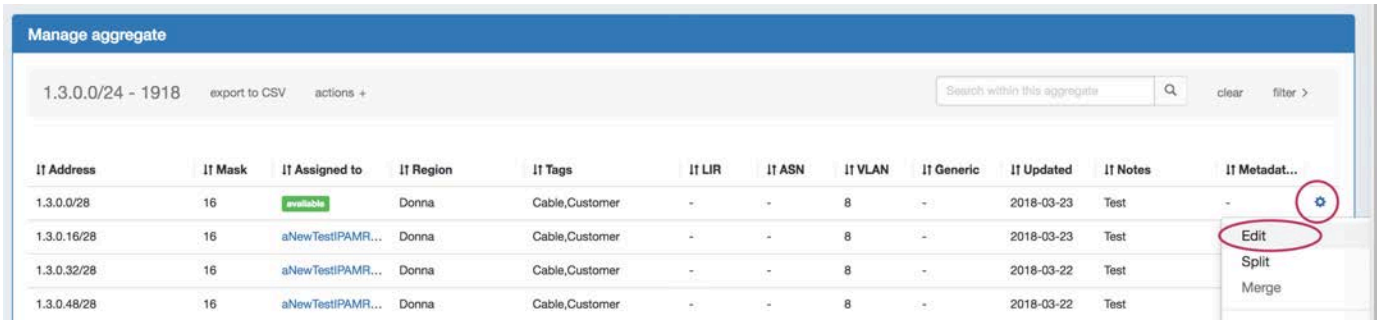


In the Edit Attributes page for the block, change the Domain to the desired item, then select the VLAN. Hit save when done.

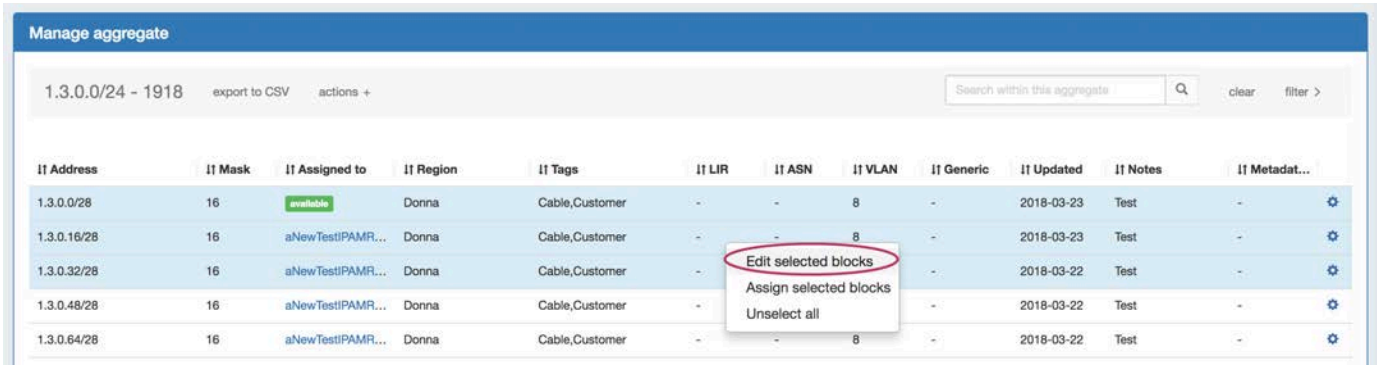


IPAM Manage:

From the IPAM Manage screen, right click on the Action Menu (gear icon) for the desired block, the select "Edit".

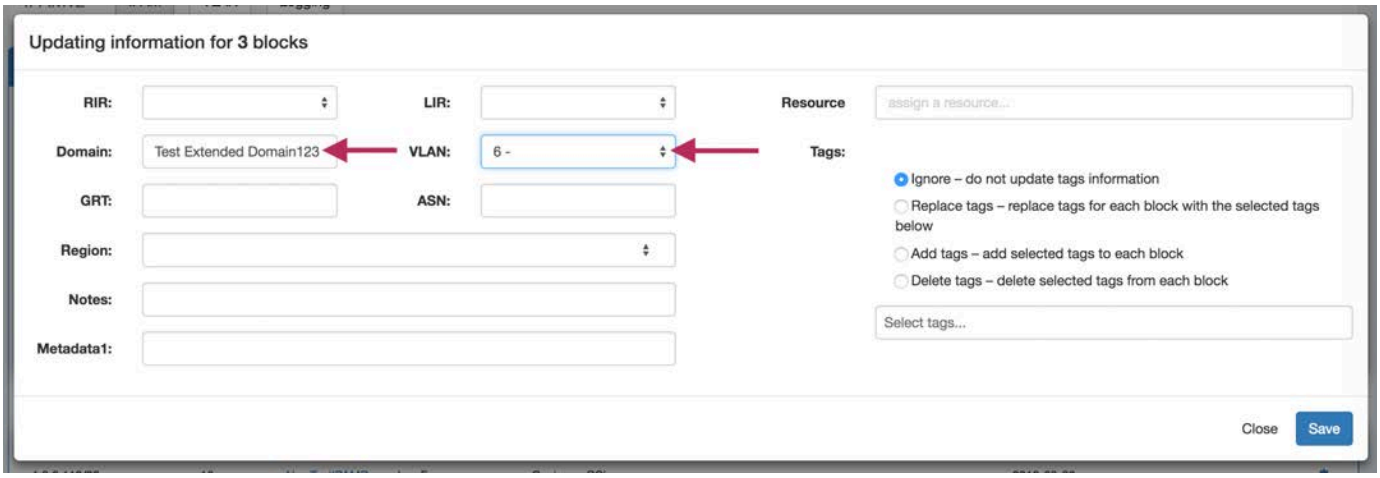


To select multple blocks to edit, hold down the shift key and click on the blocks you wish to edit - the selected blocks will be highlighted. Right click anywhere in the blue highlighted area, and select "Edit selected blocks".



The "Edit Block" modal window will pop up for the selected block(s). Edit the domain by clicking in or typing in the domain search box, and select the desired Domain.

Then, select the VLAN for the blocks from the enabled VLAN list. When done, click "Save".



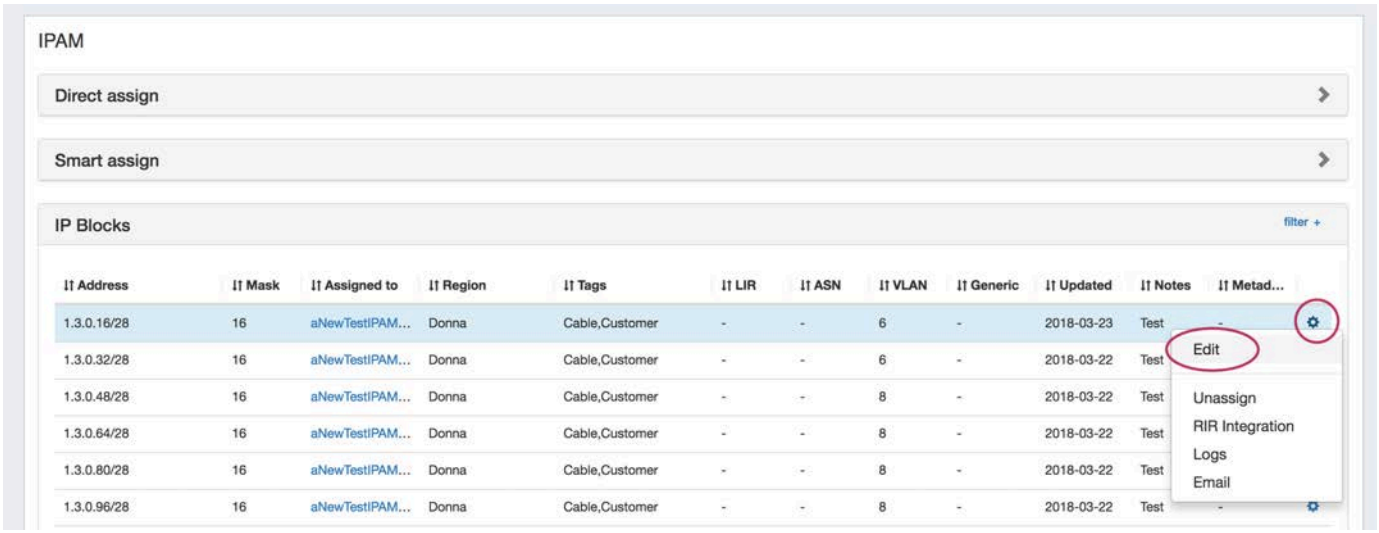
VLANs and Child Blocks

NOTE: Child blocks whose Parent blocks already have an assigned Domain / VLAN, may not have domains and VLANs changed to be different than the Parent Block's values. If you attempt to save domain / VLAN changes to a child block under this situation, an error will occur.

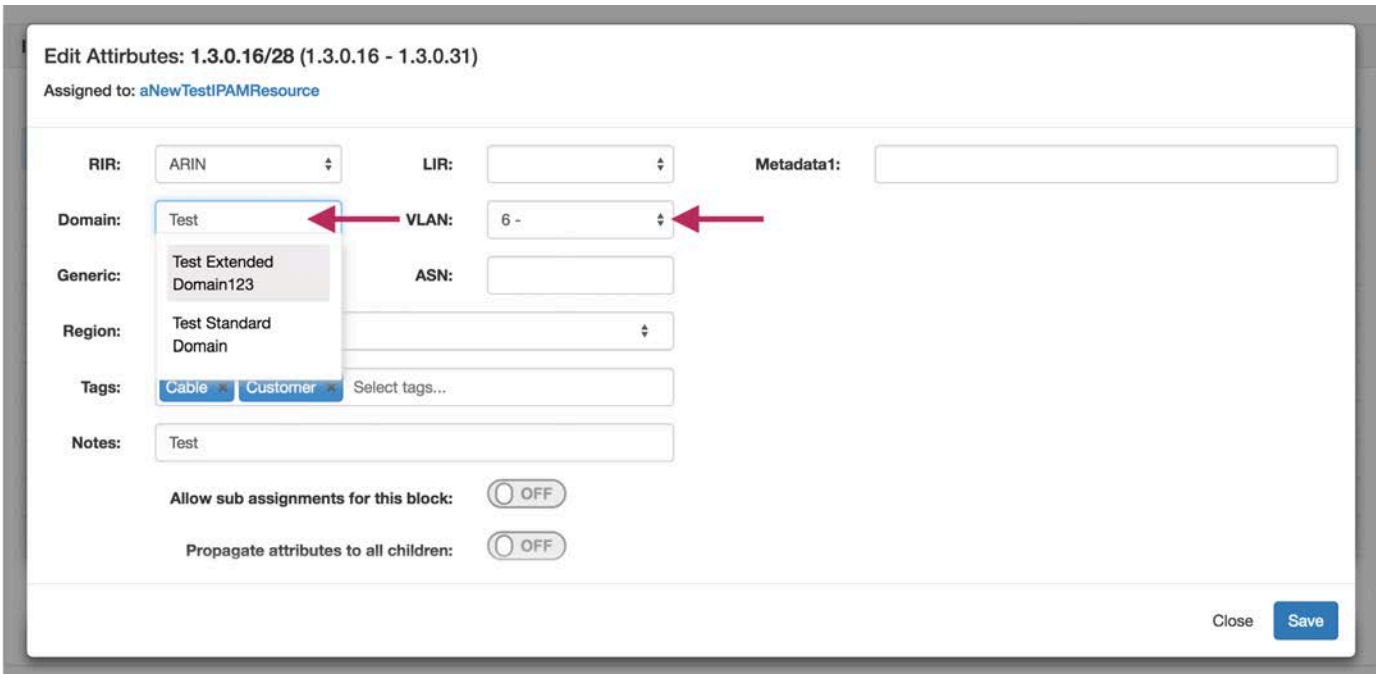
i.e., Parent and Child IP Blocks must have matching Domain / VLAN values.

Add / Update VLAN to Blocks from IPAM Gadget:

You may also add / edit Domain /VLAN information when editing a block in the IPAM Gadget. From the IP block's Action Menu (gear icon), select "Edit".



Then, in the Edit Attributes box that pops up, search for or type the desired Domain, select VLAN information, and click "save".

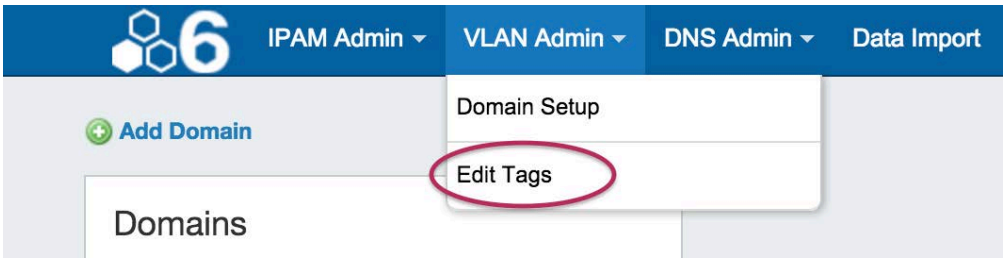


Working with VLAN Tags - (Optional)

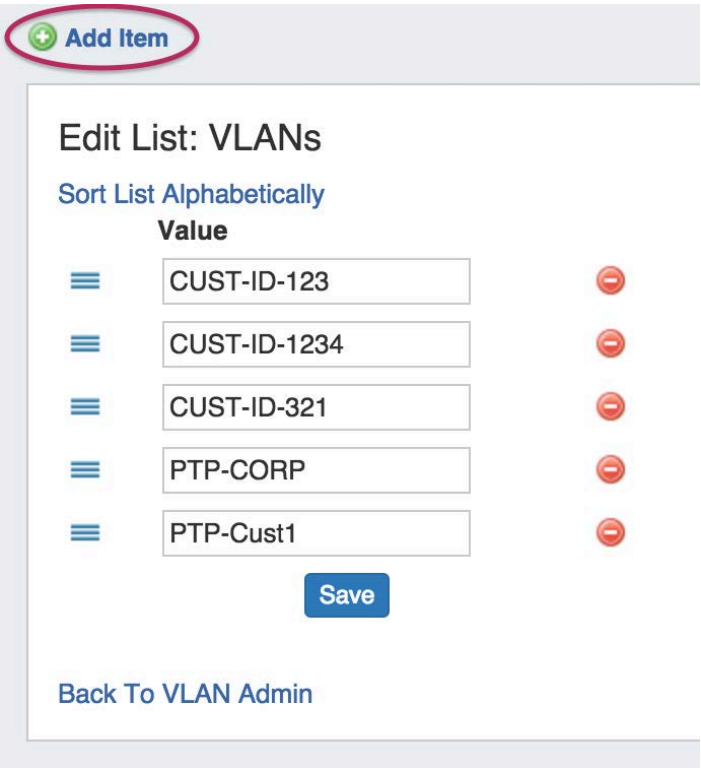
VLAN tags provide further organizational / search criteria when searching for VLANs to enable.

Creating VLAN Tags

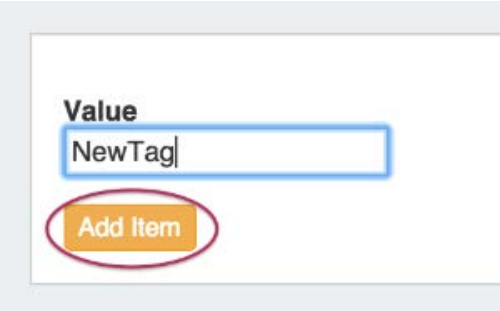
Create VLAN Tags from the VLAN Admin Tab, in the Admin area of ProVision, and select "Edit Tags" from the dropdown.



To add a new VLAN tag, click on "Add Item" at the top of the Edit List: VLAN Tags page.

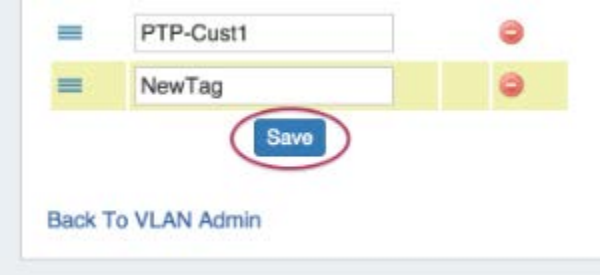


Then, type in the desired name value for the new tag, and hit "Add Item". The new tag will be added to the list below.



To **edit** a tag, simply type your changes in to the text box with the tag name. Tags with unsaved changes will be highlighted until saved.

To **delete** a tag, click on the red "delete" symbol to the right of the tag name.



When complete, be sure to click on the "Save" button to save your changes.

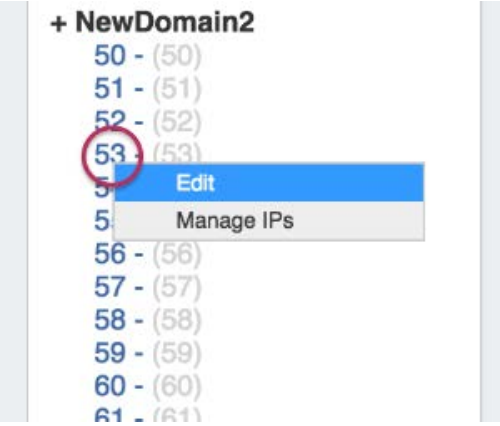
Once VLAN tags have been created, they will be available to add to VLAN ranges and use in the "Add VLAN" search function.

Add VLAN Tags to VLANs

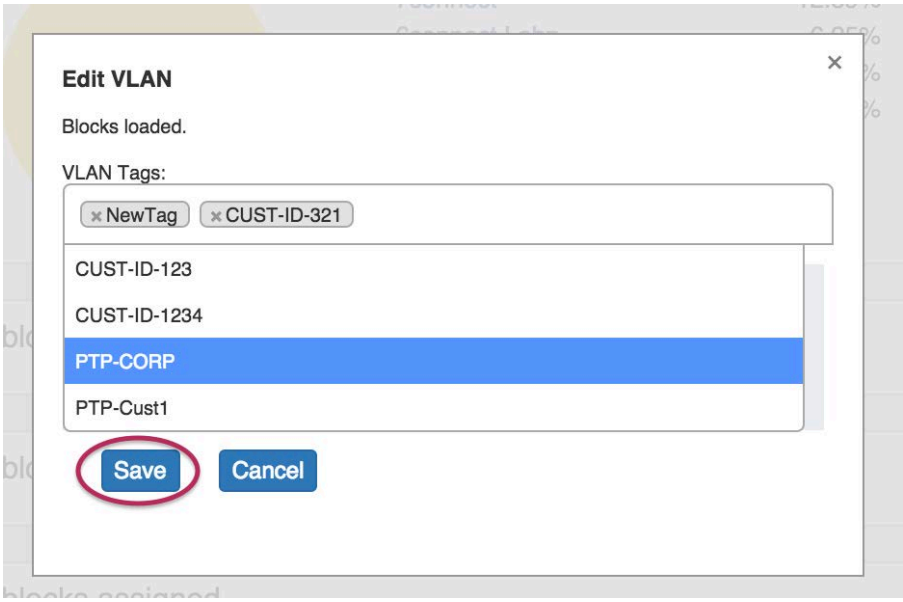
VLAN tags may be added to VLANs in two ways: Editing an already-enabled VLAN (VLAN Edit), or by assigning the tag to a range of VLANs from the Domain VLAN pool (un-enabled VLANs, which are then available to search by VLAN tag to enable).

Add Tag to an Enabled VLAN

Edit an existing, already enabled VLAN by clicking on the VLAN in the VLAN List sidebar, then selecting "Edit".



Apply a VLAN tag from clicking in the "VLAN Tags" box, and selecting one or more VLAN Tags.

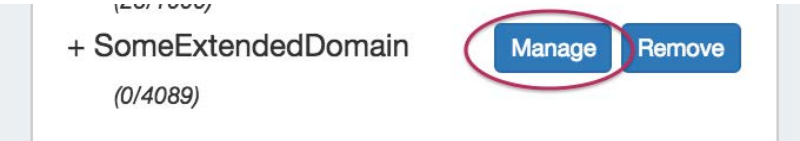


After making your changes, click the "Save" button. Tagged VLANs will be viewable in the VLAN Chart a single tag points under VLAN Admin - Domain Manage.

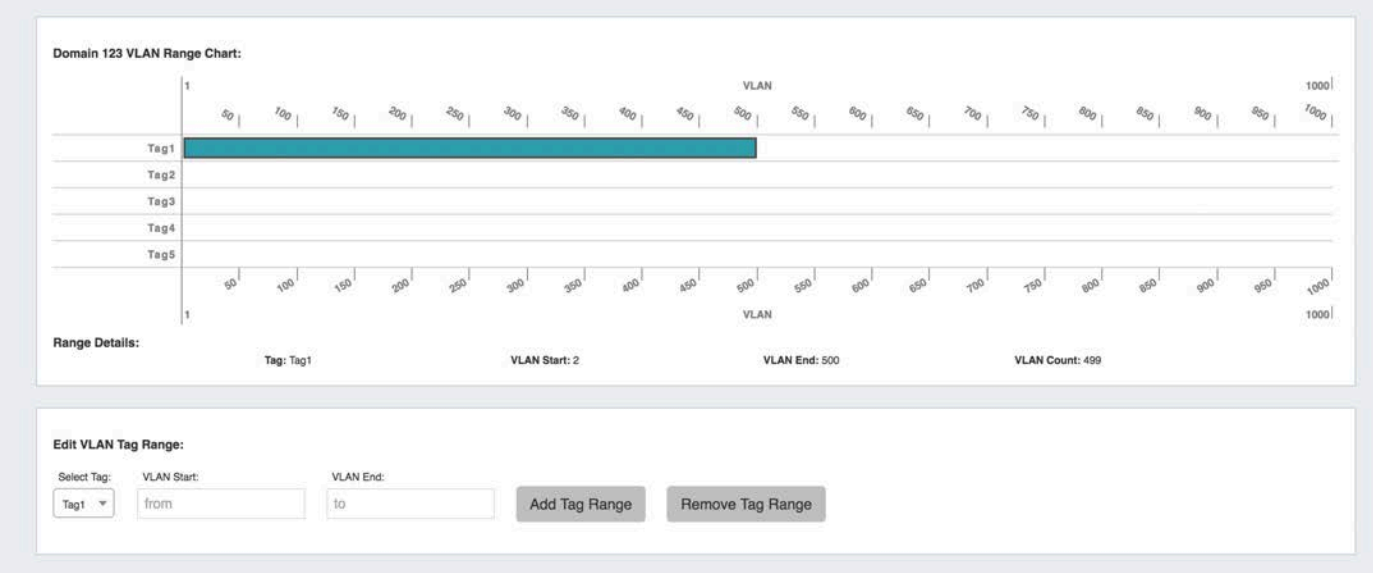
Add Tag to a Range of the VLAN Pool

You can also tag a range of the (un-enabled) VLANs in a domain's VLAN pool. Then, when moving on to enable a set of VLANs for a Domain, you can select those VLANs by tag.

Go to the VLAN Admin Tab in the Admin area of ProVision. Then, next to the desired domain, click "Manage".



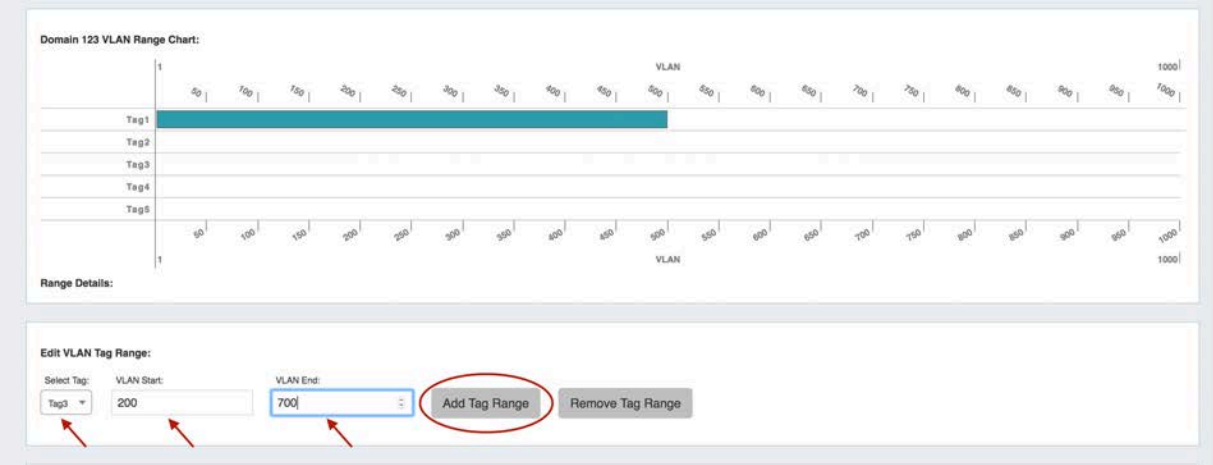
This brings up the page to edit the domain or view a chart of VLAN tags (y axis) to VLANs in the available pool (x axis).



Hovering the mouse over a tag range will show details such as the range start, end, and count of vlans in the range.

Under the "Edit VLAN Tag Range" section, you can bulk assign tags to ranges by:

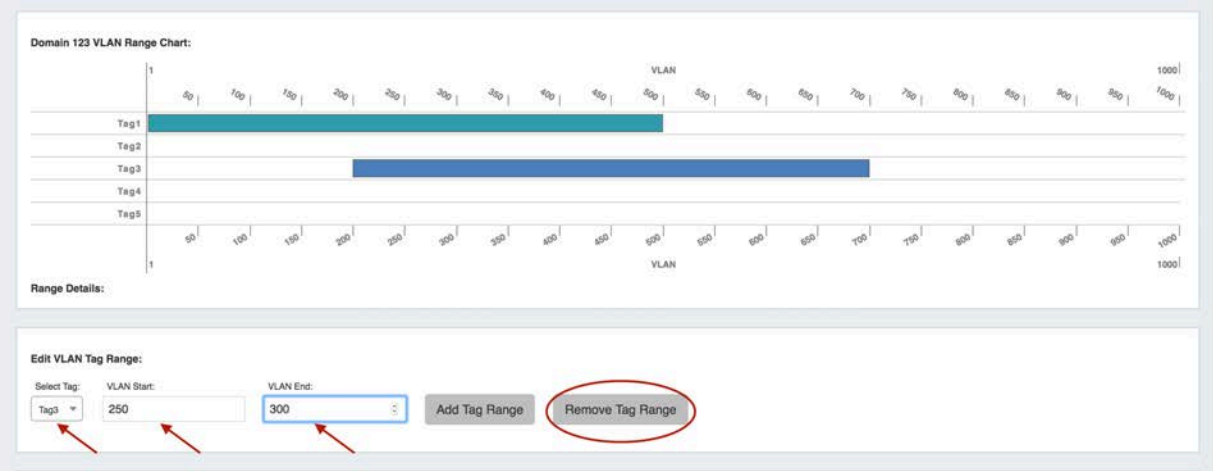
1. Select the tag to edit in dropdown below chart
2. Type desired VLAN start and end value for the range
3. Click "Add Tag Range" to set and save.



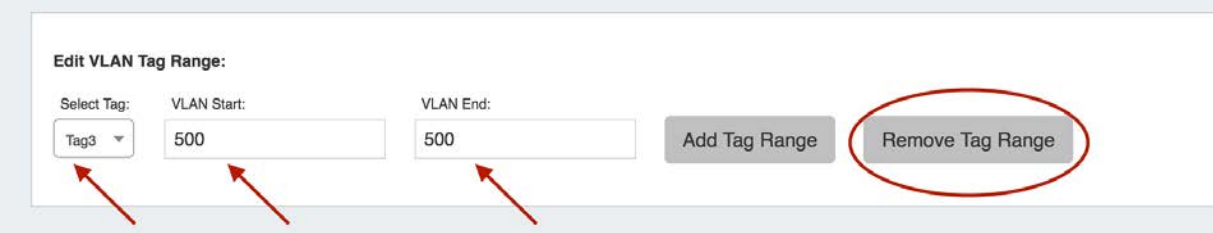
To remove a range of tags, under "Edit VLAN Tag Range":

1. Select tag to edit in dropdown below chart
2. Type desired VLAN start and end value for the range to remove

3. Click "Remove Tag Range" to set and save.



4. To untag a single VLAN, simply enter the same VLAN value in the VLAN Start and VLAN End boxes, then hit "Remove Tag Range".



5. Changes to tag ranges are automatically saved when clicking "Add Tag Range" or "Remove Tag Range", so after all desired changes are made, you may immediately exit the page.

Once assigned, the tags show as a property of the VLAN block in the VLAN edit menu and may be used for VLAN Searches / Assignments.

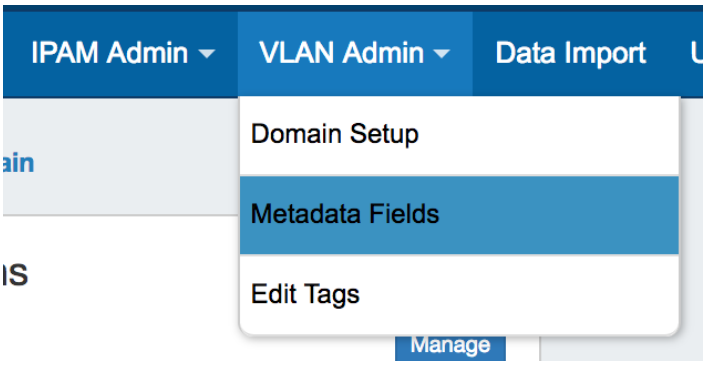
Working with VLAN Metadata - (Optional)

Customizable metadata fields are available to hold VLAN-specific information such as Location, Client, Firewall, Gateway, and more.

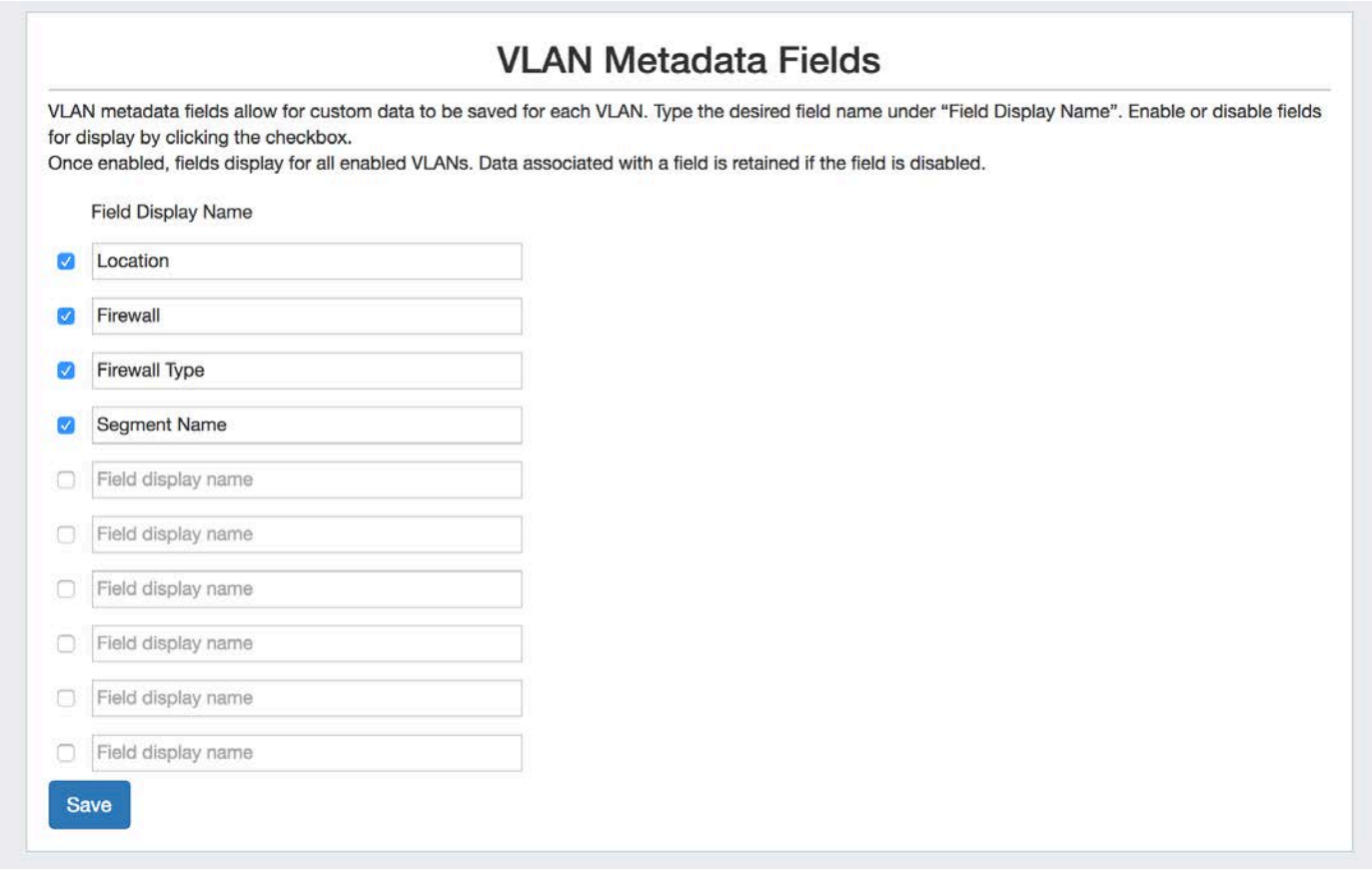
These fields can be enabled and disabled in the VLAN Admin section of ProVision, and viewed / edited under "Edit VLAN" in the VLAN Tab.

Enable / Disable VLAn Metadata Fields

VLAN Metadata fields can be enabled and disabled under the **VLAN Admin** area of ProVision.



To enable a new Metadata Field, navigate to the Admin section of ProVision, and select "Metadata Fields" from the dropdown.



To enable a new field, type the desired field name in an empty text input box under "Field Display Name", and select the check box next to the new name. Then, click "Save" to save your changes. Field display names may be edited at any time by simply typing the desired change into the "Field Display Name" box and saving your changes.

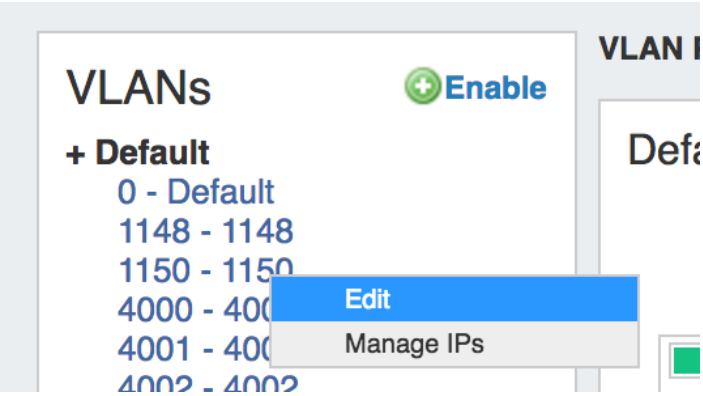
Once enabled, a metadata field displays for all enabled VLANs.

To disable a metadata field, deselect the check box next to the Field Display Name and click "Save". Any data that exists under a disabled field is retained, and will reappear once the field is re-enabled.

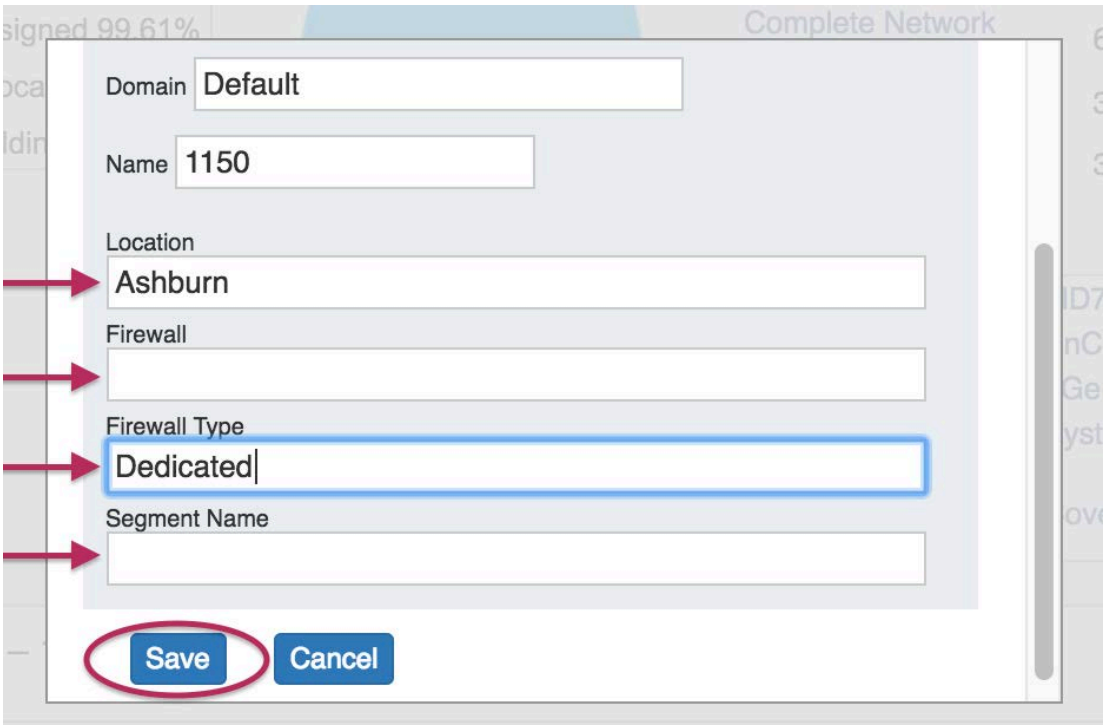
View / Edit Metadata Field Information

VLAN Metadata may be viewed and edited in from the IPAM - VLAN Tab.

While in the IPAM Tab -> VLAN Page, expand the desired domain from the left sidebar and click on a VLAN. Then select "Edit".

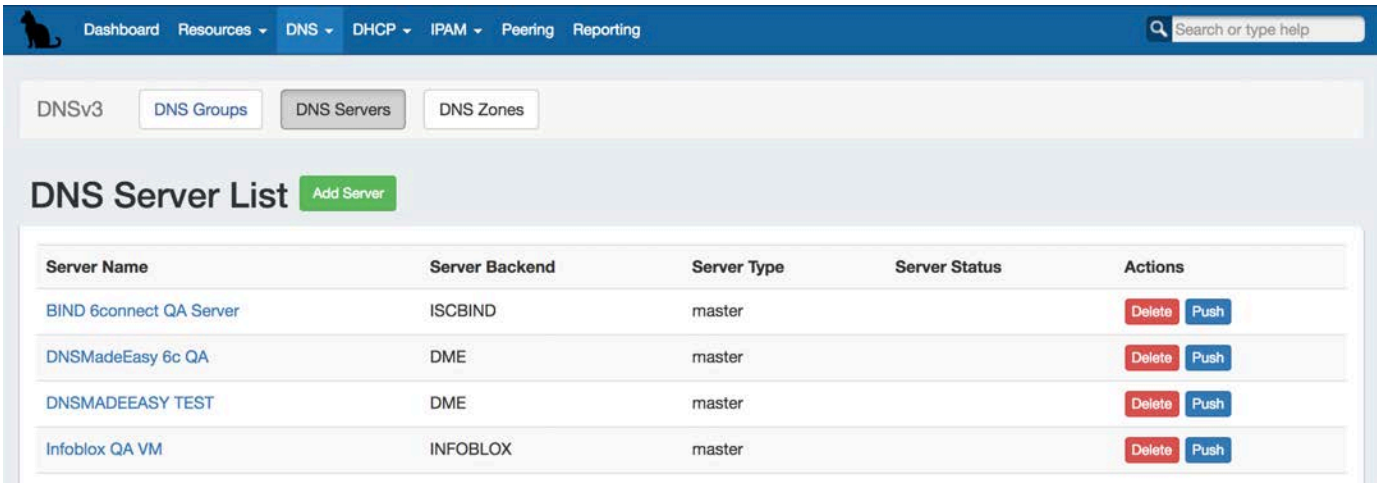


In the Edit VLAN screen, you may view the Metadata or enter your desired changes and click "Save" when done.


Complete Network
Domain Default
Name 1150
Location
Ashburn
Firewall
Firewall Type
Dedicated
Segment Name
Save
Cancel

DNS Administration

DNS Administration



DNS Administration in DNSv3 is primarily handled under the **DNS** tab **DNS Servers** sub-tab.

The **DNS Servers** tab is only accessible to Admin users, and contains functions for adding, updating, and managing DNS servers as well as scheduling server tasks.

Additional DNS Admin tasks occur in other ProVision areas, and not all management tasks require Admin-level permissions - some only require appropriate resource permissions on the DNS Groups, Zones, and servers involved. See additional sections on this page for more detailed information:

- DNS Administration
  - DNSv3 Overview
    - DNSv3 Permissions
    - DNSv3 Administrative Tasks
  - Manage DNS Servers
  - ACLs / Views
  - DNS Zone Transfers (Pushes)
    - Manual DNS Pushes:
    - Scheduled DNS Pushes:
  - DNS Record Types
  - Global DNS Settings (Local Installation Only)
    - DNS Global Settings:
    - DNSSEC Tools:
  - DNS Export Functions
    - Exporting Zones
  - Importing DNS Zones:
  - Additional Information:
    - System Information for Local Installations
    - Additional Sections:

DNSv3 Overview

DNSv3 reorganizes ProVision's DNS system into a more unified and accessible interface, combining both admin and non-admin DNS tasks together under the **DNS** tab.

In DNSv3, zones are gathered under DNS Groups, servers are attached to those Groups, and Nameservers, Default SOA values, and ACLs are managed at a per-Group level.

Users can then view and manage Groups, individual zones, default SOA values, ACLs, attach servers, and perform pushes all on the same page.

DNSv3 Zone Name Response Change

Note: In DNSv3, zone name responses will enforce and return a trailing period (i.e., "example.com" will be converted and returned as "example.com.").

DNSv3 is designed to reflect RFC 1035 standards for valid data formats and will return error messaging for data not meeting those formats.

DNSv3 Permissions

DNSv3 (6.0.0 release), restructures DNS to incorporate DNS zones and Groups into ProVision's Resource System. Zones and Groups are Resources just like Customers, Servers, Routers, or Contacts (See Resource Concepts for a more detailed explanation of Resources in ProVision).

This allows for DNS zones and Group permissions to be managed similarly to other ProVision resources, where users with Resource permissions (Create / Read / Update / Delete) on the parent resource of the DNS Group can create groups and zones, manage those groups and zones, push (if a server is attached), and delete.

A user with full Resource permissions on a DNS Server, as well as the parent resource of a Group, may view and attach that server to a Group.

Users with Admin permissions can access the **DNS Servers** area under the **DNS** tab manage DNS server creation, edits, and deletion.

For more information on setting up permissions groups in ProVision, see Users & Permissions.

DNSv3 Administrative Tasks

Admin-only tasks in DNSv3 include setting DNS Globals, importing zones, scheduling pushes, and managing DNS Servers.

These tasks may be accessed in the following areas:

- DNS Global Settings (Local Installs): Admin Preferences page.
- Importing DNS Zones: the **Data Import** tab, see Import DNS Zones.
- Scheduling DNS Pushes:
  - May be set up from the Scheduler or,
  - Set by server from the **DNS Servers** page, or
  - Set by DNS Group from the **DNS Groups** page, or
  - Set by individual zone from the View Zone page
- Managing DNS Servers: The **DNS Servers** page

Tasks such as working with zones, DNS Groups, adding ACLs, attaching servers to Groups, and manually pushing zones do not require ProVision Admin permissions.

Instead, these tasks simply require that the user be included in a User Group that has appropriate permissions (either direct, or inherited) on the DNS zones, Groups, and servers involved.

Manage DNS Servers

Adding, configuring, and managing DNS Servers occurs in the Admin-only **DNS Servers** page under the **DNS** Tab.

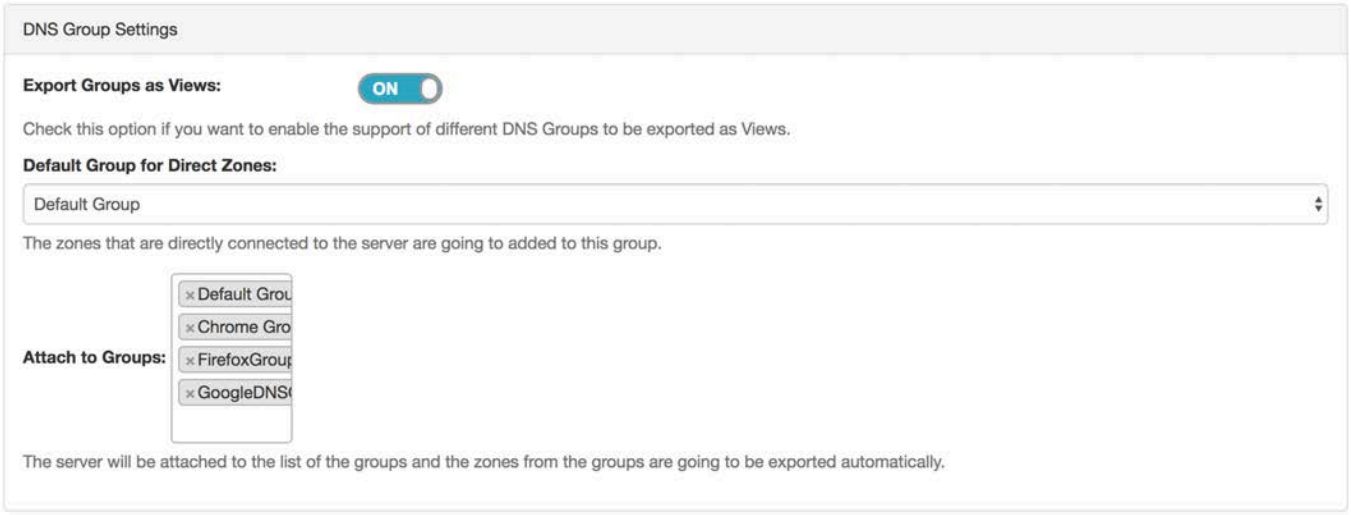
Supported DNSv3 server types include:

- ISC BIND
- Secure 64 (Authority, x86 Authority, KNOT Authority, Cache, Signer)
- PowerDNS (BIND and MySQL)
- InfoBlox
- Knot DNS
- NS ONE
- DynDNS (Beta)
- DNSMadeEasy
- Dummy (a fake server entry used for servers outside of ProVision access, in order to manage master / slave configurations)

For detailed information on adding, editing, and deleting DNS Servers, see Working with DNS Servers.

ACLs / Views

In DNSv3, Views may be created by selecting the "Export Groups as Views" toggle and selecting a DNS Group under the server details sections "DNS Group Settings".



ACLs are managed on a per-DNS Group level, and may be added, edited, or removed by any user with appropriate resource permissions on the DNS Group.



For details on working with ACLs, see Configuring Split Horizon/Views.

DNS Zone Transfers (Pushes)

In DNSv3, zones may be pushed manually or scheduled for a future time, and may be performed for a single zone, all zones in a DNS Group, or all zones on a server.

Scheduling DNS pushes requires administrative access, but manual pushes only require permissions on the DNS Group / DNS Zone, and the server(s) attached.

Manual DNS Pushes:

Manual pushes may be performed from the following locations in ProVision:

- Single Zone: From the DNS Groups Zone list "Push" button. See Working with DNS Zones - Pushing Zones
- Single Zone: From the DNS View Zone page "Push Zone Now" button. See Working with DNS Zones - Pushing Zones
- Group: From the DNS Groups Zone list "Push Group" button. See Working with DNS Groups - Pushing a Group
- Server: From the DNS Servers list "Push" button. See Working with DNS Servers - Pushing a server
- Server: From the DNS Server Settings page "Push Zones" button. See Working with DNS Servers - Pushing a server

Scheduled DNS Pushes:

Scheduled pushes may be performed from the following locations in ProVision:

- May be set up from the Scheduler for single zones, all zones in a Group, or all zones on a server.

- Set by server from the **DNS Servers** page, or
- Set by DNS Group from the **DNS Groups** page, or
- Set by individual zone from the View Zone page

DNS Record Types

DNSv3 Record Types are a static list comprised of:

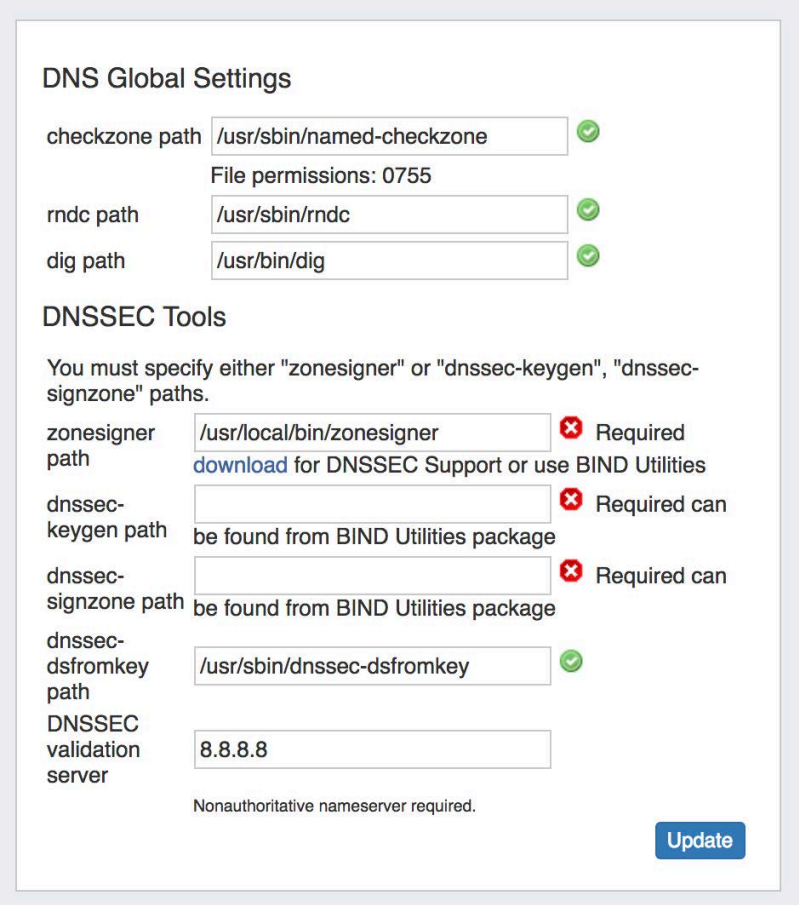
- A, AAAA, CNAME, DNAME, DNSKEY, DS, MX, NAPTR, NS, PTR, SOA, SRV, SPF, HINFO, TXT.

Any user with appropriate resource permissions on the DNS Zone / DNS Group may add, edit, or delete DNS Records.

See Working with DNS Zones - Common Tasks for detailed information on managing DNS Records.

Global DNS Settings (Local Installation Only)

DNS Global Settings is accessible from the Admin Preferences page by users with Admin level permissions.



DNS Global Settings:

- **Checkzone path:** Path to checkzone
- **rndc path:** Path to rndc
- **dig path:** Path to dig

DNSSEC Tools:

- **zonesigner path :** Enter the zonesigner path that will be used for DNS. Zonesigner is required if dnssec-keygen and dnssec-signzone are not set.
- **dnssec-keygen path:** Enter the keygen path. Required if zonesigner is not set.
- **dnssec-signzone path:** Enter the signzone path. Required if zonesigner is not set.
- **dnssec-dsfromkey path :** Enter the dnssec-dsfromkey path that will be used for DNS. Required in all cases.

- **DNSSEC validation server:** Enter the IP of the nonauthoritative DNSSEC validation nameserver. Optional.

DNS Export Functions

Exporting Zones

Individual Zones:

DNS zones may be exported individually from the zone's View Zone page "Export Zone" button, by any user with appropriate zone permissions (see: Editing DNS Zones).

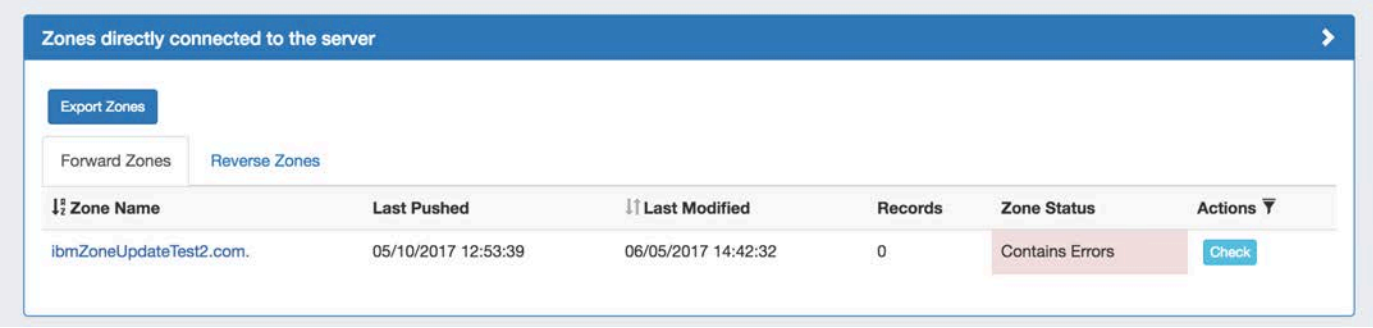
Zones in a DNS Group:

All zones under a DNS Group may be exported from the DNS Group "Export Zones" button, by any user with the appropriate group permissions (see: Working with DNS Groups).

Zones directly connected to a server:

Zones directly connected to a server may be exported by Admin users from the DNS Server Details page, under the section "Zones connected directly to the server".

Expand the module by clicking the (>) arrow, and then click the "Export Zones" button.



Importing DNS Zones:

ProVision offers six DNS zone import options, available under the **Data Import** tab in the Admin section. For more information on importing DNS zones, see Importing your Data and Import DNS Zones.

BIND Zone Upload / Import: Imports DNS zones using the named.conf configuration file tied to the zones you are uploading, a .zip or .tar file of the zones themselves, and an optional .csv file mapping zones to customers and DNS Servers. This is the simplest and most commonly used import method. Refer to Import DNS Zones for more information.

PowerDNS Zone Import: This tool is available after configuring a PowerDNS server with a MySQL backend. The PowerDNS Import connects to the selected server and imports all zones.

InfoBlox Zone Import: Imports DNS zones using a provided Host, Username, and Password. The InfoBlox import pulls all zones on the InfoBlox LOCAL grid and adds them to a designated Group. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

NS ONE Zone Import: Imports DNS zones using a NS One API Key. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

Dyn DNS Zone Import: Imports DNS zones using a Dyn DNS Customer Name, Username, and Password. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

DNSMadeEasy Zone Import: Imports DNS zones using a DNSMadeEasy API Key and API Secret. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

IPPlan Zone Import: Imports DNS zones using IPPlan MySQL database options. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

Additional Information:

System Information for Local Installations

Zones are stored in the 6connect web root under /zones.

DS keys are stored in the 6connect web root under /keys.

Global DNS Settings may be set from the Admin Preferences page.

Additional Sections:

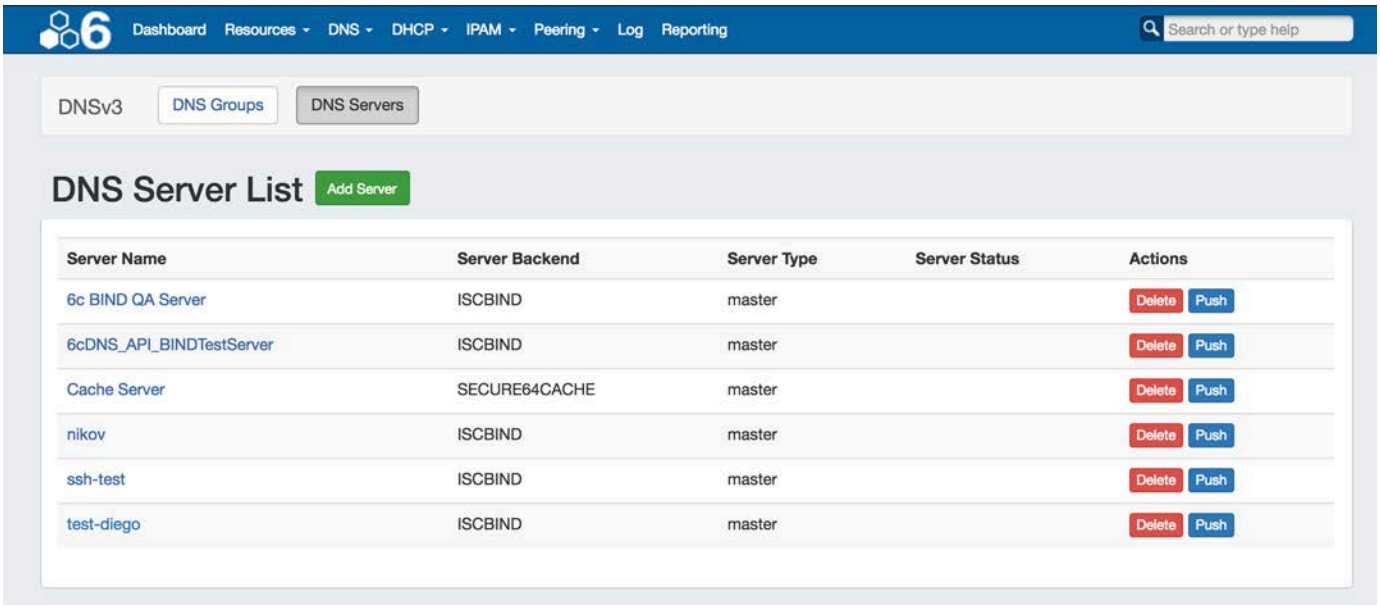
For more information on DNS and configurations, see the following sections:

- Working with DNS Servers
- Configuring ISC BIND Support
- Configuring PowerDNS Support
- Configuring Secure64 Support
- Configuring Split Horizon/Views
- Configuring DNSSEC

Working with DNS Servers

DNS Servers

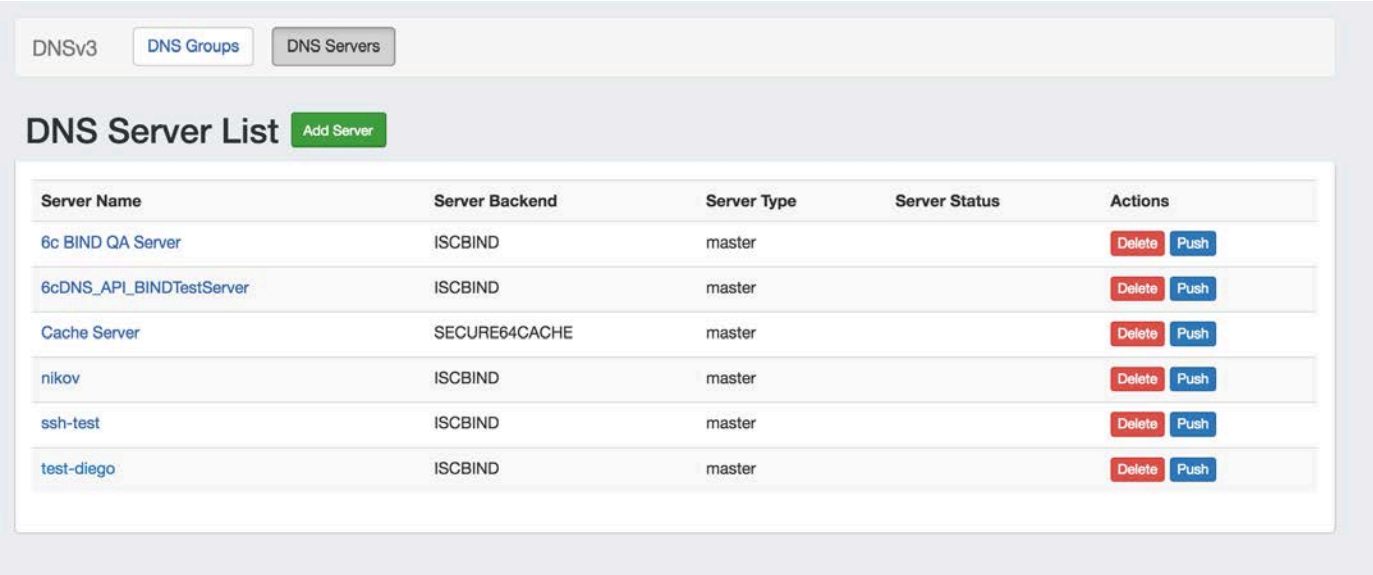
ProVision's DNSv3 combines server management, group organization, and zone management under the **DNS** tab.



The **DNS Servers** tab is only accessible to Admin users, and contains functions for adding, updating, and managing DNS servers as well as scheduling server tasks.

- DNS Servers
  - DNS Server List Interface
- Working with DNS Servers
  - Add a Server
    - Server Settings
      - 1) Set Server Common Settings
      - 2) Set Server Specific Settings
      - 3) Set DNS Group Settings for Server
      - 4) Save Changes
  - Edit Servers
  - Review Zones Connected to a Server
    - Zones Directly Connected to the Server
    - Zones Connected via a Group
  - Pushing a Server
    - Manual Push
    - Scheduled Push
  - Delete a Server

DNS Server List Interface



1) Add Server Button: Opens a dialog for creating a DNS server.

2) Server List:

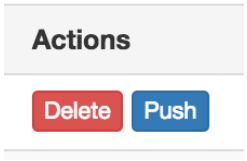
3) Server Name: Name of the DNS server. Click to open server details.

4) Server Backend: The DNS Service backend type for the server.

5) Server Type: Whether the server is a master or slave type.

6) Server Status: Currently a placeholder column for future display of server error and connection status messages.

7) Actions: The actions that may be performed on each server:



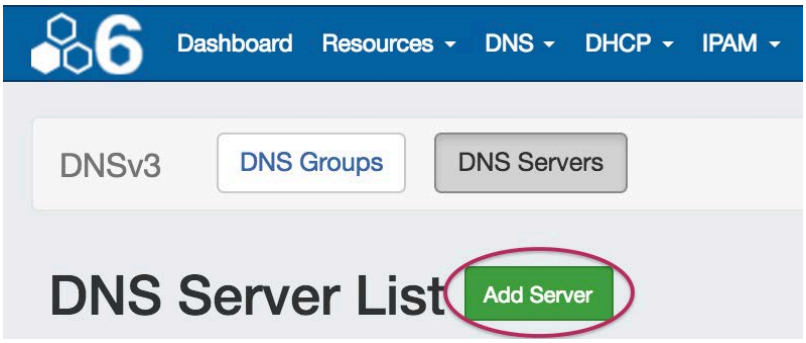
8) Delete: Deletes the server from ProVision.

9 Push: Pushes all zones associated with the selected server.

Working with DNS Servers

Add a Server

To create a new server, start from the **DNS** Tab, select the **DNS Servers** sub menu. Then, click the "Add Server" button next to "DNS Server List".

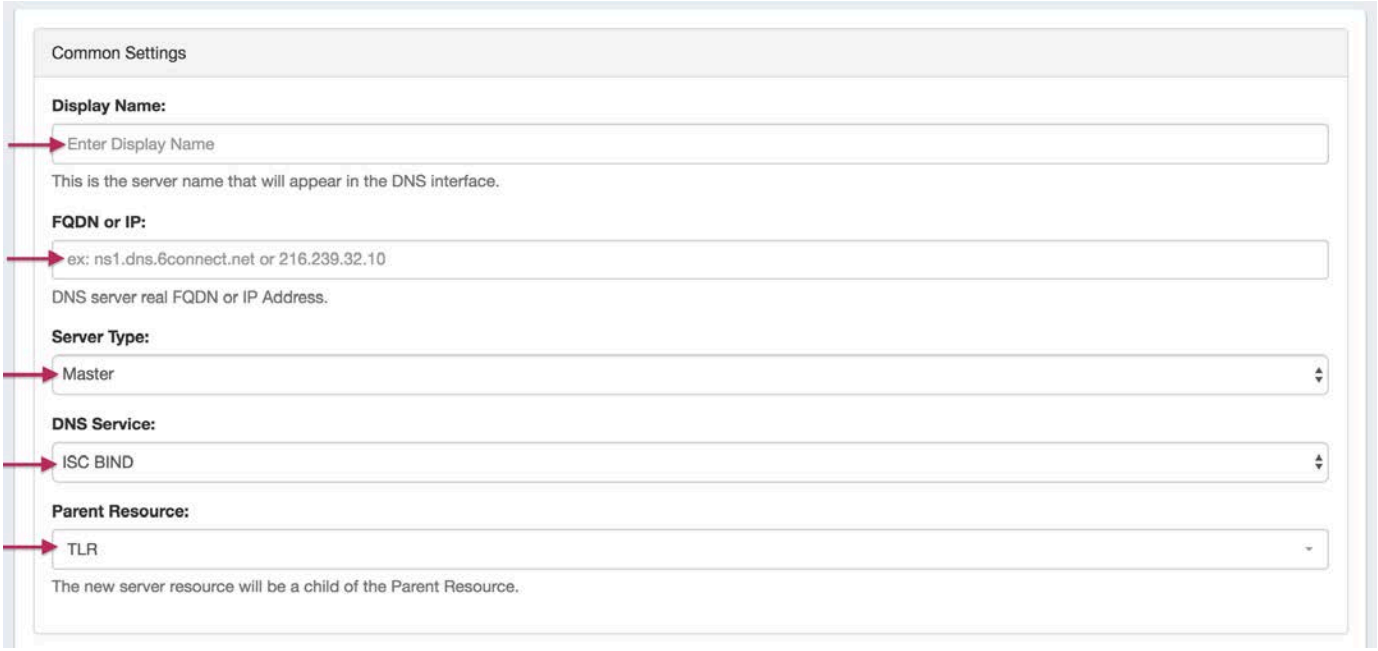


This will open the "Server Settings" page.

Server Settings

1) Set Server Common Settings

In the "Common Settings" section of Server Settings, enter the new server's Display Name (the name that will appear on the ProVision interface), the FQDN / IP, server type, service type, and desired parent Resource (may be left at the default Top Level Resource).



Display Name: Name you want the server to display.

FQDN or IP: The FQDN or ip address of the DNS server.

Default: Specify if the server should be added to new zones by default or not.

Server Type: Specify if the server is a master or slave. Different configuration files are created master vs. slave on the Bind, PowerDNS /Bind, and Secure64 platforms.

DNS Service: Select the DNS service type (ISC Bind, Secure64, KnotDNS, etc).

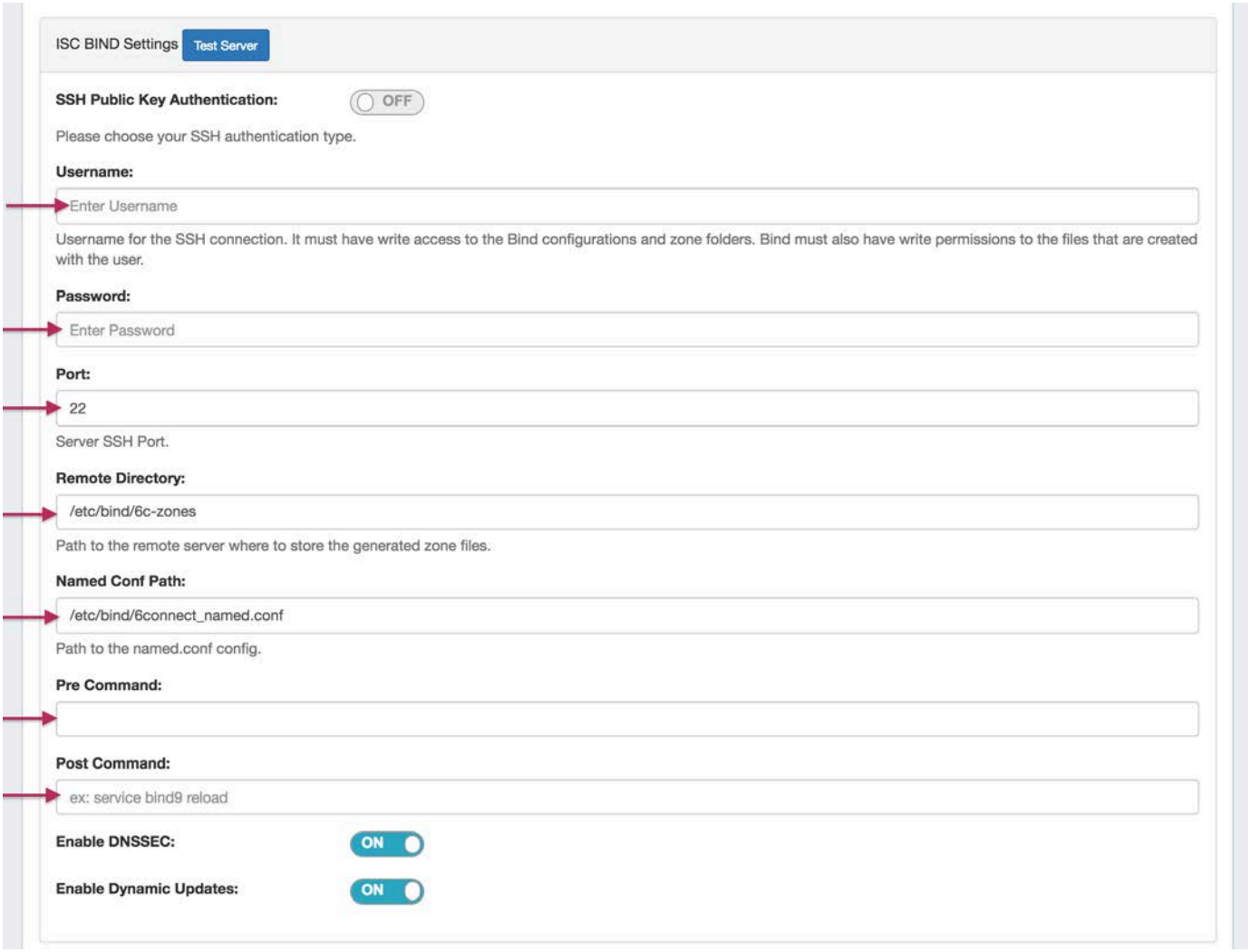
Parent Resource: Select the ProVision resource to be the "parent" of the server - typically TLR (Top Level Resource), but may be a lower level resource such as a Customer or Location. The parent resource selection is the basis of access permissions for the server.

2) Set Server Specific Settings

The next section is entering server service-type specific settings. The options visible in this section will depend on the "DNS Service" type chosen under "Common Settings".

Here, we see the fields for ISC BIND server settings. Enter the server Username, Password, Port, Remote Director, Named Conf. Path, and Pre /Post Command (if desired). Your fields may vary for other server types.

For SSH Public Key Authentication, DNSSEC, and Dynamic Option updates, click on the ON / OFF toggle to select "ON" or "OFF" for each as needed.



SHH Public Key Authentication: If applicable, toggle "On" or "Off"

Username: Login/username for the target DNS server. The specified account needs to be valid, and have write permission to the remote directory and execute permission for any pre/post commands.

Password: Password for the target account. All passwords are stored encrypted in the database.

Port: Port to contact the target server on. This is port used for SSH on Bind and Secure64 server types.

Remote Directory: The target directory to transfer zone files to on the DNS system.

Named Conf Path: The path to other zones on the Bind systems.

Pre Command: Any valid system command on the target DNS system. This command will be run before any files are transferred.

Post Command: Any valid system command on the target DNS system. This command will be run after any files are transferred. For example, on a Bind system you would need to run "rndc reload" to reload the zones.

Enable DNSSEC: If available for the server type, toggle to "On" or "Off". See Configuring DNSSEC for additional information.

Enable Dynamic Updates: Toggle to "On" or "Off", if the server allows dynamic updates.

After entering the server-specific settings in this section, you can click the "Test Server" button at the top of the section to test the server connection and authentication.

The "Test Server" button will attempt to login to the target system and write to the target directory.



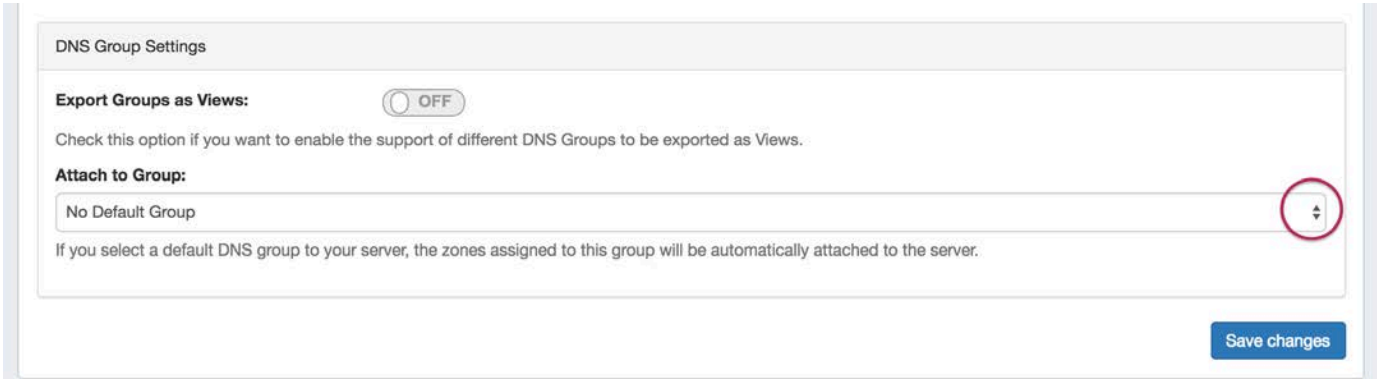
A window will pop up showing a success or failure response.

If any failures are encountered, an error will be written with some detail. If the test is successful, the word "Success!" will show verifying that files can be transferred. This does not test if the user can execute pre/post commands. This needs to be checked manually.



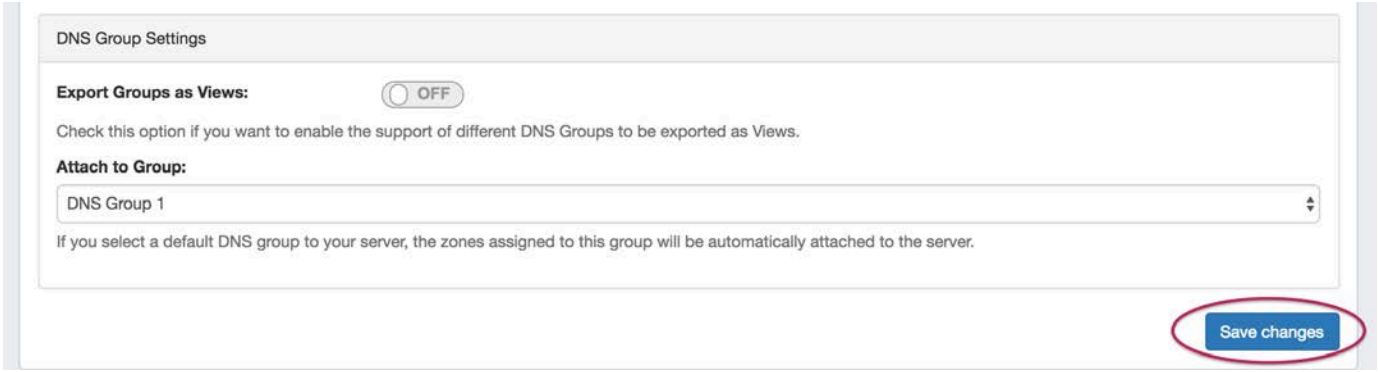
3) Set DNS Group Settings for Server

In the last section, select whether to enable support for exporting DNS Groups as Views (click to toggle ON / OFF), and select a default Group, if desired, to be associated with the server. Zones assigned to the selected Group will automatically be attached to the server.



4) Save Changes

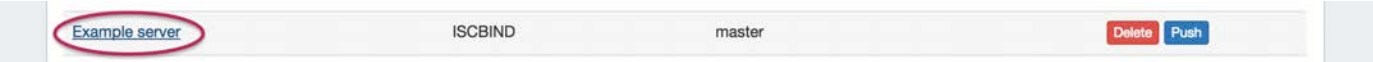
Save your changes when done! Just click the "Save Changes" button at the bottom right of the page.



The new server will now be added to the DNS Servers list. These settings may be changed at any time by selecting the server from the server list and editing the information.

Edit Servers

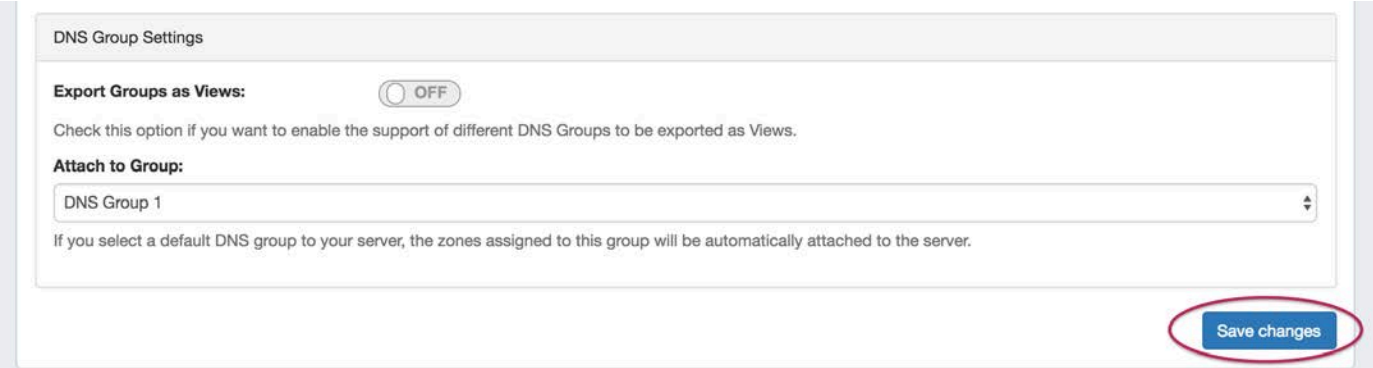
Edit an existing server by clicking once on the server name in the DNS Servers list.



The "Server Settings" page will open.



Click inside the field that you want to change, type your changes, and then click "Save Changes" at the bottom of the page.



Review Zones Connected to a Server

There are two ways that zones may be connected to a DNS server:

1) Directly connected, by attaching the zone to a server from the View Zone page.

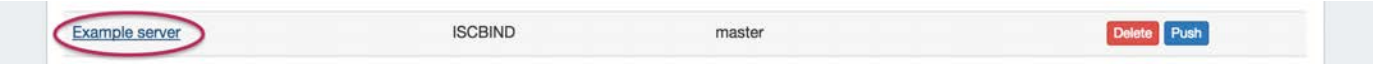
or,

2) Connected by a Group that has been set as the default DNS Group for the server, selected under "DNS Group Settings".

Both are able to be viewed on the DNS Server Settings page.

To view either, open the Server Settings page for the server by clicking on the server name in the DNS Servers list.

Edit an existing server by clicking once on the server name in the DNS Servers list.



The "Server Settings" page will open.

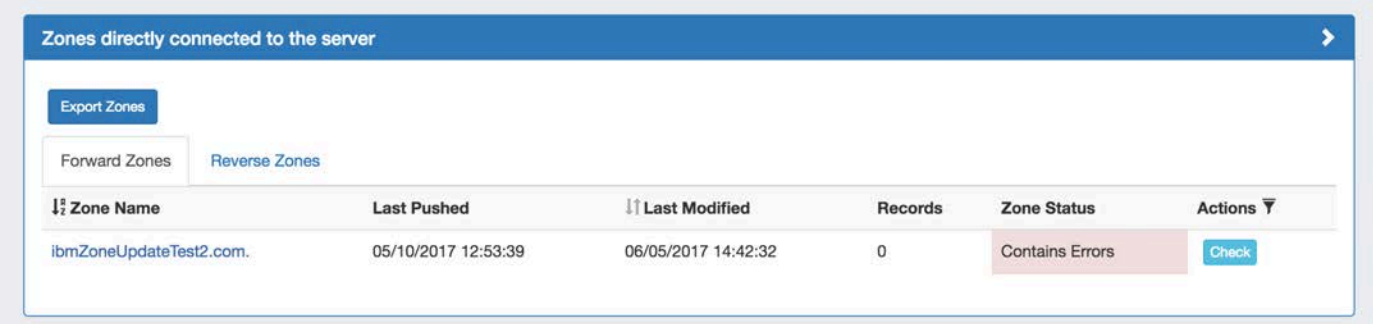
Zones Directly Connected to the Server

Scroll to the bottom of the page, and open the module titled "Zones directly connected to the server" by clicking on the expansion arrow.



A zone list will show the zone(s) that have been directly connected to this server.

Here, you may browse through forward and reverse zones by selecting the "Forward Zones" or "Reverse Zones" tabs, sort the list by Zone Name or Last Modified, open the zone's page by clicking on the name, or check the zone's status by clicking the "Check" button.

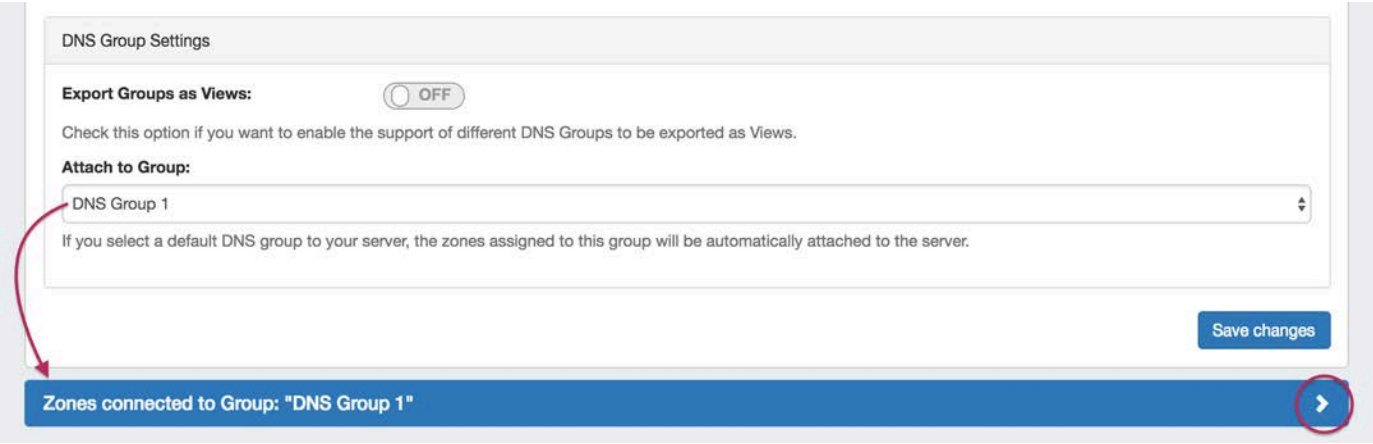


Zones may be exported by clicking the "Export Zones" button.

Zones Connected via a Group

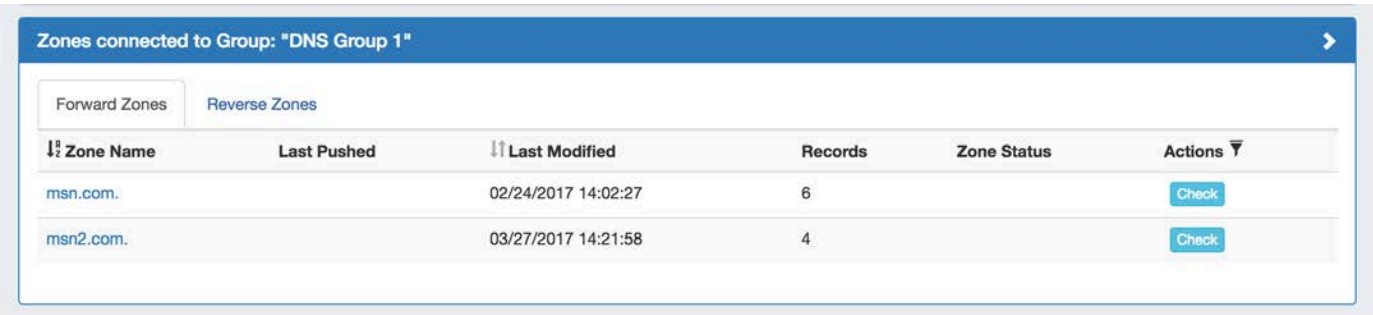
If a default Group has been selected under "DNS Group Settings" for the server, Zones under that Group will be connected to the server and able to be viewed on the Server Settings page.

Scroll to the bottom of the page, and open the module titled "Zones connected to Group '(Group Name)' " by clicking on the expansion arrow.



A zone list will show the zone(s) that connected to this server via a selected Group.

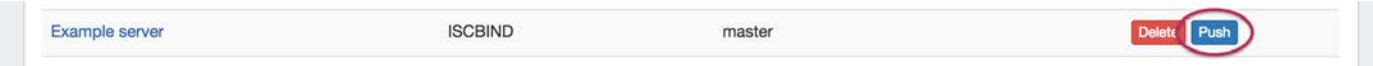
Here, you may browse through forward and reverse zones in that Group by selecting the "Forward Zones" or "Reverse Zones" tabs, sort the list by Zone Name or Last Modified, open the zone's page by clicking on the name, or check the zone's status by clicking the "Check" button.



Pushing a Server

Manual Push

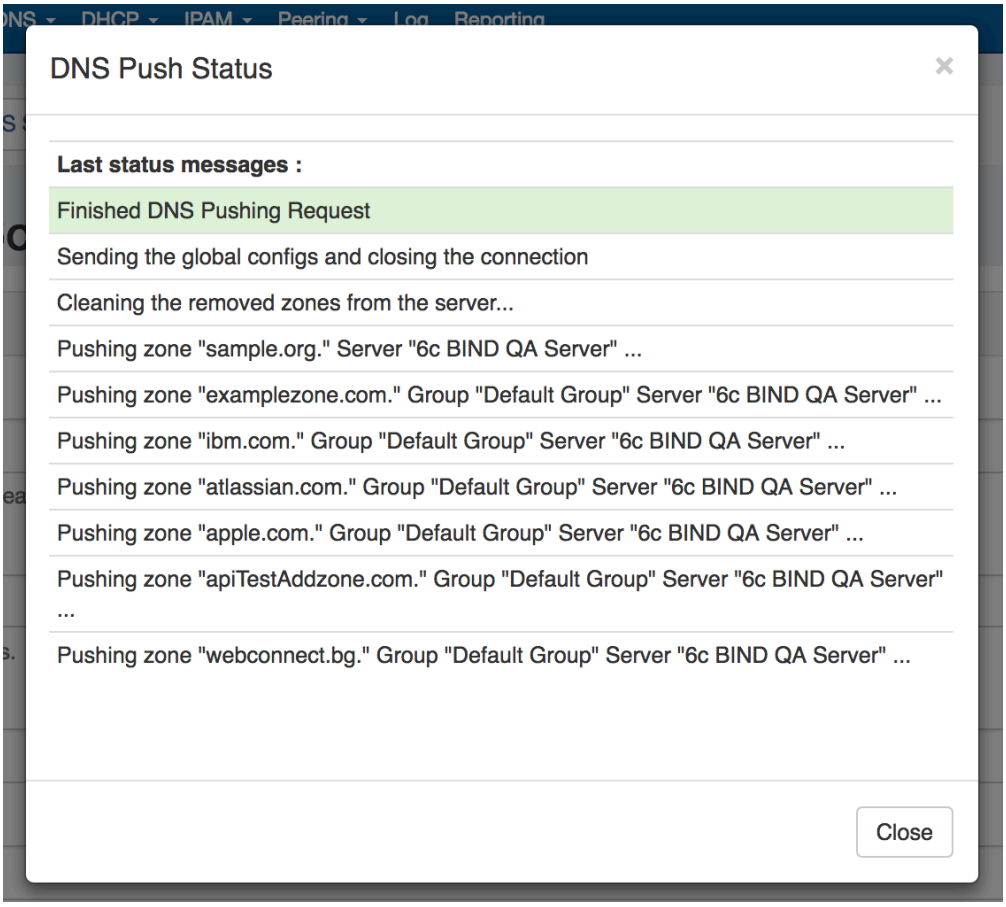
Manually pushing all zones on a server may be done directly from the DNS Server list. Under the "Actions" section of the Server List, click the "Push" button for the desired server.



Pushing may also be done while in the Server Settings page. While in the Server Settings page, click the "Push Zones" button at the top right of the page.



A "DNS Push Status" box will appear, showing the status of each zone as it is pushed. Once all zones have been pushed successfully, a green status message of "Finished DNS Pushing Request" will appear. At this point, the push is complete and the window may be closed.

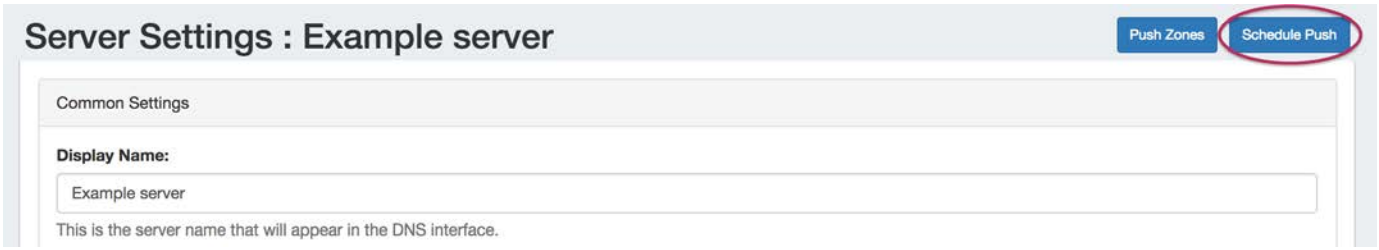


Scheduled Push

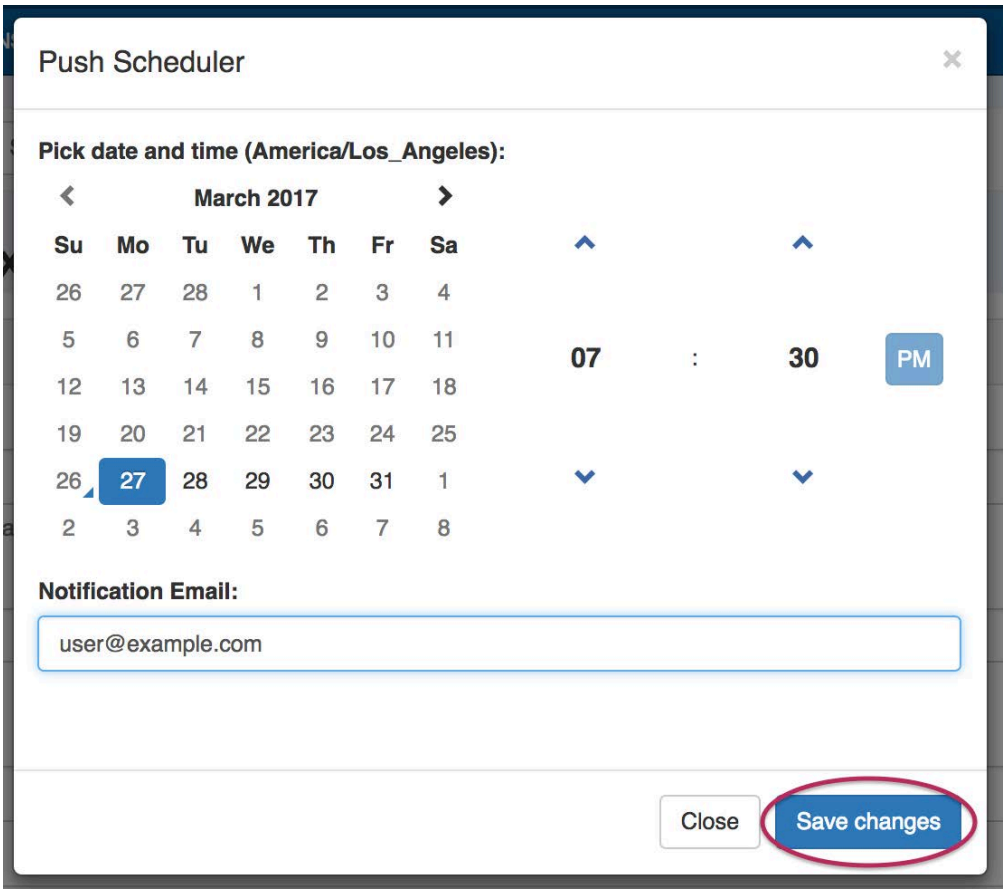
DNS server pushes may be scheduled from either the Admin Area **Scheduler** Tab, or from within the DNSv3 Server Settings page. Scheduled pushes require Admin access.

For information on scheduling a push from the **Scheduler** Tab, see Scheduler documentation.

To schedule a push from a server's Settings page, open the Server Settings page for the desired server, and click on the "Schedule Push" button.

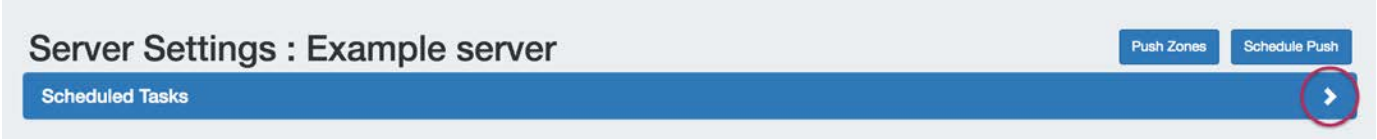


The Push Scheduler dialog will open. Click on the calendar on the left to select a date for the push, set the desired push time on the right, enter a notification email address, and then click "Save Changes".

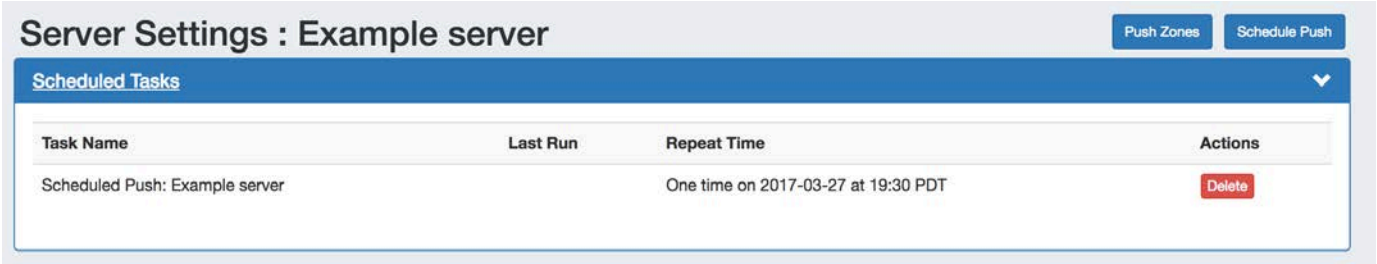


Once a schedule push has been created, a "Scheduled Tasks" module will appear at the top of the Server Settings page.

Click on the expansion arrow for the module to open and view the tasks.



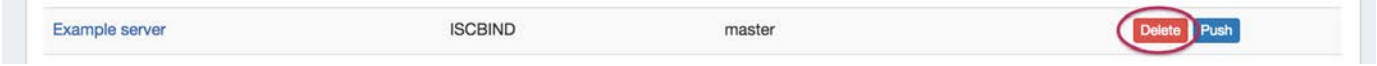
Scheduled pushes for the server will be listed in the "Scheduled Tasks" module, and may be viewed or deleted (by clicking the "Delete" button under "Actions").



If necessary, the Scheduled Push may be edited from the **Scheduler** Tab in the Admin area of ProVision. See the Scheduler for information on editing scheduled tasks.

Delete a Server

Delete a server by clicking the "Delete" button under the "Actions" section of the Server List for the desired server.



Configuring ISC BIND Support

Configuring ISC BIND Support

- Configuring ISC BIND Support
  - Getting Started
  - Adding a BIND Server to ProVision
    - Server Settings
      - 1) Set Server Common Settings
      - 2) Set Server Specific Settings
      - 3) Set DNS Group Settings for Server
      - 4) Save Changes

Getting Started

BIND Version Compatibility

BIND 9 Support is officially supported. Please note that as of May 24, 2018, ISC implemented a different release numbering scheme. From the ISC BIND 9.13 release notes (https://kb.isc.org/article/AA-01612):

Previously new feature development releases were tagged as "alpha" and "beta", leading up to the first stable release for a given development branch, which always ended in ".0".

Now, however, BIND has adopted the "odd-unstable/even-stable" release numbering convention. There will be no "alpha" or "beta" releases in the 9.13 branch, only increasing version numbers. So, for example, what would previously have been called 9.13.0a1, 9.13.0a2, 9.13.0b1, and so on, will instead be called 9.13.0, 9.13.1, 9.13.2, etc.

The first stable release from this development branch will be renamed as 9.14.0. Thereafter, maintenance releases will continue on the 9.14 branch, while unstable feature development proceeds in 9.15.

You will need a user who can log in to the DNS server and make changes to the directory in which the zones are being stored. Additionally, it is often useful for this user to have the ability to restart the DNS server. The login and password for this user will be required to configure this server on the DNS Admin page.

6connect Zone files are written out in the following format:

/path/to/zone/directory/viewName/zoneFirstLetter/zonefile.zone

If no views are configured, or if views are expressly disabled, then the default viewName "6connectGeneric" is used. The zoneFirstLetter is the first letter of the zone name, so the subdirectory 'microsoft.com.zone' is placed in would be /m/.

All 6connect-managed Zones are managed by a dedicated 6connect configuration file named 6connect_named.conf. This file is created to act a supplementary conf file to work in concert with any existing named.conf which might exist. To include the 6connect configuration file, edit named.conf and append the following line:

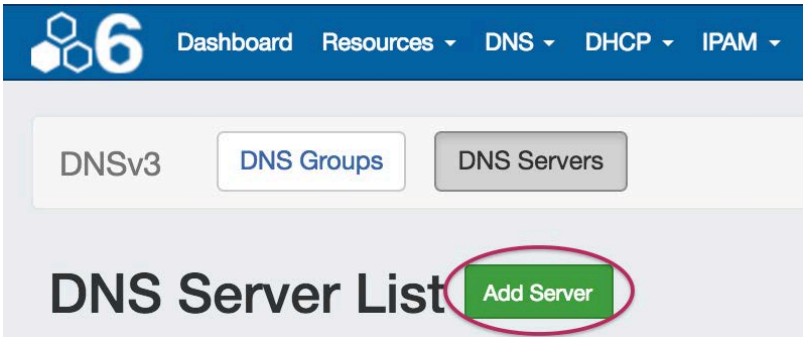
include "/path/to/conf/directory/6connect_named.conf";

You must remember to include the 6connect configuration file or none of the changes managed by 6connect ProVision will take effect!

It is also important to note that if your existing named.conf file contains zones within Split Horizon views, then the 6connect-managed zones must also be view-enabled. Likewise, if existing zones are not grouped into views, then views must be disabled on ProVision.

Adding a BIND Server to ProVision

To create a new server, start from the **DNS** Tab, select the **DNS Servers** sub menu. Then, click the "Add Server" button next to "DNS Server List".

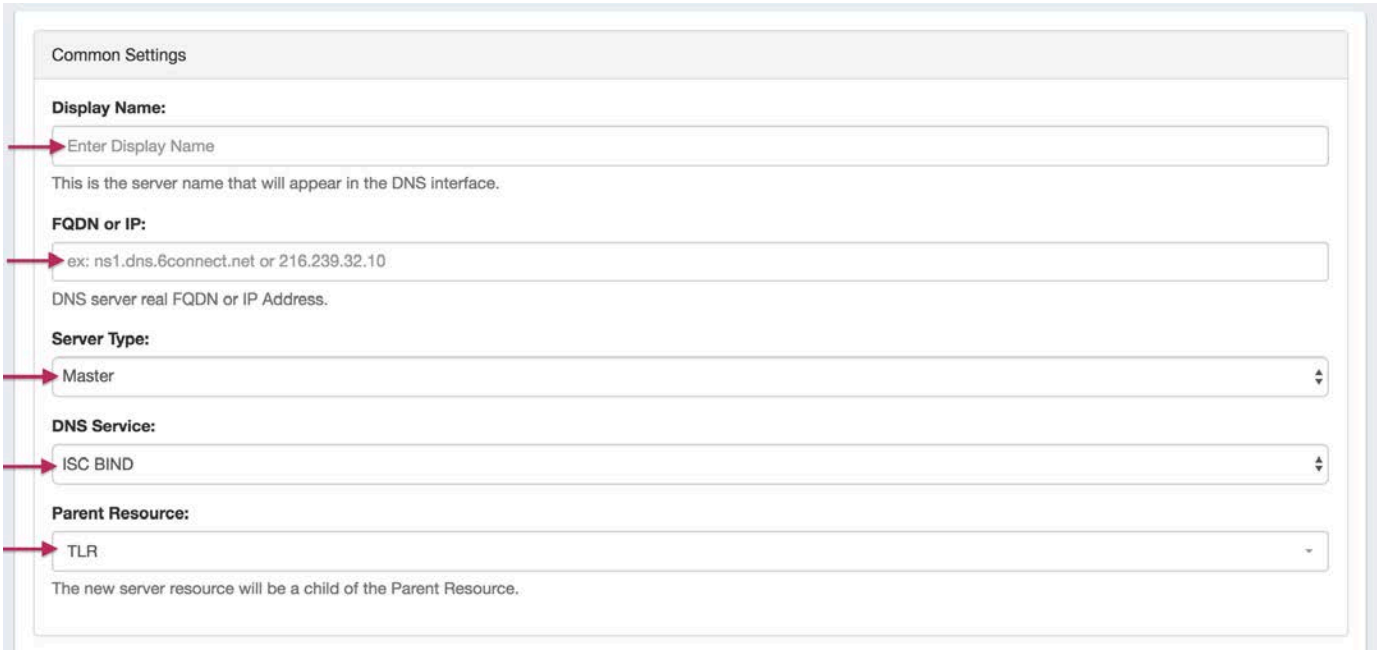


This will open the "Server Settings" page.

Server Settings

1) Set Server Common Settings

In the "Common Settings" section of Server Settings, enter the new server's Display Name (the name that will appear on the ProVision interface), the FQDN / IP, server type, service type, and desired parent Resource (may be left at the default Top Level Resource).

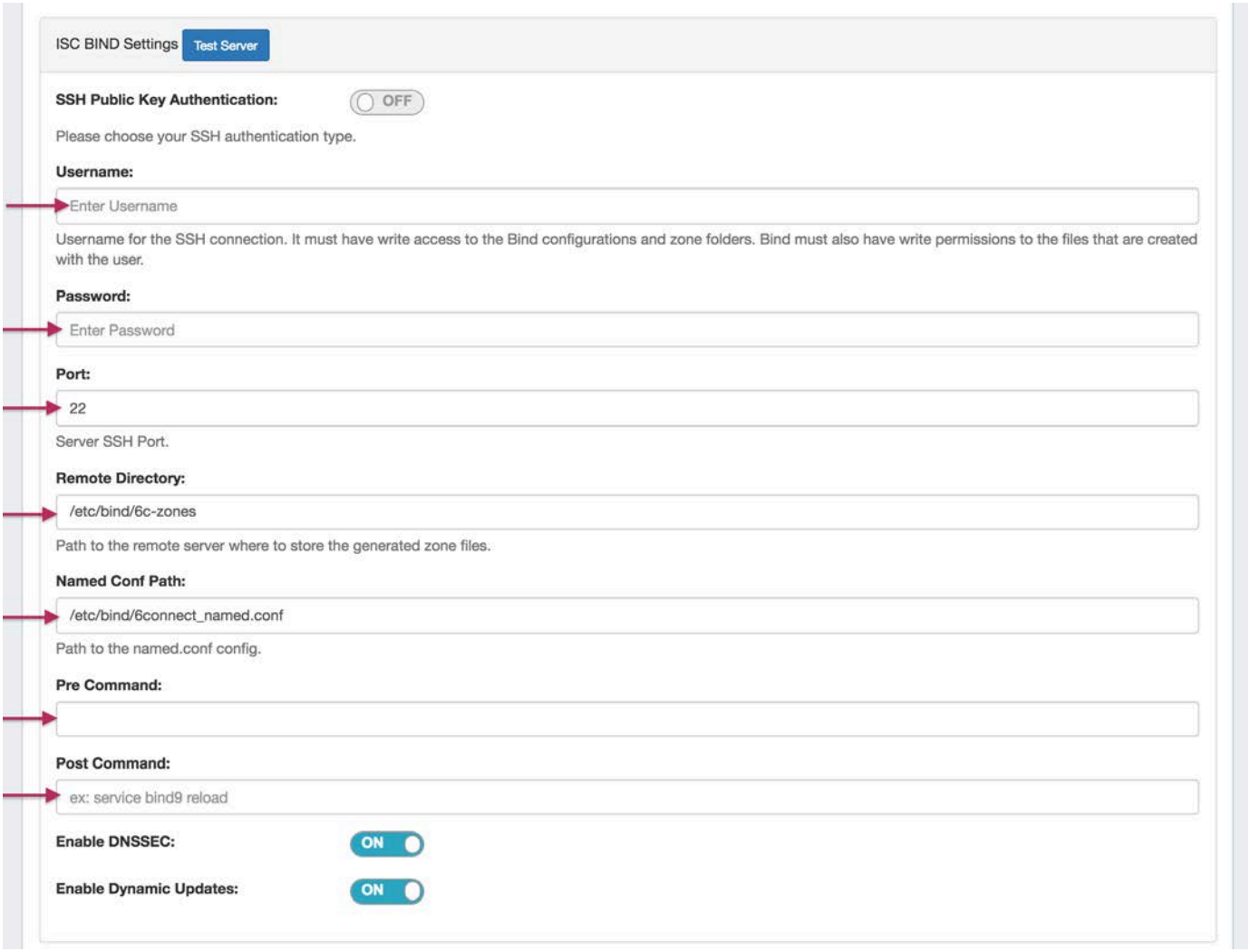


2) Set Server Specific Settings

The next section is entering server service-type specific settings. The options visible in this section will depend on the "DNS Service" type chosen under "Common Settings".

Here, we see the fields for ISC BIND server settings. Enter the server Username, Password, Port, Remote Director, Named Conf. Path, and Pre /Post Command (if desired). Your fields may vary for other server types.

For SSH Public Key Authentication, DNSSEC, and Dynamic Option updates, click on the ON / OFF toggle to select "ON" or "OFF" for each as needed.



After entering the server-specific settings in this section, you can click the "Test Server" button at the top of the section to test the server connection and authentication.

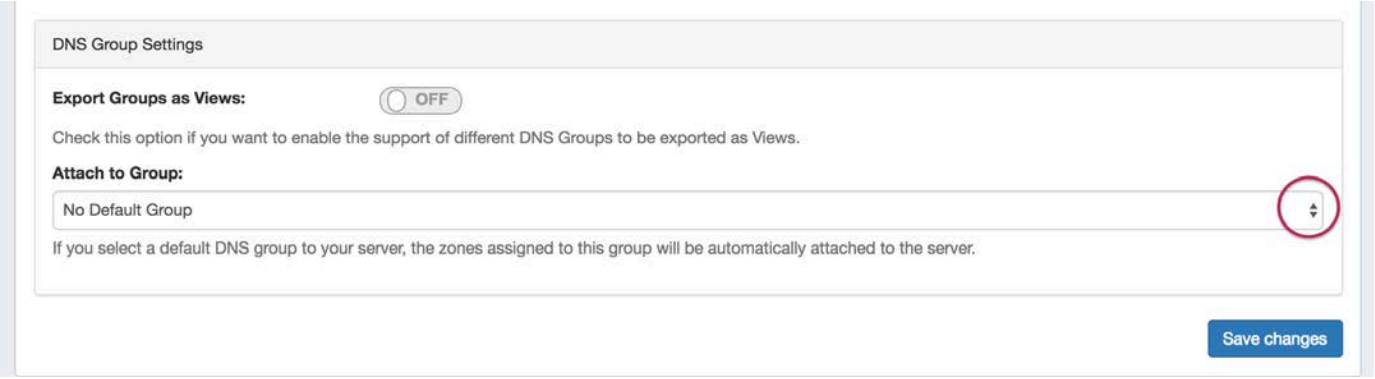


A window will pop up showing a success or failure response.



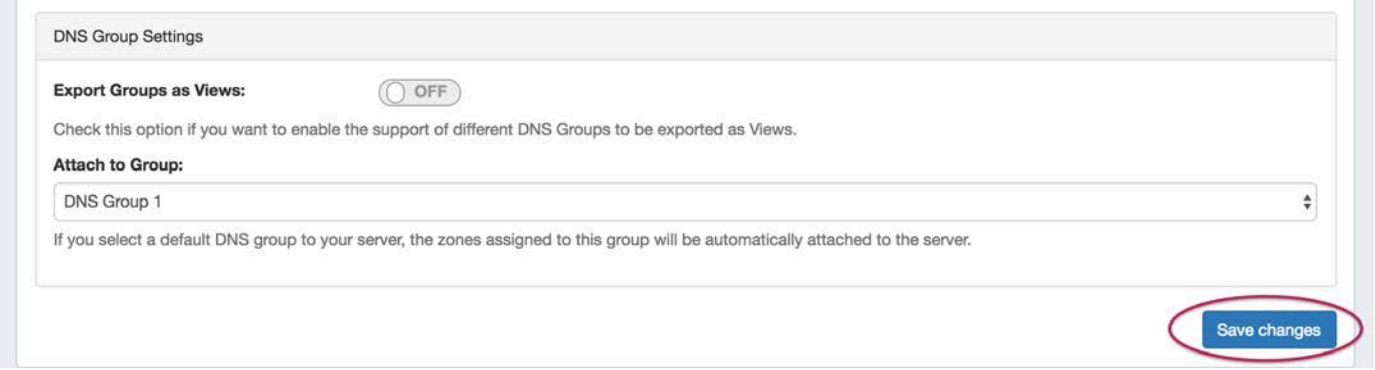
3) Set DNS Group Settings for Server

In the last section, select whether to enable support for exporting DNS Groups as Views (click to toggle ON / OFF), and select a default Group, if desired, to be associated with the server. Zones assigned to the selected Group will automatically be attached to the server.



4) Save Changes

Save your changes when done! Just click the "Save Changes" button at the bottom right of the page.



The new server will now be added to the DNS Servers list. These settings may be changed at any time by selecting the server from the server list and editing the information.

Configuring PowerDNS Support

Configuring PowerDNS Support

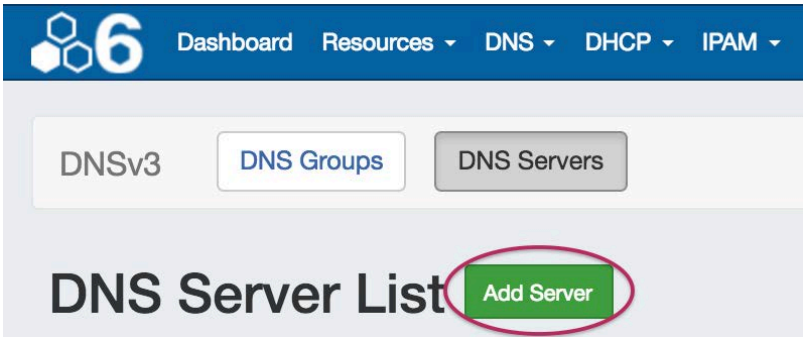
- Configuring PowerDNS Support
  - Environments supported
  - Step 1: Add a PowerDNS Server
    - Server Settings
      - 1) Set Server Common Settings
      - 2) Set Server Specific Settings
      - 3) Set DNS Group Settings for Server
      - 4) Save Changes
  - Step 2: Import PowerDNS Zones
  - Step 3: Push zones to PowerDNS
  - BIND Backend
  - MySQL Backend

Environments supported

- PowerDNS version 3.0 or above on the target server(s)
- BIND or MySQL backend

Step 1: Add a PowerDNS Server

To create a new server, start from the **DNS** Tab, select the **DNS Servers** sub menu. Then, click the "Add Server" button next to "DNS Server List".

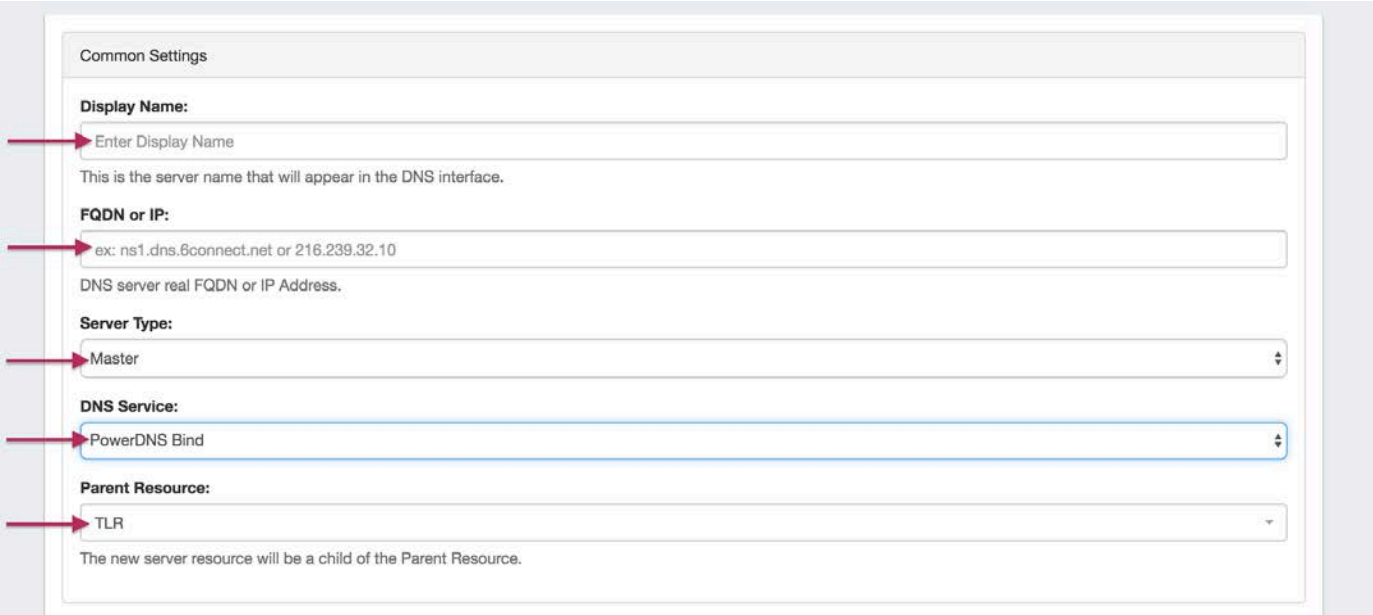


This will open the "Server Settings" page.

Server Settings

1) Set Server Common Settings

In the "Common Settings" section of Server Settings, enter the new server's Display Name (the name that will appear on the ProVision interface), the FQDN / IP, server type, DNS service type, and desired parent Resource (may be left at the default Top Level Resource). For PowerDNS servers, ensure that either "PowerDNS BIND" or "PowerDNS MySQL" is selected under DSN server type.

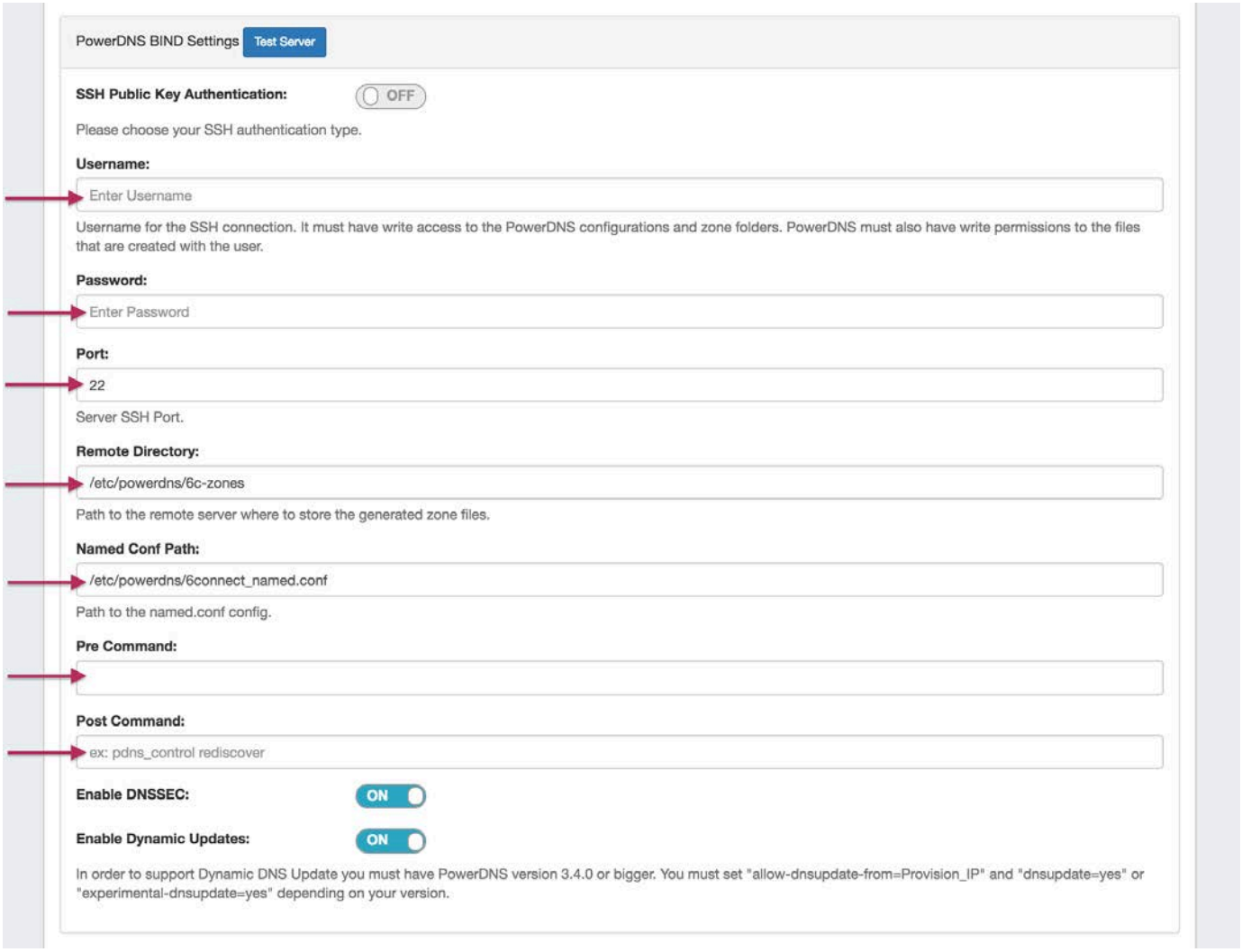


2) Set Server Specific Settings

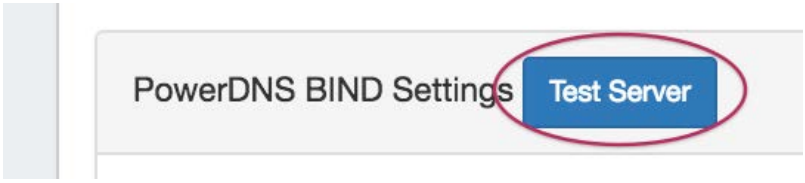
The next section is entering server service-type specific settings. The options visible in this section will depend on the "DNS Service" type chosen under "Common Settings".

Here, we see the fields for PowerDNS BIND server settings. Enter the server Username, Password, Port, Remote Director, Named Conf. Path, and Pre/Post Command (if desired). Your fields may vary for other server types.

For SSH Public Key Authentication, DNSSEC, and Dynamic Option updates, click on the ON / OFF toggle to select "ON" or "OFF" for each as needed.



After entering the server-specific settings in this section, you can click the "Test Server" button at the top of the section to test the server connection and authentication.

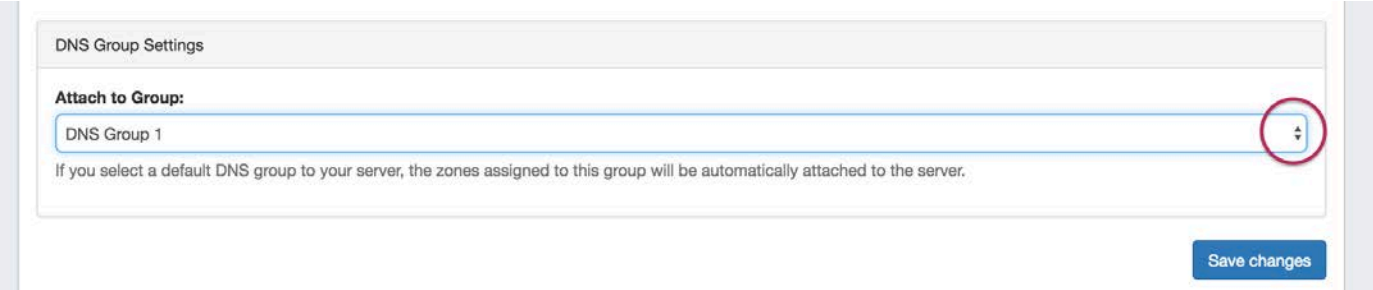


A window will pop up showing a success or failure response.



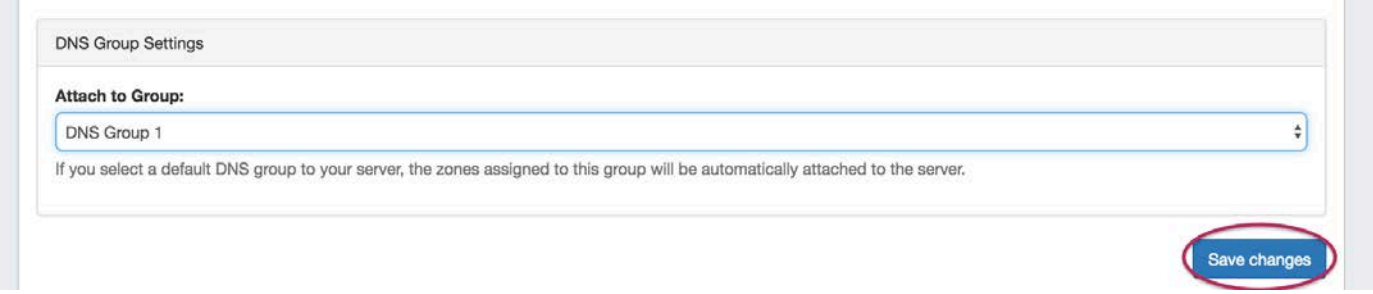
3) Set DNS Group Settings for Server

In the last section, select a default Group to be associated with the server. Zones assigned to the selected Group will automatically be attached to the server.



4) Save Changes

Save your changes when done! Just click the "Save Changes" button at the bottom right of the page.



The new server will now be added to the DNS Servers list. These settings may be changed at any time by selecting the server from the server list and editing the information.

Step 2: Import PowerDNS Zones

While in the **Admin** section, navigate to the **Data Import** Tab. Select the "Power DNS Zone Import" link.

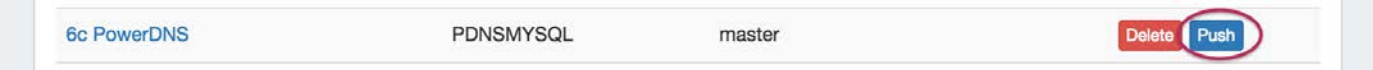
To import your data, simply choose your PowerDNS server and click "Import".



Step 3: Push zones to PowerDNS

Navigate back to the **DNS** tab, and select the "**DNS Servers**" tab.

Locate the PowerDNS server in the DNS Servers list, and then click the "Push" button under "Actions" at the end of the row.



BIND Backend

Note on SSH

The integration does not require a remote database connection, but it does require an SSH account and a writable directory. The SSH account must have access to the server. This account will also be used for DNSSEC functionality within PowerDNS.

MySQL Backend

Note on SSH

The integration requires a remote database connection, so will need a mysql user with permissions for remote administration. We highly recommend using ACLs to ensure that configuration only occurs from intended sources.

For DNSSEC functionality, you will need a standard SSH user account withing your PowerDNS user group

Please note that Views are not supported with the MySQL backend
Only BIND and MySQL backends are supported.

Configuring Secure64 Support

Configuring Secure64 Support

- Configuring Secure64 Support
  - Step 1: Create an nsd.conf file under the root directory / of your S64 Auth server
  - Step 2: Make a directory for 6connect ProVision to push zone files to on the Secure64 DNS Server
  - Step 3: Setup and Configure 6connect ProVision for your Secure64 DNS Server
    - Server Settings
      - 1) Set Server Common Settings
      - 2) Set Server Specific Settings
      - 3) Set DNS Group Settings for Server
      - 4) Save Changes
  - Step 4: Associate zones to your Secure64 DNS Server(s)
  - Step 5: Push Zones to Secure64 Server(s)
  - Step 6: Verify DNS Zone push on Secure64 Server(s)
  - Step 7: Validate Zone data in Your Infrastructure

A note on Ports

ProVision uses port 22 to communicate with and configure Secure64 infrastructure - please ensure that this is addressed in any ACLs/firewalls

ProVision also uses port 53 to do zone checks if the DNS Module is enabled and in use. Please ensure that your Secure64 infrastructure is configured to accept DNS lookups from the ProVision server

The initial setup of the Secure64 Authoritative server is as follows:

Step 1: Create an nsd.conf file under the root directory / of your S64 Auth server

DO THIS

Make sure to add the line include: 6connect_nsd.conf to the nsd.conf file

Output/Input

```
[authdnsadmin@Secure64DNS]# cat nsd.conf
server:
ip-address: 50.198.192.141

axfr-logfile: /axfr_log/axfr.log
axfr-logfile-flush-count: 1
axfr-logfile-max-size: 100000
axfr-logfile-max-size: 10

request-logfile: /request_log/request.log
request-logfile-flush-count: 10
request-logfile-max-size: 1000000
request-logfile-max-files: 10

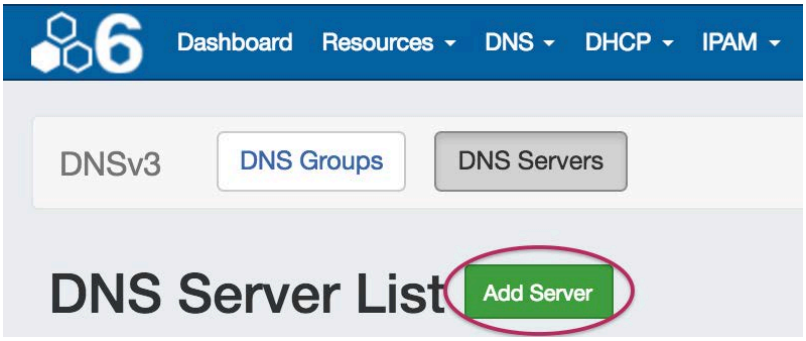
include: 6connect_nsd.conf
```

Step 2: Make a directory for 6connect ProVision to push zone files to on the Secure64 DNS Server

```
[authdnsadmin@Secure64DNS]# mkdir test12
[authdnsadmin@Secure64DNS]# ls
/:
322 2013-08-19 06:07:42 nsd.conf
<DIR> 1024 2013-08-16 17:30:12 test12
```

Step 3: Setup and Configure 6connect ProVision for your Secure64 DNS Server

To create a new server, start from the DNS Tab, select the **DNS Servers** sub menu. Then, click the "Add Server" button next to "DNS Server List".

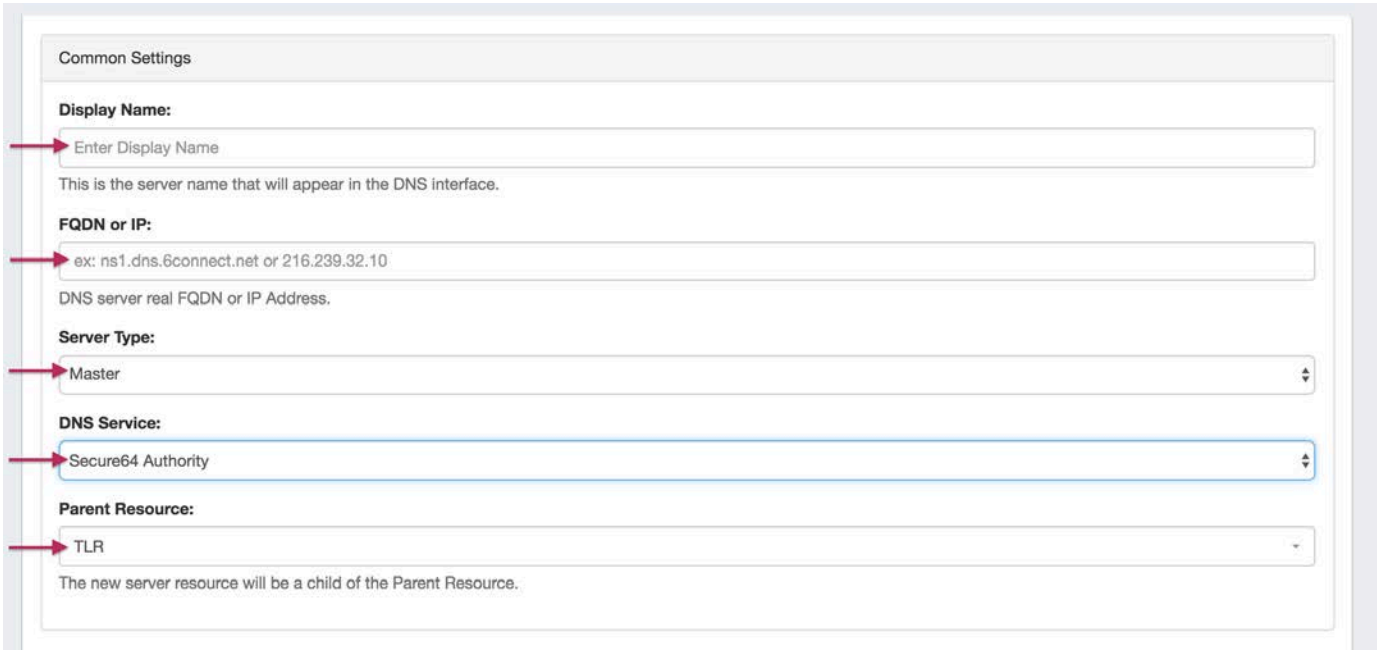


This will open the "Server Settings" page.

Server Settings

1) Set Server Common Settings

In the "Common Settings" section of Server Settings, enter the new server's Display Name (the name that will appear on the ProVision interface), the FQDN / IP, server type, DNS service type, and desired parent Resource (may be left at the default Top Level Resource). For Secure64 servers, ensure that DNS Service is set to "Secure64 Authority", "Secure64 x86 Authority", "Secure64 KNOT Authority", or "Secure64 Cache".



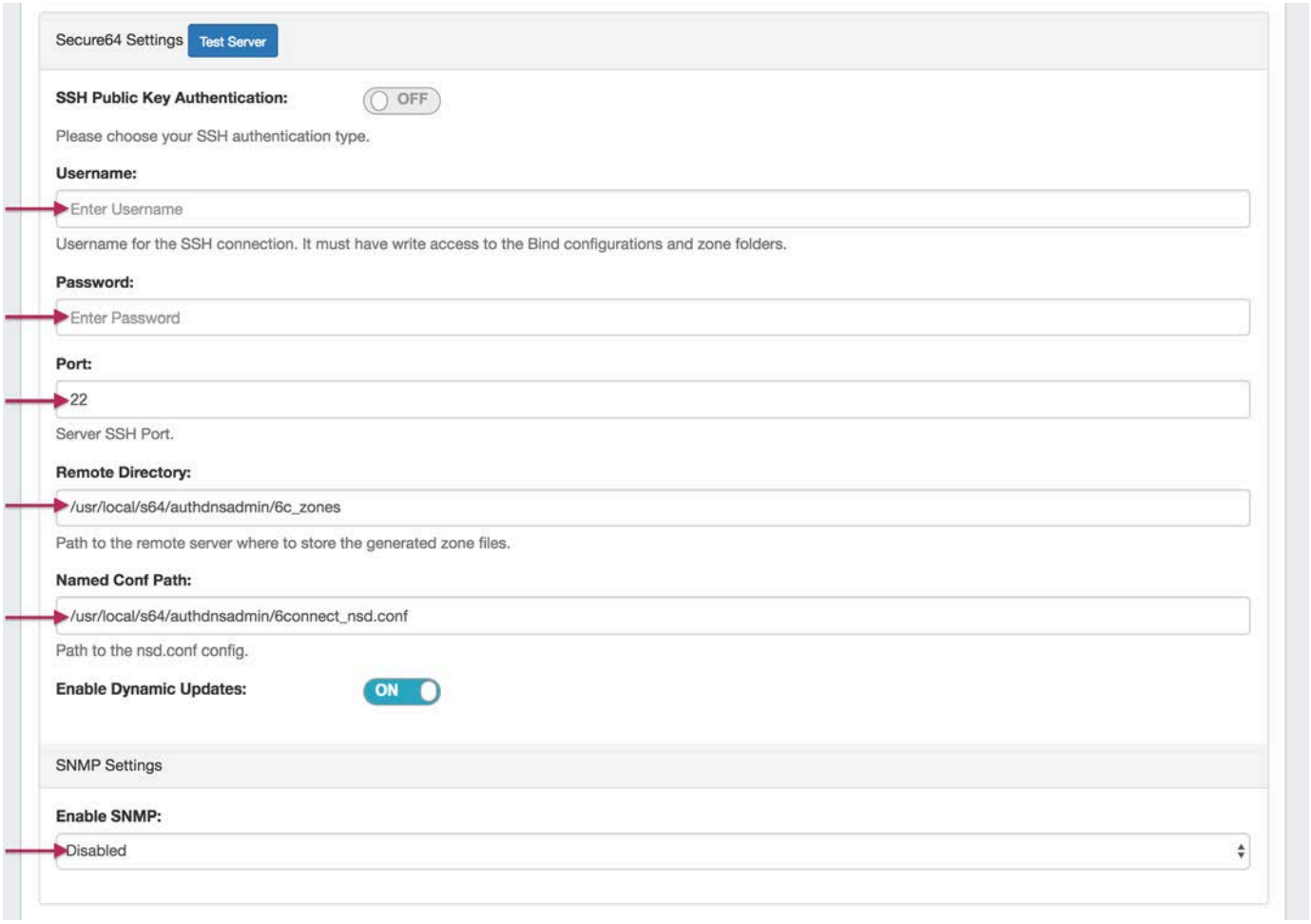
2) Set Server Specific Settings

The next section is entering server service-type specific settings. The options visible in this section will depend on the "DNS Service" type chosen under "Common Settings".

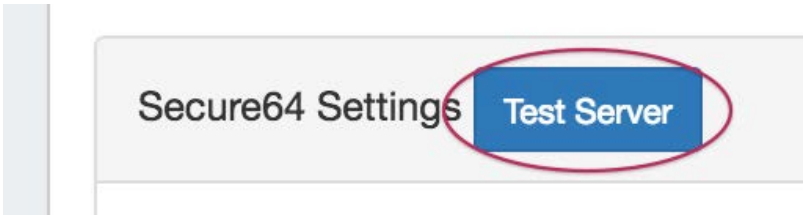
Here, we see the fields for Secure64 server settings.

Enter the server Username, Password, Port, Remote Directory, Named Conf. Path, and whether to enabled SNMP.

For SSH Public Key Authentication and Dynamic Option updates, click on the ON / OFF toggle to select "ON" or "OFF" for each as needed.



After entering the server-specific settings in this section, you can click the "Test Server" button at the top of the section to test the server connection and authentication.

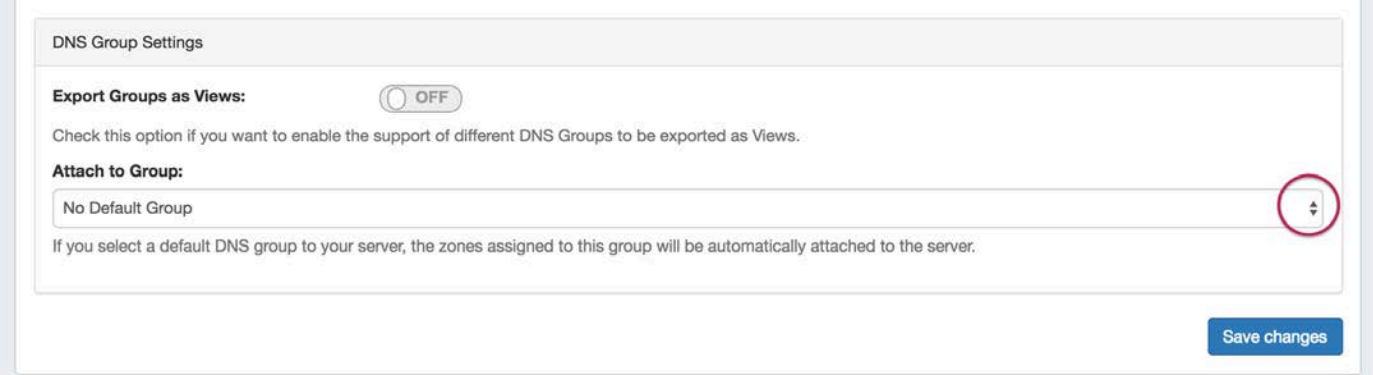


A window will pop up showing a success or failure response.



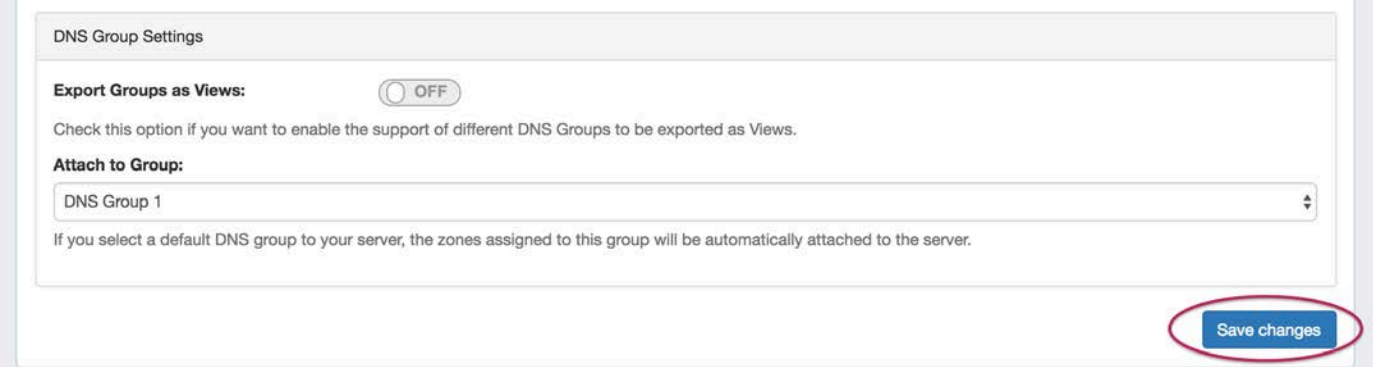
3) Set DNS Group Settings for Server

In the last section, select whether to enable support for exporting DNS Groups as Views (click to toggle ON / OFF), and select a default Group, if desired, to be associated with the server. Zones assigned to the selected Group will automatically be attached to the server.



4) Save Changes

Save your changes when done! Just click the "Save Changes" button at the bottom right of the page.

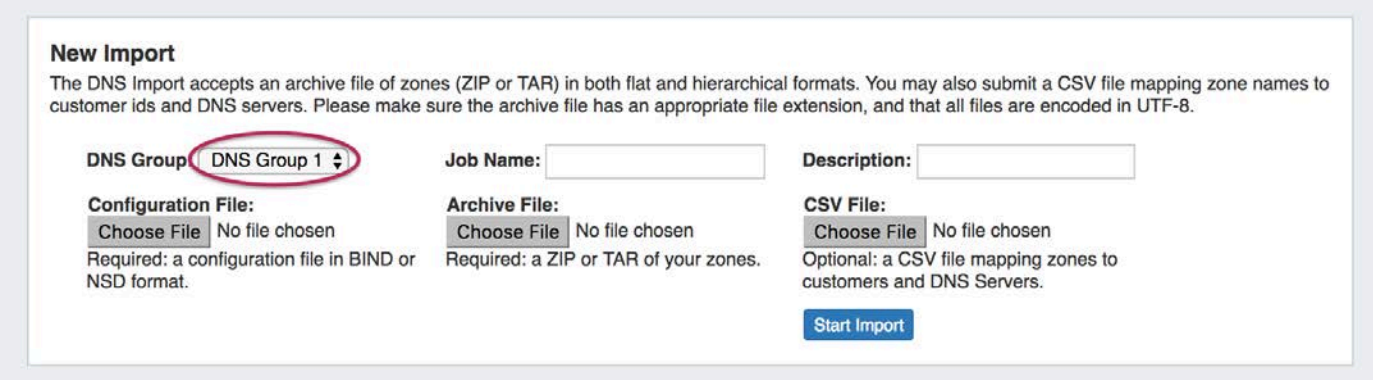


Step 4: Associate zones to your Secure64 DNS Server(s)

If a default Group was selected while setting up the S64 server, then any zones in that Group will automatically be associated with the server.

Zones may be manually added, moved, or imported into the Group - see Working with DNS Zones - Common Tasks and Working with DNS Groups for details on associating zones with Groups.

If zones are to be imported, a DNS Group may be selected during the import process to associate with the zones.

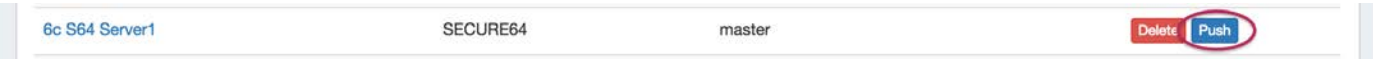


Select the group specified as the default S64 server Group, then import the zones as described in the Import DNS Zones documentation.

Step 5: Push Zones to Secure64 Server(s)

Navigate back to the **DNS** tab, and select the "**DNS Servers**" tab.

Locate the Secure64 server in the DNS Servers list, and then click the "Push" button under "Actions" at the end of the row.



Step 6: Verify DNS Zone push on Secure64 Server(s)

The result of the Push can be checked/verified by checking the Secure64 server as follows:

Verifying Zone pushes

ssh to 50.198.192.141
Login using the designated login account and password
Enable cachednsadmin
ls

Now, verify that the "788 2013-08-21 12:35:04" 6connect_nsd.conf file now exists.

```
[authdnsadmin@eval138.secure64.com]# ls
/:
6728 2013-08-13 00:15:30 nsd.conf
8416071 2013-08-21 12:35:07 nsd.db
788 2013-08-21 12:35:04 6connect_nsd.conf
<DIR> 1024 2013-08-21 12:34:50 test12
```

You can verify the Push contents by doing a cat of the 6connect_nsd.conf

[authdnsadmin@Secure64DNS]# cat 6connect_nsd.conf

AutoGenerated by 6connect ProVision. Do not manually edit.

zone:

name: atestzone.com

zonefile: /test12/6connectGeneric/m/atestzone.com.zone

zone:

name: Testzone2.com

zonefile: /test12/6connectGeneric/m/Testzone2.com.zone

In the example above, two Zones have transferred.

To look at the contents of each zone you can cd to the proper directory /test12/6connectGeneric and find the zone files in an alphabetical directory structure as follows:

[authdnsadmin@Secure64DNS]# cd 6connectGeneric

[authdnsadmin@Secure64DNS]# cd test12

changed to test12
[authdnsadmin@Secure64DNS]# ls
/test12/:
<DIR> 1024 2013-08-16 19:43:21 6connectGeneric
[authdnsadmin@Secure64DNS]# cd 6connectGeneric
changed to 6connectGeneric
[authdnsadmin@Secure64DNS]# ls
/test12/6connectGeneric/:
<DIR> 1024 2013-08-16 17:30:13 e
<DIR> 1024 2013-08-16 17:30:16 m
<DIR> 1024 2013-08-16 18:49:21 d
<DIR> 1024 2013-08-16 19:43:23 s
[authdnsadmin@Secure64DNS]# cd m
changed to m
[authdnsadmin@Secure64DNS]# ls
/test12/6connectGeneric/m/:
[authdnsadmin@eval138.secure64.com]# ls
5192 2013-08-21 15:35:01 atestzone.com.zone

6758 2013-08-21 15:35:02 Testzone2.com.zone
[authdnsadmin@Secure64DNS]#

Step 7: Validate Zone data in Your Infrastructure

Finally, do a **dig** of the zones to verify the DNS configuration has been successfully deployed.

Using dig to validate your Secure64 Server installation

[authdnsadmin@eval138.secure64.com]# dig @50.198.192.141 atestzone.com
; <<>> DiG SourceT 3.x <<>> @50.198.192.141 atestzone.com
;; Got answer:
;; ~~>>HEADER<<~~ opcode: QUERY, status: NOERROR, id: 59591
;; flags: qr aa rd; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0
;; QUESTION SECTION:
;atestzone.com. IN A
;; AUTHORITY SECTION:
atestzone.com. 3600 IN SOA ns1.dns.6connect.net. hostmaster.6connect.net. (2013082102 10800 3600 604800 38400)
[authdnsadmin@eval138.secure64.com]#

For any questions regarding the integration of Secure64 products into 6connect ProVision, please email 6connect at support@6connect.com, or Secure64 at support@secure64.com

Changing Secure64 Server IP addresses

When you setup ProVision to communicate via SSH to a Secure64 server, a ley/fingerprint is saved to the local hosts file. If you modify the IP address, but do not clear out the hosts file, then ProVision will think this is an attack and prevent communication with the Secure64 server.

To rectify this issue, you may need to reset the host file so that a new SSH host key can be added for the IP address. To do this manually, please follow the following steps:

1) The admin needs to login to the ProVision server via SSH/CLI
2) Open the file “known” in the /tmp folder in your preferred editor (vi, etc.)
3) Delete the line in the file with the server IP/fingerprint
4) Save the changes and exit the editor

To verify the functionality - attempt to connect to the DNS Server(s) using the “Test Server” button from the ProVision GUI

If you have any issues, please contact 6connect support per your Support Agreement/Plan.

Configuring Split Horizon/Views

Configuring ACL Views

ProVision's ACL module applies the specified network lists to the in the named.conf for zones inside the group and the servers that belongs to the group, when exporting the group or zone.

- Configuring ACL Views
  - Create an ACL View
  - Edit an ACL
  - Delete an ACL
  - Add an existing ACL to another Group
  - Additional Information

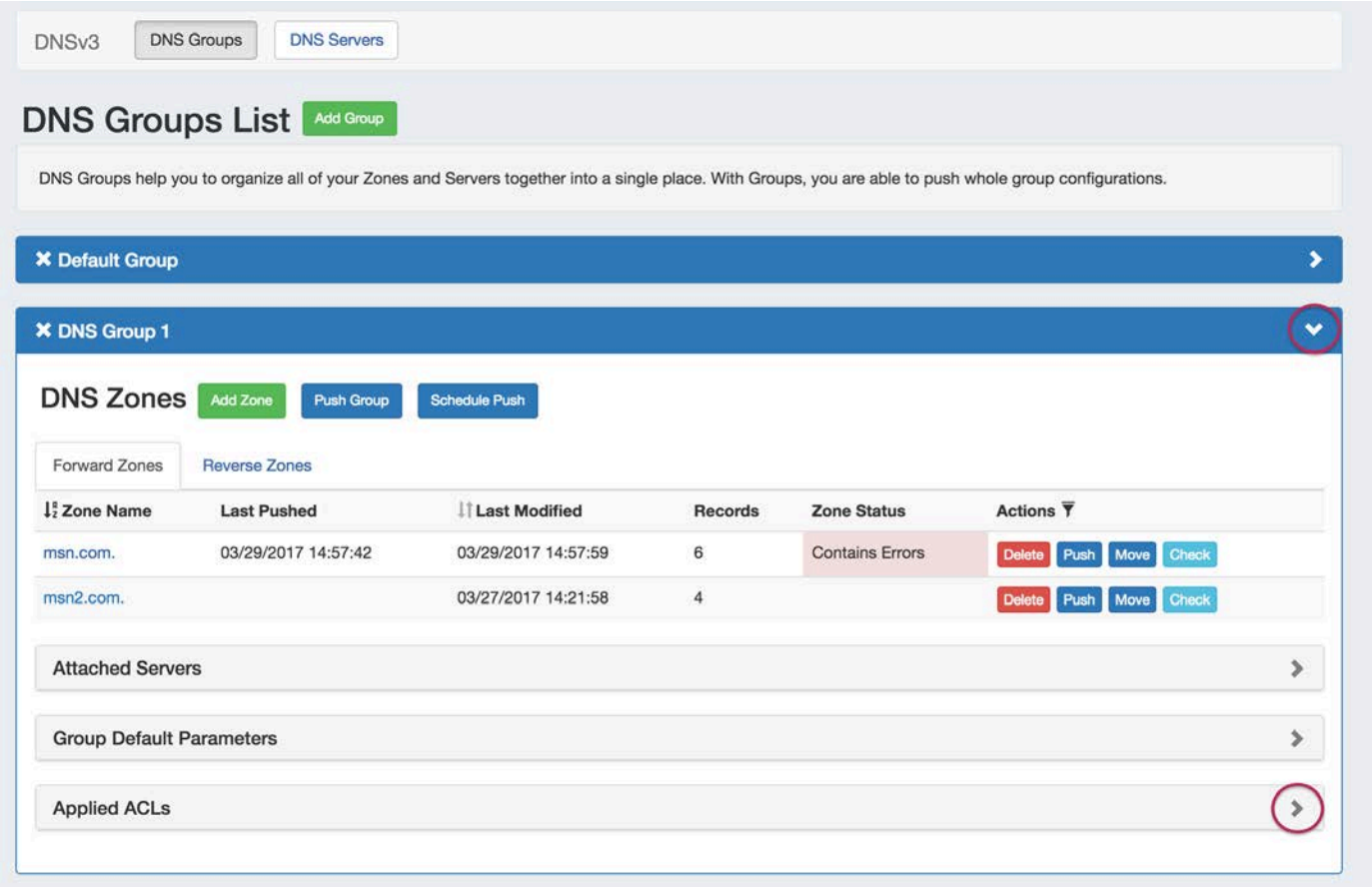
WARNING

If you see a view named "_6connectDefault" - DO NOT DELETE IT.

Create an ACL View

ACL views may be created under each DNS Group, accessed from the **DNS** tab **DNS Groups** sub-tab.

Expand the Group that you want the view applied to, and then expand the section "Applied ACLs" by clicking on the expansion arrow to the right of the section name.

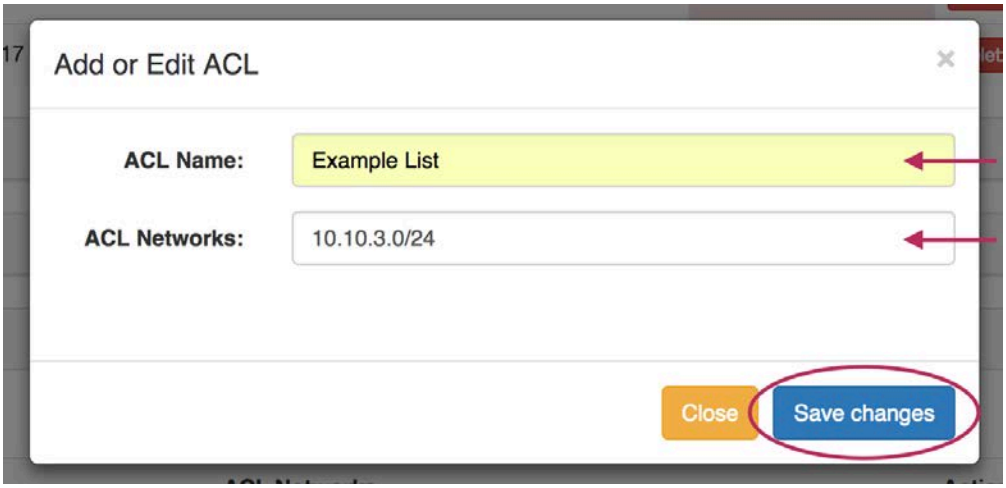


If ACLs already exist, they will be shown under this section.

To add a new list, click the "Add or Edit ACL" button.

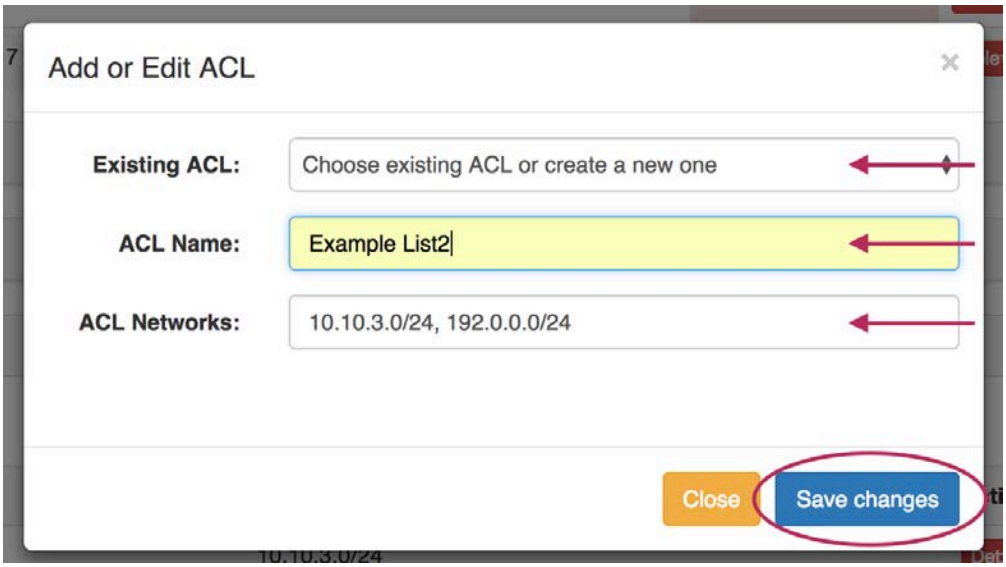


If no ACLs currently exist, the dialog will only show inputs for ACL Name and ACL Networks - enter a Name and Network and click "Save Changes".



If previous ACLs exist, a selector will be be available for "Existing ACL:".

Since we are creating a new ACL, just keep this at the default value ("Choose Existing ACL or create a new one"), and enter in the Name and Network(s) for the new ACL. Separating multiple CIDRs with a comma (,).



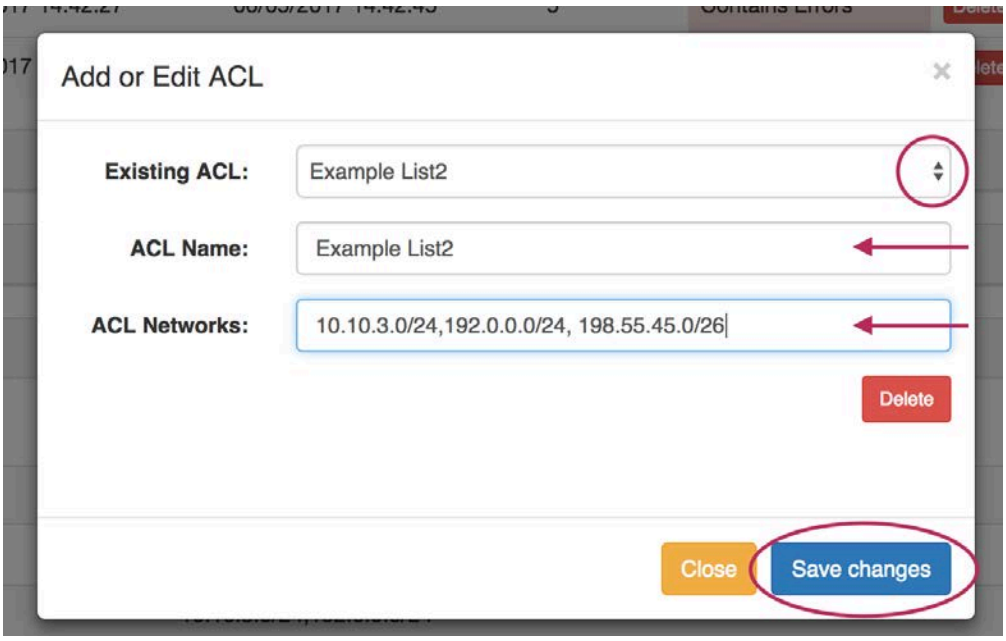
When done, save the ACL by clicking "Save changes". The new ACL will be added to the Group's Applied ACL list.



Edit an ACL

ACLs may be edited similarly to how they are created. Click the "Add or Edit ACL" button under the "Applied ACLs" module for the DNS Group.

Select the ACL you wish to edit from the "Existing ACL" Selector. Then, edit the Name or Network fields as desired.



When done, click "Save Changes".

Detach an ACL from a Group

Detaching an ACL removes it from the Group, but the ACL will still exist in ProVision for re-use in other Groups.

To detach an ACL, expand the "Applied ACLs" section under the desired Group, and click on the "Detach" button under the "Actions" column.

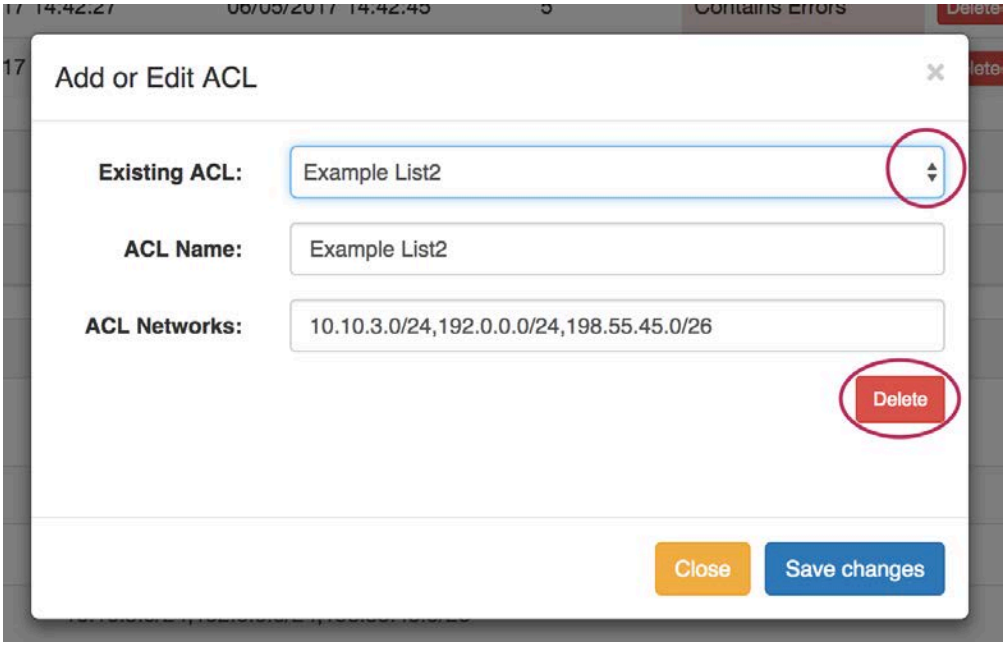


The ACL will be removed from under the Group, but will remain in other Groups it exists under, as well as be selectable from the "Add ACL" window.

Delete an ACL

ACLs may be deleted through the "Edit ACL" interface. Click the "Add or Edit ACL" button under the "Applied ACLs" module for the DNS Group.

Select the ACL you wish to delete from the "Existing ACL" Selector. Then, click the red "Delete" button under the Networks field.

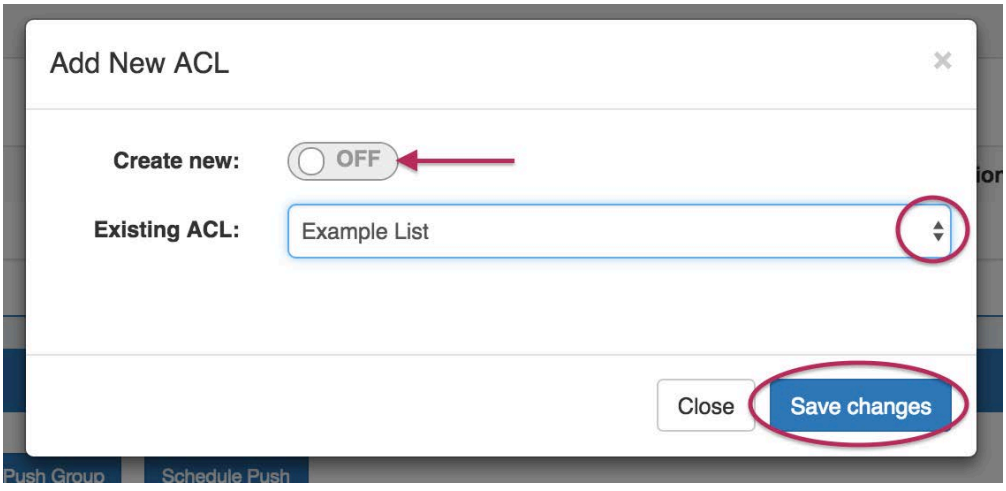


The ACL will be permanently deleted, and will no longer show under any Groups to which it has been attached, nor show in the "Existing ACLs" list.

Add an existing ACL to another Group

Existing ACLs may be added to as many Groups as desired. If ACLs already exist in ProVision, clicking the "Add ACL" button will give an additional option to "Create New".

To select a previously created ACL, turn the "Create New" toggle to the "OFF" position.



An "Existing ACL" list appears. Select the desired existing ACL from the list, and then click "Save changes" when done.

Additional Information

For more information on DNS Groups, Servers, and Zones, see:

DNS Tab

Working with DNS Groups

Working with DNS Zones - Common Tasks

DNS Administration

Working with DNS Servers

Configuring DNSSEC

Configuring DNSSEC

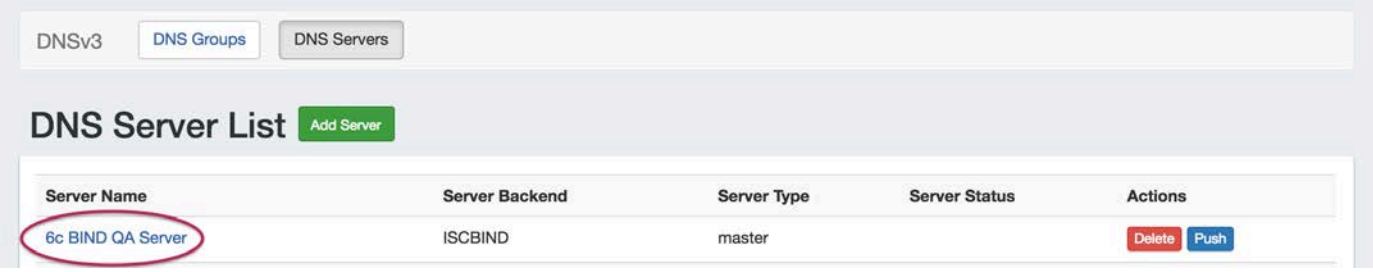
- Configuring DNSSEC
  - Enable DNSSEC for a Server (ProVision GUI)
  - Enable DNSSEC for a Zone (ProVision GUI)
    - Enabling DNSSEC (for a single zone)
    - Update Registrar and Confirm
  - For BIND server(s)
  - For Secure64 and PowerDNS

Enable DNSSEC for a Server (ProVision GUI)

DNSSEC may be enabled on a per-server basis in the ProVision DNS Server Settings.

Navigate to the DNS Tab, and select the **DNS Servers** section.

Find the desired server in the DNS Server List, and then click on the server name to open the settings for that server.



In the server settings page, scroll to the bottom of the second section, containing server-specfic settings.

Near the bottom of the section is a toggle to "Enable DNSSEC" for the server; click the toggle to the "ON" position.



Enable DNSSEC for a Zone (ProVision GUI)

DNSSEC may be enabled on a per-zone basis in the ProVision Zone Advanced Settings.

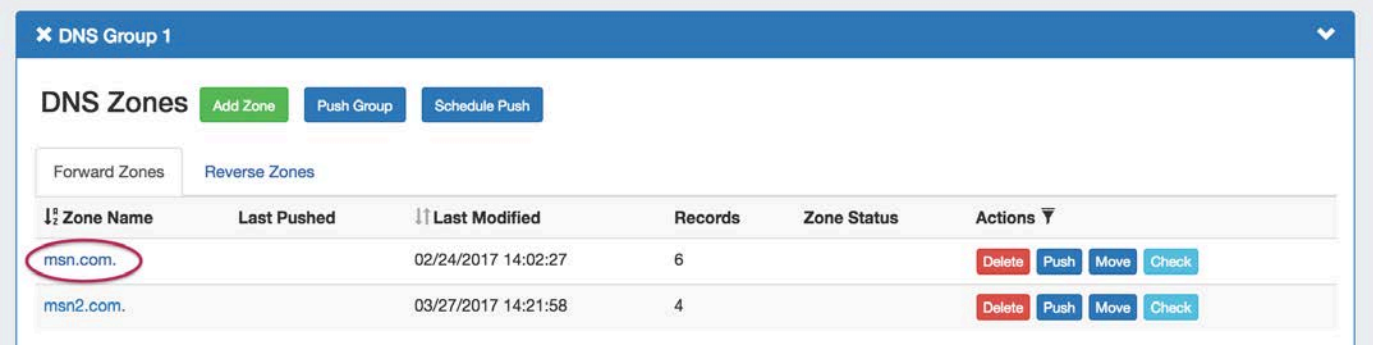
Before you enable DNSSEC for a zone, make sure that do the following:

- Make sure DNSSEC is enabled on the DNS server(s) you will be pushing zones to (see "Enable DNSSEC for a Server", above)
- run **configTest.php** to make sure that your directories/permissions are correct
- Set external server for Authenticated Data verification
- Create/Edit a zone - see Working with DNS Zones - Common Tasks for additional information.
- Ensure that the zone is associated with a DNS server(s)

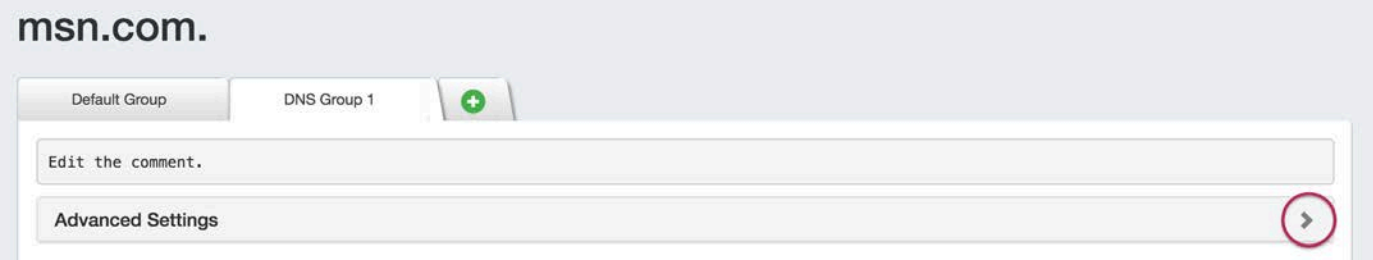
Enabling DNSSEC (for a single zone)

Navigate to the DNS Tab, and select the **DNS Groups** section.

Find the desired Group and Zone in the Groups List, and then click on the zone name to open the details for that zone.



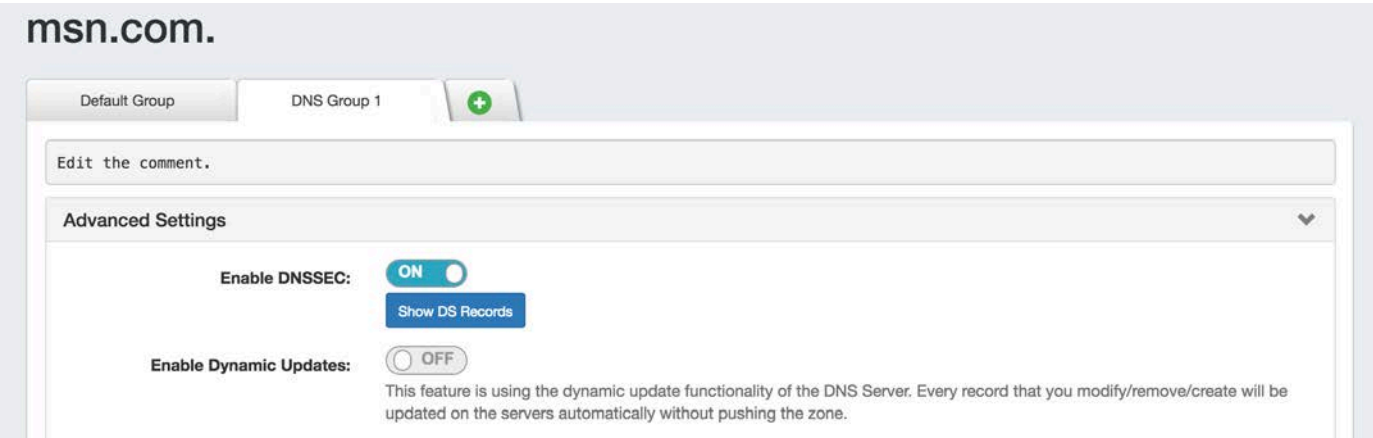
The view zone details page will open. Expand the zone details section "Advanced Settings" by clicking on the expansion arrow.



At the top of the "Advanced Settings" section will be a toggle to "Enable DNSSEC". Click the toggle to enable to the ON position.



Once enabled, the toggle will show as "ON", and a "Show DS Records" button will appear. At this point, no records exists, so clicking "Show DS Records" will result in a message telling you so.



In order for DS Records to be created, the zone must be successfully pushed. Push the zone, Group, or Server containing the zone successfully and DS records will be created (see Working with DNS Zones - Common Tasks and Working with DNS Groups for details on how to schedule and push zones).

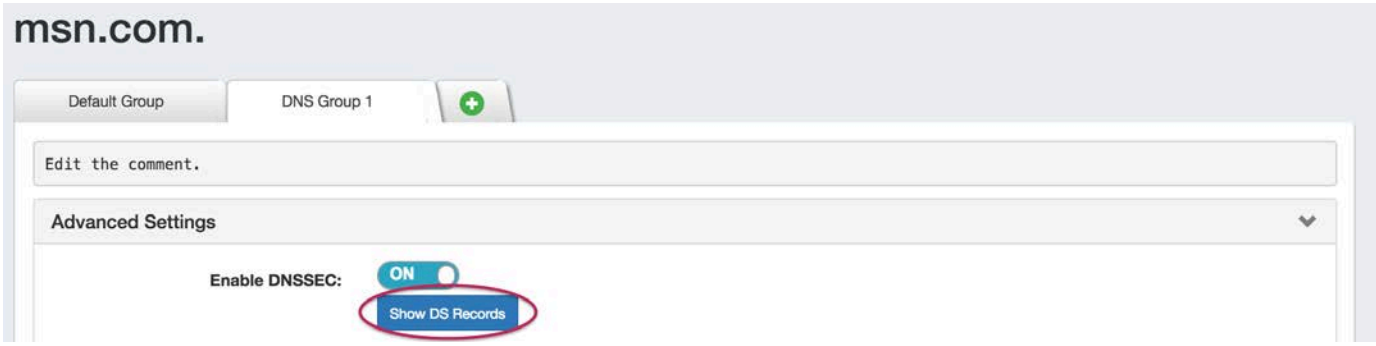
To quickly push just a single zone, go back to the DNS Groups section, and click on the "Push" button for the zone.



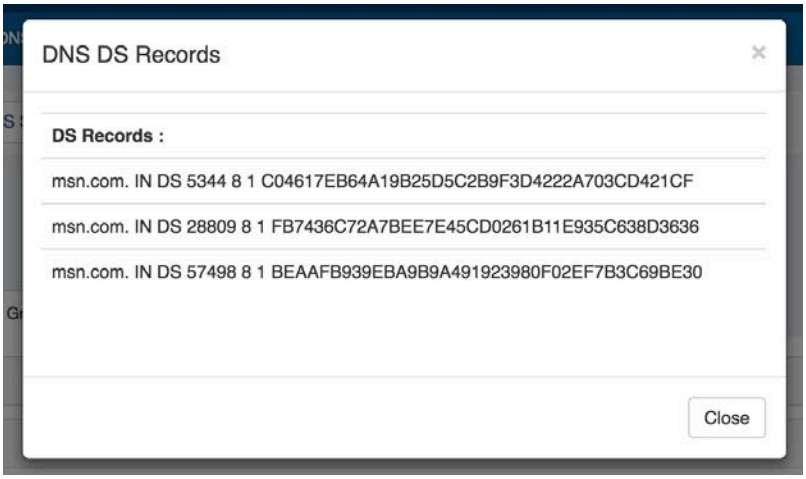
If successfully pushed, a green "Finished DNS Pushing Request" message will appear.



At this point, you will now have DS records available in the view zone details - Advanced Settings section. Open Advanced Settings for the zone and click on the "Show DS Records" button.



The DS records will be shown in a pop-up. Note these values for Registrar confirmation.



Update Registrar and Confirm

Once DS records have been created in ProVision, you will need to update and confirm the Zone Registrar:

- Upload these values to your Zone Registrar, obtained from the "Show DS Records" button
  - DS Record #, Key Tag, Algorithm, Digest Type, Digest
- Confirm values are saved at the Zone Registrar
- External sites
  - http://dnssec-debugger.verisignlabs.com/
  - http://dnsviz.net/

For BIND server(s)

To enable DNSSEC on BIND9 you need to modify **named.conf.options** with following parameters in the **options { }** section:

```
dnssec-enable yes:
dnssec-validation yes:
dnssec-lookaside auto;
```

These parameters may already be enabled in some Linux distributions by default, so please confirm before making changes.

Your DNSSEC implementation may need other options for your environment - please contact support@6connect.com if you have ny

Please note that you will need to restart the BIND service after these changes.

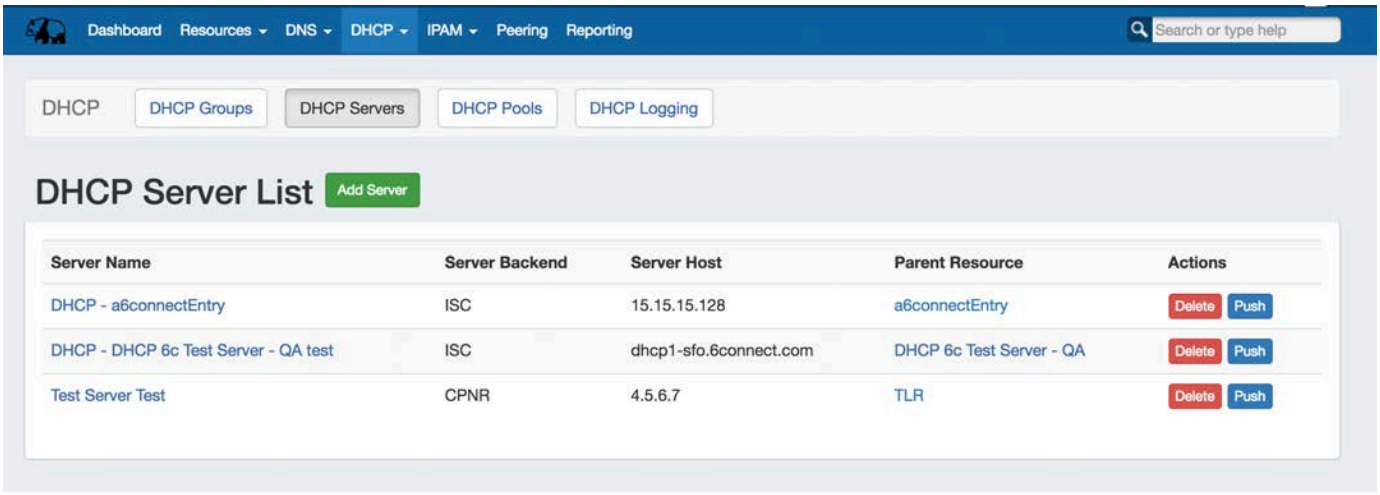
For Secure64 and PowerDNS

DNSSEC Signatures

In this scenario, 6connect ProVision uses the DNSSEC signing functions of the respective environment we write the zones to. We are evaluating how to properly integrate DNSSEC functions to ProVision for these platforms. Please contact support@6connect.com if you have feedback or specific questions.

DHCP Administration

DHCP Administration



DHCP Administration is primarily handled under the **DHCP** tab **DHCP Servers** sub-tab.

The **DHCP Servers** tab is only accessible to Admin users, and contains functions for adding, updating, and managing DHCP servers as well as scheduling server tasks.

Additional DHCP Admin tasks occur in other ProVision areas, and not all management tasks require Admin-level permissions - some only require appropriate resource permissions on the DHCP Groups, Zones, and servers involved. See additional sections on this page for more detailed information:

- DHCP Administration
  - DHCP Overview
    - DHCP Permissions
    - DHCP Administrative Tasks
  - Manage DHCP Servers
  - DHCP Pool Pushes
    - Manual DHCP Pushes:
    - Scheduled DHCP Pushes:
  - Additional Information:

DHCP Overview

The **DHCP** Tab organizes ProVision's DHCP system into a more unified and accessible interface, combining both admin and non-admin DHCP tasks together under the **DHCP** tab.

DHCP pools are gathered under DHCP Groups and servers are attached to those Groups. Pools, attached servers, and failovers are managed at a per-Group level.

Users can then view and manage Groups, individual pools, failover values, attach servers, and perform pushes all on the same page.

DHCP Permissions

DHCPv2 (ProVision 7.0.0 and later) more effectively utilizes ProVisions resource permissions system to allow fine-grain permissions in DHCP.

In order for a user to work with DHCP Groups, Pools, and servers, they must have User Group permissions for:

- One or more Parent Resources under which the DHCP Group and DHCP Pools will be associated (it can be a specific resource, or the all-encompassing "TLR")
- IPAM permissions sufficient to access and manage the desired blocks from which to assign DHCP Pools
- Permissions on the specific DHCP Server(s) in order to attach, detach, and push.

Creation and management of DHCP servers, as well as scheduled DHCP pushes, are Admin-only tasks.

DHCP Administrative Tasks

Admin-only tasks in DHCP include scheduling pushes, and creating / managing DHCP servers.

These tasks may be accessed in the following areas:

- Scheduling DHCP Pushes:
  - May be set up from the Scheduler or,
  - Set by server from the **DHCP Servers** page, or
  - Set by DHCP Group from the **DHCP Groups** page, or
  - Set by individual pool from the Pool Details page
- Managing DNS Servers: The **DHCP Servers** page

Tasks such as working with pool, DHCP Groups, editing Group Settings, attaching servers to Groups, and manually pushing pools do not require ProVision Admin permissions.

Instead, these tasks simply require that the user be included in a User Group that has appropriate permissions (either direct, or inherited) on the Parent Resources, DHCP Groups, and servers involved.

Manage DHCP Servers

Adding, configuring, and managing DHCP Servers occurs in the Admin-only **DHCP Servers** page under the **DHCP** Tab.

Supported DHCP server service types include:

- ISC DHCP
- Cisco Prime Network Registrar

For detailed information on adding, editing, and deleting DHCP Servers, see Working with DHCP Servers.

DHCP Pool Pushes

In DHCPv2, pools may be pushed manually or scheduled for a future time, and may be performed for a single pool, all pools in a DHCP Group, or a full DHCP server.

Scheduling DHCP pushes requires administrative access, but manual pushes only require permissions on the DHCP Group / Parent Resource, and the server(s) attached.

Manual DHCP Pushes:

Manual DHCP pushes may be performed from the following locations in ProVision:

- Single Pool: From the DHCP Groups pool list "Push" button. See Working with DHCP Pools - Pushing DHCP Pools
- Single Pool: From the DHCP Pool Details page "Push Now" button. See See Working with DHCP Pools - Pushing DHCP Pools
- Group: From the DHCP Groups list "Push Group" button. See Working with DHCP Groups - Pushing a DHCP Group
- Server: From the DHCP Servers list "Push" button. See Working with DHCP Servers - Pushing a server
- Server: From the DHCP Server Settings page "Push Pools" button. See Working with DHCP Servers - Pushing a server

Scheduled DHCP Pushes:

Scheduled pushes may be performed from the following locations in ProVision:

- May be set up from the Scheduler for a DHCP Group or server
- Set by server from the **DHCP Servers** page, or
- Set by DHCP Group from the **DHCP Groups** page, or
- Set by individual pool from the Pool Details page

Additional Information:

For more information on DHCP Adminsitrative tasks, see the following sections:

- Working with DHCP Servers

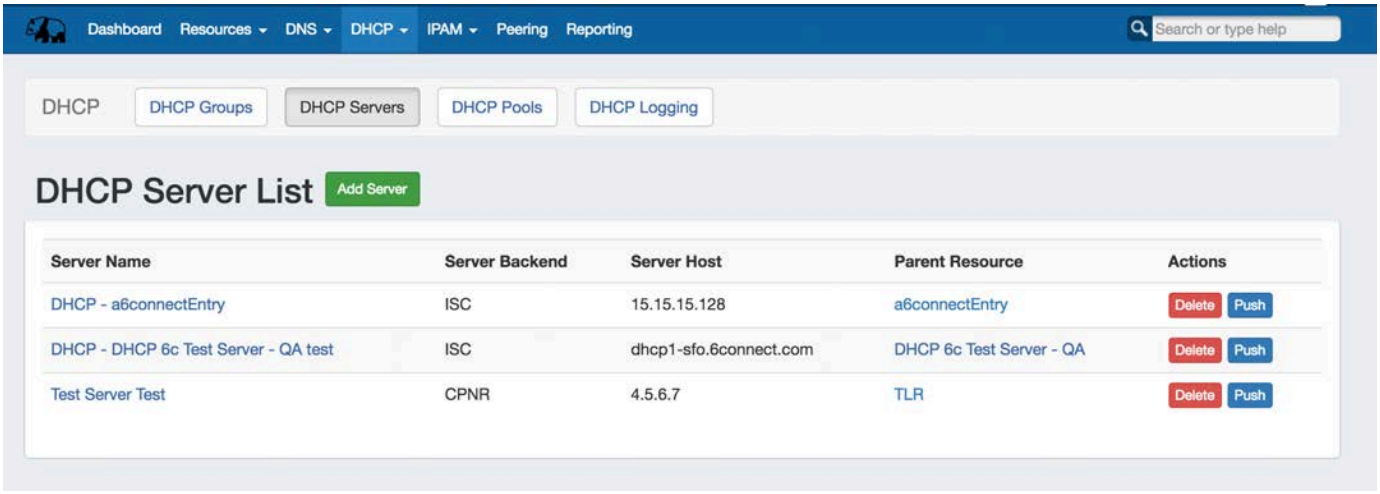
For non-admin level tasks, see:

- DHCP Tab
- Working with DHCP Groups
- Working with DHCP Pools
- Copy of DHCP Tab

Working with DHCP Servers

DHCP Servers

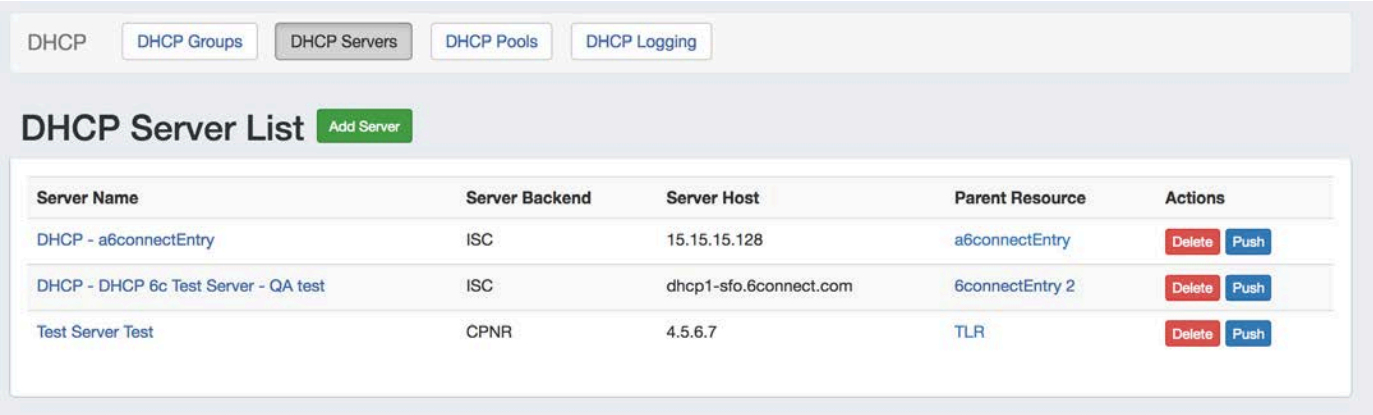
ProVision's DHCPv2 combines server management, group organization, and pool management under the **DHCP** tab.



The **DHCP Servers** tab is only accessible to Admin users, and contains functions for adding, updating, and managing DHCP servers as well as scheduling server tasks.

- DHCP Servers
  - DHCP Server List Interface
- Working with DHCP Servers
  - Add a Server
    - Server Settings
      - 1) Set Server Common Settings
      - 2) Set Server Specific Settings
      - 3) Set Advanced Server Settings
      - 4) Set DHCP Group Settings for Server
      - 5) Save Changes
  - Edit Servers
  - Review Pools Connected to a Server
    - Pools Directly Connected to the Server
    - Zones Connected via a Group
  - Pushing a Server
    - Manual Push
    - Scheduled Push
  - Review Pushed / Unpushed Configurations
  - Scan a Server
  - Delete a Server
- Additional Information

DHCP Server List Interface



1) Add Server Button: Opens a dialog for creating a DHCP server.

2) Server List:

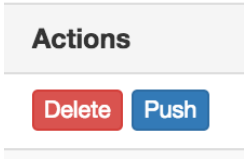
3) Server Name: Name of the DHCP server. Click to open server details.

4) Server Backend: The DHCP Service backend type for the server. Available types are ISC and CPNR.

5) Server Host: The server host.

6) Parent Resource: The ProVision resource set as the Parent for the server.

7) Actions: The actions that may be performed on each server:



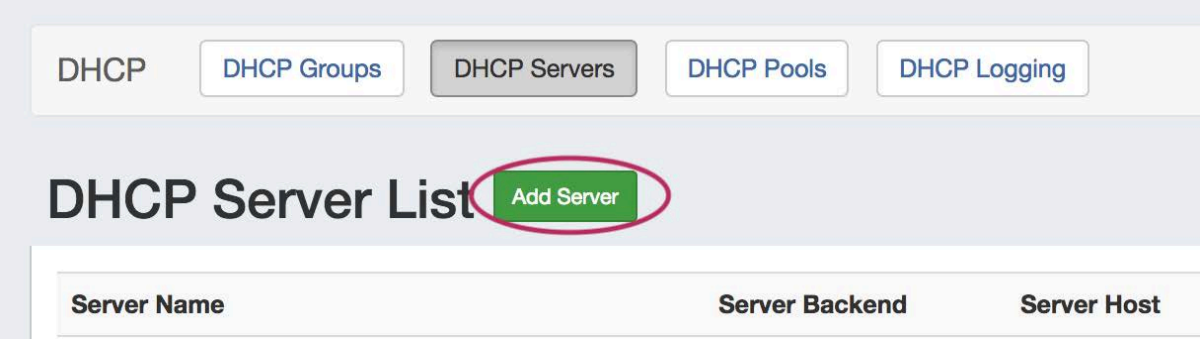
8) Delete: Deletes the server from ProVision.

9 Push: Pushes all pools associated with the selected server.

Working with DHCP Servers

Add a Server

To create a new server, start from the **DHCP** Tab, select the **DHCP Servers** sub menu. Then, click the "Add Server" button next to "DHCP Server List".

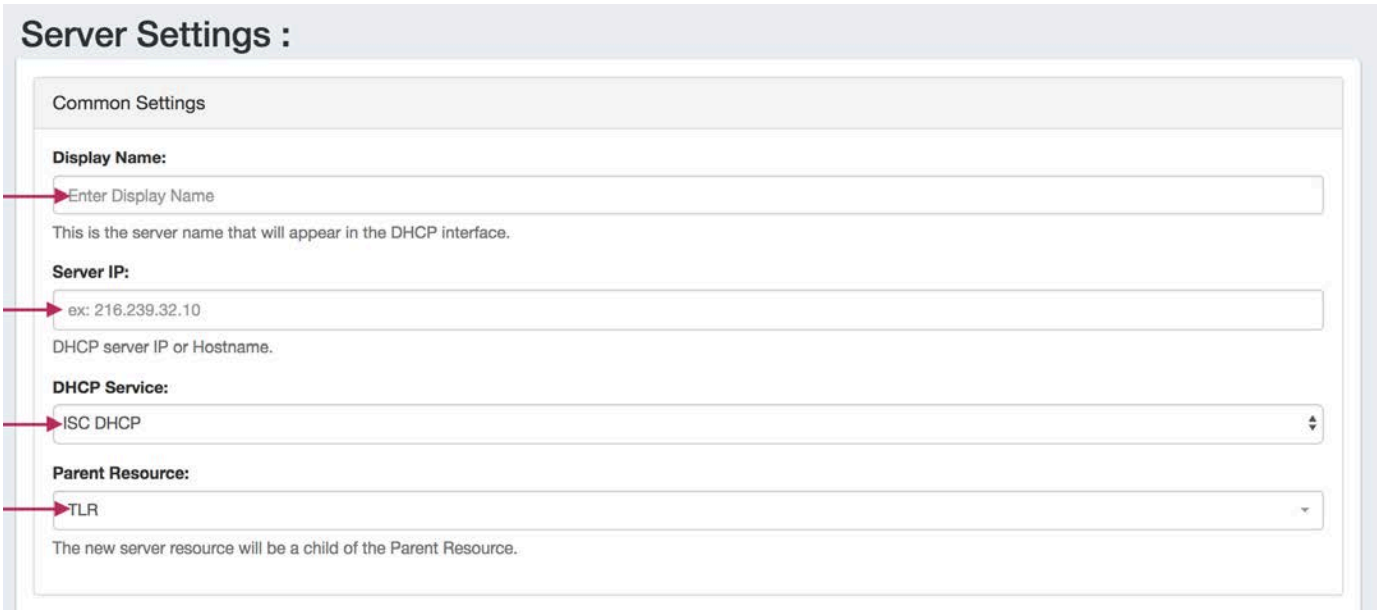


This will open the "Server Settings" page.

Server Settings

1) Set Server Common Settings

In the "Common Settings" section of Server Settings, enter the new server's Display Name (the name that will appear on the ProVision interface), the FQDN / IP, server type, service type, and desired parent Resource (may be left at the default Top Level Resource).



Display Name: Name you want the server to display.

Server IP: The IP or Hostname of the DHCP server.

DHCP Service: Select the DHCP service type - ISC DHCP, or Cisco Prime Network Registrar (CPNR).

Parent Resource: Select the ProVision resource to be the "parent" of the server - typically TLR (Top Level Resource), but may be a lower level resource such as a Customer or Location. The parent resource selection is the basis of access permissions for the server.

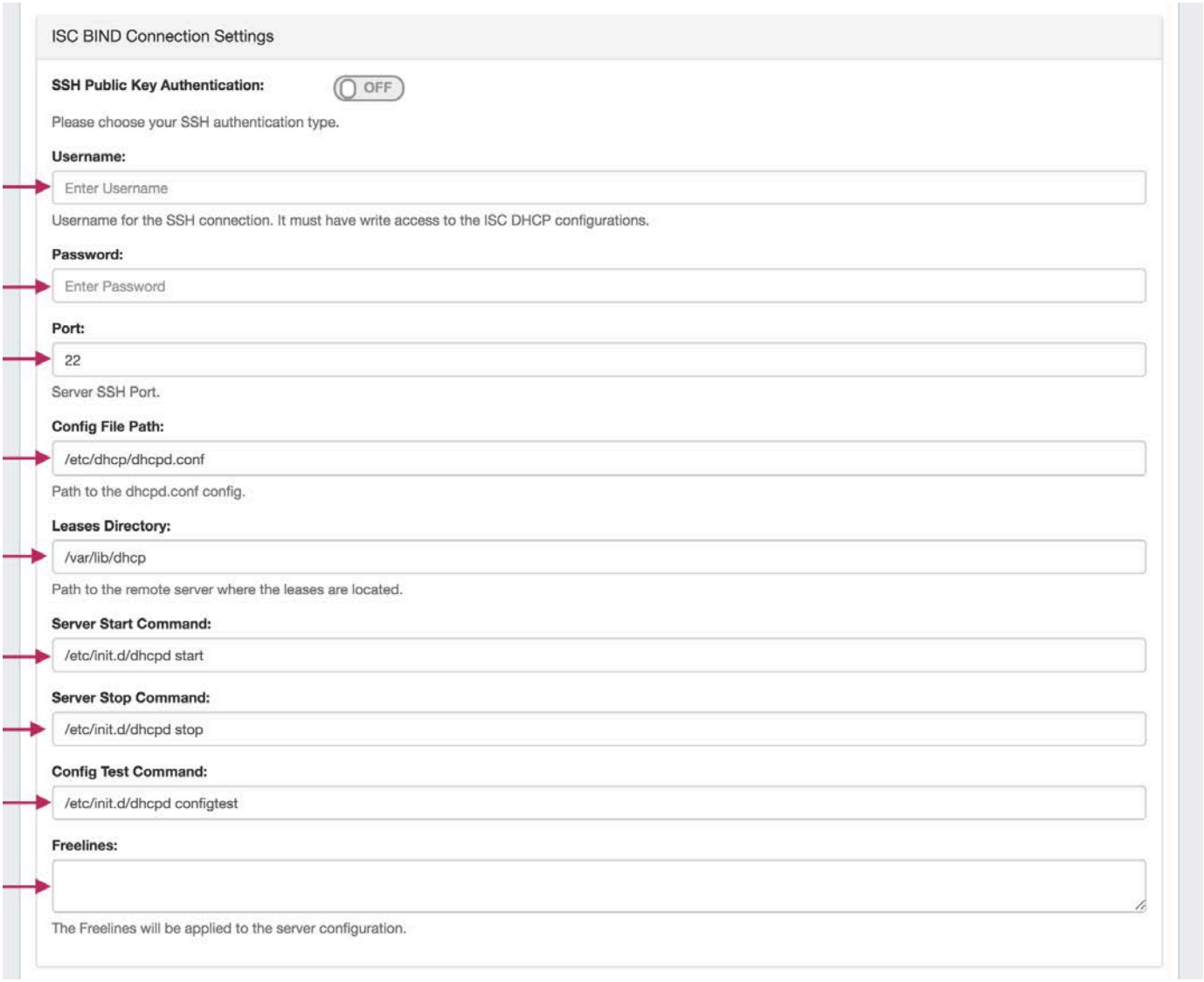
2) Set Server Specific Settings

The next section is entering server service-type specific settings. The options visible in this section will depend on the "DHCP Service" type chosen under "Common Settings".

Here, we see the fields for ISC DHCP server settings. Enter the server Username, Password, Port, Leases Directory, Server Start / Stop Commands, Server Test Command and Freelines (if desired).

Your fields may vary for other server types.

For SSH Public Key Authentication, click on the ON / OFF toggle to select "ON" or "OFF" as needed.



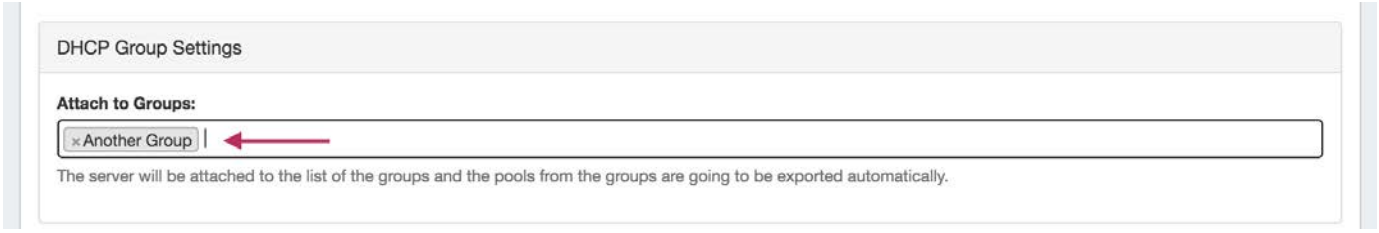
3) Set Advanced Server Settings

If desired, Advanced Settings for the DHCP Server may be entered. Toggle "Authorative Server" to "On" or "Off", and fill in the fields for Routers, Domain Name Servers, Domain Name, Default Lease Time, Max Lease Time, Local Port, or Log Facility.



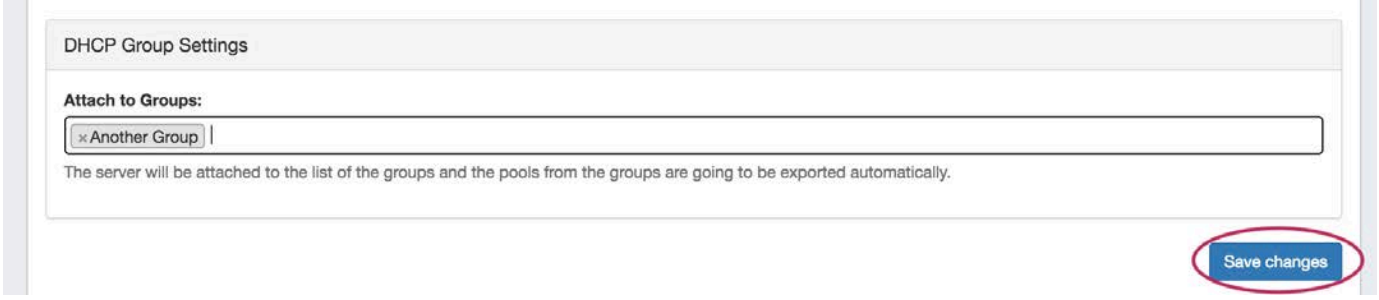
4) Set DHCP Group Settings for Server

In the last section, select whether to attach the server to an existing DHCP Group. Pools assigned to the selected Group will automatically be attached to the server.



5) Save Changes

Save your changes when done! Just click the "Save Changes" button at the bottom right of the page.



The new server will now be added to the DHCP Servers list. These settings may be changed at any time by selecting the server from the server list and editing the information.

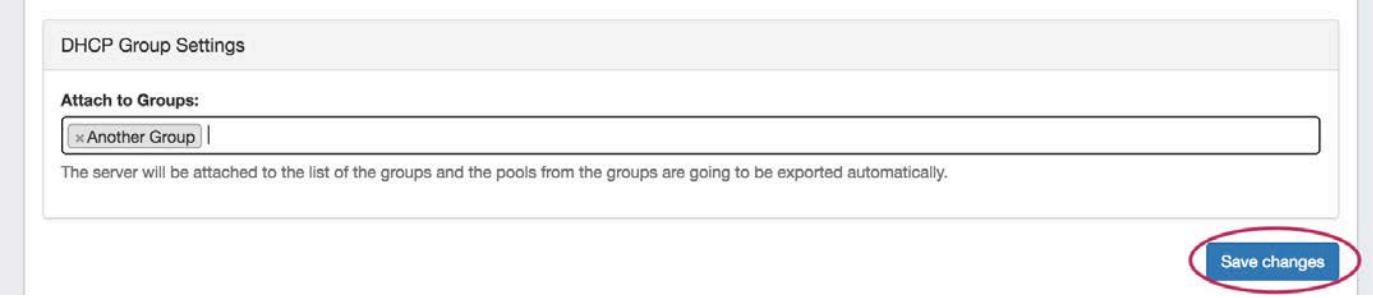
Edit Servers

Edit an existing server by clicking once on the server name in the DHCP Servers list.



The "Server Settings" page will open.

Click inside the field that you want to change, type your changes, and then click "Save Changes" at the bottom of the page.



Review Pools Connected to a Server

There are two ways that pools may be connected to a DHCP server:

1) Directly connected, by attaching the pool to a server from the Pool Details page.

or,

2) Connected by a Group that has been set as the default DHCP Group for the server, selected under "DHCP Group Settings".

Both are able to be viewed on the DHCP Server Settings page.

To view either, open the Server Settings page for the server by clicking on the server name in the DHCP Servers list.

Edit an existing server by clicking once on the server name in the DHCP Servers list.



The "Server Settings" page will open.

Pools Directly Connected to the Server

Scroll to the bottom of the page, and open the module titled "Pools directly connected to the server" by clicking on the expansion arrow.



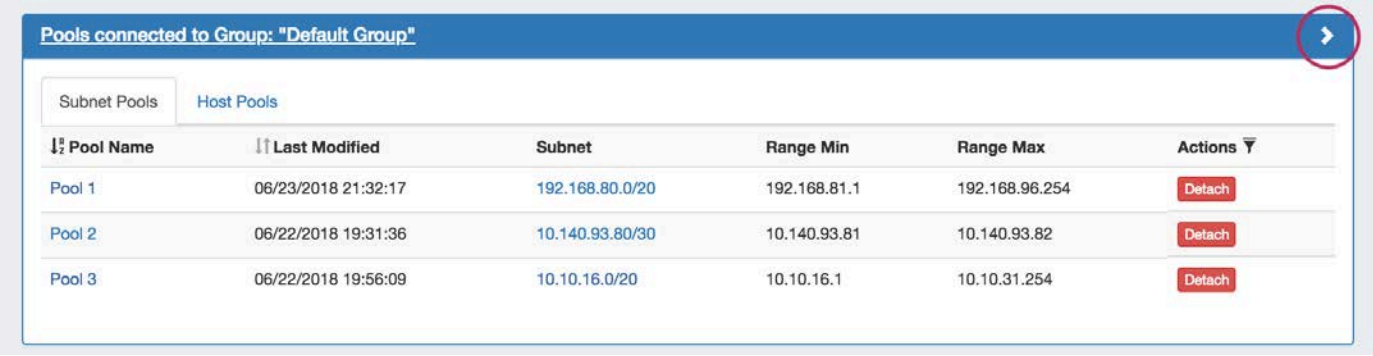
A pool list will show the pool(s) that have been directly connected to this server.

Here, you may browse through subnet and host pools by selecting the appropriate tabs, sort the list by Pool Name or Last Modified, or open the Pool Details page by clicking on the name.

Zones Connected via a Group

If a default Group has been selected under "DHCP Group Settings" for the server, pools under that Group will be connected to the server and able to be viewed on the Server Settings page.

Scroll to the bottom of the page, and open the module titled "Pools connected to Group '(Group Name)' " by clicking on the expansion arrow.



A pool list will show the pool(s) that have been directly connected to this server.

Here, you may browse through subnet and host pools by selecting the appropriate tabs, sort the list by Pool Name or Last Modified, or open the Pool Details page by clicking on the name.

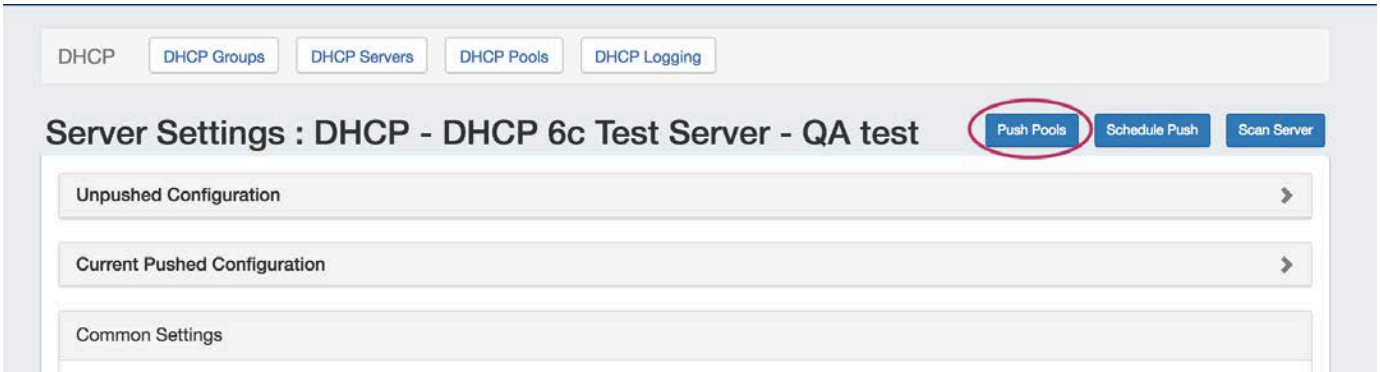
Pushing a Server

Manual Push

Manually pushing all pools on a server may be done directly from the DHCP Server list. Under the "Actions" section of the Server List, click the "Push" button for the desired server.



Pushing may also be done while in the Server Settings page. While in the Server Settings page, click the "Push Pool" button at the top right of the page.



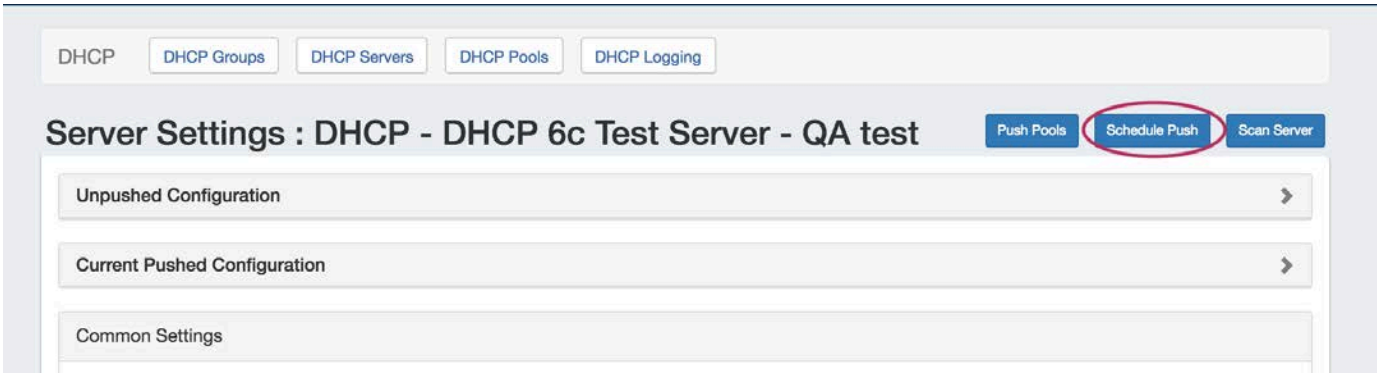
A "DHCP Push Status" box will appear, showing the status as the server is pushed. Once all pools have been pushed successfully, a green status message of "Finished DHCP Pushing Request" will appear. At this point, the push is complete and the window may be closed.

Scheduled Push

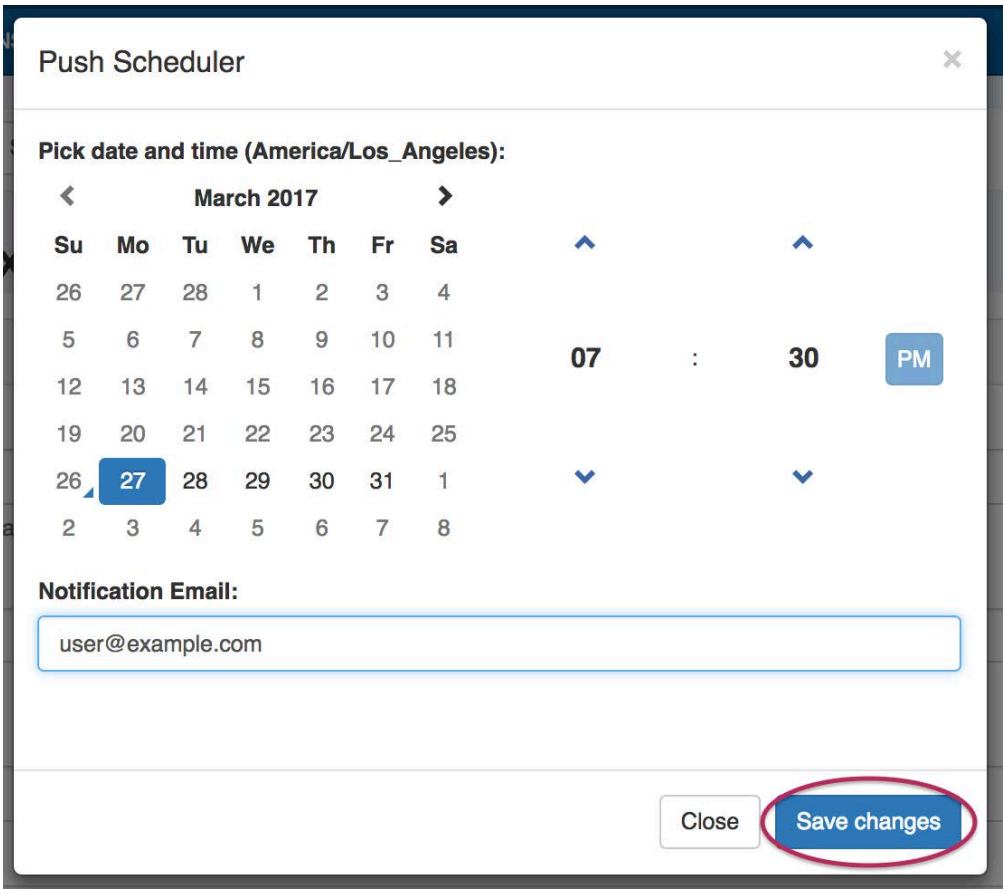
DHCP server pushes may be scheduled from either the Admin Area **Scheduler** Tab, or from within the DHCP Server Settings page. Scheduled pushes require Admin access.

For information on scheduling a push from the **Scheduler** Tab, see Scheduler documentation.

To schedule a push from a server's Settings page, open the Server Settings page for the desired server, and click on the "Schedule Push" button.

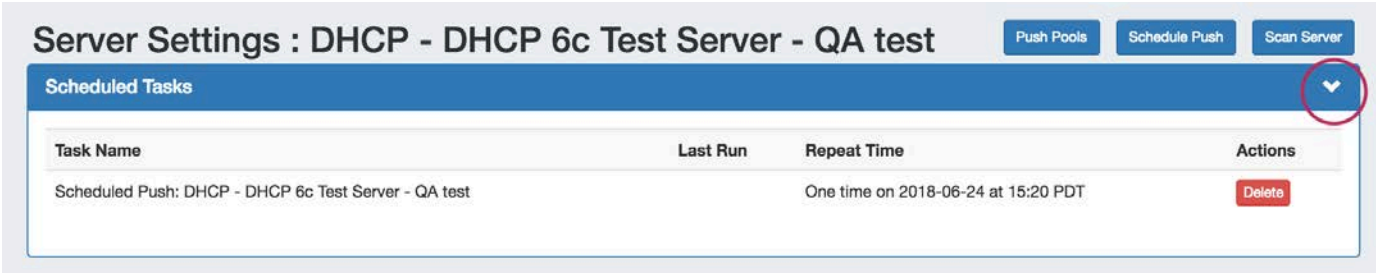


The Push Scheduler dialog will open. Click on the calendar on the left to select a date for the push, set the desired push time on the right, enter a notification email address, and then click "Save Changes".



Once a schedule push has been created, a "Scheduled Tasks" module will appear at the top of the Server Settings page.

Click on the expansion arrow for the module to open and view the tasks. Scheduled tasks may be deleted by clicking the "Delete" button under "Actions".



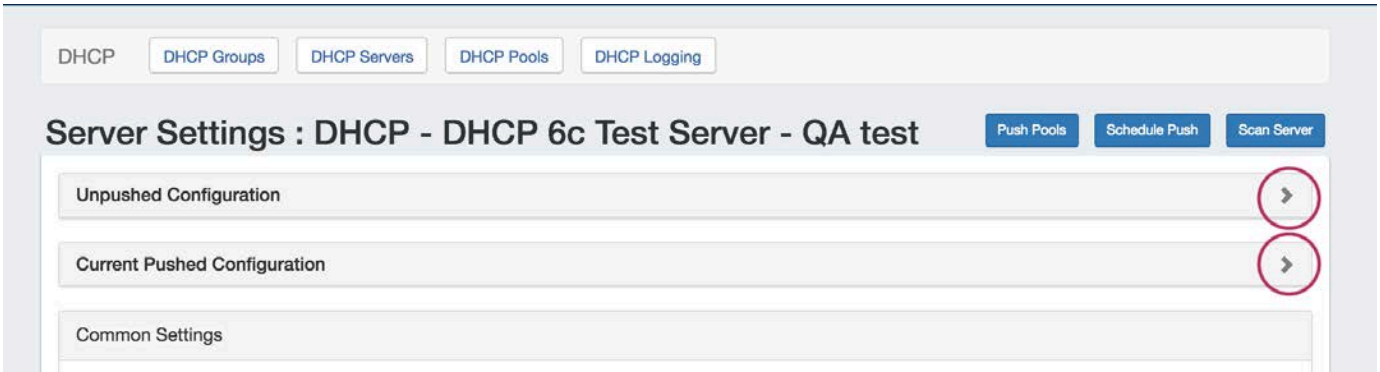
If necessary, the Scheduled Push may be edited from the **Scheduler** Tab in the Admin area of ProVision. See the Scheduler for information on editing scheduled tasks.

Review Pushed / Unpushed Configurations

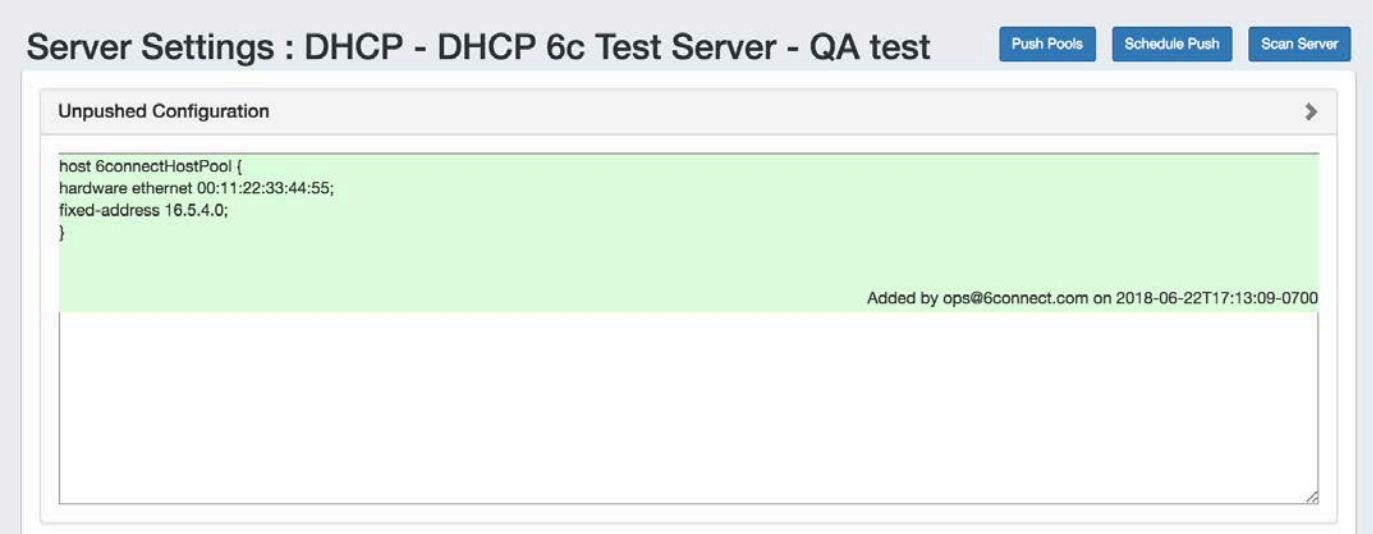
After a server has been created, unpushed edits may be viewed under the "Unpushed Configurations" module in Server Settings.

After a push, the most recent pushed configuration may be viewed under the "Current Pushed Configuration" module in Server Settings.

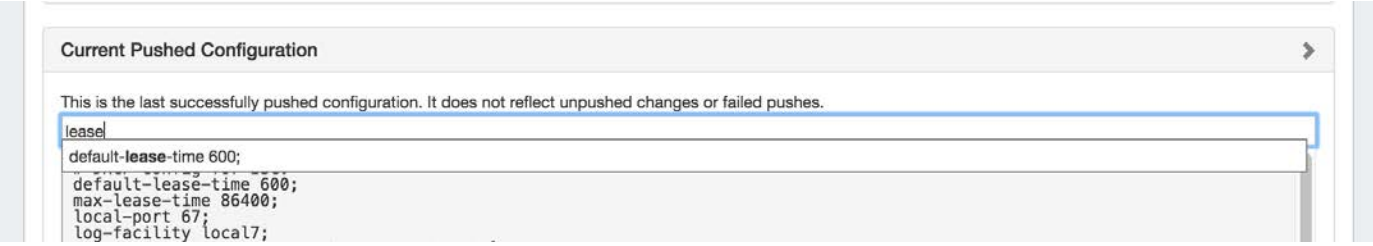
To open either, click the expansion arrow for the module on the top right of the header.



The Unpushed Configuration module will show the edits made to the configuration since the last successful push, and the user, time, and date of the change.

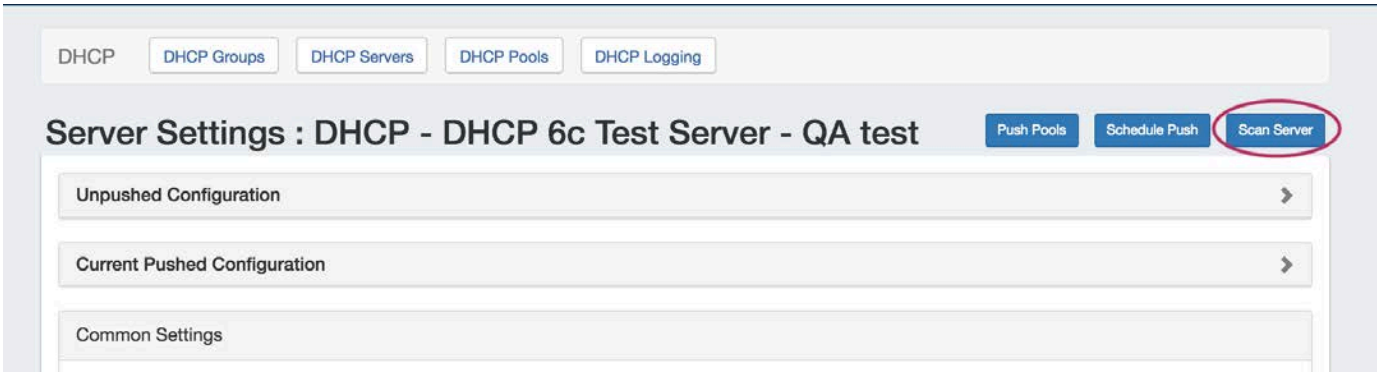


In Current Pushed Configuration, you may search for a specific line by entering a term in the search bar and hitting the "Enter" key - the config will jump to the appropriate line for review.



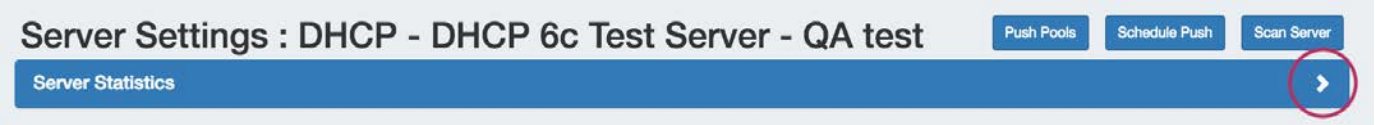
Scan a Server

Scan a DHCP Server by clicking the "Scan Server" button at the top of the Server Settings page.



When complete, a "Server Statistics" module will appear. Expand the module, and if the server permissions allow, the scanned statistics will show in the module.

If permissions or directory location fails during the scan, an error message will show instead.



Delete a Server

Delete a server by clicking the "Delete" button under the "Actions" section of the Server List for the desired server.



Additional Information

For more information on working with DHCP in ProVision, see the following areas:

- DHCP Tab
- Working with DHCP Groups
- Working with DHCP Pools
- Copy of DHCP Tab

Importing Your Data

Preparing for Data Import

Before importing your data into ProVision, there are a few steps we recommend in order to make the import process as smooth as possible:

- Preparing for Data Import
  - Step 1: Normalize your Data
  - Step 2: Prep your Data
  - Step 3: Import your Data
- Data Import Overview: Which Import Tool Should I Use?
  - Resource Import
  - Import Templates
  - Peering Import
  - IP Import
  - DNS Import
  - Additional Information

Step 1: Normalize your Data

Prior to importing your data, there is a key step of Data Normalization to ensure that information is accurate. If you need assistance with parsing your data prior to importing, 6connect can help with our Data Analyst service. Email us at support@6connect.com for more information.
You can also use off the shelf tools like Microsoft Excel, MySQL, or Google Refine if you intend to take on the task of data cleanup in house.

Common normalization steps required for ProVision imports include:

- Remove any empty rows in the data range
- Remove any empty columns in the data range
- Remove stray data from inside and outside the data range, such as:
  - notes, comments, and placeholder text as field values, when not set as their own data column
  - nested secondary information located in additional rows under a primary record
- Ensure proper data formatting
  - Check for data formatting errors - a common issue is leading zeros being dropped from a ZIP field when a file is opened. Re-specify format as needed.
  - Ensure each field has consistent formatting for values
- Ensure unique values for primary identifiers, such as Resource Name and ID Fields

  Data Encoding Format

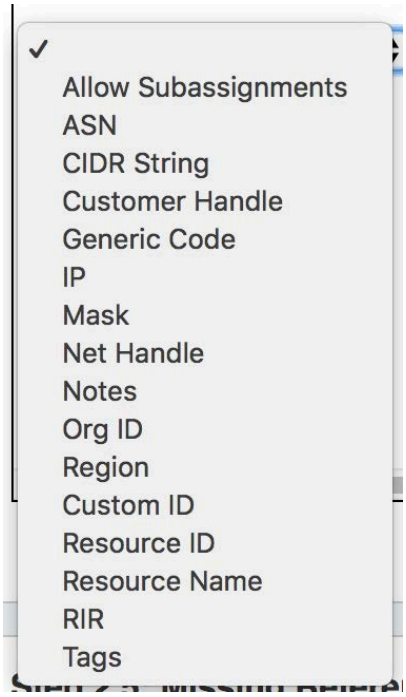
  To ensure correct importing of any special characters, make sure to use UTF-8 encoding for your CSV file!

Step 2: Prep your Data

You can download data import templates from the **Dashboard Tab** or **Data Import Tab**. We recommend that you open the CSV import templates and get familiar with the data fields that you can import into the platform.

- **For Company information** you can import relevant data including mailing/billing address information as well as ARIN specific SWIP fields, and specific DNS servers.
- **For Contact information** you can import contact records assigned to a given **Company**. We support typical fields for this data including Name, multiple email fields, phone numbers as well as Timezone and Role (Roles can be customized from the **IPAM Admin Tab)**.
- **For IP Block information** you can import the following fields:

Click here to expand...



Allow Subassignments option

When importing the field "Allow Subassignments" - the parameters accepted are "TRUE", "1", "Y", "yes"

Step 3: Import your Data

Get to the **Data Import Tab** from the **Admin button** to import your data. For larger data import runs, feel free to contact 6connect at any time for assistance at support@6connect.com.



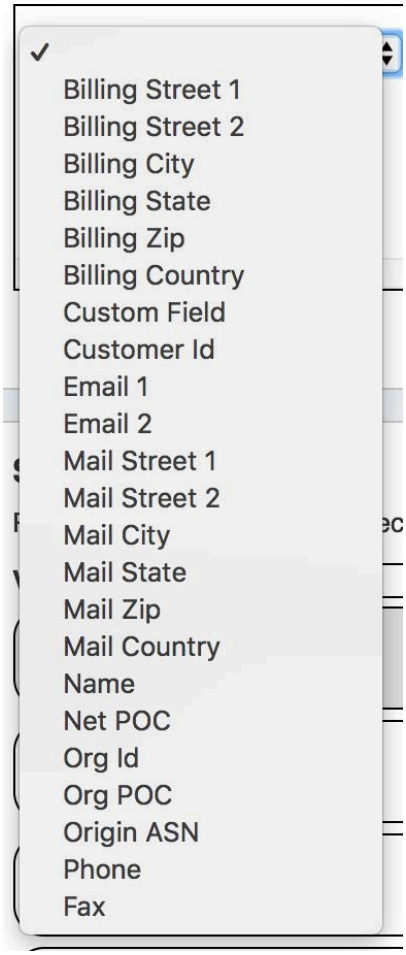
Data Import Overview: Which Import Tool Should I Use?

Resource Import

Simple Upload / Import from CSV: Use this tool if you have a simple .csv file of customer / contact information, such as Name, Address, Billing Address, Phone numbers, POC, etc. See Resource Import from CSV for more details.

This tool supports the following fields:

Click here to expand...



For Resource Import from .csv, the following fields are REQUIRED:

- Name (Unique Resource / Customer Name)
- Customer ID (Unique non-name identifier)

All other fields are optional.

Resource Import Tool: The Resource Import tool is designed for importing .csv files for any Resource type (company, physical devices, contacts, customers, etc), and allows for customized headers through integrating ProVision's Section fields. It also allows for data editing within the tool. Use the Resource Import Tool if you have resource data that does not fit under the Simple Upload / Import from CSV field options, and can equate that data to an existing or newly created Section.

Additional Information

A tutorial / walkthrough for this tool is available at the Resource Import Tool page.

Import Templates

This section links sample files for selected ProVision import tools. These may be viewed as an example of .csv setup for their respective importers, or used for tutorials.

They are:

All Import Samples - Links to the CSV import templates page of the documentation, showing all available import template .csv files.

IP Import Sample File - Download the IP-import-sample_v1.csv file, used in the IP Import: Upload / Import from CSV Tool.

Customer Sample Import - Download the customer-import-sample.csv file, used in the Resource Import: Simple Upload / Import from CSV Tool.

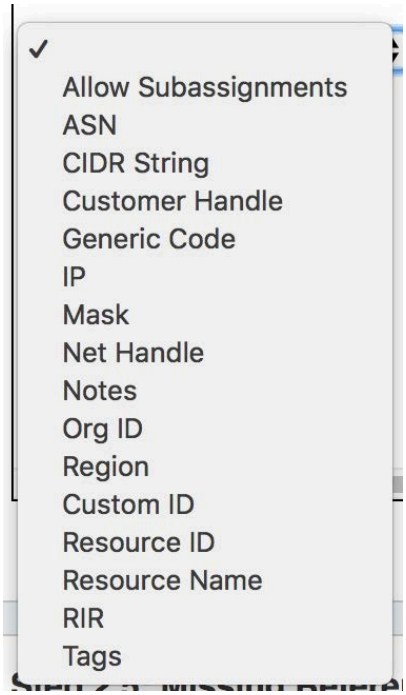
Peering Import

Import BGP Sessions: This section imports peering sessions from a selected exchange and router. Routers must be created prior to using this importer. See Peering Routers and Importing Sessions for additional information.

IP Import

Upload / Import from CSV: Use this tool if you have a .csv file of IP block information, such as CIDR, Mask, ASN, RIR, etc.
This tool supports the following fields:

Click here to expand...



For IP Import from .csv, the following fields are REQUIRED:

- CIDR String OR both IP and Mask
- RIR OR a selected default RIR

All other fields are optional.

Import from RIR: This tool automatically lookups your ARIN or RIPE information based on the IP address you are connected to. Once it identifies the blocks assigned to your company, you can import both 1918 aggregates as well as public IP space from ARIN and RIPE. See Import Aggregate Blocks for more detail on this tool.

DNS Import

BIND Zone Upload / Import: Imports DNS zones using the named.conf configuration file tied to the zones you are uploading, a .zip or .tar file of the zones themselves, and an optional .csv file mapping zones to customers and DNS Servers. This is the simplest and most commonly used import method. Refer to Import DNS Zones for more information.

PowerDNS Zone Import: This tool is available after configuring a PowerDNS server with a MySQL backend. The PowerDNS Import connects to the selected server and imports all zones.

InfoBlox Zone Import: Imports DNS zones using a provided Host, Username, and Password. The InfoBlox import pulls all zones on the InfoBlox LOCAL grid and adds them to a designated Group. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

NS ONE Zone Import: Imports DNS zones using a NS One API Key. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

Dyn DNS Zone Import: Imports DNS zones using a Dyn DNS Customer Name, Username, and Password. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

DNSMadeEasy Zone Import: Imports DNS zones using a DNSMadeEasy API Key and API Secret. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

IPPlan Zone Import: Imports DNS zones using IPPlan MySQL database options. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

Additional Information

For more details, see:

- Resource Import from CSV
- Resource Import Tool
- Import Sessions
- IP Import from CSV
- Import Aggregate Blocks
- Import DNS Zones

Resource Import from CSV

Simple Upload / Import from CSV

The Simple Upload / Import from CSV tool is used if you have a simple .csv file with customer / contact information, such as Name, Address, Billing Address, Phone numbers, POC, etc. It is accessed from the **Data Import Tab** from the **Admin** section of ProVision.



To import your customer / contact information, follow the following steps:

- Before you Begin: Prepare your Data for Import
- Step 1: Create a new Resource / Customer Import Job
- Step 2: Define Columns
- Step 3: Reviewing Data
- Additional Information

Before you Begin: Prepare your Data for Import

The Simple Upload / Import from CSV tool requires only a .csv file for importing.

Be sure to review "Preparing Data for Import" on the Importing your Data page before you begin. Verify that your .csv is correctly parsed and cleanly formatted with your customer/contact information (using your choice of the available field list shown below), and is UTF-8 encoded for best results.

Be sure to check that your data is cleaned up with the following steps:

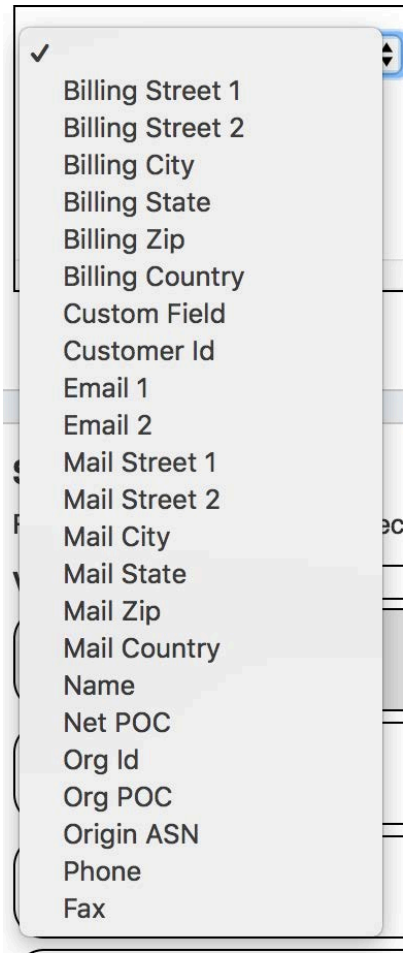
- Remove any empty rows in the data range
- Remove any empty columns in the data range
- Remove stray data from inside and outside the data range, such as:
  - notes, comments, and placeholder text as field values, when not set as their own data column
  - nested secondary information located in additional rows under a primary record
- Ensure proper data formatting
  - Check for data formatting errors - a common issue is leading zeros being dropped from a ZIP field when a file is opened. Re-specify format as needed.
  - Ensure each field has consistent formatting for values
- Ensure unique values for primary identifiers, such as Resource Name and ID Fields

  Data Encoding Format

  To ensure correct importing of any special characters, make sure to use UTF-8 encoding for your CSV file!

This Resource Import from .csv tool supports the following fields:

Click here to expand...



Note: the following fields are REQUIRED:

- Name (Unique Resource / Customer Name)
- Customer ID (Unique non-name identifier)

All other fields are optional.

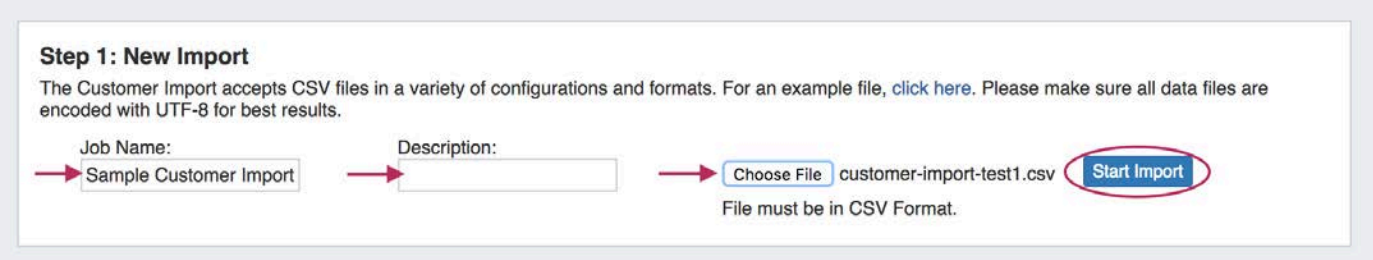
Step 1: Create a new Resource / Customer Import Job

Navigate to the **Data Import** Tab from the **Admin** button. Select "Simple Upload / Import from CSV" under "Resource Import".



Create a Job Name and Description for the import. This is especially useful to keep track of progress in cases the data arrives from multiple sources, or will require multiple stages of manual review.

Select the .csv file that you prepared above by selecting the "Choose File" button, and browsing to the correct file location. Then hit "Start Import".



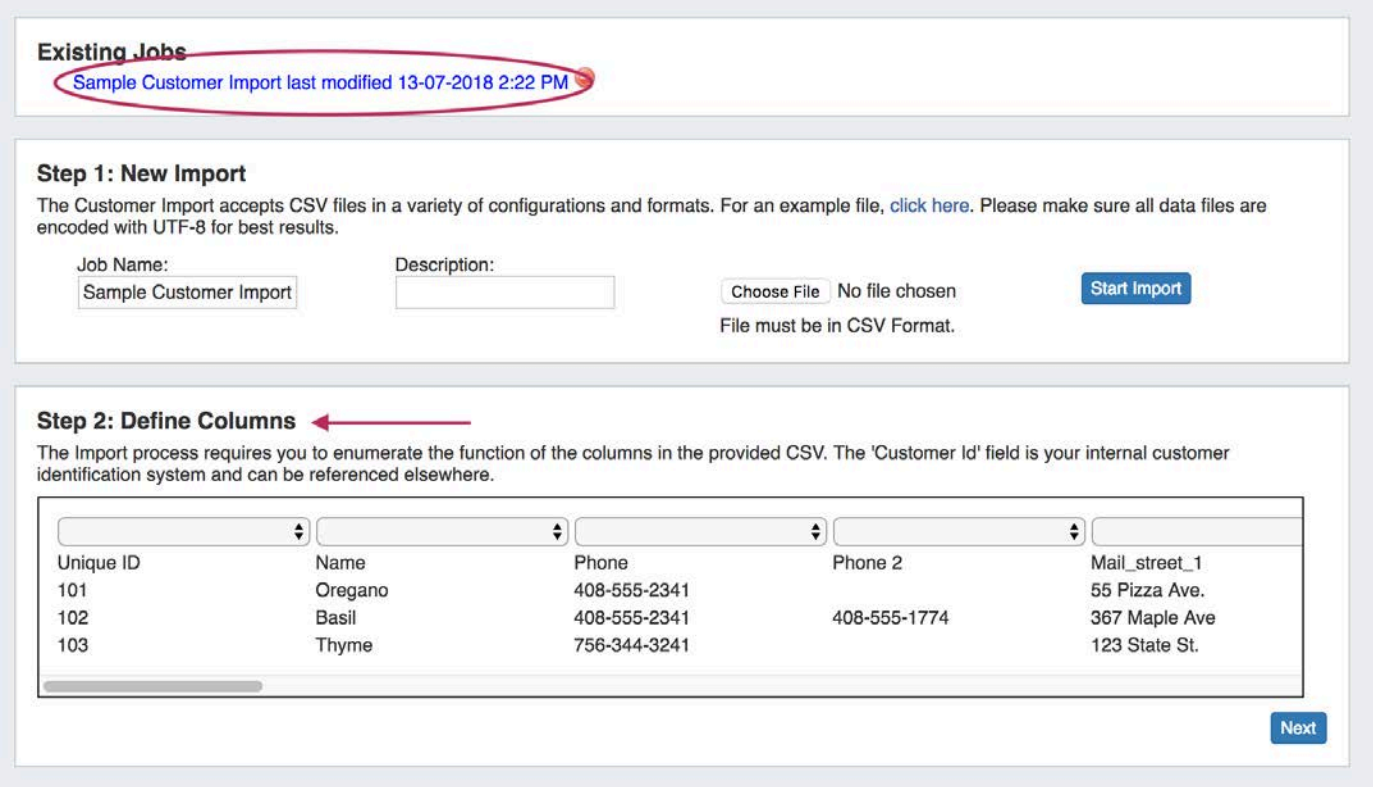
Working with Large or Multiple Data Sets

Although you cannot add new files to an existing job, for jobs with multiple sources for data (which may have different formatting), you can simply create separate jobs and descriptions for each source - no need to manually combine the data into one file before importing. The Import tool's mapping and editing functions will allow for the data to be reconciled in ProVision.

For large data sets where multiple stages of manual review might be needed, you can create a new job using the same set of data files in order to work in parallel on a different portion of the data.

After importing, the new job will appear under the "Existing Jobs" section.

To continue working with this job, select it from the list (by clicking on the link) and the next step ("Define Columns") will appear at the bottom of the page.



Step 2: Define Columns

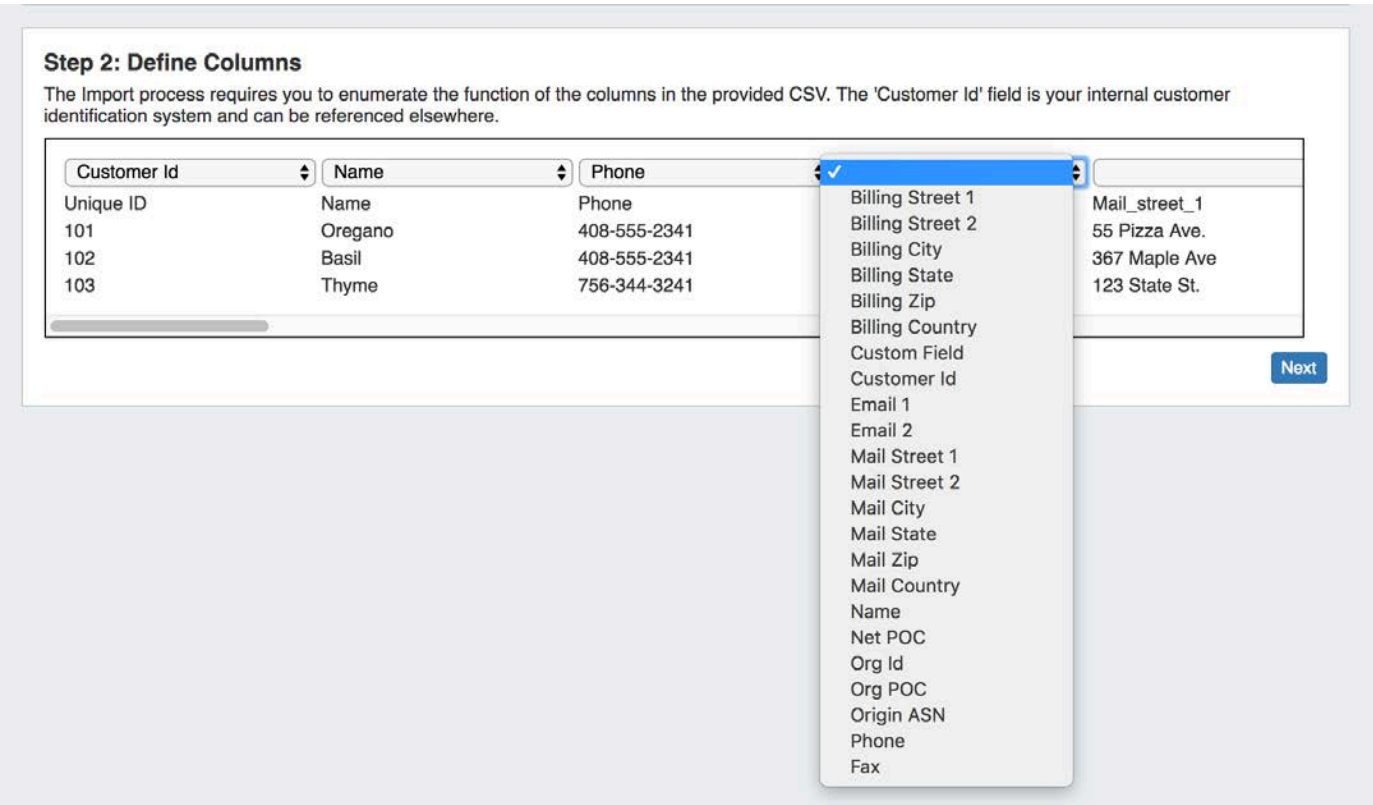
Using the dropdown menu above each data field, select the appropriate definition for each of the imported columns. **Phone** and **Fax** may have multiple columns associated with the data. Other columns which do not apply under the available definitions should be left as blank, and will be skipped during the upload process.

Note: the following fields are REQUIRED:

- Name (Unique Resource / Customer Name)
- Customer ID (Unique non-name identifier)

Make sure that you have defined all desired fields by using the scroll bar below your data to view additional columns.

When completed, hit "Next".



Step 3: Reviewing Data

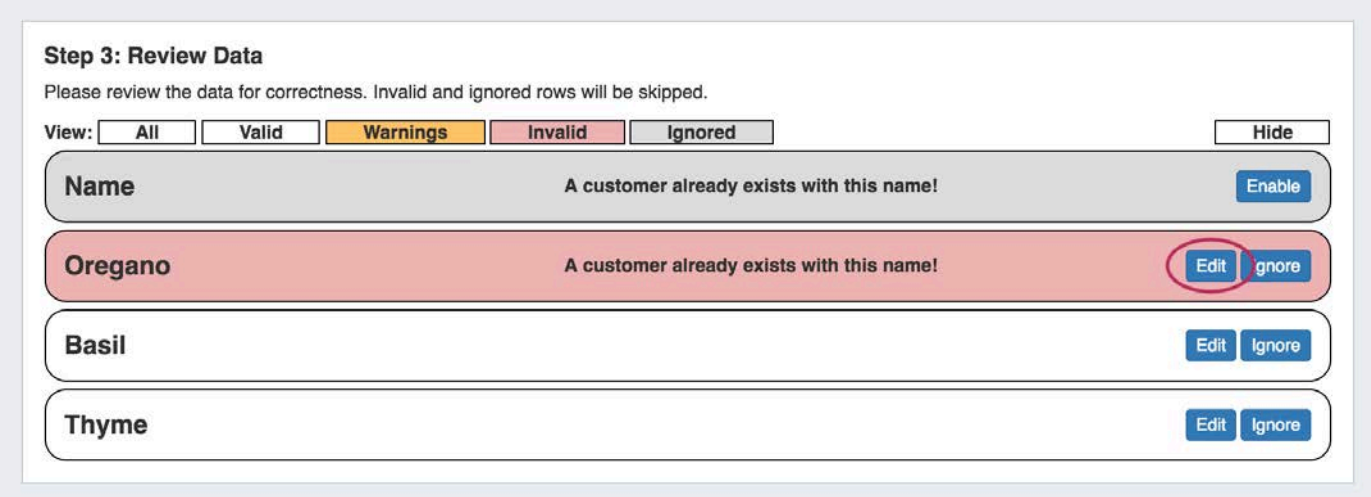
After supplying the file and defining columns, a review step is provided. Records with errors will show as color coded, and can be filtered to be viewed by All, Valid, Warnings, Invalid, or Ignored.

From here, the records can be edited or ignored. Select "Ignore" for records that you do not wish to import at this time. Records that are Ignored or Invalid will automatically be skipped.

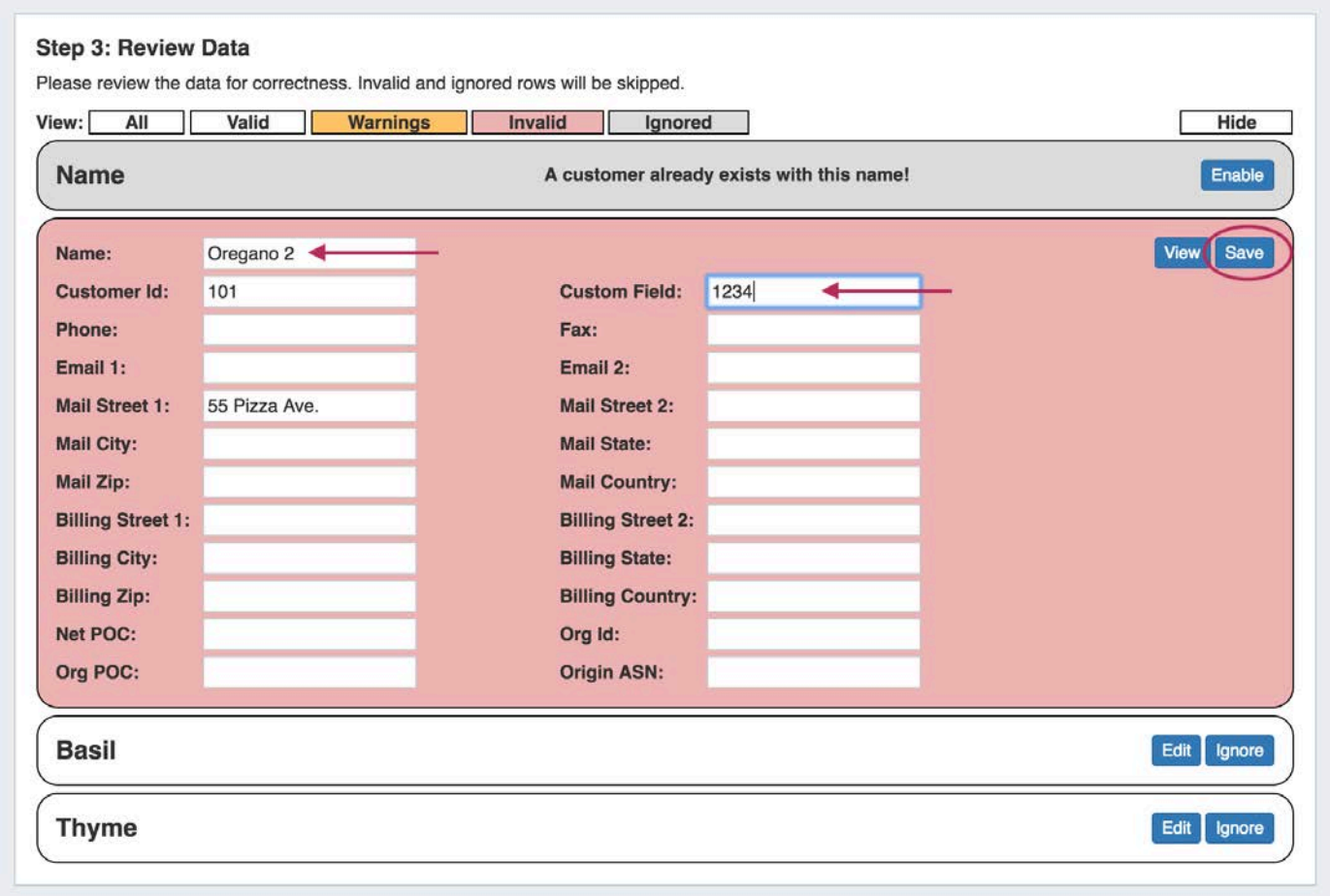
Header Rows

If your .csv has a header row as the first line, that row will be shown as the first record in review data as well. Simply click "Ignore" on the first record to disregard the row.

For rows with an "Invalid" or "Warning" status, you may wish to edit the record to change information to a valid option.



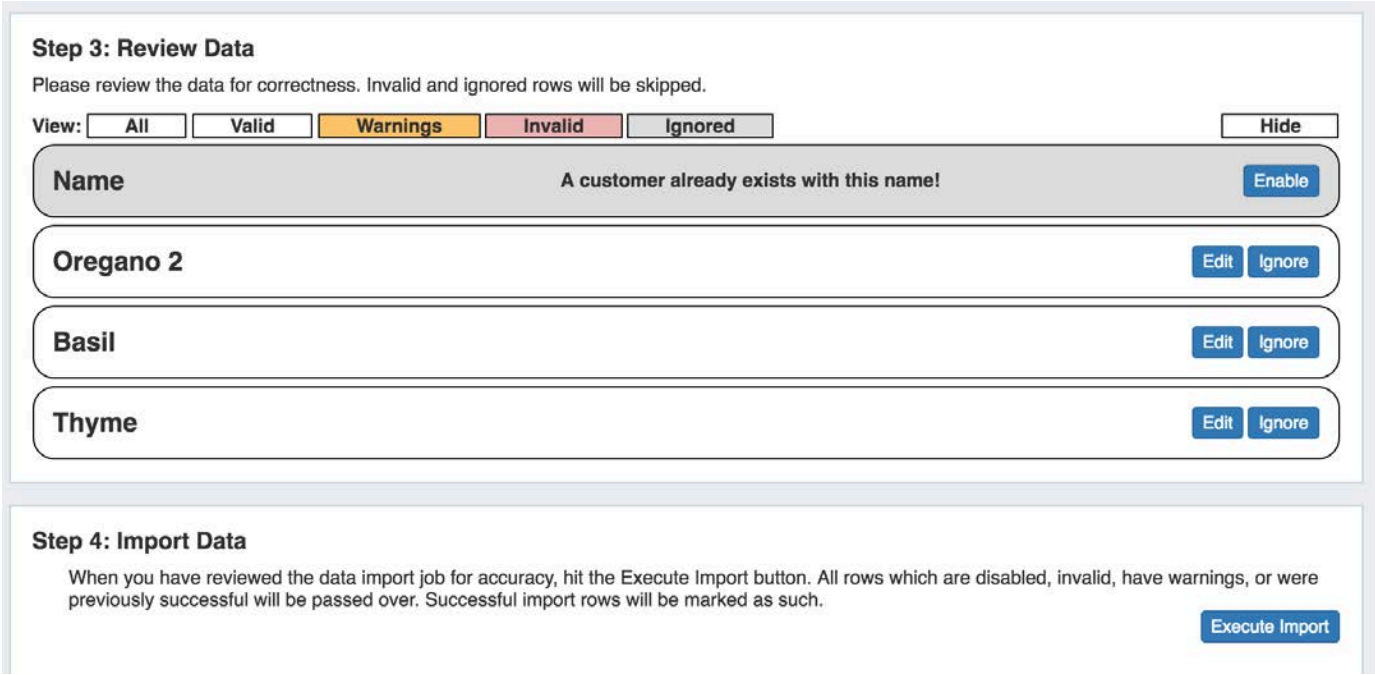
Hitting the "Edit" button for the record provides options to change or add information for available fields.



In this example, as "Name" was the invalid field, it has been updated to a unique value. The "Custom Field" value has been added as well.

After making the desired edits, hit "Save".

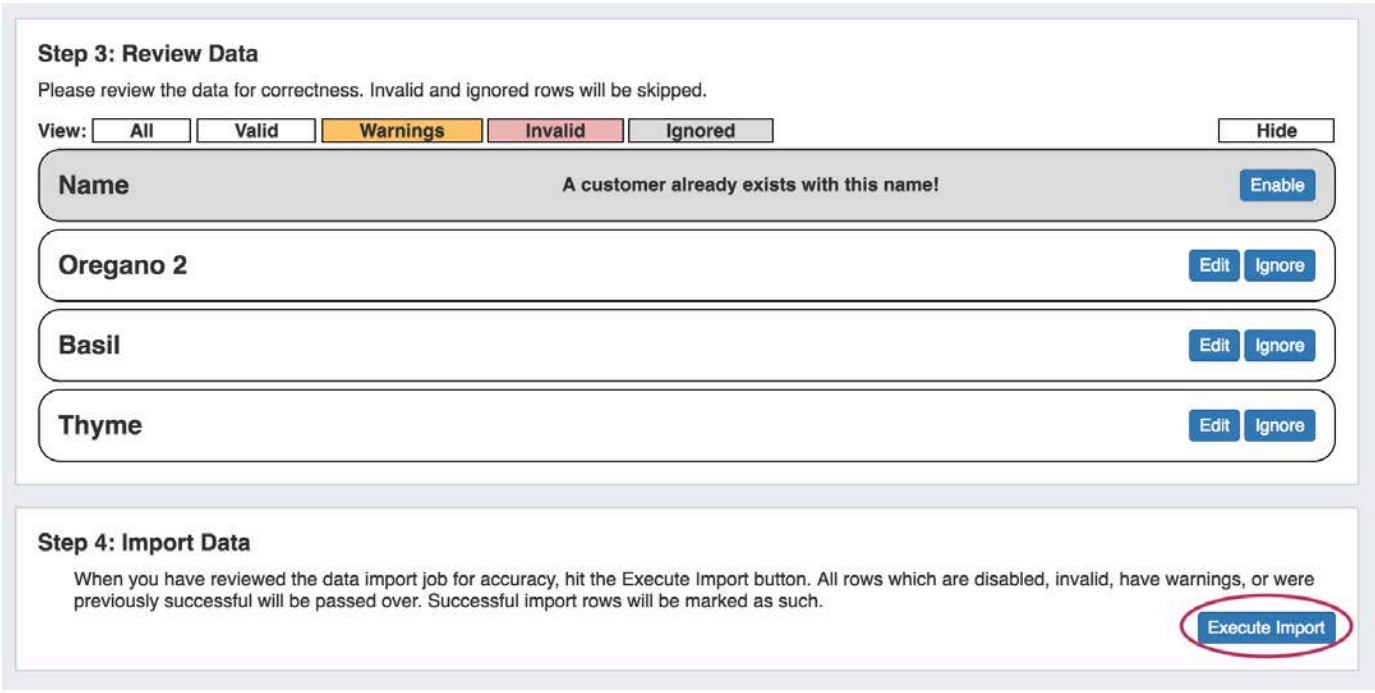
The data record will be re-checked for validity, and will have a white background if it is now valid for import.



After data has been reviewed for accuracy, you may execute the import.

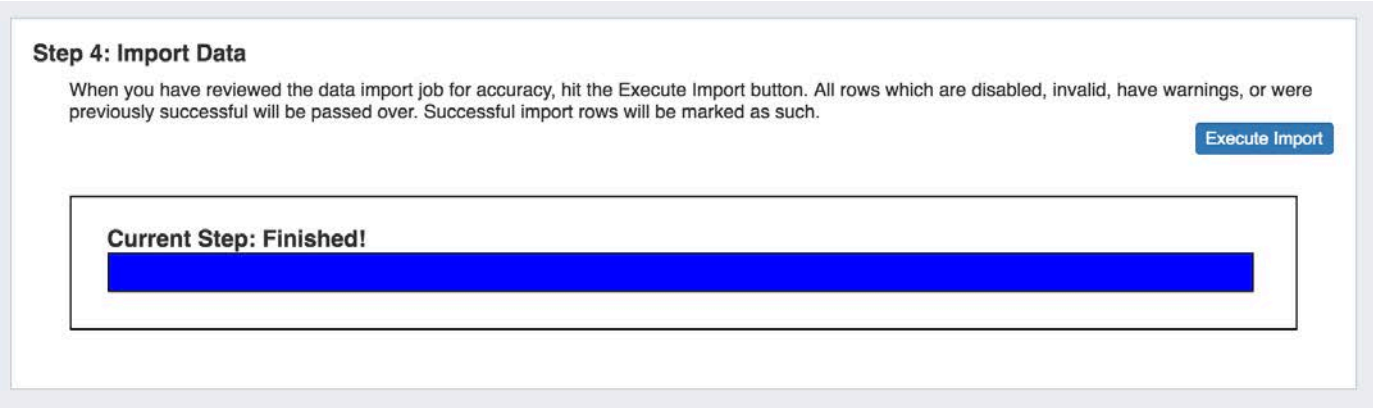
Step 4: Execute Import

When the review step is completed, hit the "Execute Import" button.



A progress bar will appear to and display the records as they are imported.

When the bar reaches 100%, and displays "Current Step: Finished!" the import is complete, and you may leave the page.



Additional Information

For more import options, see:

- Resource Import Tool
- Import Peering Sessions
- IP Import from CSV
- Import Aggregate Blocks
- Import DNS Zones

Resource Import Tool

Importing Resources

- Importing Resources
  - The Resource Import Tool
  - Before You Begin
  - Opening the Resource Import Tool
    - The Resource Import Tool UI
  - Resource Importer Walkthrough

The Resource Import Tool

The Resource Import Tool (in beta) allows you to import resource data from a .csv file into ProVision. In the Resource Import Tool, you can open one or more user-created .csv spreadsheets, perform basic editing functions if needed, associate the data to a specific Section, and correlate the data columns to specific Section Fields.

In ProVision, since Resources can be any desired entity, and Sections can be anything from "customers" to "firewalls" to "racks", you have total flexibility in what type of data to import with the Resource Importer to meet your specific company needs. Check out Working With Resources, Customizing Sections, and Customizing Fields for more details on how to fit these elements to your business.

Before You Begin

There are a few items that you will need have set up prior to using the Resource Importer Tool. Ensure that you have:

- The .csv document you wish to import saved with UTF-8 encoding. Windows, Mac, and Linux type .csv files are supported.
- A header row for the data in the .csv.
- The .csv file should be "clean", that is, only contain the data to be imported and a header row for that data.
- A Section created in ProVision with fields that correlate to the import data. For example, if you wish to import a list of contact information, there will need to be a Section in ProVision created for "Contacts", with fields such as "First Name", "Last Name", "email address", "Phone number", and so on. To create a new Section, or edit an existing Section, refer to Working With Resources, Customizing Sections, and Customizing Fields.

  If the above preconditions are not met, the Resource Importer Tool may not be able to correctly read the .csv file or complete the import. Verify UTF-8 .csv encoding, a clean dataset with a header row, and that an appropriate Section exists in ProVision prior to import.
  Best Practice

  To ensure a fast and straightforward resource import, best practice is to verify ahead of time that your .csv data is correct and contains all the necessary column information for the Section. This includes a top-level Name and Unique ID, as well as a column per Section field. Data edits and column adjustments can be performed inside the Resource Importer Tool if necessary, but will require additional time and steps.

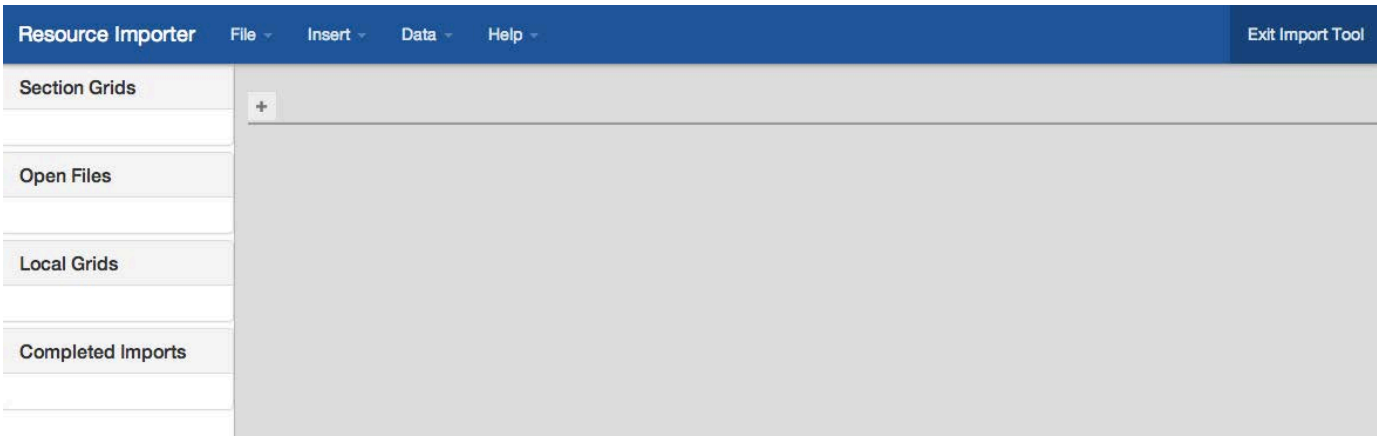
Opening the Resource Import Tool

To open the Resource Import Tool, navigate to the **Data Import Tab** from the **Admin button** to import your aggregate blocks. Select "Resource Import Tool" under "Resource Import".



The Resource Import Tool UI

When you first open the Resource Importer, you will be given the option to view a short on-screen guide to using the tool. After stepping through the guide and/or exiting out of it, the tool will look like this:



On the top are standard menu options of "File", "Insert", and Data" and "Help". Under those menus, you may see greyed-out functions listed. Those functions are items under development, or not available to use at the current Importer step.

On the left side of the screen is a listing of currently opened files:

Sections Grids lists grids currently open that were created from a ProVision Section

Open Files lists the current user created .csv spreadsheets that are open

Local Grids lists any grids that were created in the tool itself, instead of opened from an external file

Completed Imports show imports which have been completed and imported into ProVision

If, at any time, you need to leave the Resource Importer Tool, select the "Exit Import Tool" in the top right corner of the screen, and you will be taken back to the ProVision Dashboard.

Exiting the Resource Importer Tool prior to completing the import process will result in the current open grids being discarded.

Resource Importer Walkthrough

For a step by step walkthrough of the Resource importer, continue on to the Resource Importer Walkthrough , which shows how to import a sample contact list and perform minor editing tasks.

Resource Importer Walkthrough - Step 1 Upload your .csv data file

Resource Importer Walkthrough - Step 2 Open a Template Grid from an existing Section

Resource Importer Walkthrough - Step 3 Reorder .csv columns to match the Section Grid column order

Resource Importer Walkthrough - Step 4 Edit Data as Needed

Resource Importer Walkthrough - Step 5 Drag rows from the .csv Grid to the Section Grid

Resource Importer Walkthrough - Step 6 Import into ProVision

Resource Importer Walkthrough - Step 1

Importing Resources

Before You Begin

Ensure that you are familiar with the overview and "Before you Begin" requirements listed on the Resource Import Tool page.

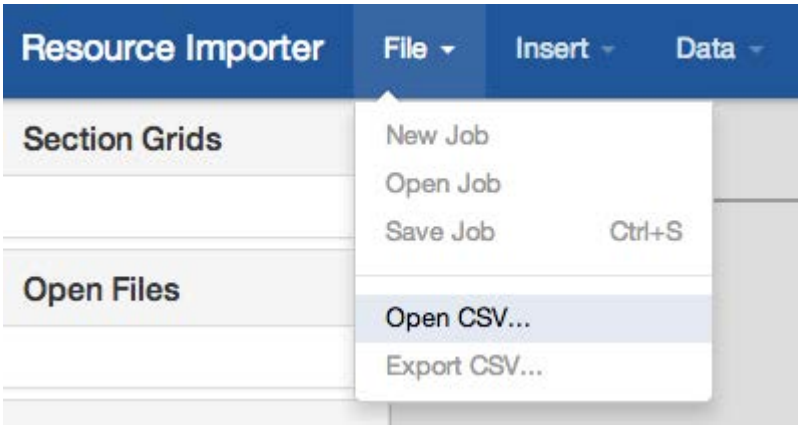
For this tutorial, we will be using the Contact Import Sample .csv available on the Import Templates page as an example, and associating it to an existing Section called "Contact" having the fields: First Name, Last Name, Email, 2nd Email, Phone, 2nd Phone, Mobile Phone, Role, and Time Zone. To create this Section, or edit an existing Section, refer to Working With Resources, Customizing Sections, and Customizing Fields.

In order to illustrate the abilities of the Resource Importer to edit data and adjust for formatting issues, the Contact Import Sample .csv is used intentionally leaving a few less-than-ideal conditions (much like you may encounter in real life) such as leaving typos, having an extra data column, and missing a needed column. If you follow the "Before you Begin" requirements and "Best Practice" notes, however, you may be able to skip any editing or column adjustment steps.

When you are ready to begin, open the Resource Importer and proceed to Step 1.

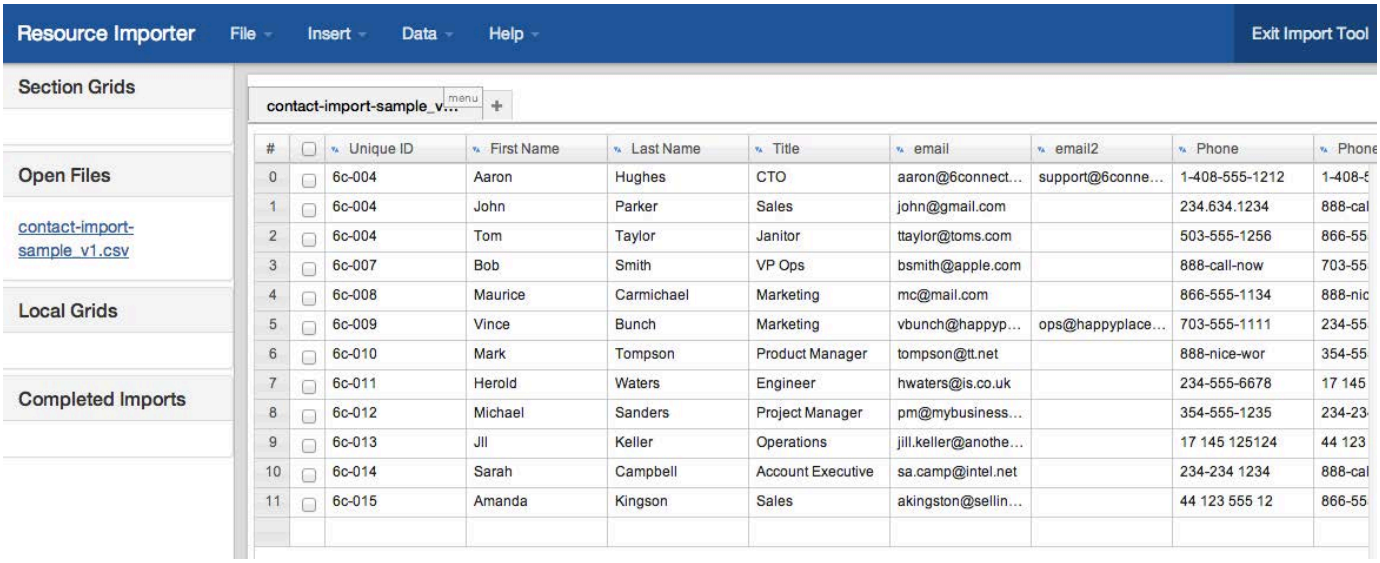
Step 1: Upload your .csv data file

Under the "File" Menu, select "Open .csv". Browse to and select your UTF-8 encoded data file.





After hitting "OK", your file should be visible in the workspace, as well as listed under "Open Files" like this:



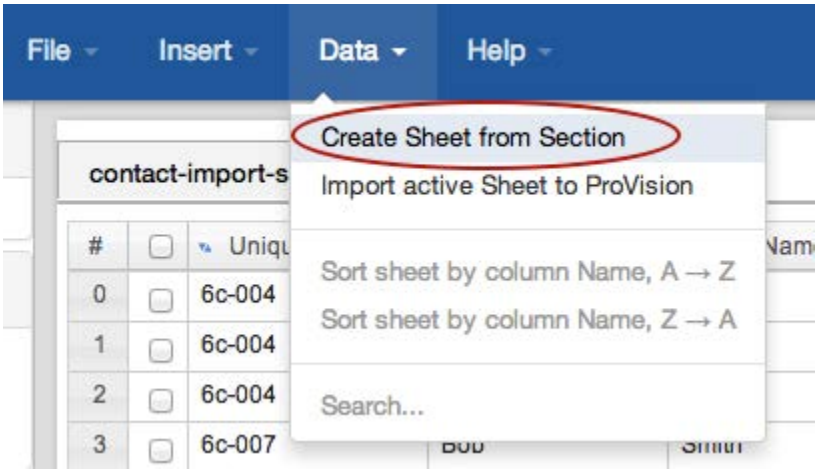
After opening your .csv grid, proceed to Step 2 - Open a template grid from an existing Section

Resource Importer Walkthrough - Step 2

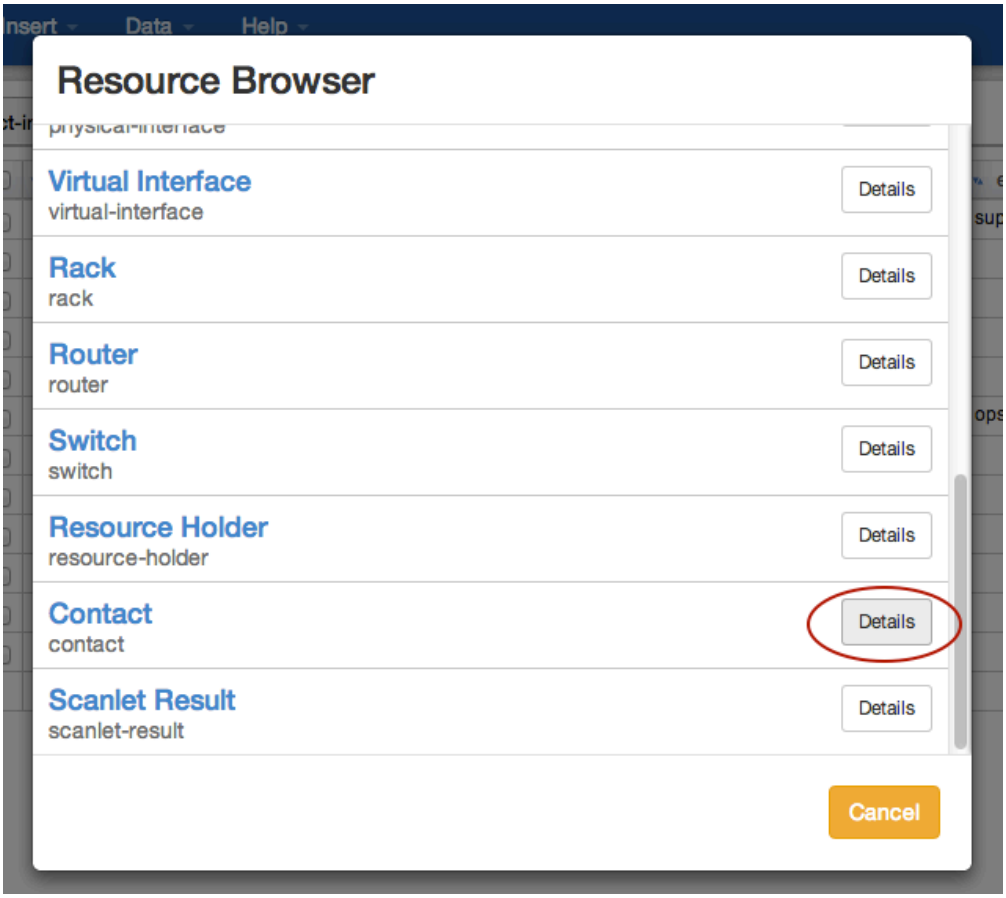
Importing Resources

Step 2: Open a Template Grid from an existing Section

Under the "Data" menu, select "Create Sheet from Section".

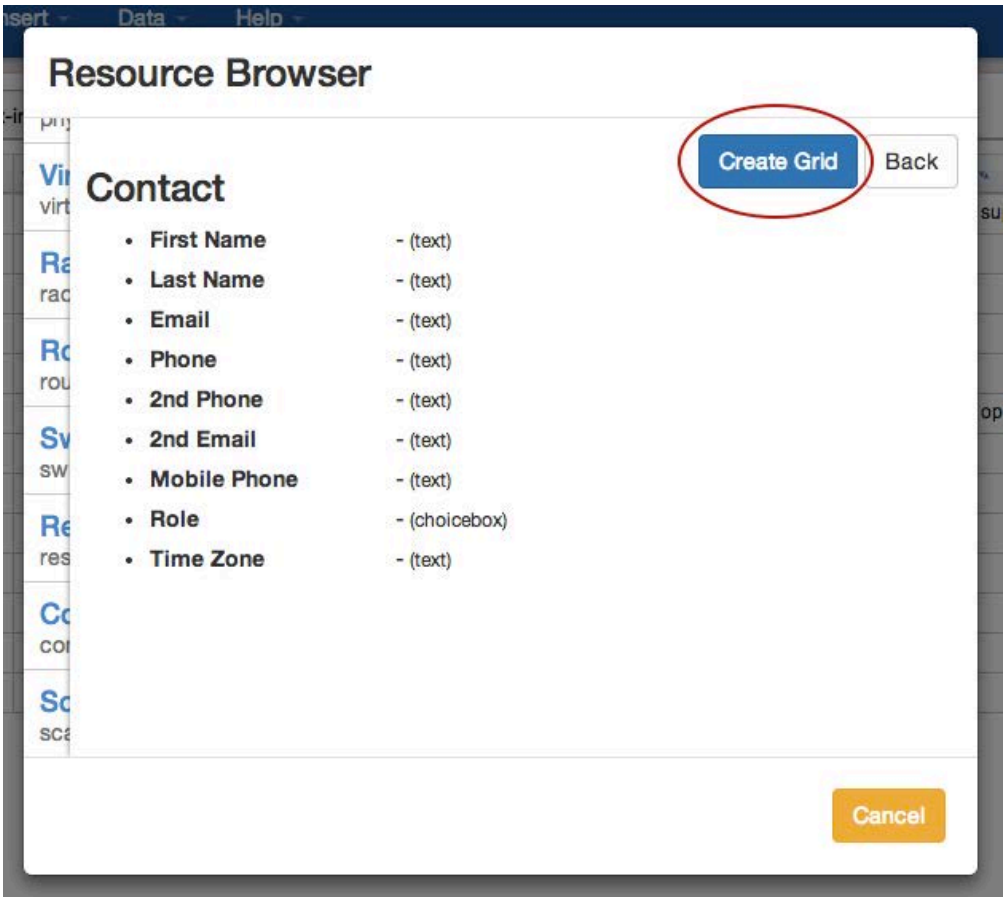


The Resource Browser will pop up, showing the list of Sections currently available in Provision. Clicking on the "Details" button will show the fields for that Section.

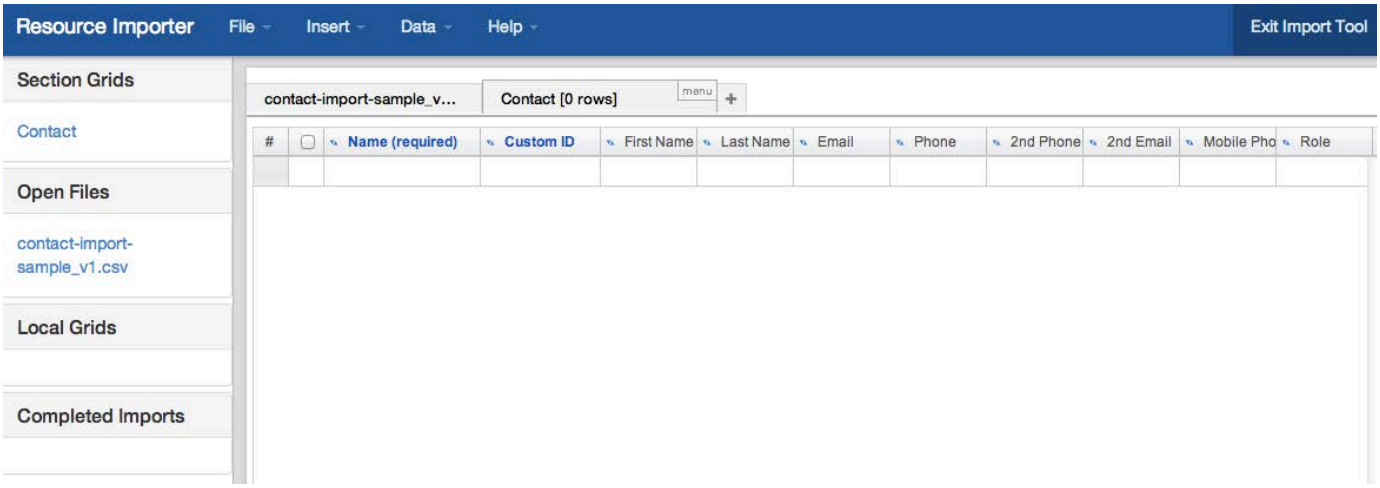


Verify that the Section and available fields match the type of data you are trying to import. In this case, the Section "Contact" has the fields that correlate to our spreadsheet data.

Select "Create Grid" to create a grid based off this Section.



When the Section Grid has been created, required fields will show in blue font with (required) after the header, in this case, "Name" is a required field. The "Custom ID" field is metadata allowing for a unique ID to be associated with each entry, but is not necessary for a successful import. The remainder of the headers directly match the Section's fields.



After you have opened your Section Grid, proceed to Step 3: Reorder .csv columns to match the Section Grid

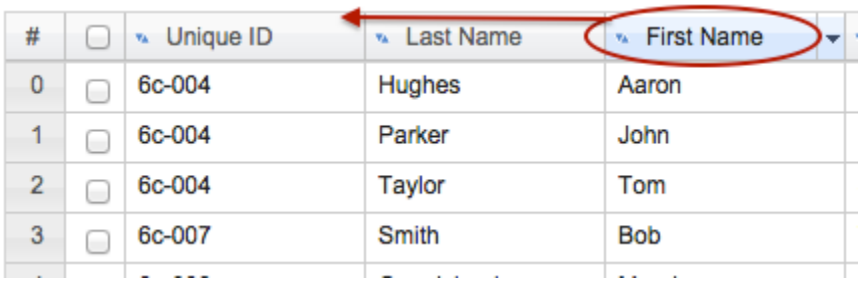
Resource Importer Walkthrough - Step 3

Importing Resources

Step 3: Reorder .csv columns to match the Section Grid column order

One of the most important steps is to reorder the columns from the .csv data to match the order of the Section Grid headers - think of the importer as copying and pasting the csv data into the "Contact" Section grid- we want to ensure that the data is under the correct headers!

Click on the column header to Drag and Drop to the desired location:



#		Unique ID	Last Name	First Name
0	☐	6c-004	Hughes	Aaron
1	☐	6c-004	Parker	John
2	☐	6c-004	Taylor	Tom
3	☐	6c-007	Smith	Bob

Click back and forth between the tabs to verify the column order, then click on a header and drag and drop into the desired order. This moves not only the header, but also the data below it.

Common Column Editing Questions

What if just my column headers are in the wrong place?

What if I have too many / too few columns in my .csv to match the Section Grid?

If you see any of these issues, proceed to Step 4 - Edit data as needed.

Otherwise, if your columns match up perfectly and none of the data needs editing, skip to Step 5 - Drag rows to the Section Grid

Resource Importer Walkthrough - Step 4

Importing Resources

Step 4: Edit data as needed

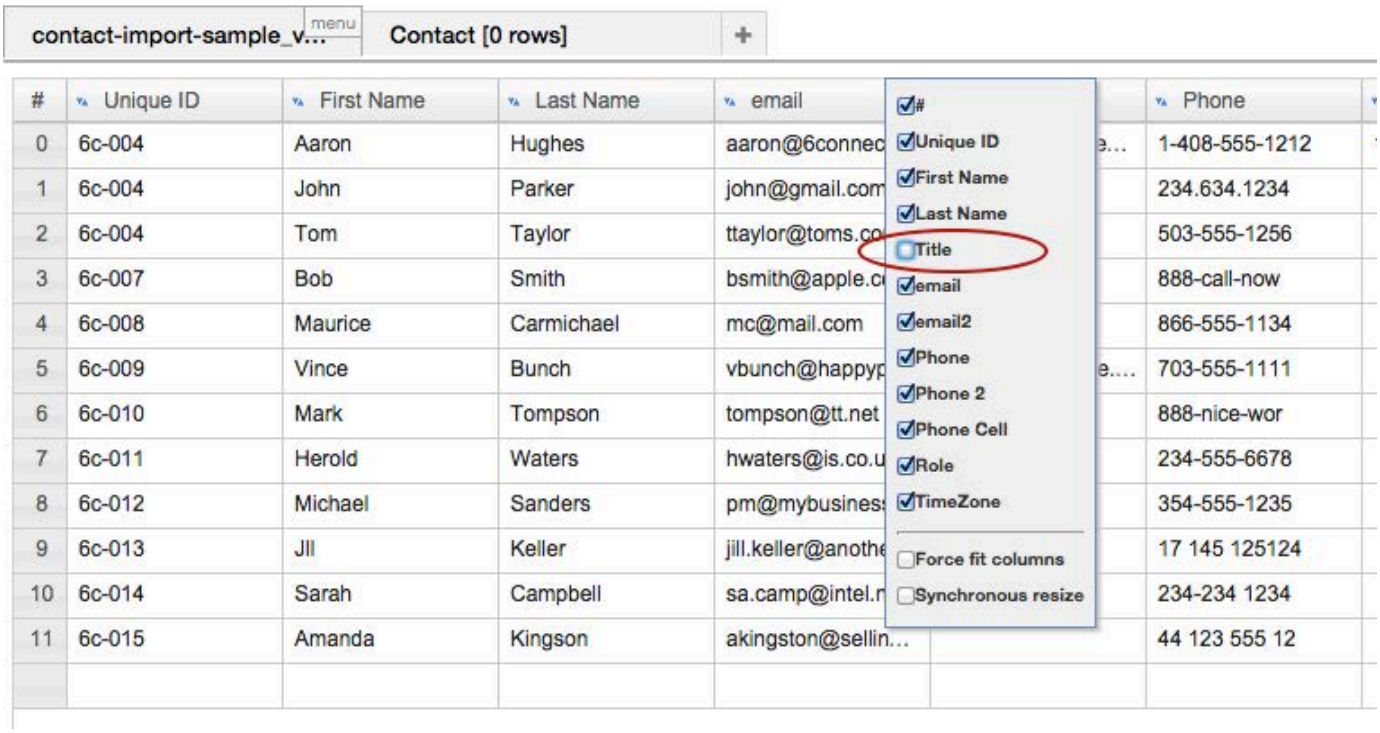
As you may have noticed in Step 3, with this example we have a couple of columns that don't quite match up to the Section Grid. The "Title" column in the .csv data is an additional column we are not tracking in our Section. Also, although we have a "First Name" and "Last Name", we are missing a data column for the top-level "Name" required in the Section Grid.

Common Editing Questions:

- What if I have too many / too few columns in my .csv to match the Section Grid?
- What if I see a typo in the .csv data?
- What if just my column headers are in the wrong place?

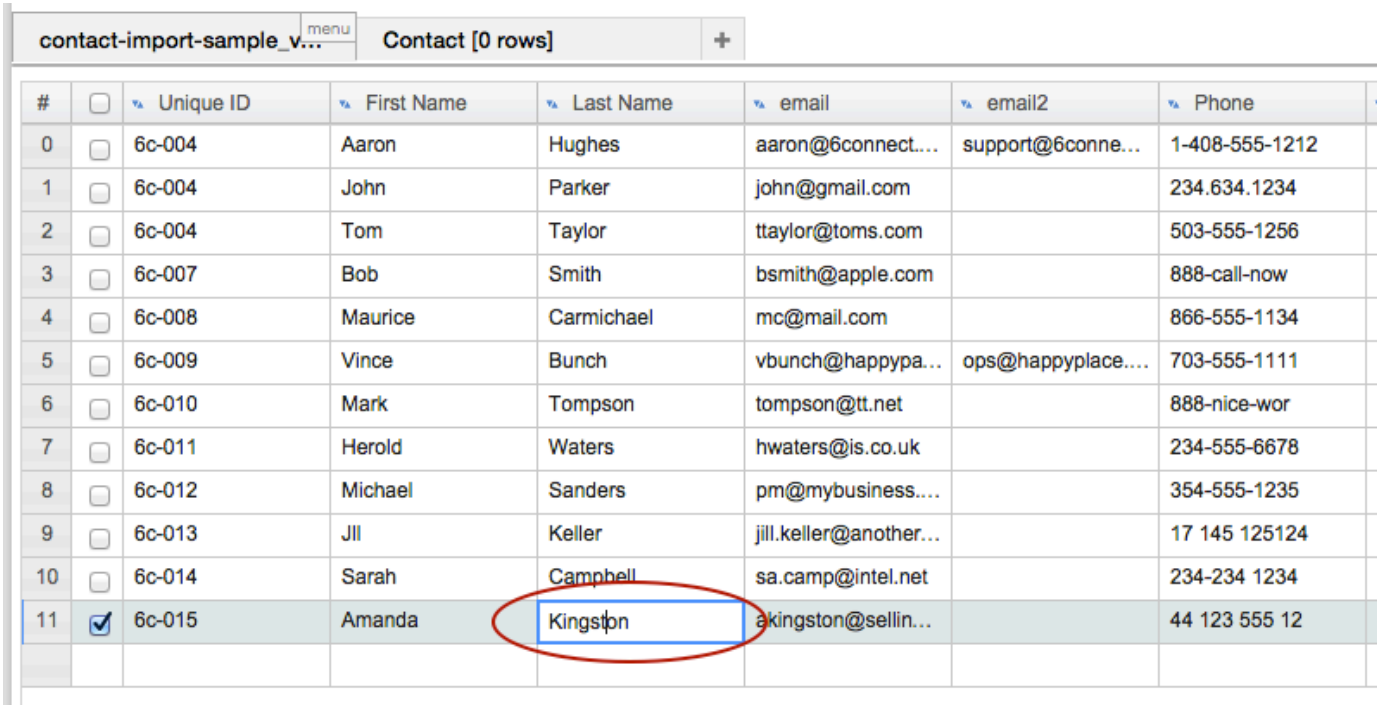
To hide extraneous column information:

Right click on a header and deselect the check box for the column you wish to hide. In this case, we want to hide "Title".



To Edit Data in the Resource Importer

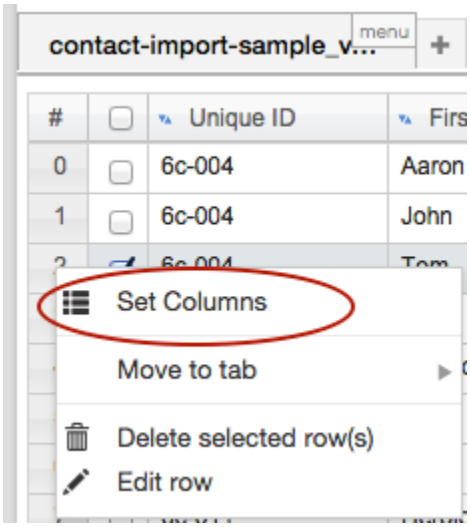
Data in the grids can be edited directly by clicking on the cell(s). In our example, we can see that "Amanda Kingson" should really be "Amanda Kingston". Let's fix that! Click in the cell, type in the edit you wish to make, and then click outside of the cell to exit edit mode. To edit a full row of data, you can right click on the row, select "Edit" row, and make multiple changes in the form box.



If a column header is over the wrong data:

If just the header is in the wrong spot (doesn't match the data below it), you can move just the column header in the Resource Importer, without moving the data below it.

1) Right click on a row of the grid to edit and select "Set Columns":



2) In the "Change Column Header" dialog box, drag and drop the column header(s) into the desired order. Remember, this only moves the headers, not the data below them! Then, hit "OK".



If your .csv data is missing a data column needed for the Section grid:

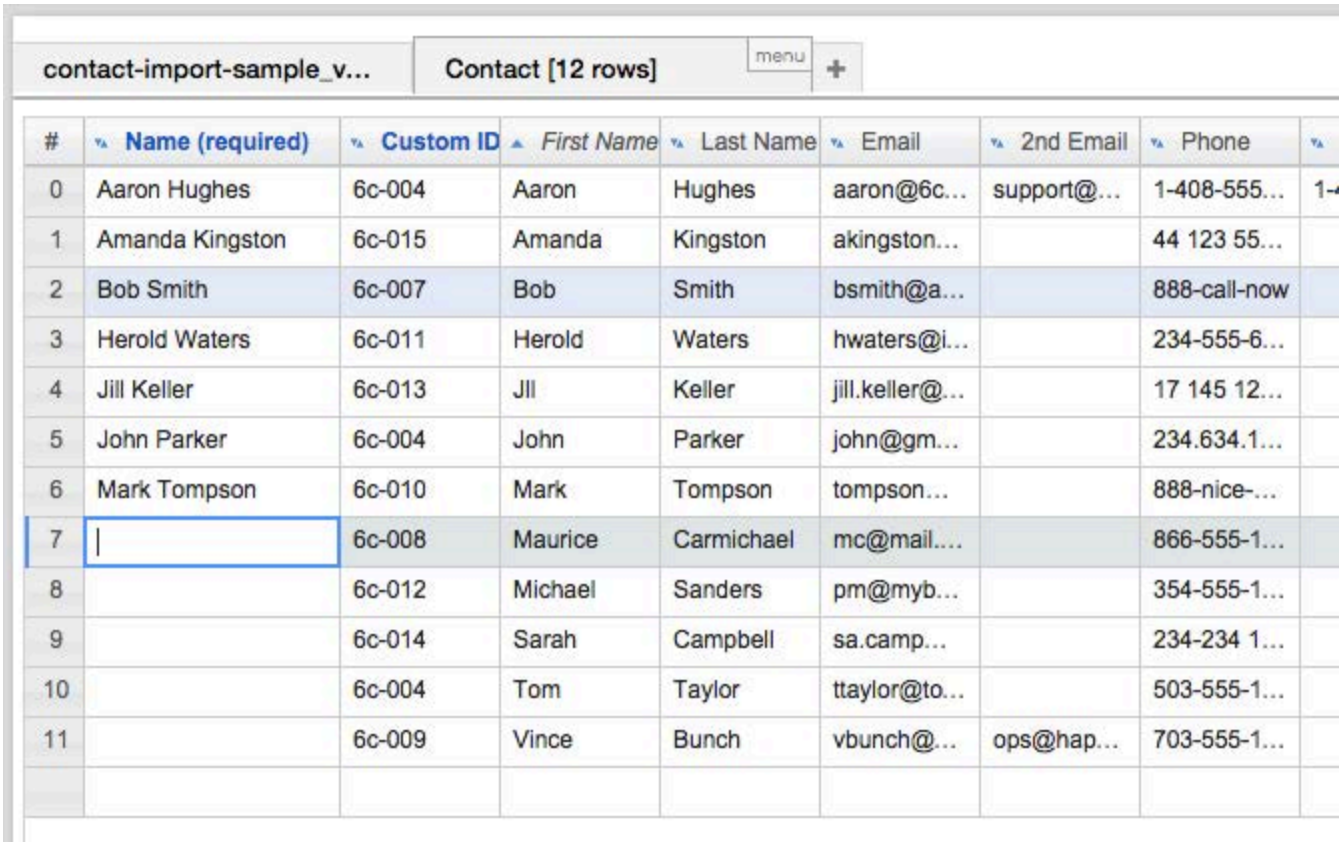
In our case, the .csv data is missing the required "Name" column for the Section grid. Think of the "Name" as the information you would want to search for in Provision. We wouldn't want to search just for "Bob" or "Smith" when looking down a list of names, so under the "Name" column, we need to see the full first and last names, like "Bob Smith".

Currently, our options to fix this are:

1) Edit the .csv directly in your spreadsheet program: (Recommended) Simply revise the .csv to include another column for "Name", and re-open the .csv in the importer. The benefit to this method is your .csv file will be set up as a template for future imports.

Or:

2) In the Resource Importer, temporarily hide the extra column in the Section Grid: Make the columns between the .csv and the Section Grid match exactly by temporarily hiding the column (in this case, "Name") in the Section Grid, proceed to move the data into the Section grid (see Step 5), then unhide the "Name" column and manually add the data as needed prior to completing the import.



#	Name (required)	Custom ID	First Name	Last Name	Email	2nd Email	Phone	
0	Aaron Hughes	6c-004	Aaron	Hughes	aaron@6c...	support@...	1-408-555...	1-
1	Amanda Kingston	6c-015	Amanda	Kingston	akingston...		44 123 55...	
2	Bob Smith	6c-007	Bob	Smith	bsmith@a...		888-call-now	
3	Herold Waters	6c-011	Herold	Waters	hwaters@i...		234-555-6...	
4	Jill Keller	6c-013	Jll	Keller	jill.keller@...		17 145 12...	
5	John Parker	6c-004	John	Parker	john@gm...		234.634.1...	
6	Mark Tompson	6c-010	Mark	Tompson	tompson...		888-nice-...	
7		6c-008	Maurice	Carmichael	mc@mail....		866-555-1...	
8		6c-012	Michael	Sanders	pm@myb...		354-555-1...	
9		6c-014	Sarah	Campbell	sa.camp...		234-234 1...	
10		6c-004	Tom	Taylor	ttaylor@to...		503-555-1...	
11		6c-009	Vince	Bunch	vbunch@...	ops@hap...	703-555-1...	

When edits and adjustments are complete, move to Step 5 - Drag rows to the Section Grid

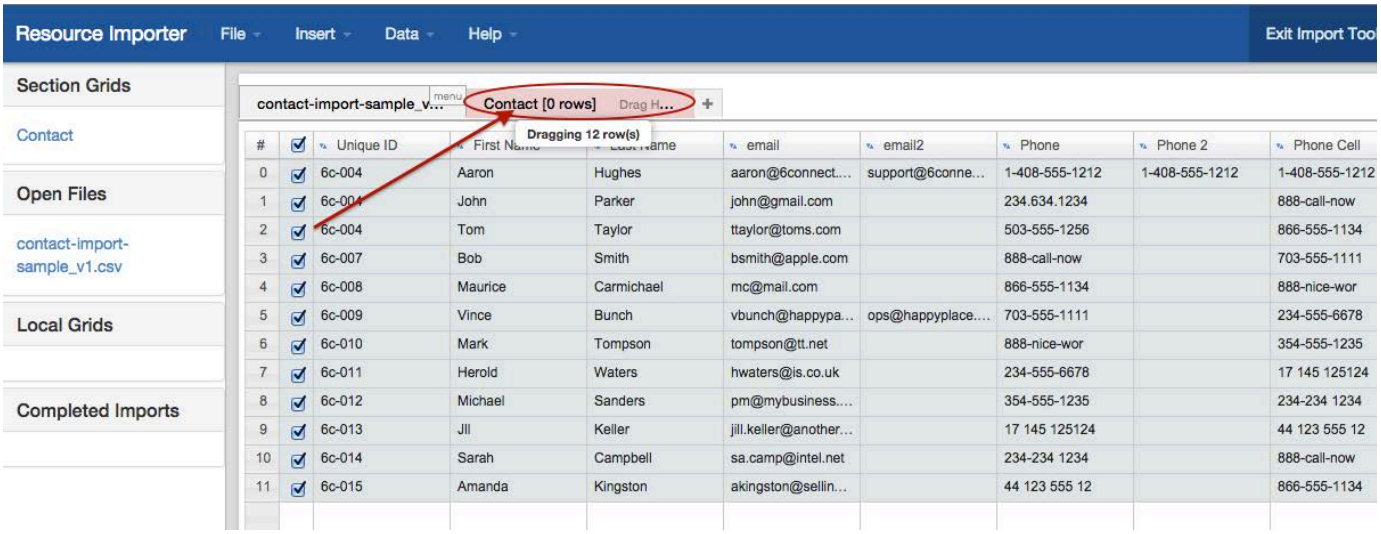
Resource Importer Walkthrough - Step 5

Importing Resources

Step 5: Drag rows from the .csv Grid to the Section Grid

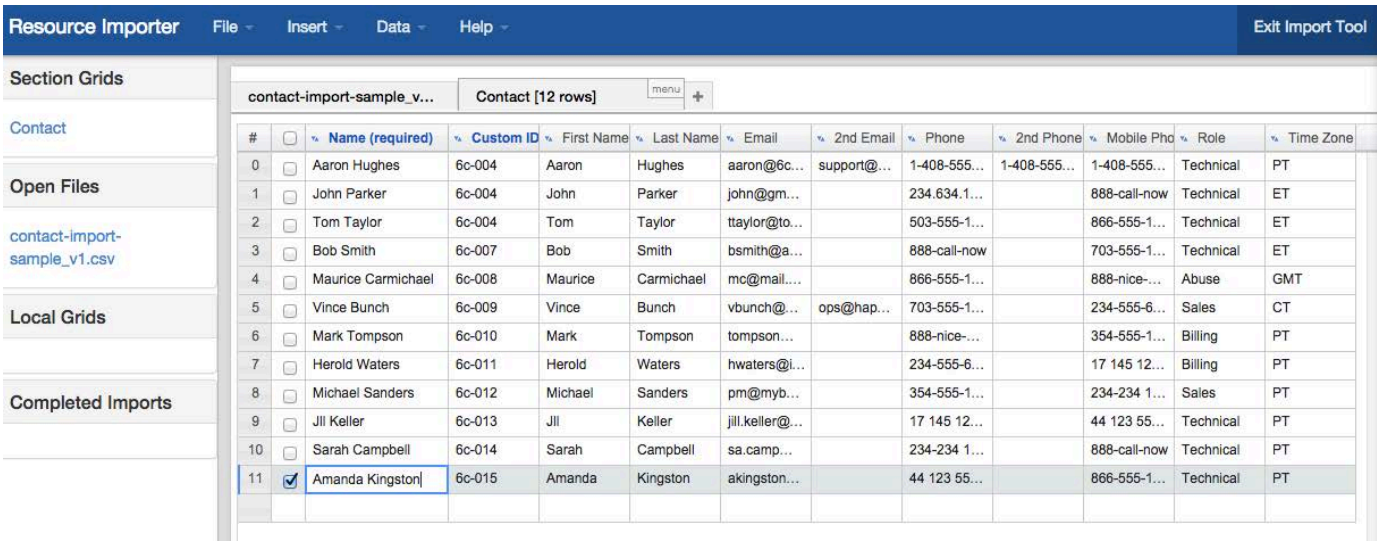
Once you have set the columns to match exactly between the .csv Grid and the Section grid, it's time to pull in the data from one to the other.

Simply click the checkboxes for the rows you wish to import (or use the "Select all" checkbox at the top), click anywhere on the row, and drag & drop onto the Section Grid tab ("Contact"). The tool will tell you how many rows you are moving as you drag them.



Click on the "Contact" tab when you are done, and you will now see your data in there, instead of the original .csv.

If you had to hide columns in the Section Grid prior to moving the .csv data, verify that all columns are visible and the required data filled in. In this case, we filled in the "Name" Column that was missing in the original .csv.



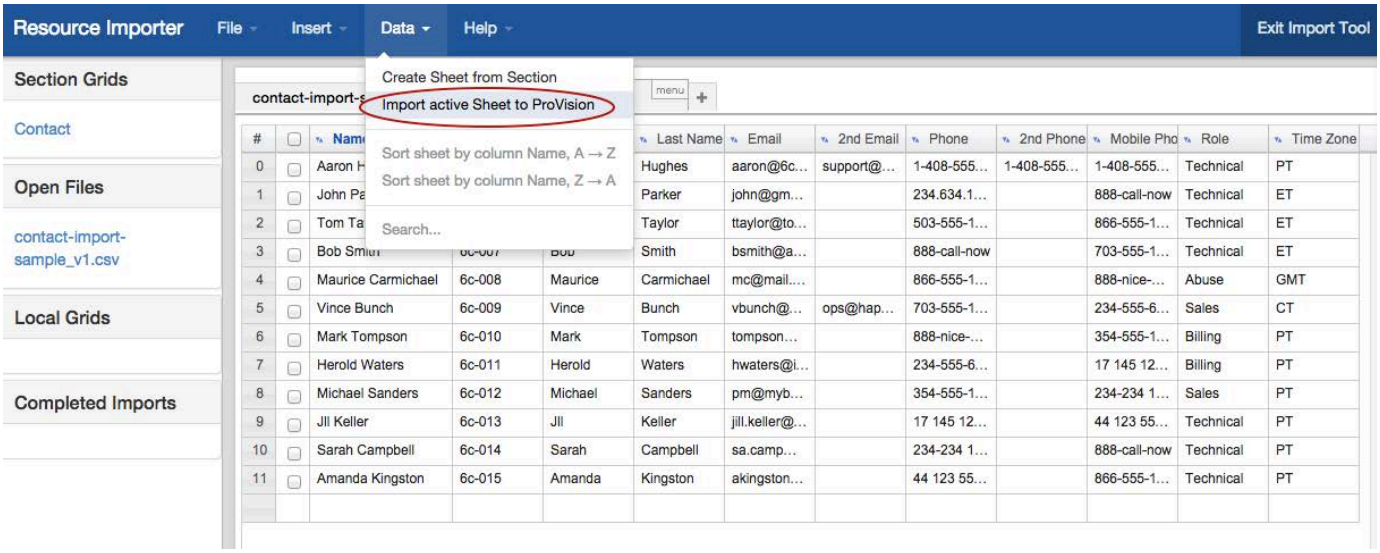
After moving your data into the Section grid, proceed to Step 6 - Importing into ProVision.

Resource Importer Walkthrough - Step 6

Importing Resources

Step 6: Import into ProVision

When all of the data is under the Section Grid tab, and any required field data filled in, you can import the data into Provision! From the Data menu, select "Import active Sheet into ProVision". You will see an import progress bar. Once complete, you data will be in provision, filled into the Section fields for your chosen Resource.





Import Sessions

Importing Sessions

- Importing Sessions
  - Peering Import
    - Load Router Sessions
    - Edit Sessions (Optional)
    - Select Groups and Sessions
  - Additional Information

Peering Import

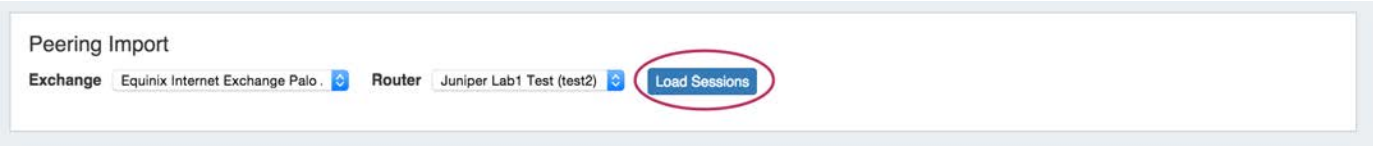
Importing peering sessions requires Admin-level permissions, and is accessible only from the Admin section of ProVision.

From the Admin section of ProVision, navigate to the **Data Import Tab.** Under Peering Import, select **Import BGP Sessions.** This will take you to the Peering Import section of ProVision.



Load Router Sessions

First, select the desired exchange and router. Routers with Logical Systems information will show up as the router name with the Logical System info in parenthesis (e.g. "Juniper (test)"). Then click "Load Sessions".



Peer Group and Sessions will then display below your selections.

Edit Sessions (Optional)

The available peer Groups and Sessions will display below your selected exchange and router.

If edits need to be made to the session prior to import, click on the wrench icon to open field edits.



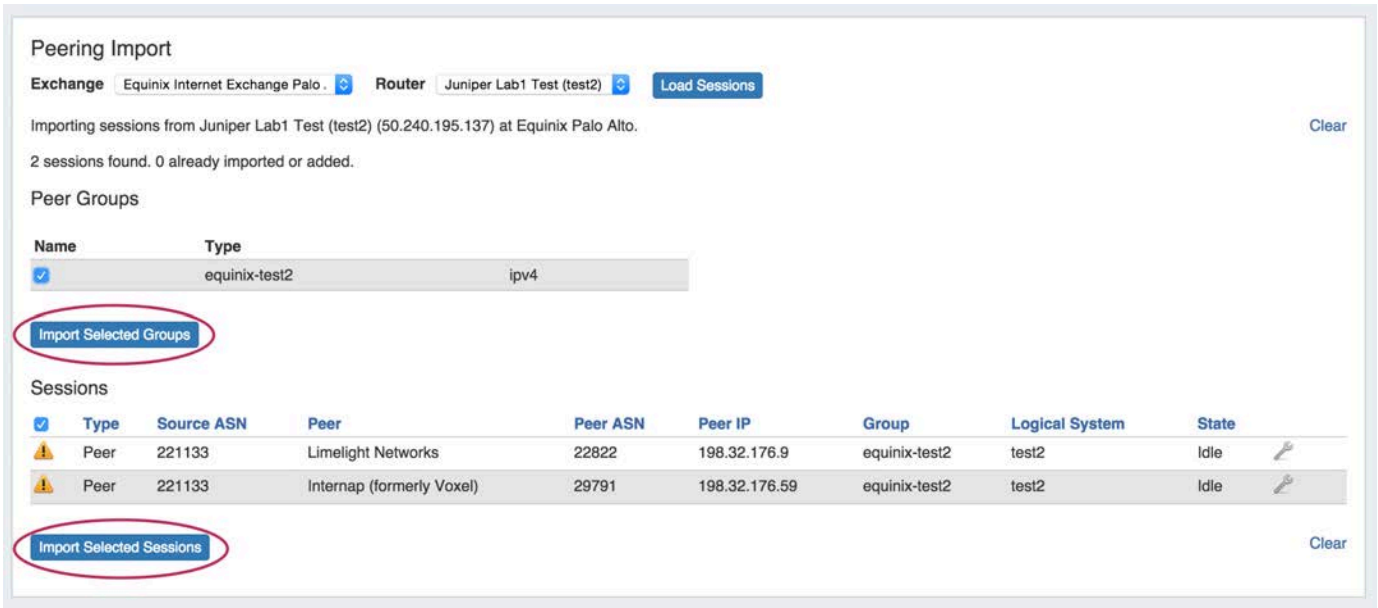
Make the changes to desired field(s), then click "Done" to save your changes and proceed to importing.



Select Groups and Sessions

Lastly, select the check box next to each Session to import (or the check box at the top to select all sessions) and click "Import Selected Sessions".

Successful imports will then display with a green check mark at the beginning of the row.



Once imported, you can manage and configure your sessions from the Peering Tab.

Additional Information

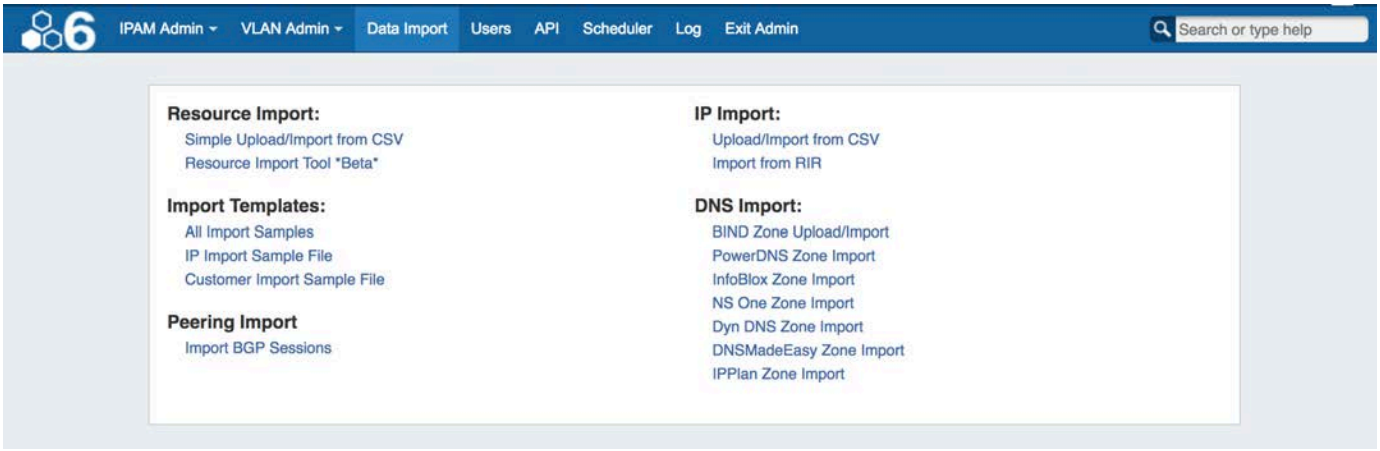
See the following areas for more information on working with Peering tasks:

- Peering v2
- Peering Exchanges
- Peering Routers
- Peering Sessions

IP Import from CSV

Simple Upload / Import from CSV

The Upload / Import from CSV tool is used if you have a simple .csv file with IP block information, such as CIDR, Mask, ASN, RIR, etc. It is accessed from the **Data Import Tab** from the **Admin** section of ProVision.



To import your IP block information, follow the following steps:

- Simple Upload / Import from CSV
  - Methodology
  - Before you Begin: Prepare your Data for Import
  - Import Your Data
    - Step 1: Create a new IP Import Job
    - Step 2: Define Columns
    - Step 2.5: Missing References
    - Step 3: Reviewing Data
    - Step 4: Execute Import
    - Post Import Followup
  - Additional Information

Methodology

The Import from CSV tool will create Top-Level Aggregates and place blocks under those aggregates based on the following method:

First, the importer will parse through the provided data, order all blocks from largest to smallest, then attempt to split the largest block out of an existing block matching the IP space and RIR. If that fails (no larger block exists), then that block is added in the system as its own Top-Level-Aggregate. Subsequent blocks will undergo the same process.

This method ensures that your list of IP blocks does not need to be organized in any particular order.

> If your import includes a large number of small blocks (/30s, /32s), be sure you've included at least one large block which encompasses them (/24, /22). This will ensure that the smaller blocks are neatly organized under the larger block, rather than imported as their own Top-Level Aggregates.

Before you Begin: Prepare your Data for Import

The Upload / Import from CSV tool requires only a .csv file for importing.

Be sure to review "Preparing Data for Import" on the Importing your Data page before you begin. Verify that your .csv is correctly parsed and cleanly formatted with your IP block information (using your choice of the available field list shown below), and is UTF-8 encoded for best results.

Common normalization steps required for ProVision IP imports include:

- Remove any empty rows in the data range
- Remove any empty columns in the data range
- Remove stray data from inside and outside the data range, such as:
  - notes, comments, and placeholder text as field values, when not set as their own data column
  - nested secondary information located in additional rows under a primary record

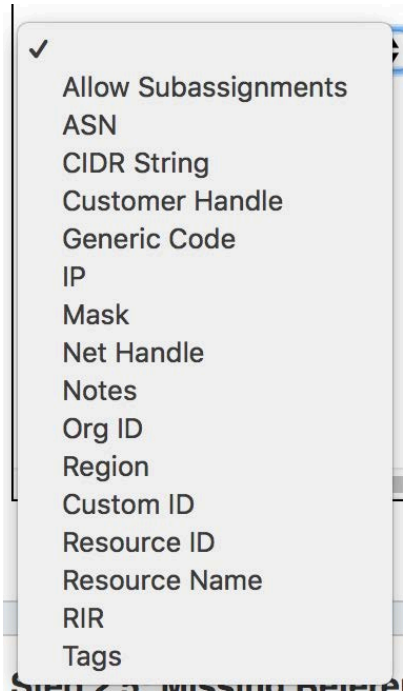
- Ensure proper data formatting
  - Check for data formatting errors - a common issue is leading zeros being dropped from a ZIP field when a file is opened. Re-specify format as needed.
  - Ensure each field has consistent formatting for values

Data Encoding Format

To ensure correct importing of any special characters, make sure to use UTF-8 encoding for your CSV file!

This tool supports the following fields:

Click here to expand...



Required Fields

For IP Import from .csv, the following fields are REQUIRED:

- CIDR String OR both IP and Mask
- RIR OR a selected default RIR

All other fields are optional.

Import Your Data

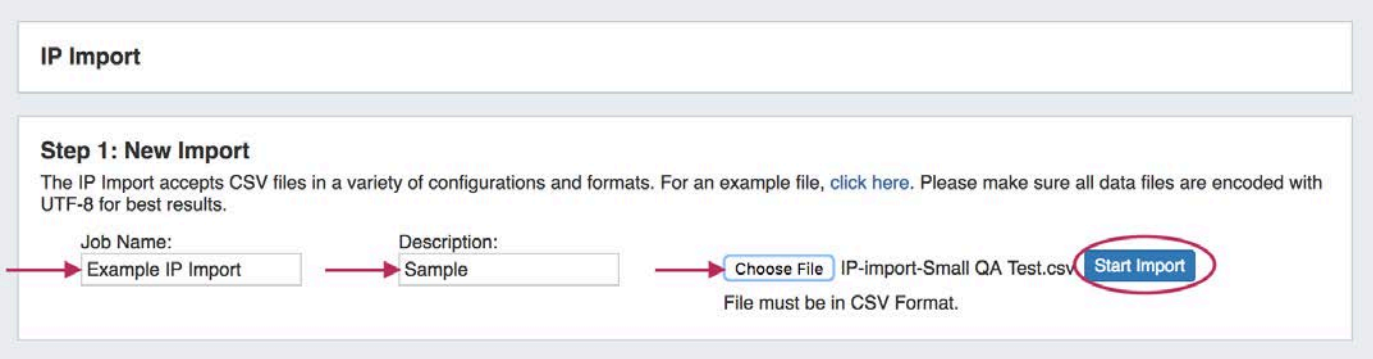
Step 1: Create a new IP Import Job

Navigate to the **Data Import** Tab from the **Admin** button. Select "Upload / Import from CSV" under "IP Import".



Create a Job Name and Description for the import. This is especially useful to keep track of progress in cases the data arrives from multiple sources, or will require multiple stages of manual review.

Select the .csv file that you prepared above by selecting the "Choose File" button, and browsing to the correct file location. Then hit "Start Import".

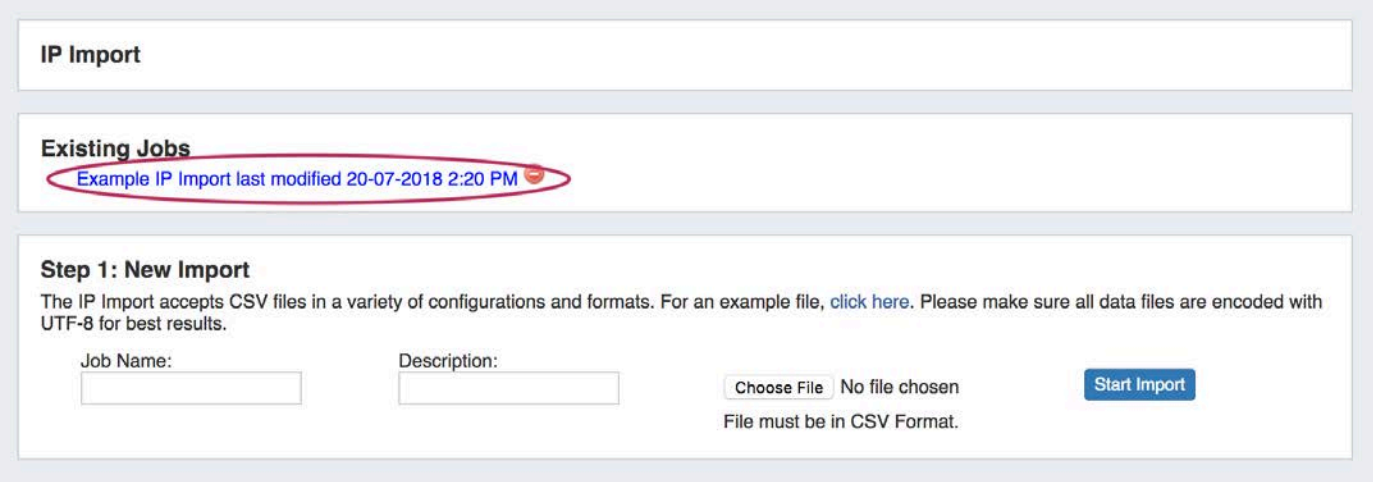


Working with Large or Multiple Data Sets

Although you cannot add new files to an existing job, for jobs with multiple sources for data (which may have different formatting), you can simply create separate jobs and descriptions for each source - no need to manually combine the data into one file before importing. The Import tool's mapping and editing functions will allow for the data to be reconciled in ProVision.

For large data sets where multiple stages of manual review might be needed, you can create a new job using the same set of data files in order to work in parallel on a different portion of the data.

After importing, the new job will appear under the "Existing Jobs" section. To continue working with this job, select it from the list (by clicking on the link) and the next step will appear on the page.

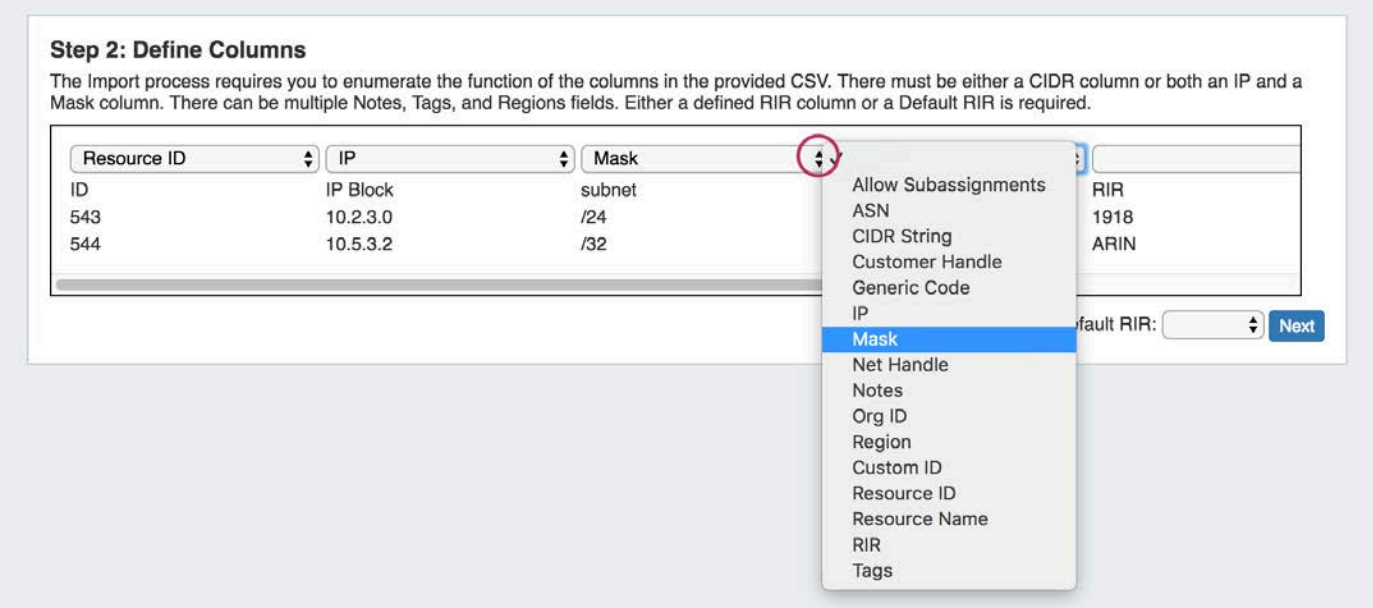


Step 2: Define Columns

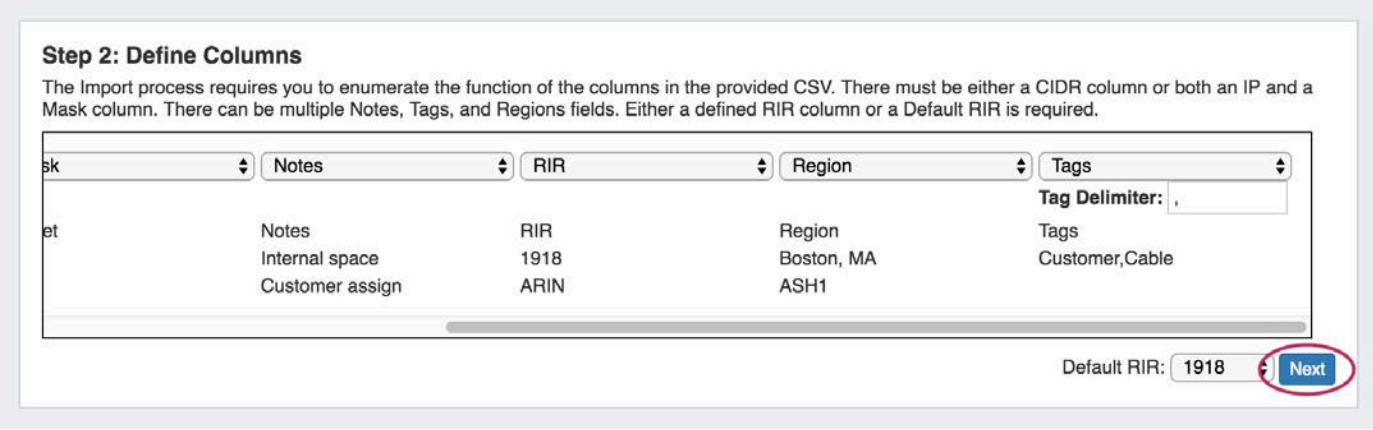
Using the dropdown menu above each data field, select the appropriate definition for each of the imported columns.

- **CIDR** or **IP/Mask** fields must be provided
- **Notes, Tags,** and **Regions** may have multiple columns associated with the data.
- If you do not have a defined **RIR** column, you must select a default RIR to associated the blocks to.

Other columns which do not apply under the available definitions should be left as blank, and will be skipped during the upload process.



Make sure that you have defined all desired fields by using the scroll bar below your data to view additional columns.



When completed, hit "Next".

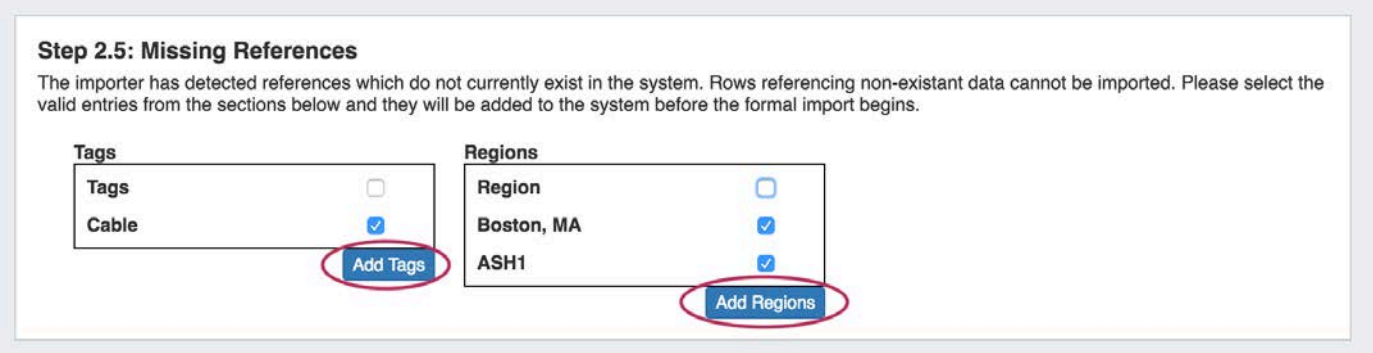
Step 2.5: Missing References

If references exist in your data that do not currently exist in ProVision (such as new tags or resources) the import tool will alert you to the missing references, and give you the option to create those references in ProVision. If you wish to add these elements, select the checkboxes next to the element and click the "Add (Element name)" button below.

Header Rows

If your .csv has a header row as the first line, that row may give information that produces a missing reference notice - such as "Tag", "Region", "Block", and so forth.

If this is the case, simply uncheck that item and add only the desired references.



In this example, we get a notification that we are missing a tag named "Tags". However, we know by looking at our data that is simply an item that was part of the header line in the .csv, so we want to bypass this step.

Step 3: Reviewing Data

After supplying the file and defining columns, a review step is provided. Records with errors will show as color coded, and can be filtered to be viewed by All, Valid, Warnings, Invalid, or Ignored.

From here, the records can be edited or ignored. Select "Ignore" for records that you do not wish to import at this time. Records that are Ignored or Invalid will automatically be skipped.

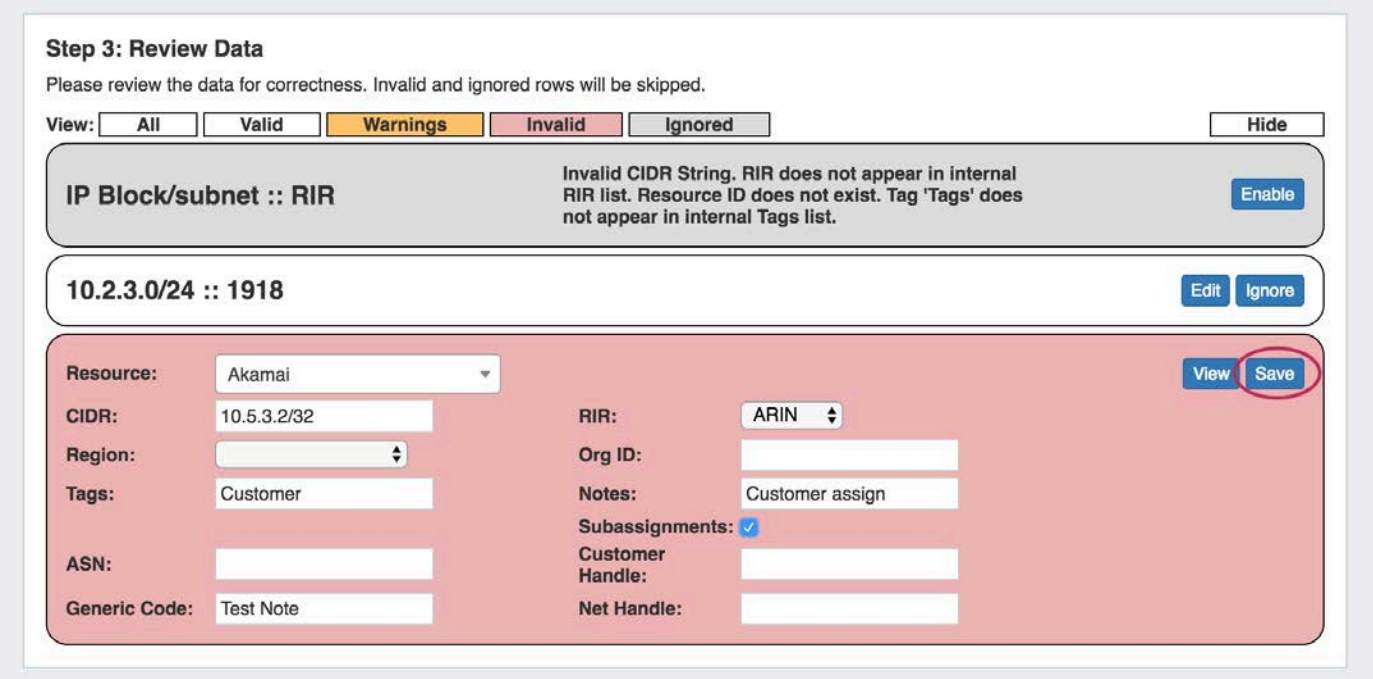
Header Rows

If your .csv has a header row as the first line, that row will be shown as the first record in review data as well. Simply click "Ignore" on the first record to disregard the row.

If warnings or invalid entries are found, you may edit the data by clicking the "Edit" button.

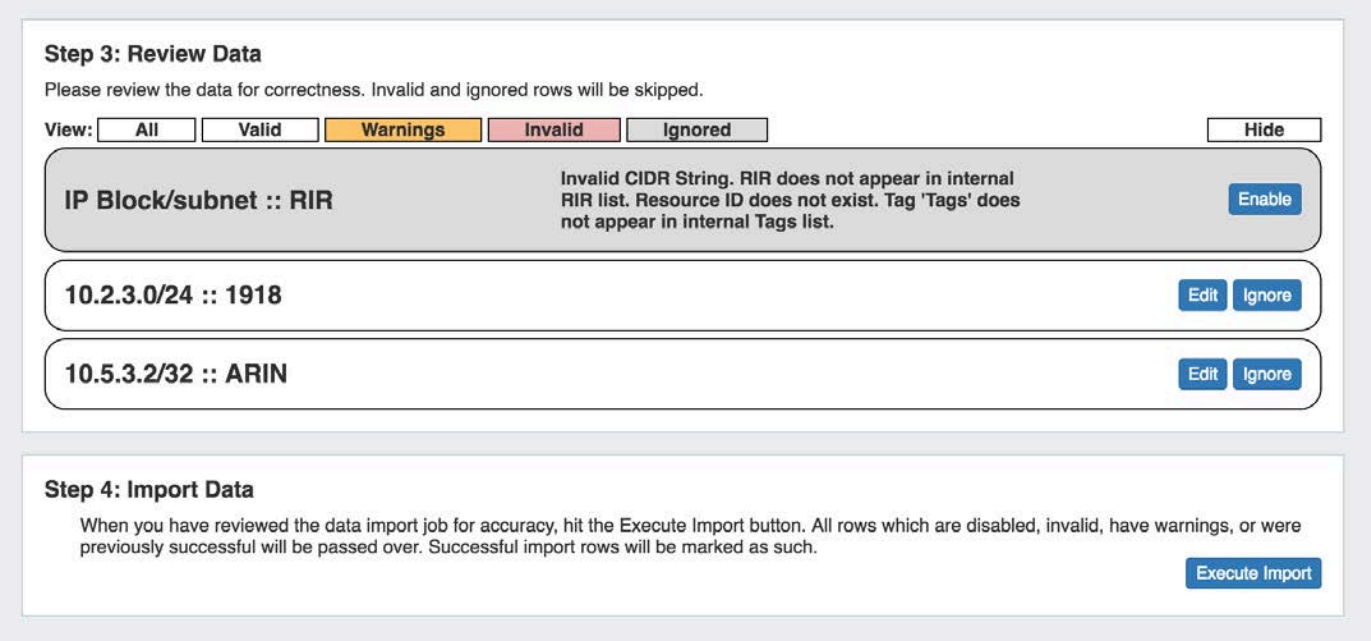


Here you can edit the invalid field, or add /change information for any other available field.



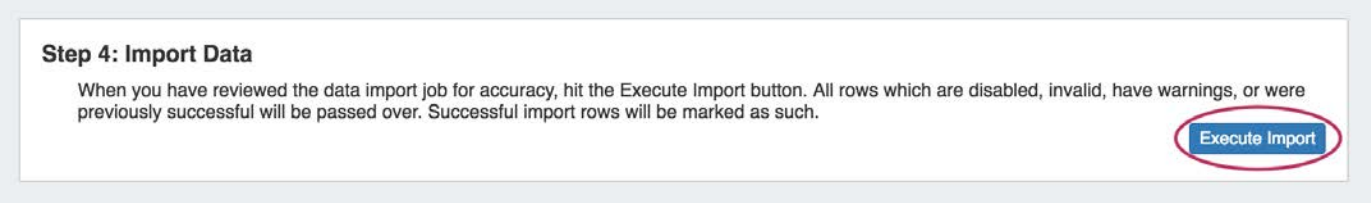
After editing, hit "Save". If the edits are valid, the background color for the data row will be white.

Continue reviewing and editing data as desired. When done, proceed to Step 4.

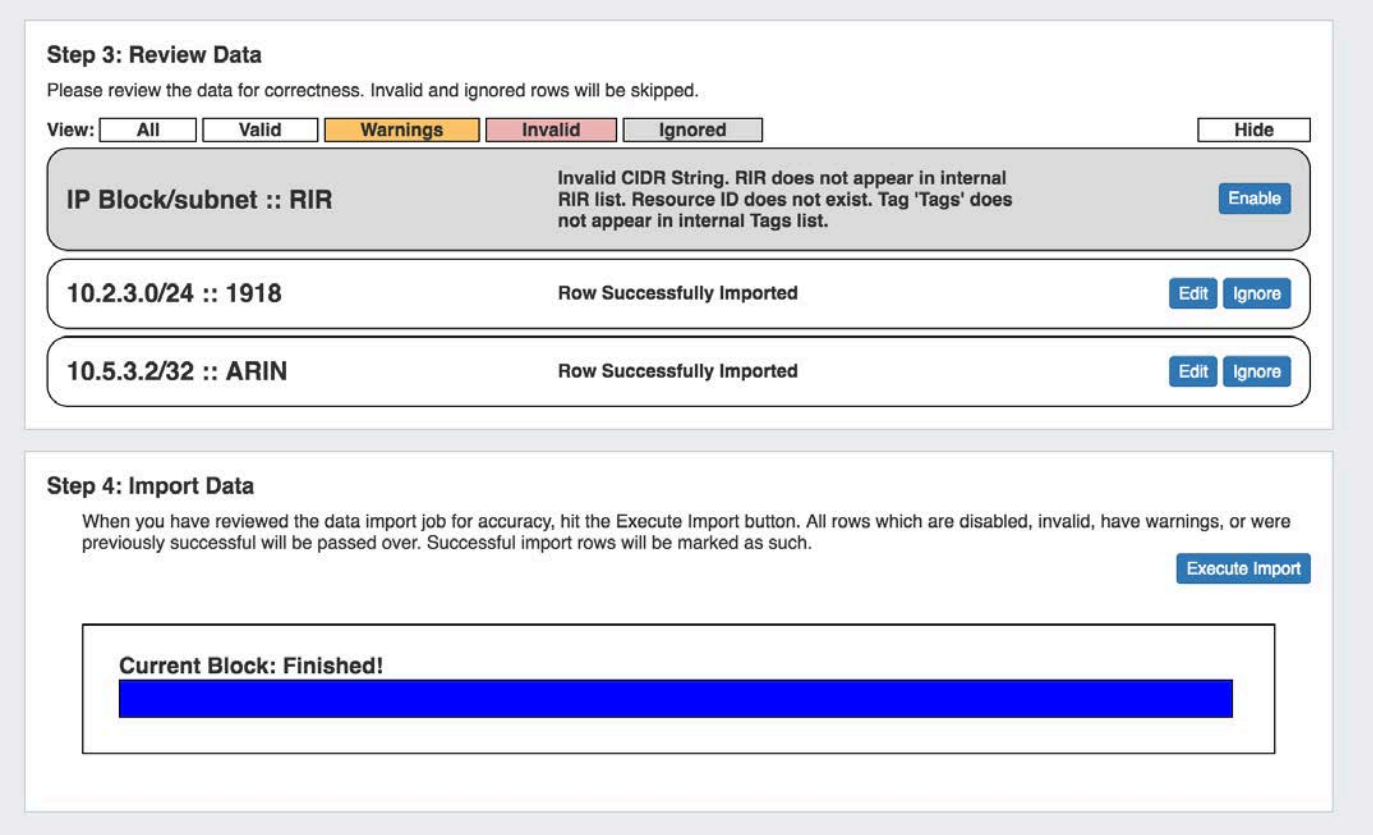


Step 4: Execute Import

When the review step is completed, hit the "Execute Import" button. A progress bar will appear to show progress and note errors if they occur.



When the bar reaches 100%, the import is complete. The Data Review section will show which data rows were successfully imported.

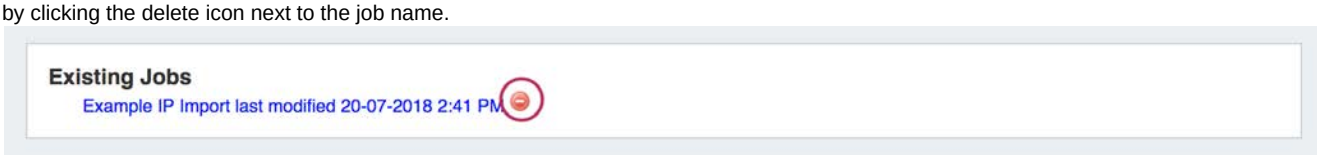


Post Import Followup

After the Import is complete, there are a few followup options for working with the import job or imported blocks. We recommend:

- Reviewing the imported blocks in the IPAM Tab, adding additional IPAM data, custom metadata, rules, or subassignments. See Working with IP Blocks and Working with IP Rules.
- If the IP Import was only partially successful or requires additional review, you may keep the import job in the "Existing Jobs" list and revise the failed / invalid blocks at a later time. Re-running the import from the same job will allow you to re-attempt import on only the previously unsuccessful / ignored / edited blocks - successfully imported and unchanged blocks will not be affected.

If the IP Import was fully successful, and you do not wish to retain the job data, the Import Job may be deleted from the "Existing Jobs" list by clicking the delete icon next to the job name.



Deleting an import job is permanent and non-recoverable!

When working with large or sensitive import jobs, consider performing a backup of your instance before and after the import job as a safeguard.

Additional Information

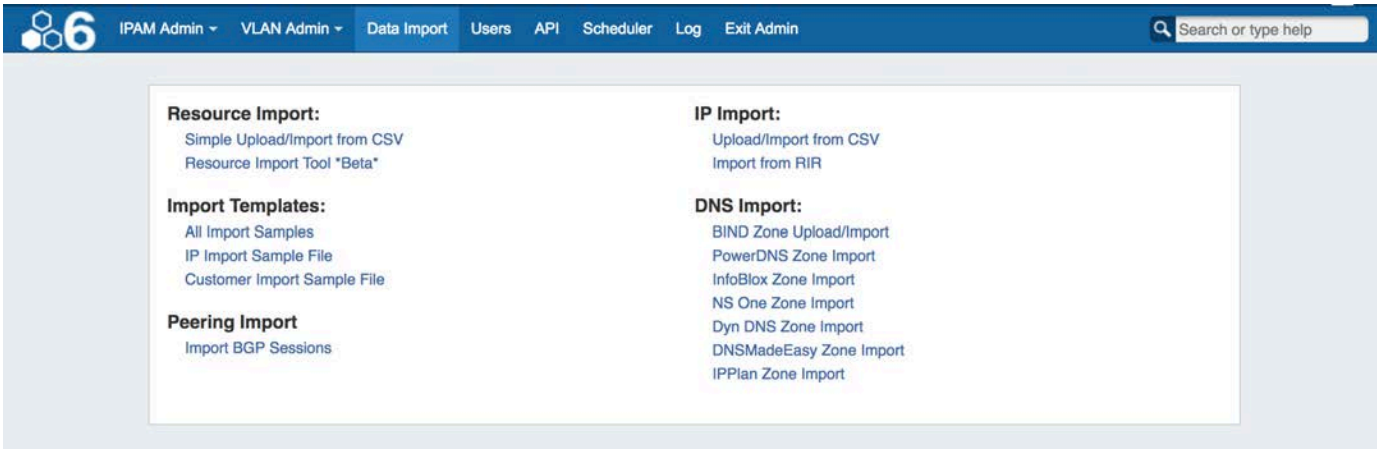
See the following areas for information on working with IPAM / IP Blocks post-import:

- IPAM Tab
- Working with IP Blocks
- Working with IP Rules
- IPAM Administration

Import Aggregate Blocks

Import Aggregates

Importing blocks from your RIR Source IP is available from the **Data Import Tab** in the Admin area of ProVision. This feature uses the RIR Org ID to identify a Source IP and determine blocks assigned to your organization that are available for import.



- Import Aggregates
  - Import from RIR
    - Step 1: Lookup from Source IP
    - Step 2: Import your aggregate blocks
    - Step 3: Customizing Block Data
  - Additional Information

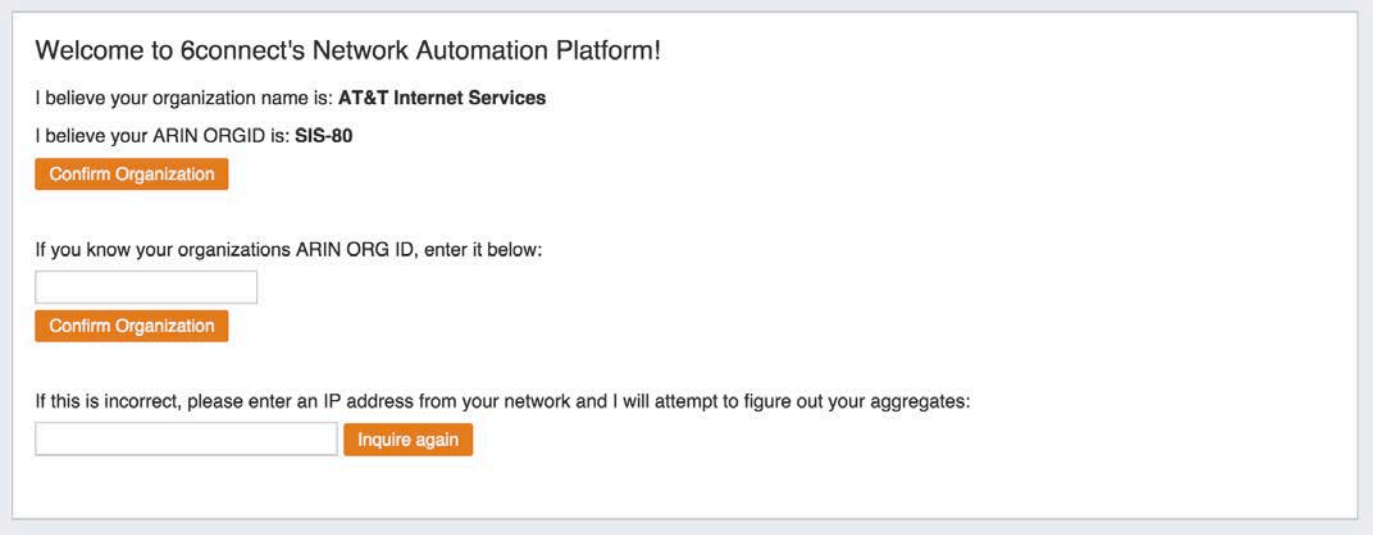
Import from RIR

From the **Data Import Tab,** select "Import from RIR" under "IP Import".



Step 1: Lookup from Source IP

We automatically lookup your ARIN or RIPE information based on the IP address you are connected to:

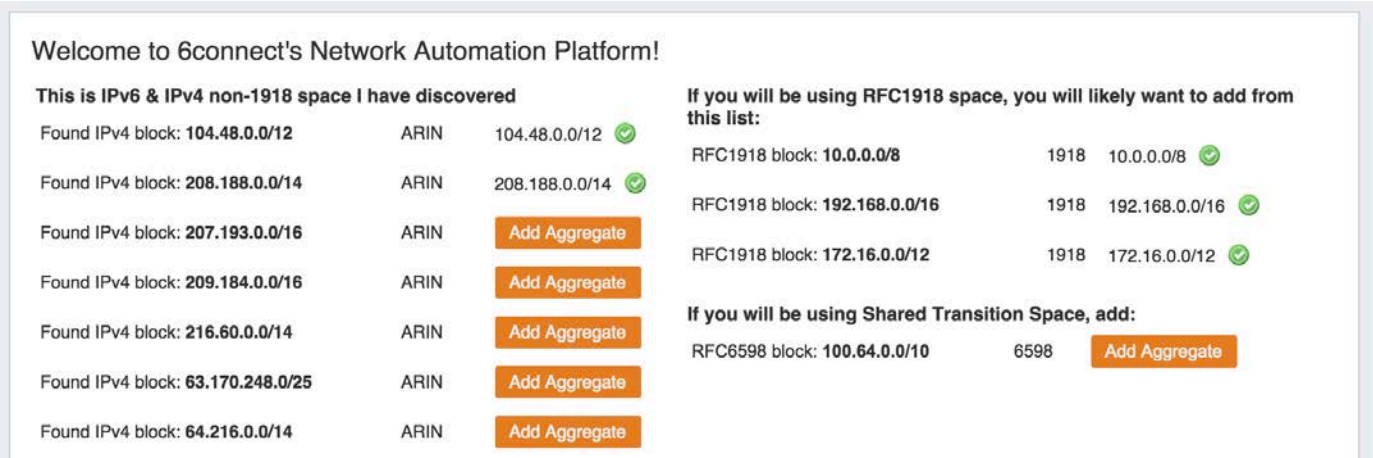


If you have another source IP that you would like to use for the lookup function, you can edit the IP and click on the "Inquire Again" button.
If the organization name and ORGID are correct, then click on the "Confirm" button to go to the next screen.

Step 2: Import your aggregate blocks

Once we have identified the blocks assigned to your company, you can import the aggregates by pressing the "Add Aggregate" buttons.

This page allows you to add both 1918 aggregates as well as public IP space from ARIN and RIPE.



When done adding aggregates, hit the "Continue to ProVision" button at the bottom of the page.



You will be redirected to the **IPAM** tab to manage the aggregates as desired.

Step 3: Customizing Block Data

With your aggregates added, you are now ready to customize the tool and import additional data! From here, you can manage your aggregates under the **IPAM** tab, edit administration functions under **IPAM Admin**, or import resources using the Resource Import Tool.

Additional Information

See the following areas for information on working with IPAM / IP Blocks post-import:

- IPAM Tab
- Working with IP Blocks
- Working with IP Rules
- IPAM Administration

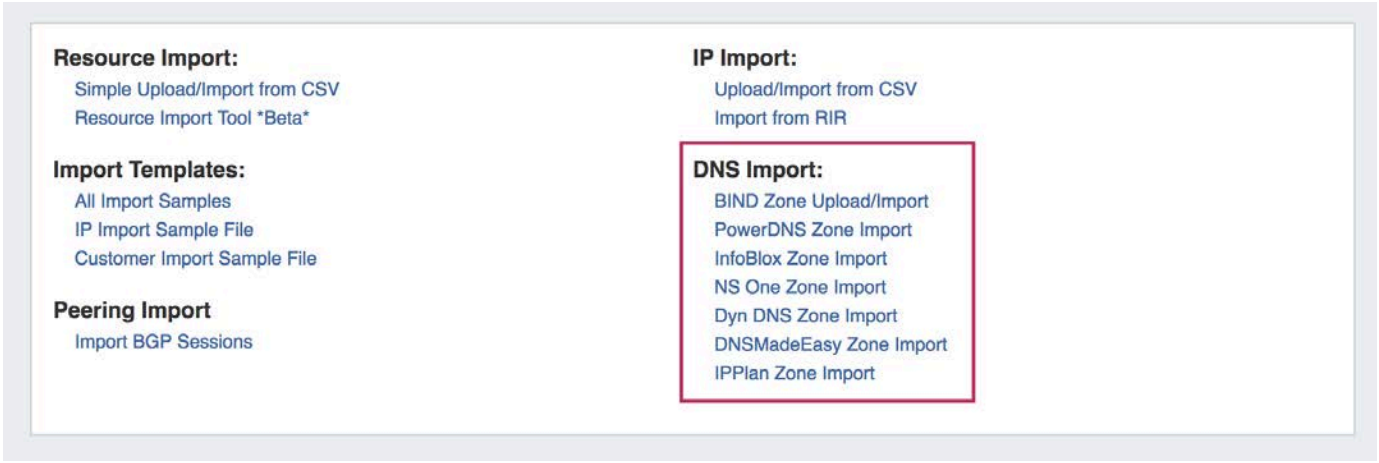
Import DNS Zones

Importing DNS Zones

ProVision offers multiple DNS zone import options, available under the Data Import tab in the the Admin section of ProVision.



DNS Import Options



ProVision offers various DNS zone import options available under the **Data Import** tab in the the Admin section of ProVision:

BIND Zone Import

- Imports using the named.conf configuration file tied to the zones you are uploading, a .zip or .tar file of the zones themselves, and an optional .csv file mapping zones to customers.

PowerDNS Zone Import

- Option is available after configuring a PowerDNS server with a MySQL backend. Connects to the selected server and imports all zones.

InfoBlox Zone Import

- Imports DNS zones using a provided Host, Username, and Password. The InfoBlox import pulls all zones on the InfoBlox LOCAL grid and adds them to a designated Group. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

NS One Zone Import

- Imports DNS zones using a NS One API Key. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

Dyn DNS Zone Import

- Imports DNS zones using a Dyn DNS Customer Name, Username, and Password. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

DNSMadeEasy Zone Import

- Imports DNS zones using a DNSMadeEasy API Key and API Secret. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

IPPlan Zone Import

- Imports DNS zones using IPPlan MySQL database options. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

Continue to the following sections for details on performing each import method:

- BIND DNS Zone Upload / Import
- PowerDNS Zone Import
- InfoBlox Zone Import
- NS One Zone Import
- Dyn DNS Zone Import
- DNSMadeEasy Zone Import
- IPPLAN Zone Import

BIND DNS Zone Upload / Import

BIND DNS Zone Import

The BIND DNS Zone Upload / Import tool uses the named.conf configuration file tied to the zones you are uploading, a .zip or .tar file of the zones themselves, and an optional .csv file mapping zones to customers. The following steps are used when importing BIND DNS zones:

- BIND DNS Zone Import
  - Preparing your DNS Zones for Import
  - Importing your DNS Zones (BIND)
    - Video Walkthrough
    - Step 1: Create a new DNS Import Job
    - Step 2: Map Data Columns (Optional)
    - Step 3: Reviewing Data
    - Step 4: Execute Import

Preparing your DNS Zones for Import

If your zone data is currently in BIND format - this is very straightforward.

There are three components for the upload process:

1) The named.conf configuration file tied to the zones you are uploading (required)

This tells the importer the Zone Name and where the zone file is written. It could be as simple as a multi-line file:

Simple DNS Config File

```
zone "my-zone.com" { type master; file "my-zone.com.zone"; };
zone "my-other-zone.com" { type master; file "my-other-zone.com.zone"; };
zone "my-third-zone.com" { type master; file "my-third-zone.com.zone"; };
```

or could be more complex like this file structure directory:

Complex DNS Config File

```
zone "my-zone.com" { type master; file "/usr/local/zones/my-zone.com.zone"; };
zone "my-other-zone.com" { type master; file "/usr/local/zones/more/my-other-zone.com.zone"; };
zone "my-third-zone.com" { type master; file "/usr/local/zones/more/even more/my-third-zone.com.
zone"; };
```

This configuration file can be taken directly from the DNS server, and can be in either ISC BIND or NSD format. The system auto-detects which one is being supplied.

For a sample Simple Config: conf.conf

2) A ZIP or TAR file of the DNS zones themselves (required)

This is as it sounds - a file archive where we can find the zones and it should match the configuration file uploaded in Step 1.

Zone Order

These zone files can be in any order, or in sub-directories, so long as the configuration file (Step 1) correctly points to them

For a sample simple ZIP: zones.zip

3) Match CSV for assigning DNS Zones to Resources (optional)

This file allows the administrator to "assign" zone files to a given Resource. If you have Imported a group of Resources, they have Resource IDs associated with them. You can then import DNS zones and assign them to those Resource IDs. When complete, you will be able to pull up the Resource Record and see the DNS Zones associated to that Resource ID.

Sample CSV File

```
my-zone.com,test-01,fun stuff, 174.23.14.4, 174.23.14.9
my-otherzone.com,test-02,great stuff, dns1.dns.net, dns2.dns.net
even-reverse-zones.arpa,test-03,amazing stuff
```

Note the columns are the "Zone Name", the "Resource ID", "Notes", "Master Server", "Slave Server"

Importing DNS Server Linkages

When importing zones, you can use the "Master Server" and "Slave Server" columns to assign zones to specified DNS Servers. Please note that the IP address or FQDN of the DNS Server is supported in this field.
To successfully map to a DNS server, that server must already exist within Provision.

For a sample CSV: config.csv

Importing your DNS Zones (BIND)

Video Walkthrough

Note: Video walkthrough is from ProVision version 5.1.x - however, import steps and descriptions still apply to current version. Updated videos are in progress!

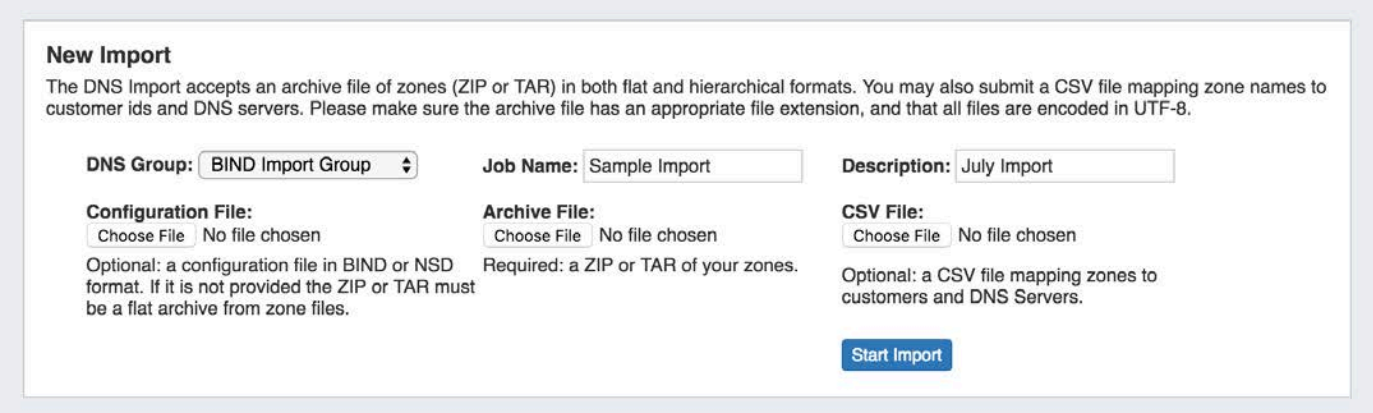
Step 1: Create a new DNS Import Job

Navigate to the **Data Import** Tab from the **Admin** button to import your data. Select "BIND Zone Upload/Import" under "DNS Import".



Under the "New Import" section, select the DNS Group under which you want the zones to be imported, then create a Job Name and Description for the import. This is especially useful to keep track of progress in cases the data arrives from multiple sources, or will require multiple stages of manual review.

Select the appropriate **Configuration File** (required if the .zip is not a flat archive file), **Archive File** (required), and **CSV File** (optional) that you prepared above by selecting the "Choose File" button(s) under each section, and browsing to the correct file location. Then hit "Start Import".

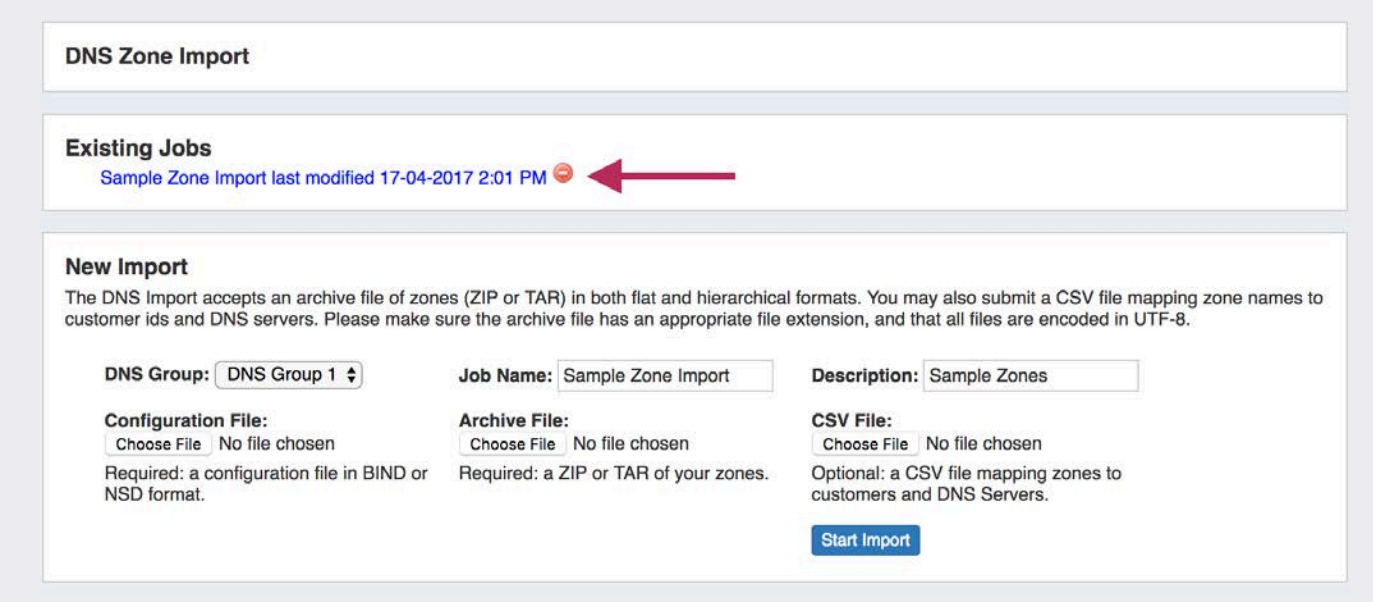


Working with Large or Multiple Data Sets

Although you cannot add new files to an existing job, for jobs with multiple sources for data (which may have different formatting), you can simply create separate jobs and descriptions for each source - no need to manually combine the data into one file before importing. The Import tool's mapping and editing functions will allow for the data to be reconciled in ProVision.

For large data sets where multiple stages of manual review might be needed, you can create a new job using the same set of data files in order to work in parallel on a different portion of the data.

After importing, the new job will appear under the "Existing Jobs" section. To continue working with this job, select it from the list and the next step will appear on the page.



Step 2: Map Data Columns (Optional)

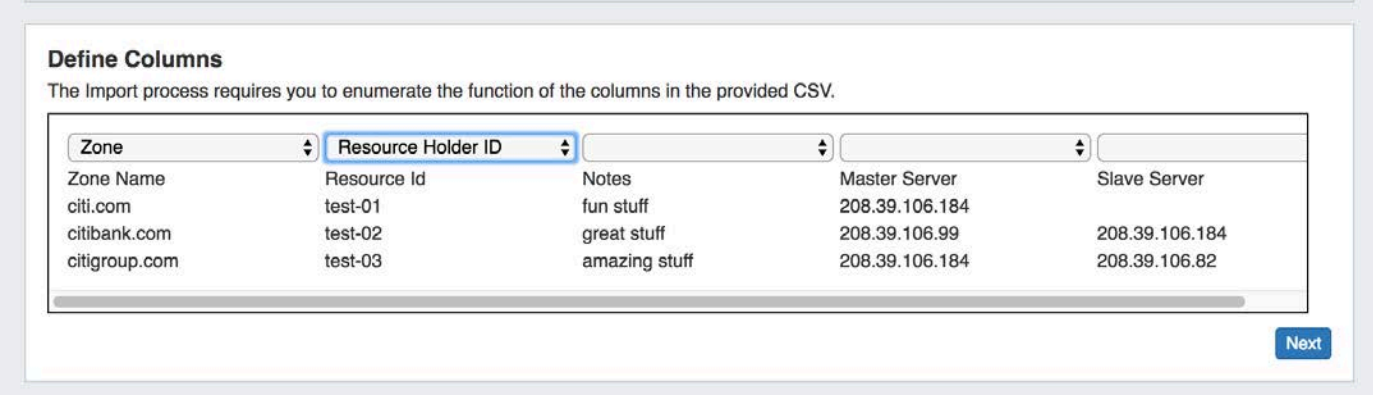
If you chose to load an optional match CSV file to assign DNS Zones to Resource, a mapping step will be available. Otherwise, proceed to Step 3: Reviewing Data.

For DNS imports, two column definitions are available: **Zone** and **Resource Holder ID**. Using the dropdown menu, select the appropriate column for the zone name and Resource Holder (customer) associated with zone. The Resource Holder ID field will be used to set the Parent Resource of the individual zone once imported.

Server associations will not be imported. Instead, the associated server will be the server attached to the selected DNS Group.

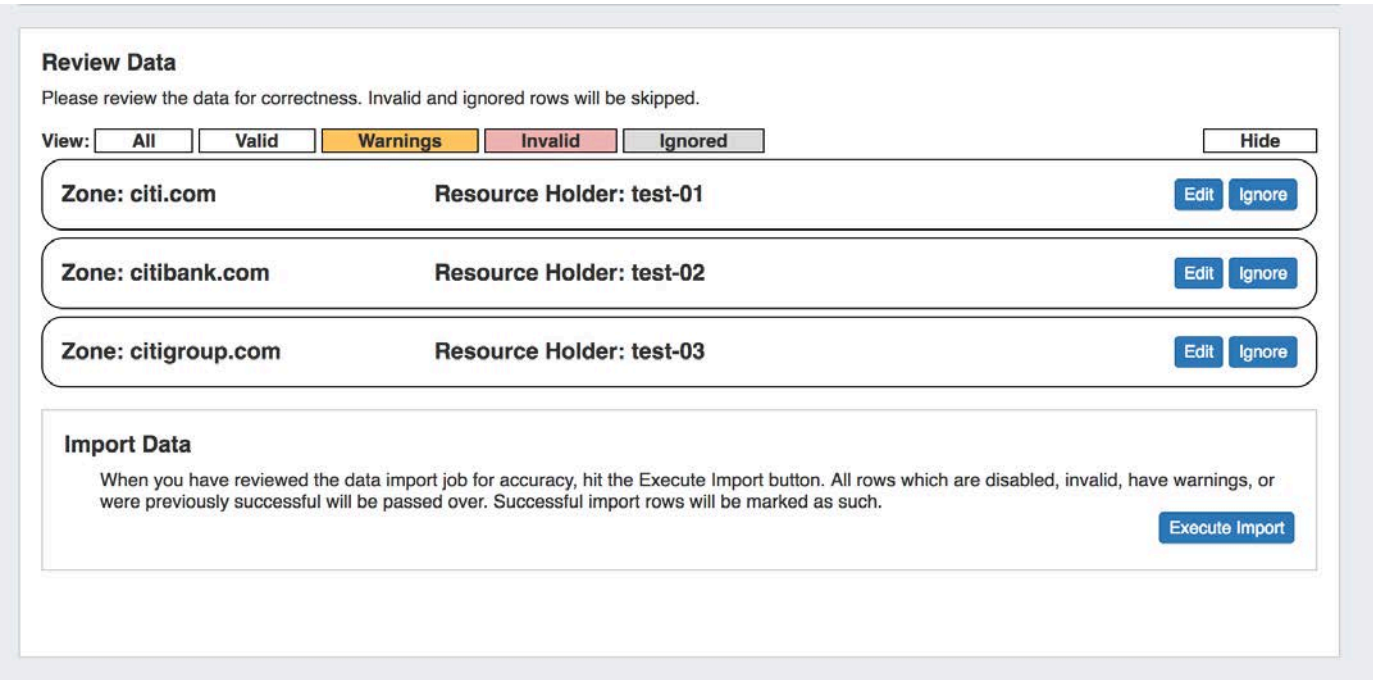
Other columns which do not apply under the available definitions should be left as blank, and will be skipped during the upload process.

When completed, hit "Next".



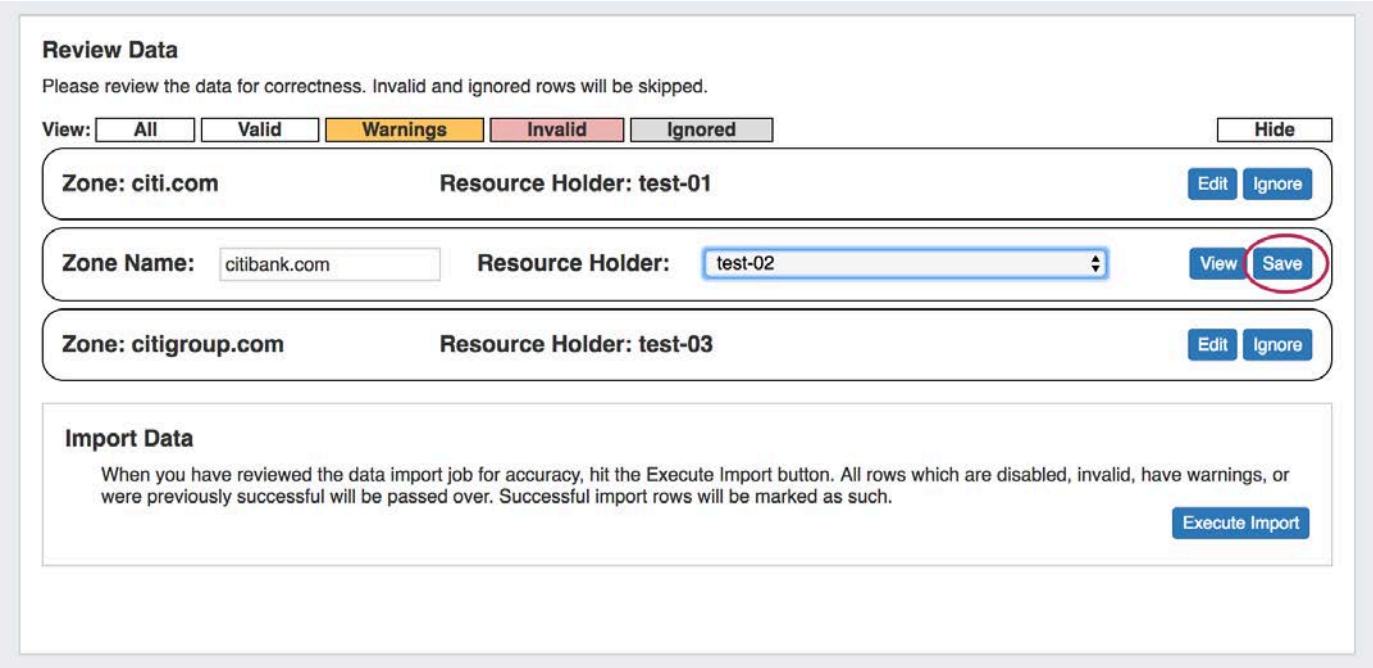
Step 3: Reviewing Data

After supplying the file set and defining columns (if applicable), a review step is provided. The configuration file is broken into individual jobs, scanned for errors, and shown by row (in batches of 100) to be reviewed. Zones with errors will show as color coded. and can be filtered to be viewed by All, Valid, Warnings, Invalid, or Ignored. From here, the zone can be edited or ignored.



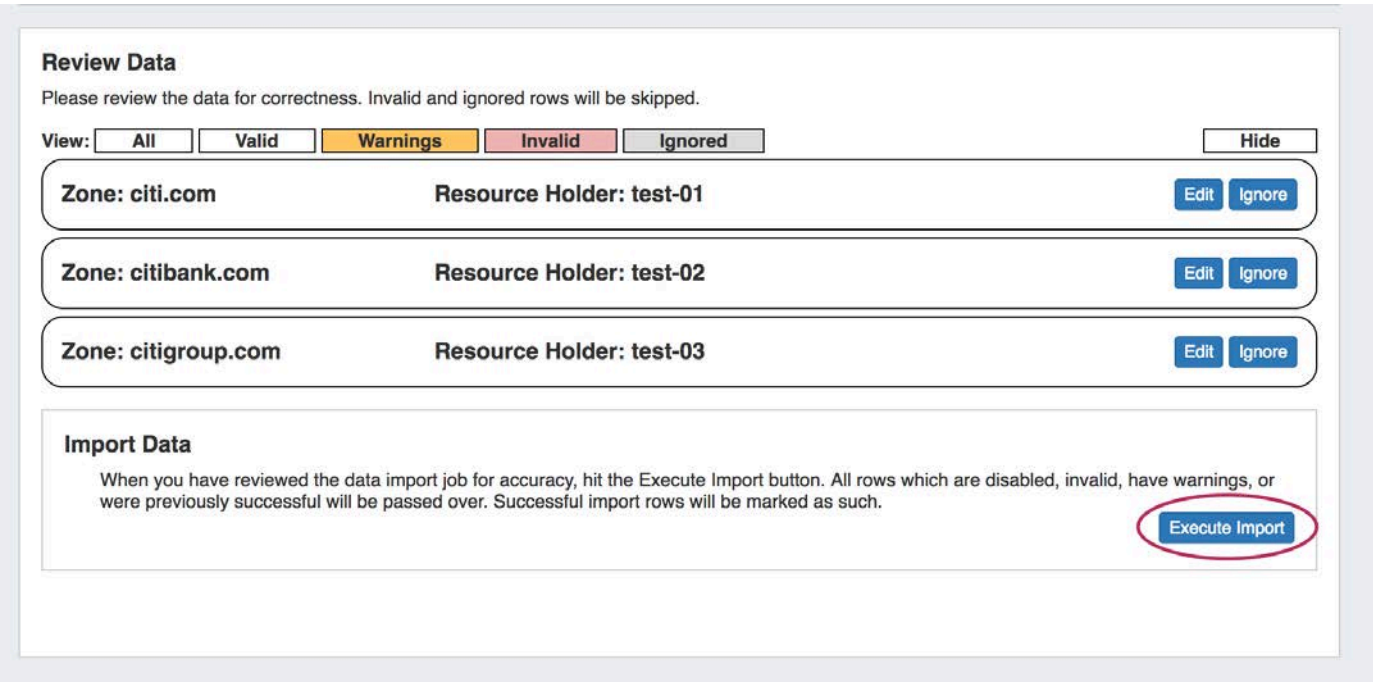
To edit the zone name or select a different Resource Holder, hit "Edit" on the zone record.

After editing, hit "Save", and continue reviewing / editing data as desired.

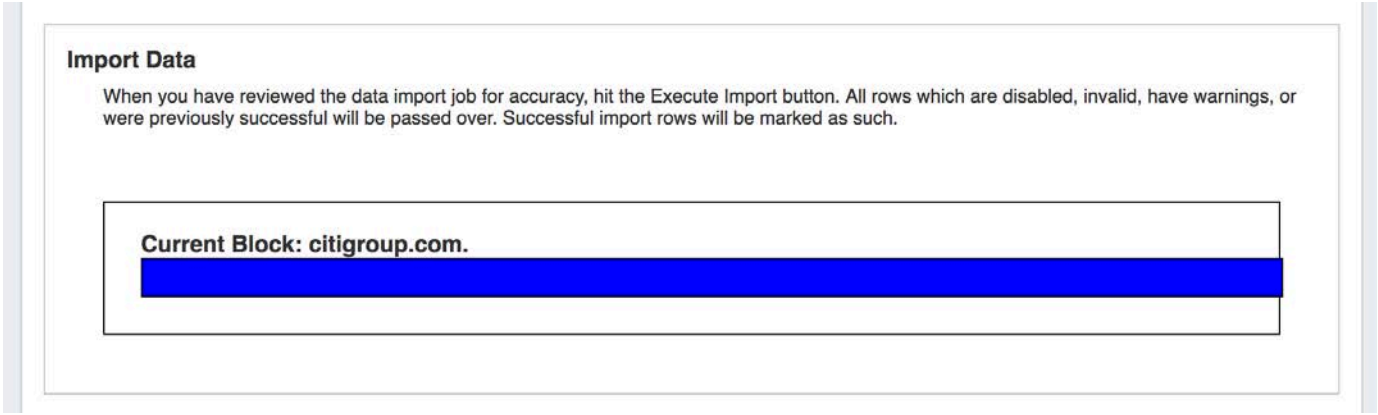


Step 4: Execute Import

When the review step is completed, hit the "Execute Import" button. A progress bar will appear to show progress and note errors if they occur.



When the bar reaches 100%, the import is complete.


Import Data
When you have reviewed the data import job for accuracy, hit the Execute Import button. All rows which are disabled, invalid, have warnings, or were previously successful will be passed over. Successful import rows will be marked as such.
Current Block: citigroup.com.

PowerDNS Zone Import

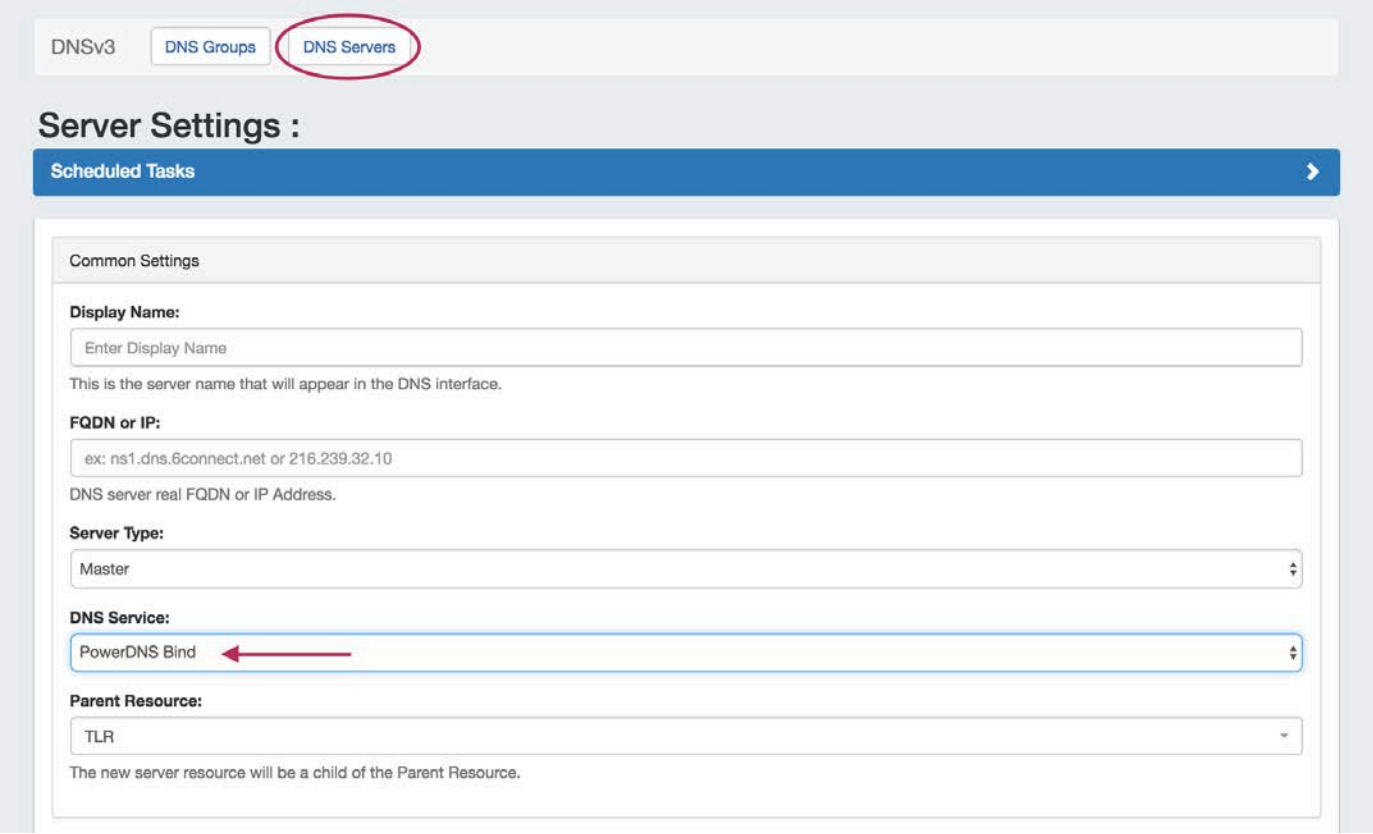
PowerDNS Zone Import

The PowerDNS Zone Import Option is available after configuring a PowerDNS server with a MySQL backend. It connects to the selected server and imports all zones.

- PowerDNS Zone Import
  - Step 1: Verify PowerDNS server setup
  - Step 2: Import your PowerDNS zones

Step 1: Verify PowerDNS server setup

To import PowerDNS zones, first ensure the PowerDNS server has been set up under the **DNS** Tab, DNS Servers section (see: Working with DNS Servers and configuring a PowerDNS server).

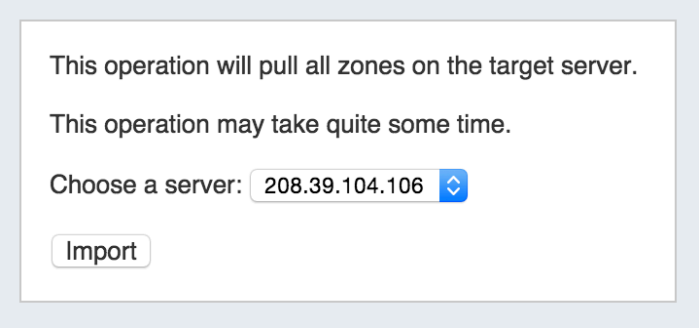


Step 2: Import your PowerDNS zones

Once server setup has been verified, navigate to the **Data Import** Tab in the Admin section. Select the "Power DNS Zone Import" link.



Then, to import your data simply choose your PowerDNS server and click "Import".



InfoBlox Zone Import

InfoBlox Zone Import

The ProVision Data Import InfoBlox option imports InfoBlox DNS zones using a provided Host, Username, and Password. It pulls all zones on the InfoBlox LOCAL grid and adds them to a designated DNS Group. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

- InfoBlox Zone Import
  - Step 1: Verify DNS Group setup
  - Step 2: Verify InfoBlox server setup
  - Step 3: Import your InfoBlox zones

Step 1: Verify DNS Group setup

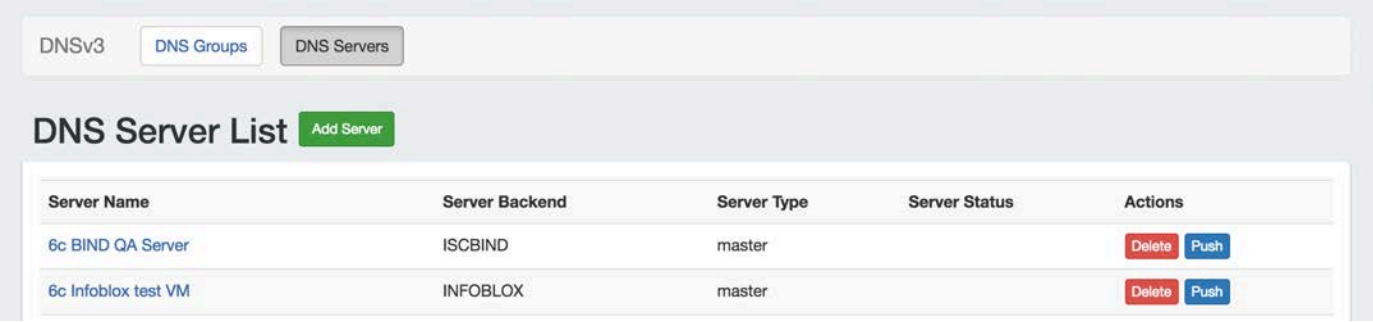
It is recommended to have a DNS Group created in advance with the proper default parameters and records to be inherited by the imported InfoBlox zones and records.

Review the existing DNS Groups in the **DNS** tab, and create a new DNS Group if necessary. See Working with DNS Groups for details on editing or creating DNS Groups.

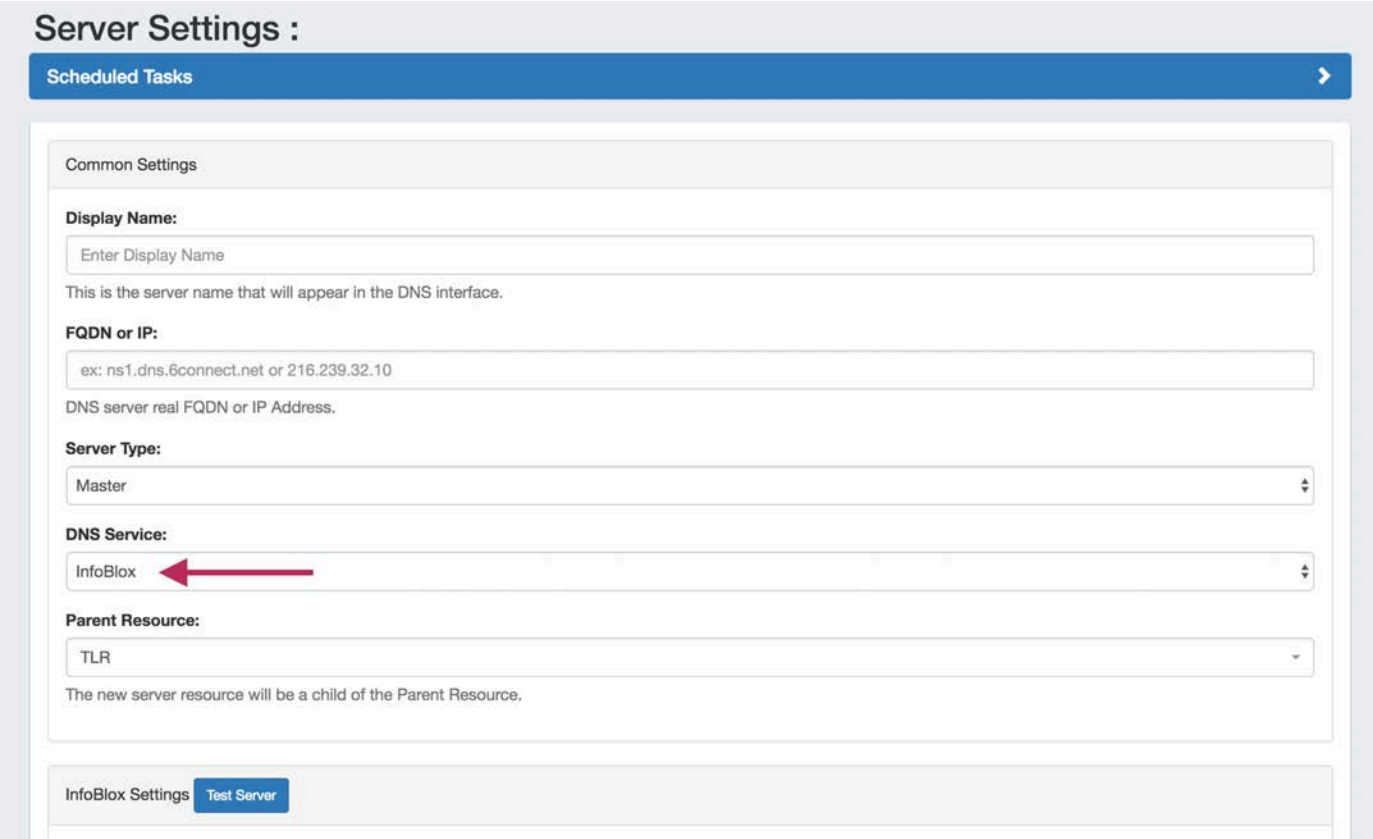
Step 2: Verify InfoBlox server setup

Although it is not strictly necessary to have the InfoBlox server created in ProVision prior to import, creating or verifying the server at this point gives you an opportunity to test the connection and attach the InfoBlox server to the DNS Group selected in step 1.

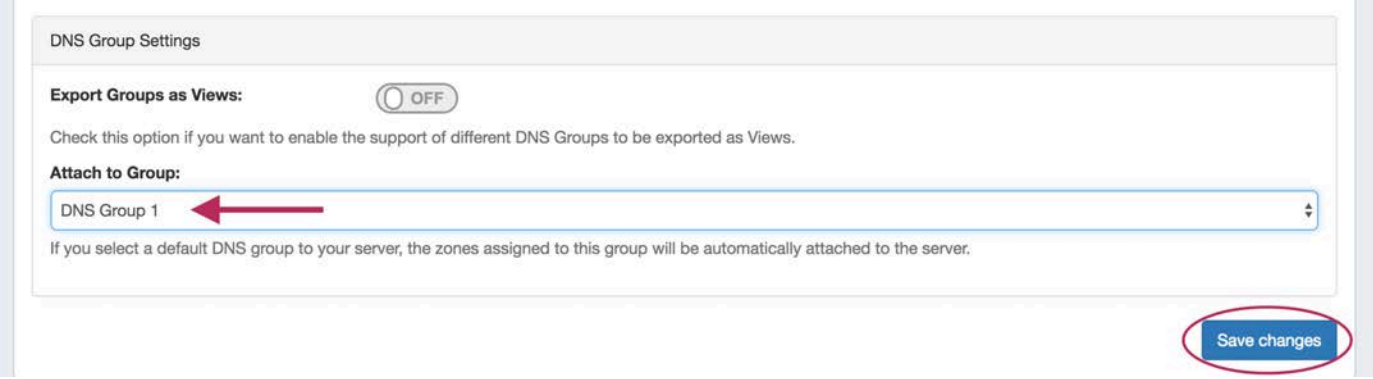
Go to the **DNS** Tab, DNS Servers section, and either verify that the desired server is shown in the server list, or click the "Add Server" button to create a new InfoBlox server.



If creating a new server, ensure that "InfoBlox" is selected for the DNS Service.



Under the "DNS Group Settings" section of the server detail page, ensure that the DNS Group from step 1 is selected as the server's Attach to Group setting, and click "Save".



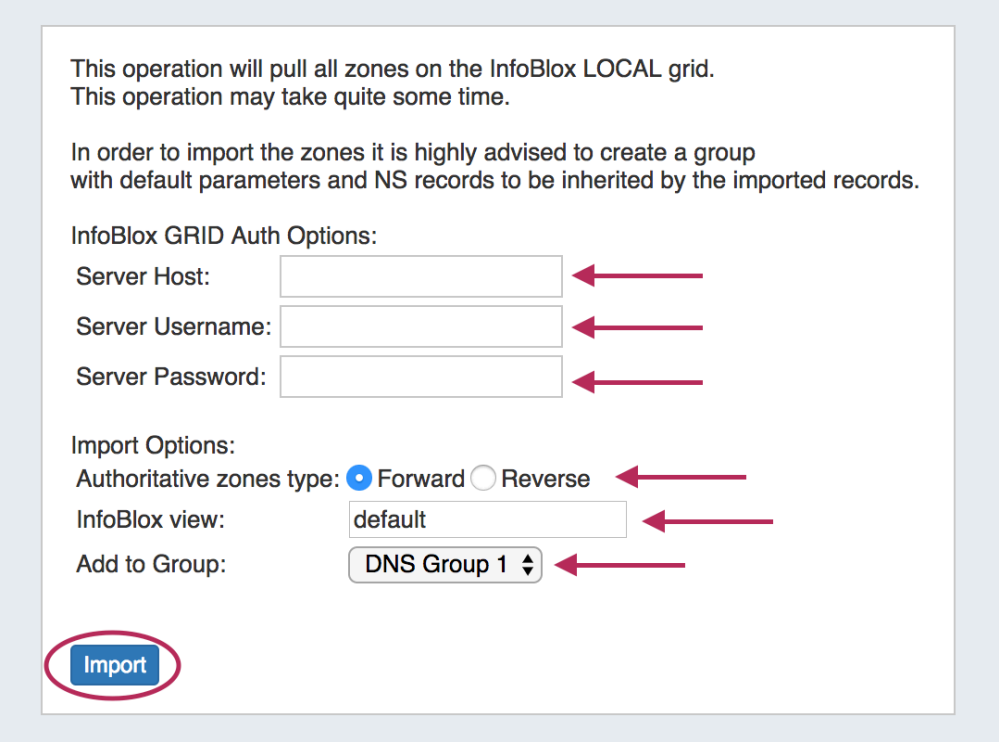
Step 3: Import your InfoBlox zones

Once server setup has been verified, navigate to the **Data Import** Tab in the Admin section. Select the "InfoBlox Zone Import" link.



Then, enter the InfloBlox server Host, Username, and Password. Select Forward or Reverse zone type, the InfoBlox view, and the ProVision DNS Group to add the zones to.

Once all information has been entered, click the "Import" button.



The import may take a couple of minutes. If errors occur, they will show under the Pull in Progress message.

When completed, a successful import will show a green Complete! status message.

Verify the zones by exiting the Admin area of ProVision, clicking on the **DNS** tab, and reviewing the zones under the Group previously selected for the import.

NS One Zone Import

NS One Zone Import

The ProVision "NS One Data Import" option imports NS One DNS zones using a provided NS One API key. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

- NS One Zone Import
  - Step 1: Verify DNS Group setup
  - Step 2: Verify NS One server setup
  - Step 3: Import your zones

Step 1: Verify DNS Group setup

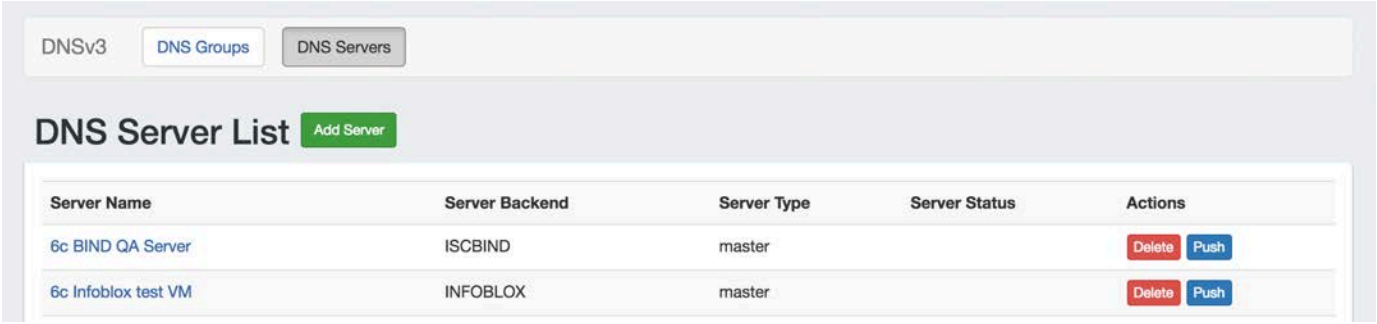
It is recommended to have a DNS Group created in advance with the proper default parameters and records to be inherited by the imported NS One zones and records.

Review the existing DNS Groups in the **DNS** tab, and create a new DNS Group if necessary. See Working with DNS Groups for details on editing or creating DNS Groups.

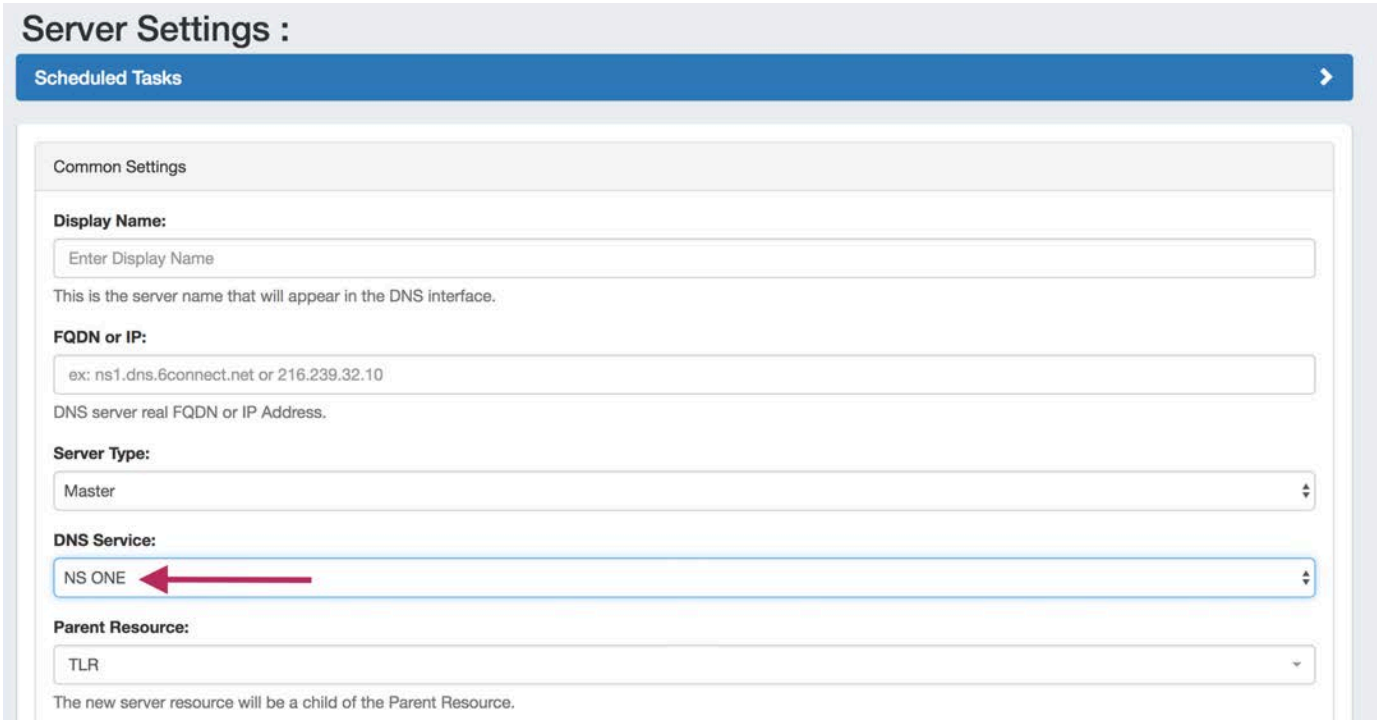
Step 2: Verify NS One server setup

Although it is not strictly necessary to have the NS One server created in ProVision prior to import, creating or verifying the server at this point gives you an opportunity to test the connection and attach the NS One server to the DNS Group selected in step 1.

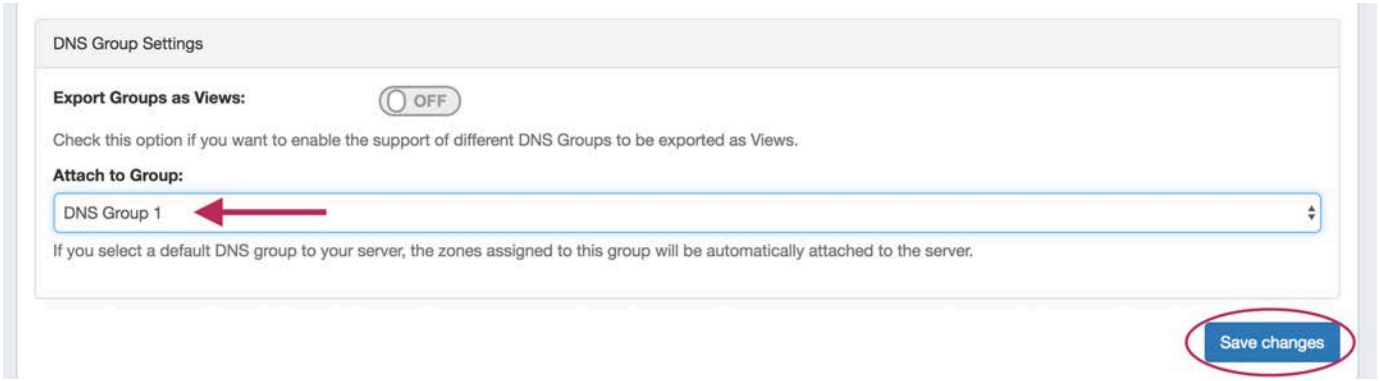
Go to the **DNS** Tab, DNS Servers section, and either verify that the desired server is shown in the server list, or click the "Add Server" button to create a new NS One server.



If creating a new server, ensure that "NS One" is selected for the DNS Service.



Under the "DNS Group Settings" section of the server detail page, ensure that the DNS Group from step 1 is selected as the server's Attach to Group setting, and click "Save".



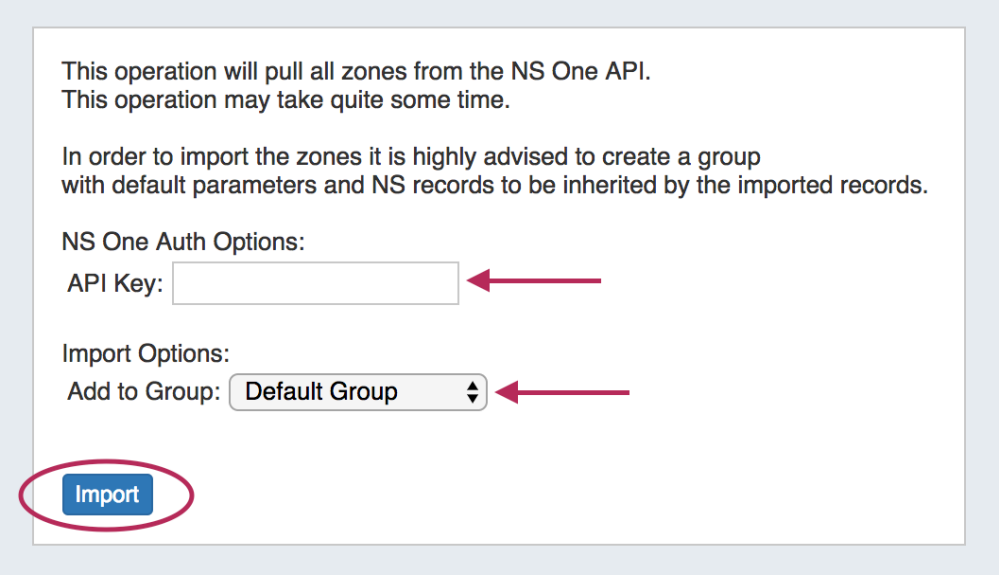
Step 3: Import your zones

Once server setup has been verified, navigate to the **Data Import** Tab in the Admin section. Select the "NS One Zone Import" link.



Then, enter the NS One API key and the ProVision DNS Group to add the zones to.

Once all information has been entered, click the "Import" button.



The import may take a couple of minutes. If errors occur, they will show under the Pull in Progress message.

When completed, a successful import will show a green Complete! status message.

Verify the zones by exiting the Admin area of ProVision, clicking on the **DNS** tab, and reviewing the zones under the Group previously selected for the import.

Dyn DNS Zone Import

Dyn DNS Zone Import

The ProVision "Dyn DNS Zone Import" option imports Dyn DNS zones using a provided Customer Name, Username, and Password. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

- Dyn DNS Zone Import
  - Step 1: Verify DNS Group setup
  - Step 2: Verify server setup
  - Step 3: Import your DynDNS zones

Step 1: Verify DNS Group setup

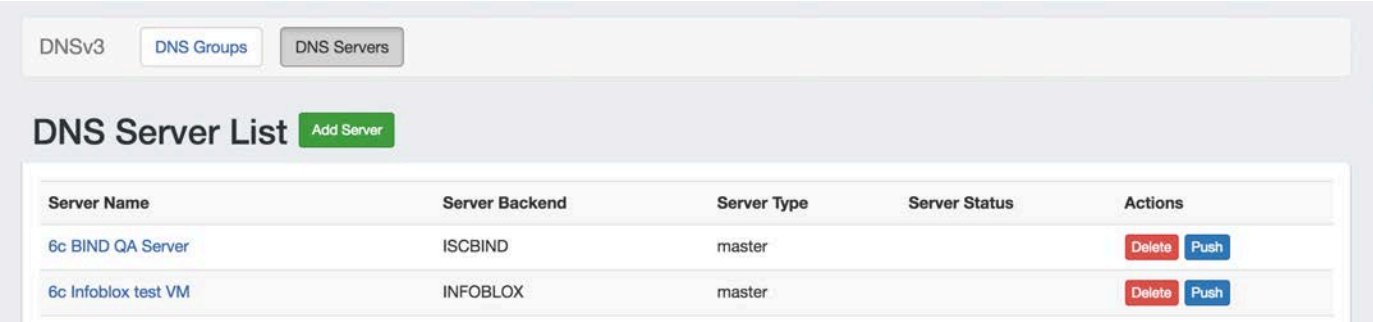
It is recommended to have a DNS Group created in advance with the proper default parameters and records to be inherited by the imported Dyn DNS zones and records.

Review the existing DNS Groups in the **DNS** tab, and create a new DNS Group if necessary. See Working with DNS Groups for details on editing or creating DNS Groups.

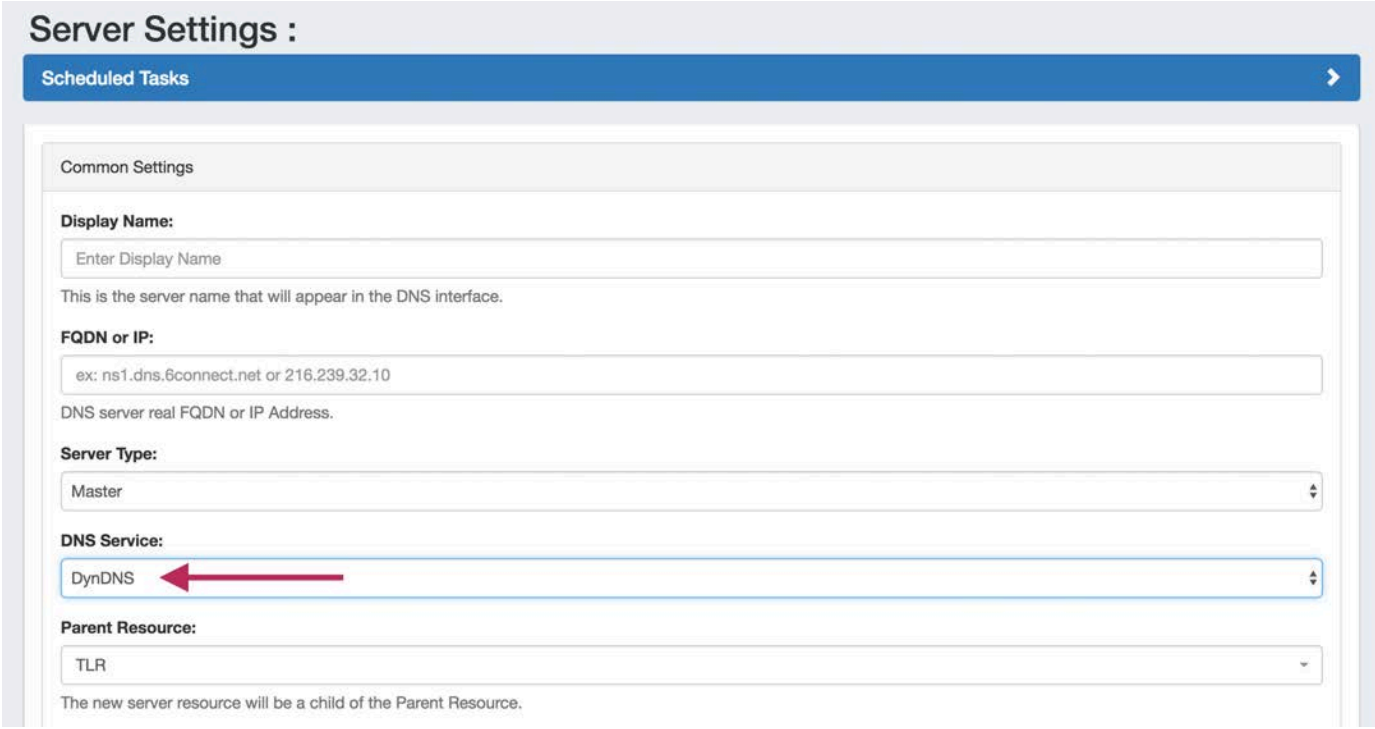
Step 2: Verify server setup

Although it is not strictly necessary to have the Dyn DNS server created in ProVision prior to import, creating or verifying the server at this point gives you an opportunity to test the connection and attach the Dyn DNS server to the DNS Group selected in step 1.

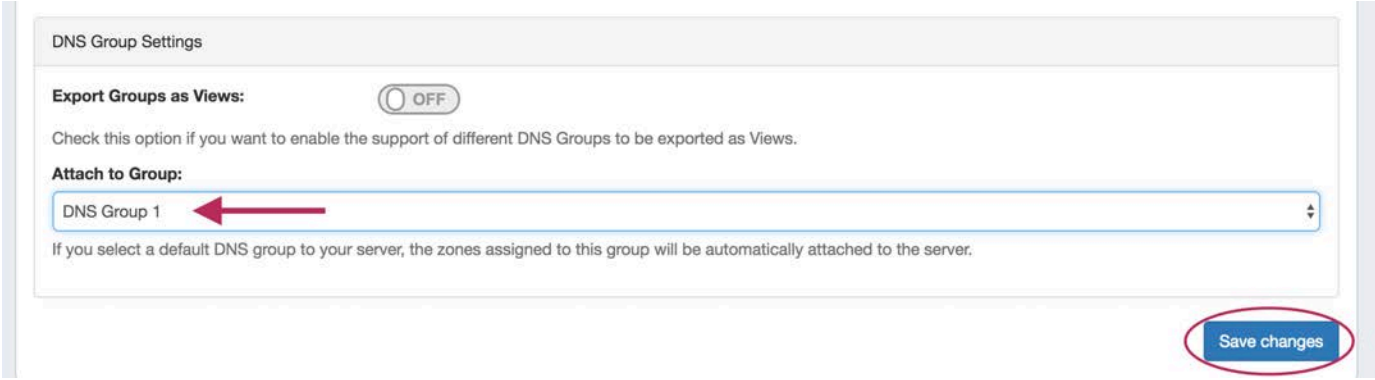
Go to the **DNS** Tab, DNS Servers section, and either verify that the desired server is shown in the server list, or click the "Add Server" button to create a new Dyn DNS server.



If creating a new server, ensure that "Dyn DNS" is selected for the DNS Service.



Under the "DNS Group Settings" section of the server detail page, ensure that the DNS Group from step 1 is selected as the server's Attach to Group setting, and click "Save".



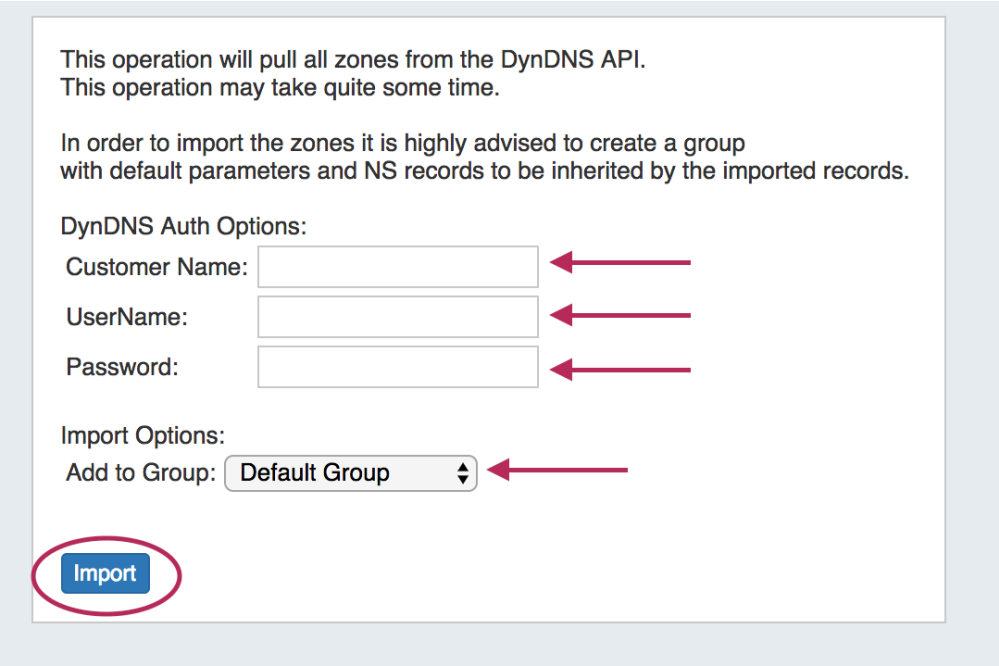
Step 3: Import your DynDNS zones

Once server setup has been verified, navigate to the **Data Import** Tab in the Admin section. Select the "Dyn DNS Zone Import" link.



Then, enter the Dyn DNS Customer Name, Username, and Password. Then, select the ProVision DNS Group to add the zones to.

Once all information has been entered, click the "Import" button.



The import may take a couple of minutes. If errors occur, they will show under the Pull in Progress message.

When completed, a successful import will show a green Complete! status message.

Verify the zones by exiting the Admin area of ProVision, clicking on the **DNS** tab, and reviewing the zones under the Group previously selected for the import.

DNSMadeEasy Zone Import

DNSMadeEasy Zone Import

The ProVision "DNSMadeEasy Zone Import" option imports DNSMadeEasy zones using a provided API Key and API Secret. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

- DNSMadeEasy Zone Import
  - Step 1: Verify DNS Group setup
  - Step 2: Verify server setup
  - Step 3: Import your DNSMadeEasy zones

Step 1: Verify DNS Group setup

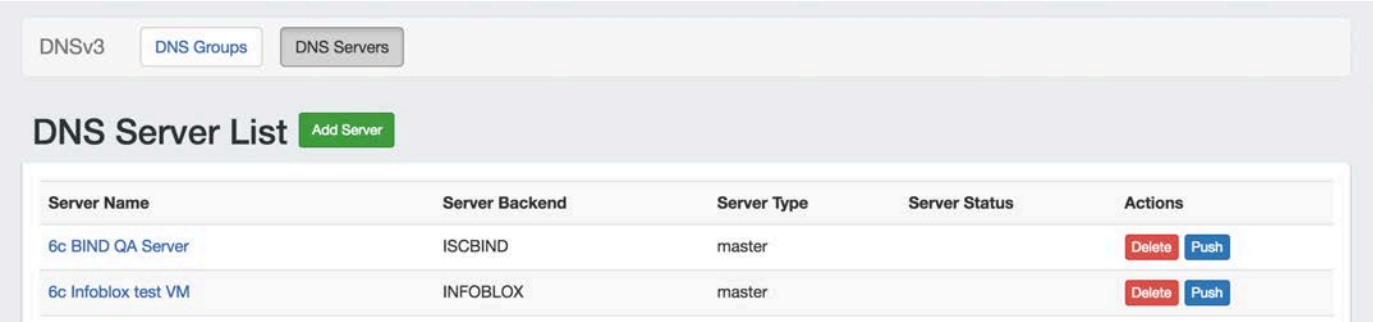
It is recommended to have a DNS Group created in advance with the proper default parameters and records to be inherited by the imported zones and records.

Review the existing DNS Groups in the **DNS** tab, and create a new DNS Group if necessary. See Working with DNS Groups for details on editing or creating DNS Groups.

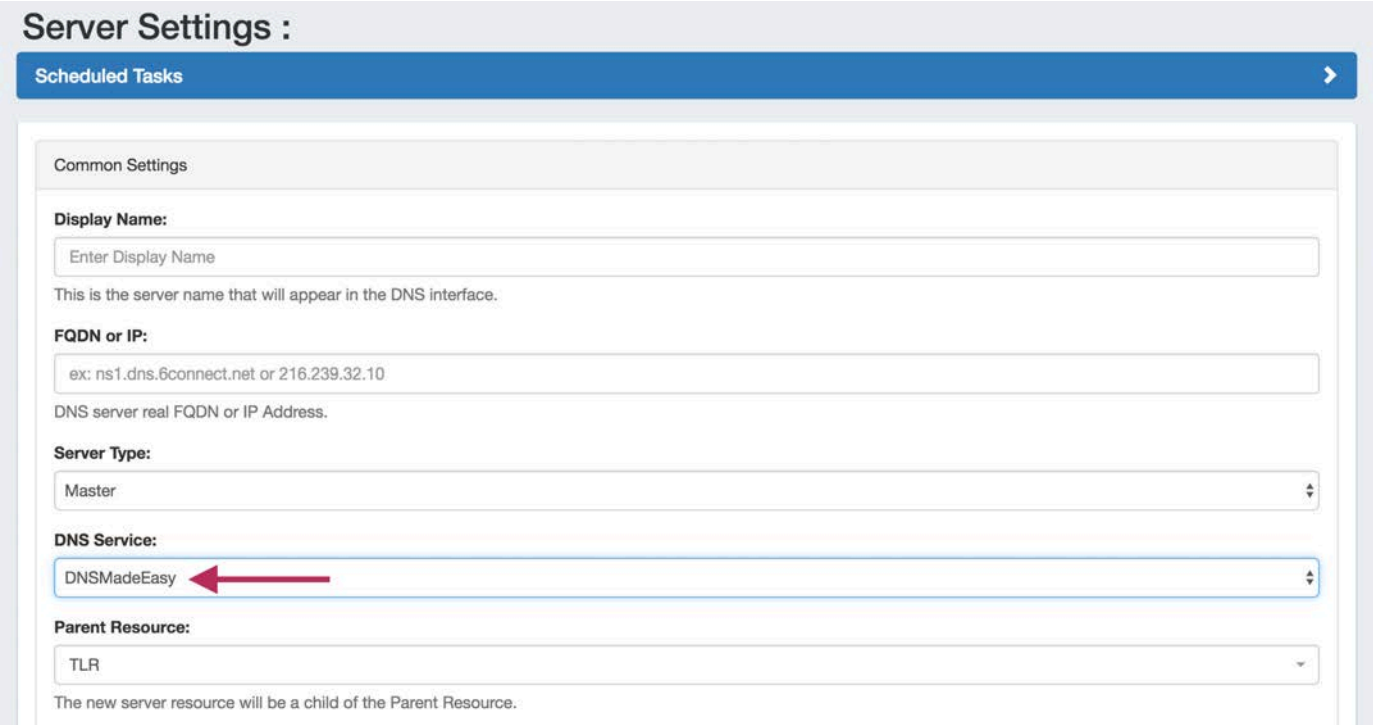
Step 2: Verify server setup

Although it is not strictly necessary to have the DNSMadeEasy server created in ProVision prior to import, creating or verifying the server at this point gives you an opportunity to test the connection and attach the DNSMadeEasy server to the DNS Group selected in step 1.

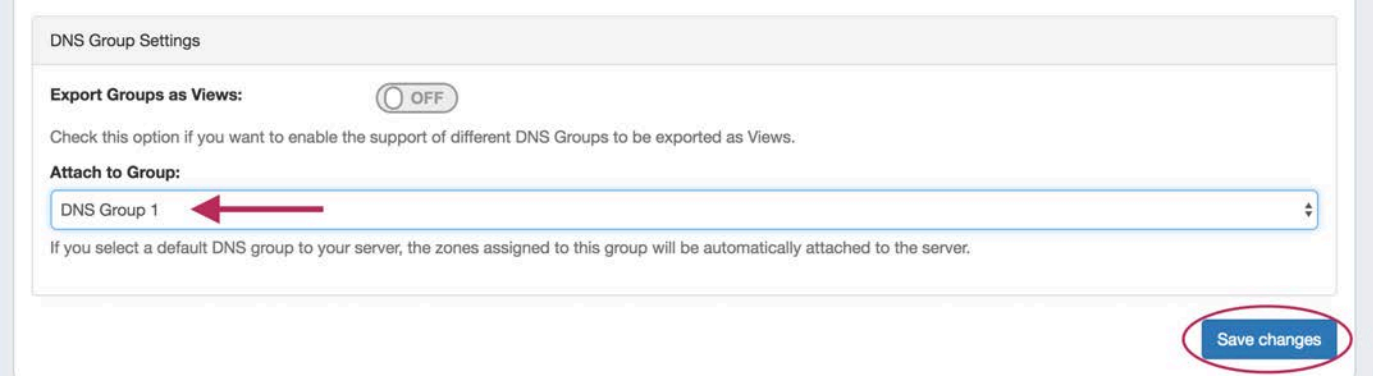
Go to the **DNS** Tab, DNS Servers section, and either verify that the desired server is shown in the server list, or click the "Add Server" button to create a new DNSMadeEasy server.



If creating a new server, ensure that "DNSMadeEasy" is selected for the DNS Service.



Under the "DNS Group Settings" section of the server detail page, ensure that the DNS Group from step 1 is selected as the server's Attach to Group setting, and click "Save".



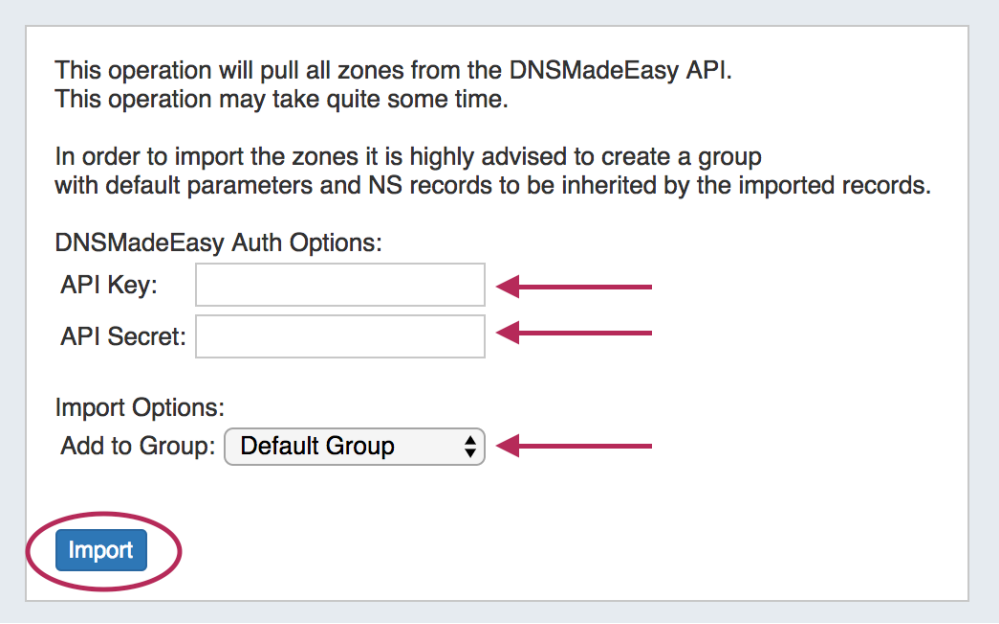
Step 3: Import your DNSMadeEasy zones

Once server setup has been verified, navigate to the **Data Import** Tab in the Admin section. Select the "DNSMadeEasy Zone Import" link.



Then, enter the DNSMadeEasy API Key, Secret Key, and the ProVision DNS Group to add the zones to.

Once all information has been entered, click the "Import" button.



The import may take a couple of minutes. If errors occur, they will show under the Pull in Progress message.

When completed, a successful import will show a green Complete! status message.

Verify the zones by exiting the Admin area of ProVision, clicking on the **DNS** tab, and reviewing the zones under the Group previously selected for the import.

IPPLAN Zone Import

IPPlan Zone Import

The ProVision "IPPlan Import" option imports IPPlan zones using provided MySQL database credentials. It is advised to create a DNS Group prior to the import with default parameters and NS records to be inherited by the imported records.

- IPPlan Zone Import
  - Step 1: Verify DNS Group setup
  - Step 2: Import your IPPlan zones

Step 1: Verify DNS Group setup

It is recommended to have a DNS Group created in advance with the proper default parameters and records to be inherited by the imported zones and records.

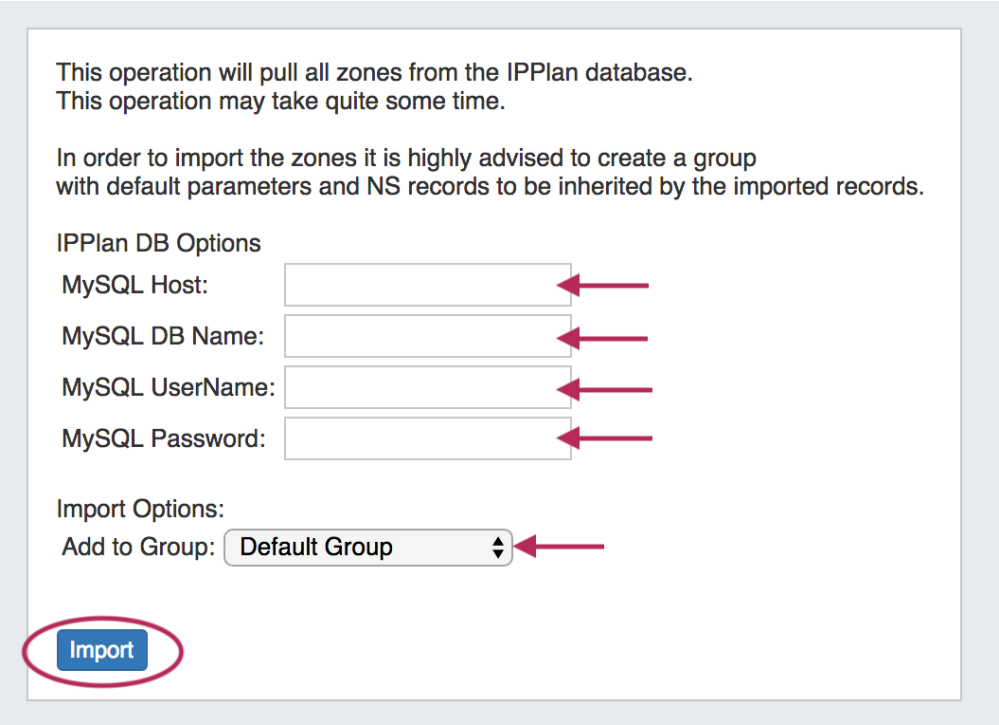
Review the existing DNS Groups in the **DNS** tab, and create a new DNS Group if necessary. See Working with DNS Groups for details on editing or creating DNS Groups.

Step 2: Import your IPPlan zones

Once DNS Group setup has been verified, navigate to the **Data Import** Tab in the Admin section. Select the "IPPlan Zone Import" link.



From there, enter the IPPLAN DB Options - Host, Database Name, MySQL Username, and MySQL Password, then select the desired DNS Group to add the zones under.



The import may take a couple of minutes. If errors occur, they will show under the Pull in Progress message.

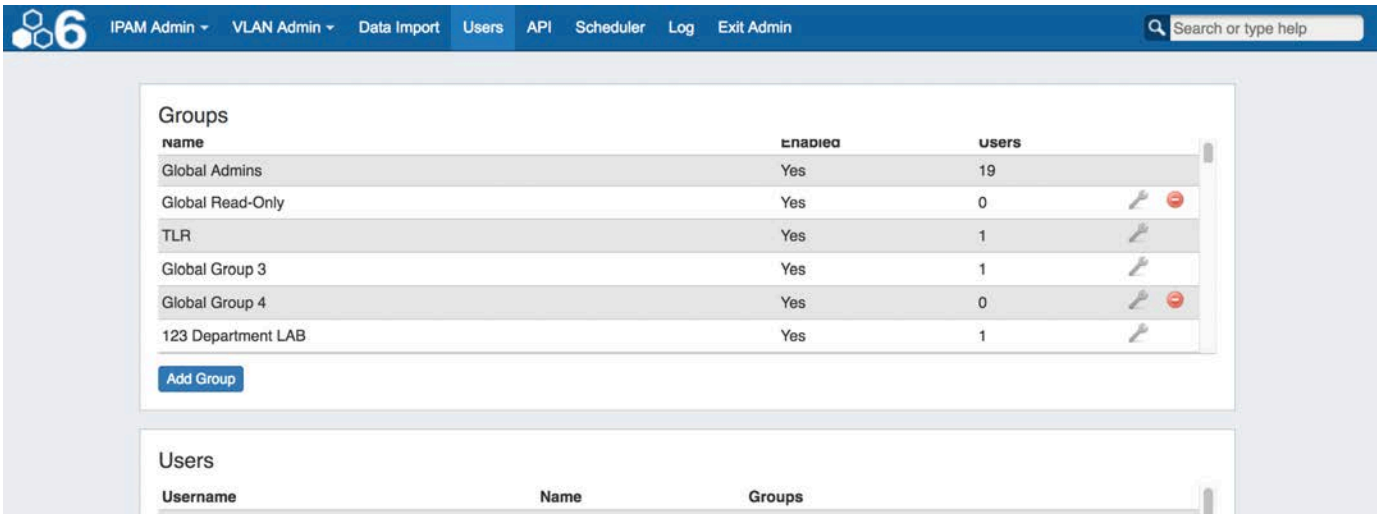
When completed, a successful import will show a green Complete! status message.

Verify the zones by exiting the Admin area of ProVision, clicking on the **DNS** tab, and reviewing the zones under the Group previously selected for the import.

Users & Permissions

Overview

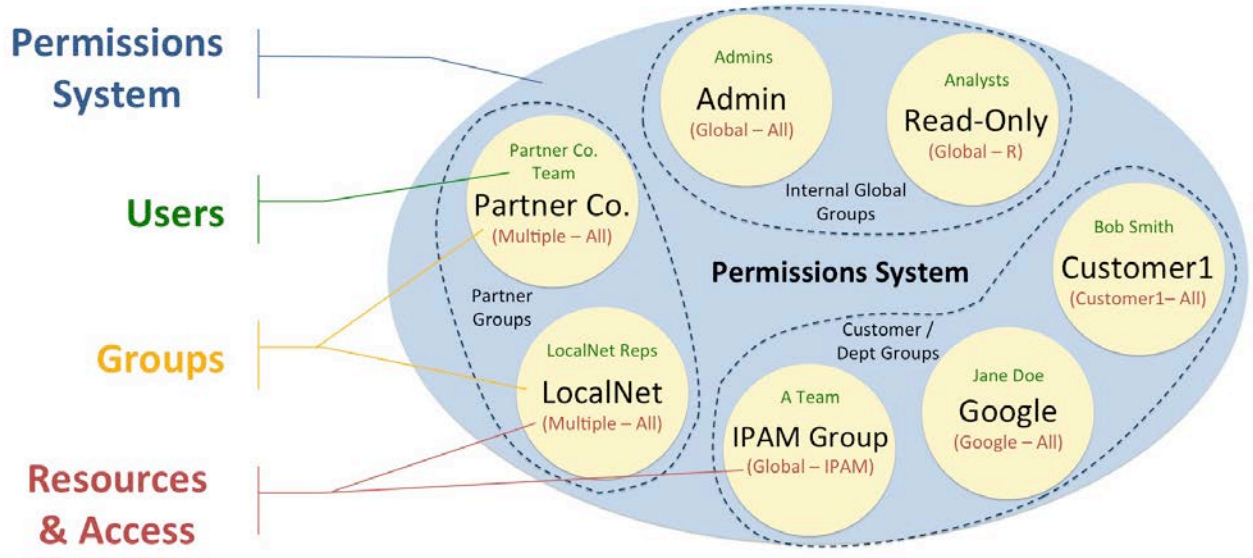
Users & Permissions is accessed from the Admin screen under the **Users** tab. Here, you will find tools for adding and managing permissions groups, users, and running queries for verifying a user's specific permissions.



The Permissions Structure

The Permissions structure in ProVision is designed to give you as much flexibility as you need to accommodate most use cases. When mapping out the permissions structure for your organization, keep in mind who you want to access to application:

- Internal Users and Roles (Admins, Read Only, etc.)
- Partners related to multiple specific Resources/Accounts
- Customers/Departments with limited view to only their respective Resources/Accounts



In this diagram, we have created groups for each of those scenarios – we have internal groups, Partner Groups, and Customer groups. Each of these groups has access to different resources, permission levels, and users assigned to them.

The components of the Permissions System include:

Users: A User is a single login account that accesses ProVision. Users are assigned to Groups.

Groups: A Group is a set of permission conditions that apply to selected Users. Allowed Resources and access levels (C/R/U/D permissions) are set inside the Group.

Resources & Access: Inside a Group, Resource access may be set to Global (applies to all Resources), or to the Resource level (applies to only the selected Resources). For each Resource selected, access permissions can be set with C/R/U/D permissions under each ProVision functional area (IPAM, DNS, Resource, Peering).

As a whole, this makes up the ProVision permissions system. The Permissions system allows you to fine-tune access to resource data to be as detailed as you need.

Video Overview

Video overview applicable to ProVision version 5.3.3 and earlier.

Permission Levels

Global Permissions

When you see a reference to a "TLR" - that is a "Top Level Resource". This Is the primary Resource under which all other resources fall under. ProVision currently only allows a single level of administrator permissions: Global Administrator.

Users with "Admin" access can assign/modify permissions for other users.

See Global Permissions for more details on configuring these elements.

Resource Permissions

An administrator can also set respective permissions for a given Resource (single or multiple). These permissions fall under Groups. So a Group is configured for the given group of Resource permissions, and then the User account is added.

See Users and Groups to learn how Resource Permissions are assigned.

See Resource Permissions for more details on configuring these elements.

Table of contents:

- Global Permissions
- Resource Permissions
- Users and Groups
- Verifying Permissions

Global Permissions

Global Permissions

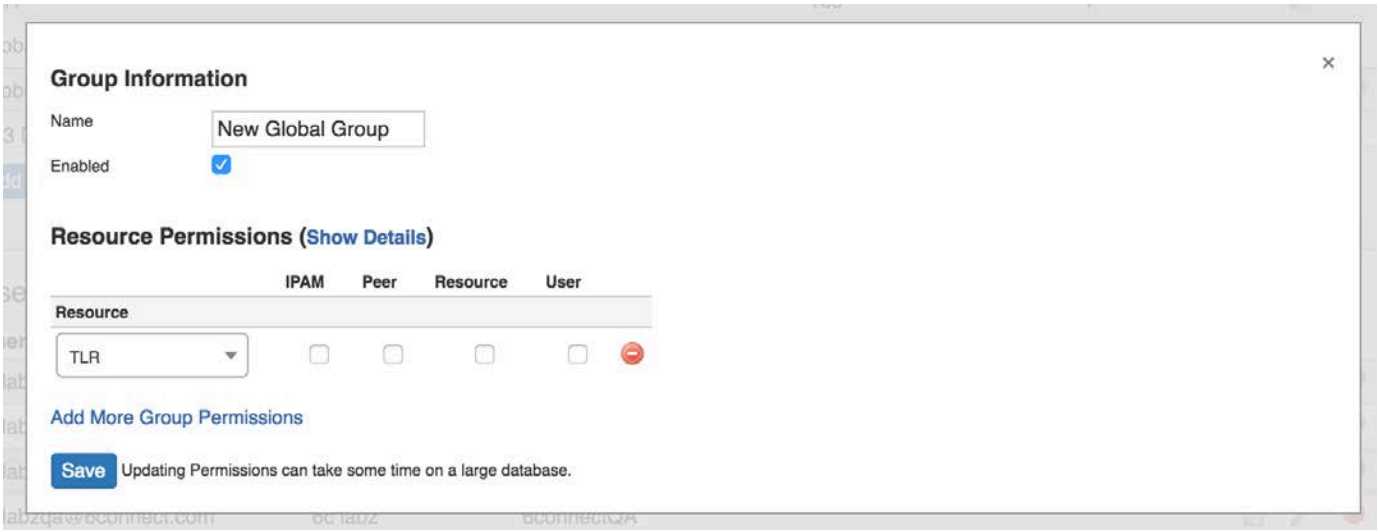
Global Permissions apply to the "TLR" or "Top Level Resource" within ProVision. By default, ProVision includes two groups with Global Permissions access - Global Admins and Global Read-Only. Initial Users are typically placed in the "Global Admins" group, and have administrative access to the entire platform. Global Read-Only users have full access to the platform, but with only read permissions.

Administration of these permissions require Administrative privileges. As an Admin, the user can then assign global permissions to groups and users. Depending on the requirement, the user can also have Resource specific permissions depending on how their group is configured.

Global Permission Details

Global groups are visible under the "Groups" section of the **Users** tab. In addition to the two default Global groups, new Global groups may be created through the "Add Group" button.

Under the resource selector, chose the "TLR" Resource ("Top Level Resource"), and then check permissions as desired.



Global groups may also be edited just like standard groups, through selecting the Action Menu (Wrench Icon) to bring up the group information details. Groups may be deleted by selecting the red circle icon.



Group details are the same for Global groups as for non-global groups, excepting that the resource selected is TLR (Top Level Resource). You may choose to edit the name, enable/disable the group, show or hide C/R/U/D permission details, and view users assigned to that group. Be sure to save any changes after editing.



Details on each global permission option is as follows:

Global Permission	Description
Create	Ability to create records of a certain type
Read	Ability to read records of a certain type
Update	Ability to update existing records of a certain type
Delete	Ability to delete records of a certain type

Functional Area	Description
IPAM	IP Address Management functionality - this covers the IPAM Tab in addition to the IPAM "Gadget" that can be present in Resources.
Peer	Peering functionality - covers the Peering Tab, both the Communication Manager and the Session Manager.
Resource	Resource functionality - this controls access for Resources depending on either the TLR or the individual Resource(s) selected. DNS zones, records, and servers are included as "Resources".
User	User/Group management - this controls access for User and Group functions within the administrative area for ProVision.
SWIP*	This affects the SWIP/RPSL integration for ARIN/RIPE. This way a user can either be enabled to have this capability or not.
Admin*	This controls whether a user is a administrator for the global ProVision application.

*

SWIP and Admin functions are only visible when Show Details is selected

Additional Information

For more information on Users and Groups, see the following areas:

Users & Permissions

Resource Permissions

Users and Groups

Verifying Permissions

Resource Permissions

Resource Permissions

Resource Permissions apply to designated Resources within ProVision. Rather than allowing a user full access to the ProVision platform, you can choose to limit a user's access to only certain resources or functional areas.

Administration of these permissions require Administrative privileges. As an Admin, the user can then assign resource permissions to groups and users.

Resource Permission Details

Resource Permission groups are visible under the "Groups" section of the **Users** tab. New groups may be created through the "Add Group" button.

Under the resource selector, chose one or more resources for which you want to define permissions, and then check permissions as desired.

Additional Resources may be added to the list by clicking on "Add more Group Permissions".

Keep in mind when working with individual Resource permissions that tasks which require moving entities (zones, IPAM blocks, attaching servers, etc) from one resource to another, require the user to have permissions on both the originating Resource and the destination Resource.

Thus, moving IP blocks from "6connect Labz" to "7connect Labs" requires the user to have IPAM and Resource permissions on both Lab resources.

Similarly, attaching a DNS server to a DNS Group requires Resource permissions on both the DNS Group and the intended DNS server to attach.

Here, we show making a group called "Some Lab Group", whose users we want to be able to access two resources: 6connect Labz and 7connect Labs.

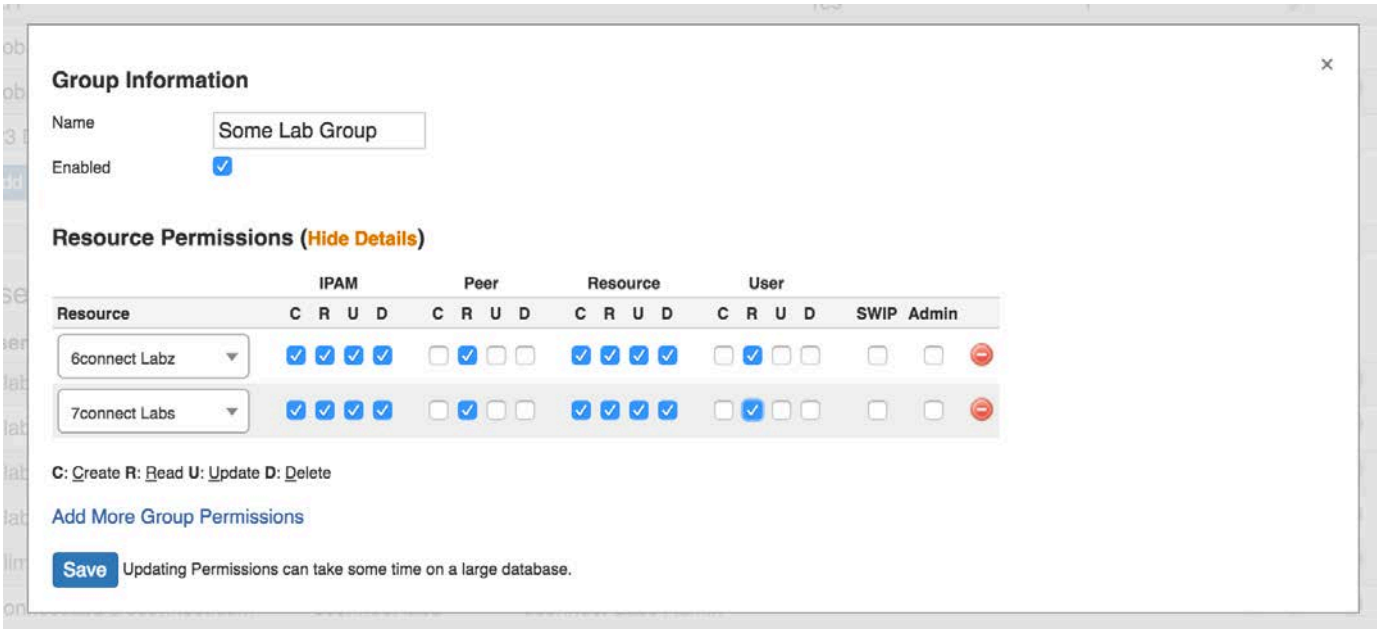
These users will be working extensively in IPAM and Resources, so we give them full access to those areas of ProVision.



However, we also want them to view other information in Peering and Users, but not edit it.

Click on "Show Details" to fine-tune the permissions, and then check the "R" column under Peer and User.

Lastly, hit "Save" to save our changes.



Details on each resource permission option is as follows:

Resource Permission	Description
Create	Ability to create records of a certain type
Read	Ability to read records of a certain type
Update	Ability to update existing records of a certain type
Delete	Ability to delete records of a certain type

Functional Area	Description
IPAM	IP Address Management functionality - this covers the IPAM Tab in addition to the IPAM "Gadget" that can be present in Resources.
Peer	Peering functionality - covers the Peering Tab, both the Communication Manager and the Session Manager.
Resource	Resource functionality - this controls access for Resources depending on either the TLR or the individual Resource(s) selected. DNS zones, records, and servers are included as "Resources".
User	User/Group management - this controls access for User and Group functions within the administrative area for ProVision.
SWIP*	This affects the SWIP/RPSL integration for ARIN/RIPE. This way a user can either be enabled to have this capability or not.
Admin*	This controls whether a user is a administrator for the global ProVision application.

*

SWIP and Admin functions are only visible when Show Details is selected

Additional Information

For more information on Users and Groups, see the following areas:

Users & Permissions

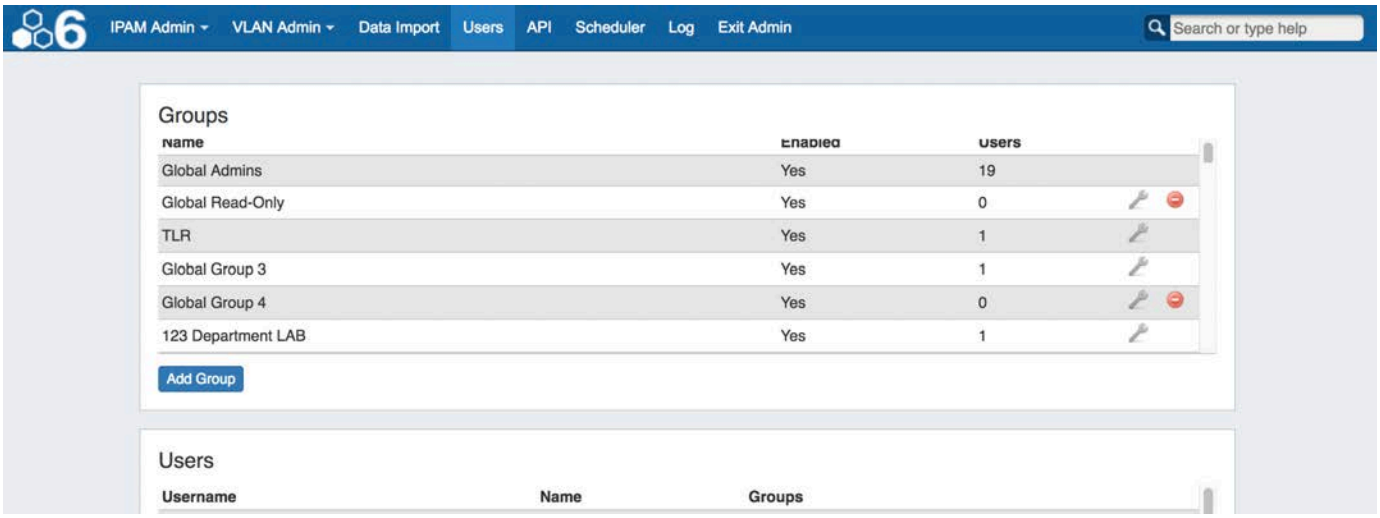
Global Permissions

Users and Groups

Users and Groups

Managing Users and Groups

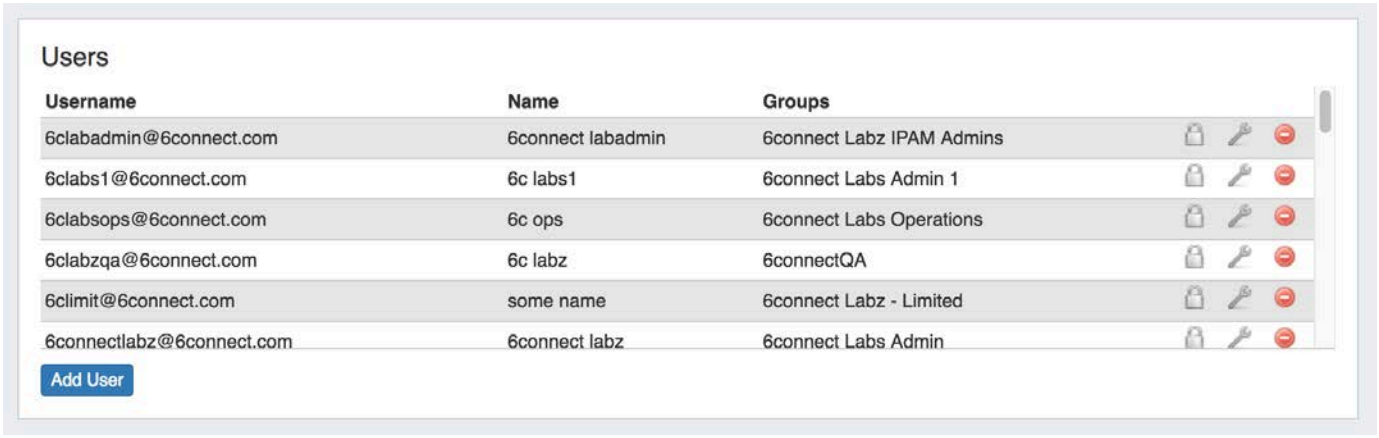
Users and Groups are managed from the Admin area of ProVision, under the **Users** tab. Both Global and Resource level groups are shown.



- Managing Users and Groups
  - Users Overview
    - Creating/Editing User Accounts
    - Setting/Resetting User Passwords
    - Removing a User
  - Groups
    - Default Groups
    - Add or Edit a Group
    - Removing a Group
  - Additional Information

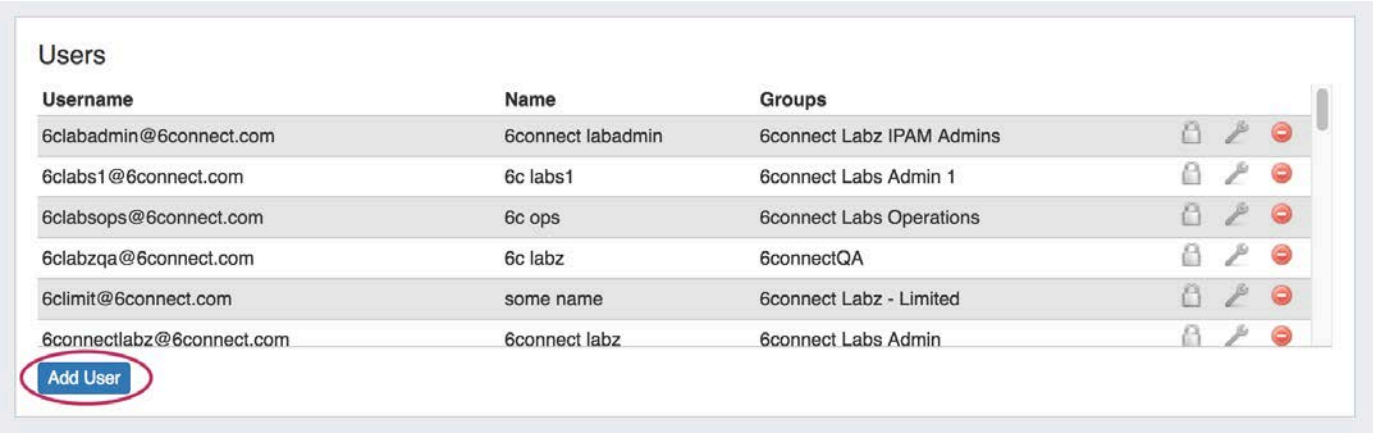
Users Overview

A User is defined as a single login account that accesses ProVision. In the Users section, users may be added, edited, assigned to groups, have password information reset, or be removed.

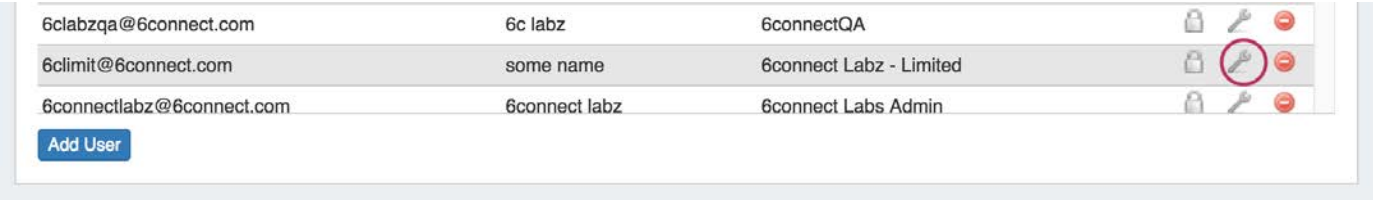


Creating/Editing User Accounts

Add Users by click on the "Add User" button at the bottom of the "Users" section of the page. The Edit User dialog will pop up.

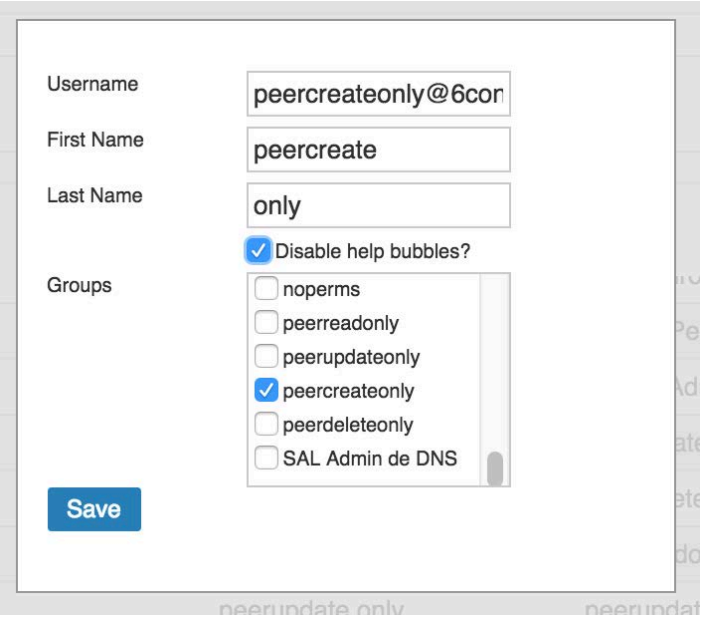


You can also bring up the same dialog to edit a user's information by clicking on the Action Menu (wrench icon) at the end of the row for the user.



When creating or editing User accounts, you will be presented with the following options.

You may enter or edit the Username, First Name, Last Name, whether to enable help bubbles, and select one or more permissions groups to which to assign the user.



When assigning users to multiple permissions groups, be sure to review the group permission details to ensure that they do not conflict with eachother!

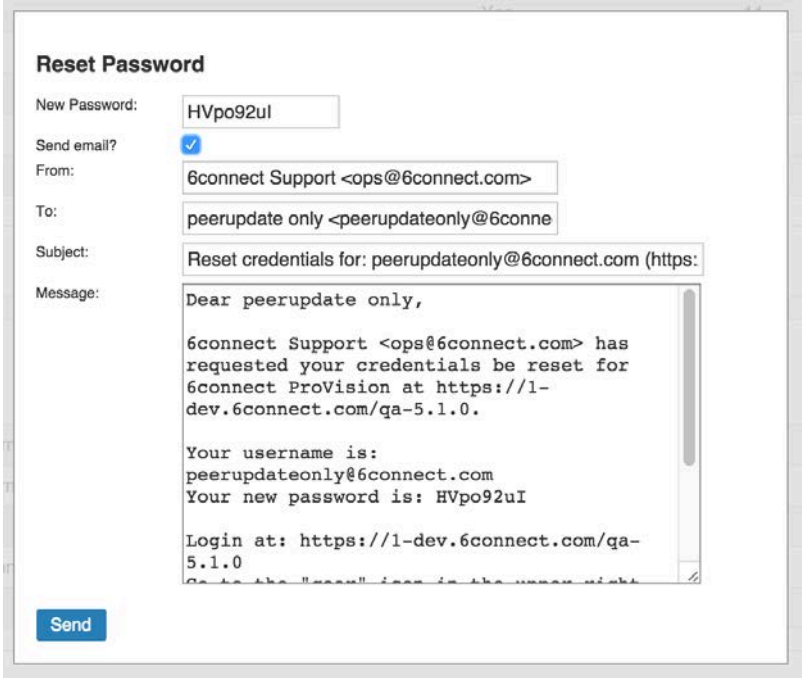
Setting/Resetting User Passwords

When you click on the padlock icon, you will be presented with options to set a new password and/or send a password reset email to the intended user account.



From here, you can choose a new password, or use the automatically generated random password.

If you select the checkbox next to "Send email?", email fields will appear that may be edited as desired. When complete, hit "Send".



Removing a User

To remove a user from the Users list, click on the red circle "delete" icon next to that user.



Groups

ProVision administrators can also create permission groups to assign users to. This allows more control over user roles.

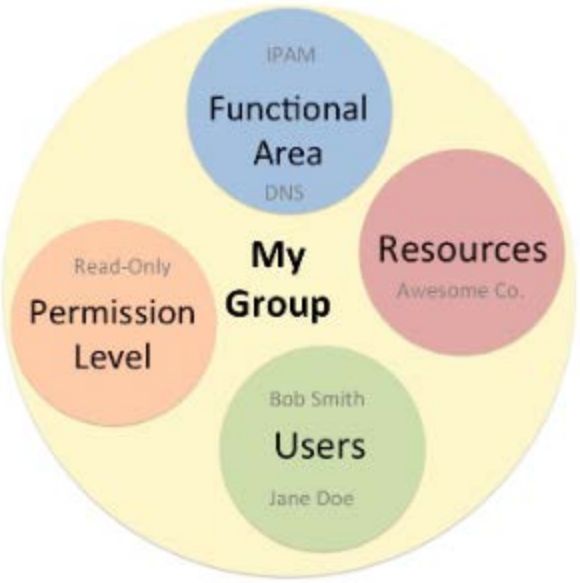
A Group has four elements associated with it:

Users: The Users that are assigned to the group, and will be limited by the selected permissions.

Resources: Resources access may be set to Global TLR (applies to all resources), or to the level of individually selected resources.

Functional Area: The ProVision functional area (IPAM, Resources, Peering, etc) for the selected resource(s) that permissions are set under.

Permission Level: Create, Read, Update, and Delete (C/R/U/D) permissions may be set under each functional area for each resource selected.



In the example above, the group “MyGroup” allows the users Bob Smith and Jane Doe to only read IPAM and DNS data for the Resource “Awesome Co”.

Through the use of these four elements, Groups can create permissions structures that are as flexible as you need.

Default Groups

Two default groups are available initially in ProVision:

- Global Admin
- Global Read-Only

These Global groups allow for access to the entire ProVision platform. You may add additional Global groups by creating a new group using the TLR (Top Level Resource). You may also add detailed resource level group permissions by selected individual resources.

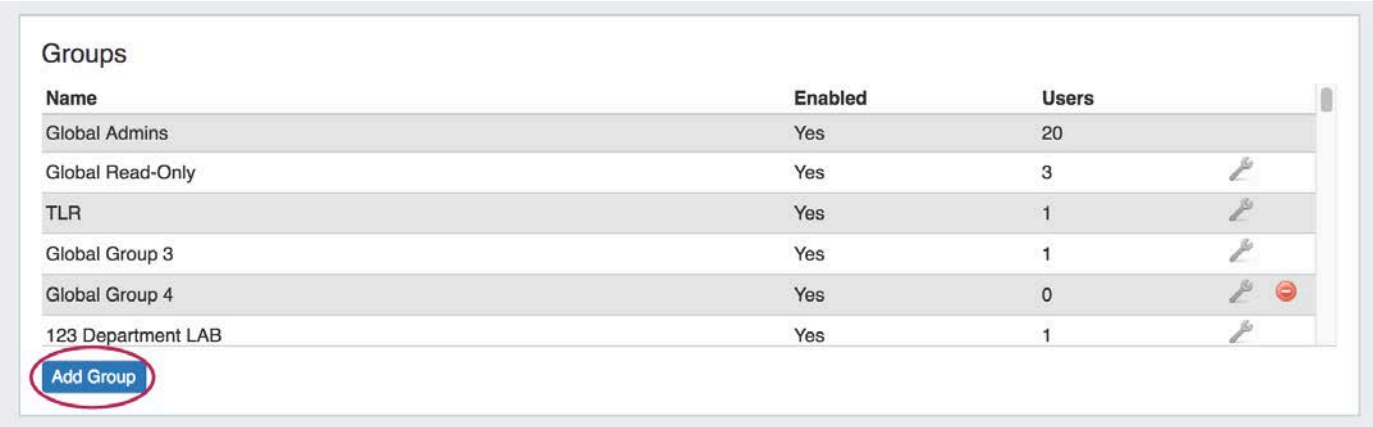
For more detail on top-level and resource permissions, see Global Permissions and Resource Permissions.

Overlapping group and user permissions

Permissions are inherited based on the hierarchy of the objects, unless you specify a different permission!

Add or Edit a Group

New Groups can be created by ProVision administrators by pressing the green "Add Group" button. To edit a group, you may click on the Action Menu (wrench icon) for the group.



After hitting the "Add Group" button or wrench icon, the Group Information screen will pop up.



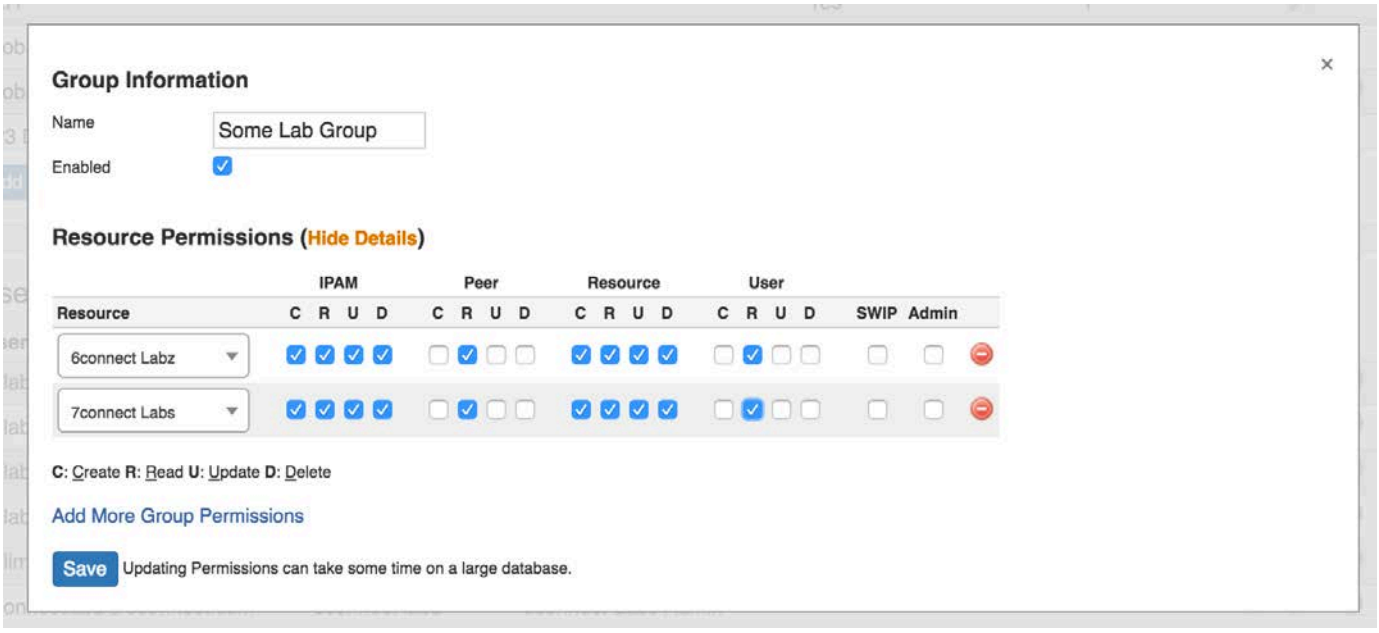
Add in the name of the new group, and set the permissions for that group by defining the resource(s) in the dropdown menu, checking the functional areas that you want accessible.

Click "Show Details" to fine tune the functional areas into Create/Read/Update/Delete level permissions.

To add permissions for additional Resources, click "Add More Group Permissions", select the Resource, and check the desired permissions.

To delete a Resource from the permissions list, simply click the red icon.

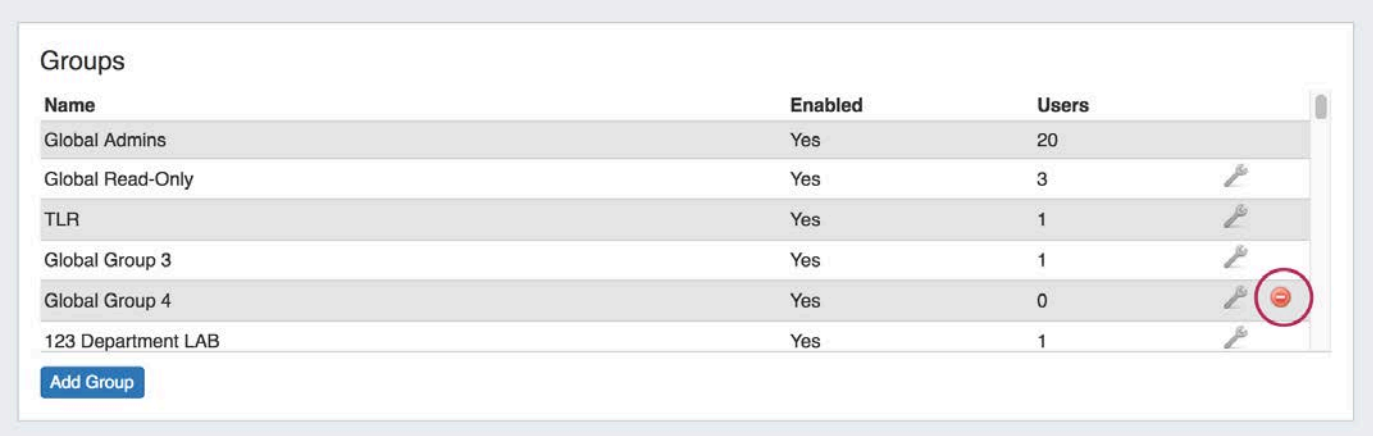
In the example below, we make a group called "Some Lab Group", whose users we want to be able to access two resources: 6connect Labz and 7connect Labs. These users will be working extensively in IPAM and Resources, so we give them full access to those areas of ProVision. However, we also want them to view other information, but not edit it. We click on "Show Details" to fine-tune the permissions, and then check the "R" column under Peer and User.



Click "Save" when complete. After adding the group, you can add users to the group by selecting the group when editing a user account.

Removing a Group

To remove a user from the Users list, click on the red circle "delete" icon next to that user.



Additional Information

For more information on Users and Groups, see the following areas:

Users & Permissions

Global Permissions

Resource Permissions

Verifying Permissions

Verifying Permissions

Verifying Permissions

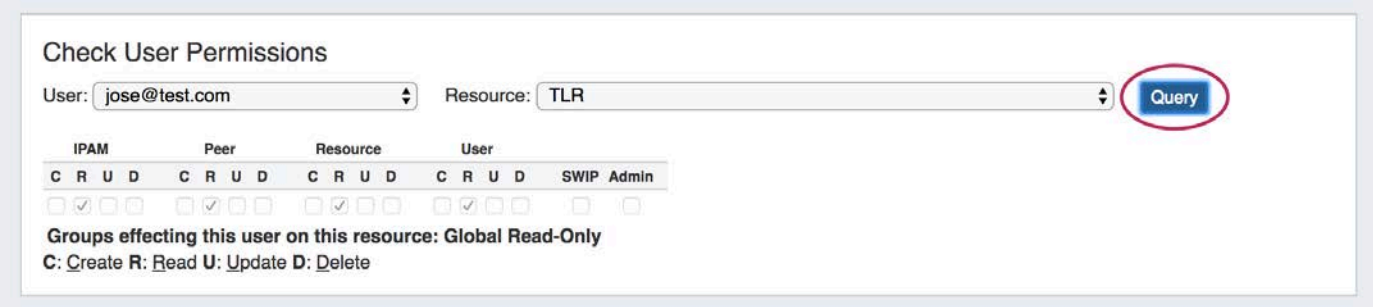
- Verifying Permissions
  - Check User Permissions
  - User Permissions Chart
  - Additional Information

Check User Permissions

Users & Permissions is accessed from the Admin screen under the **Users** tab. Here, you will find tools for adding and managing permissions groups, users, and running queries for verifying a user's specific permissions.

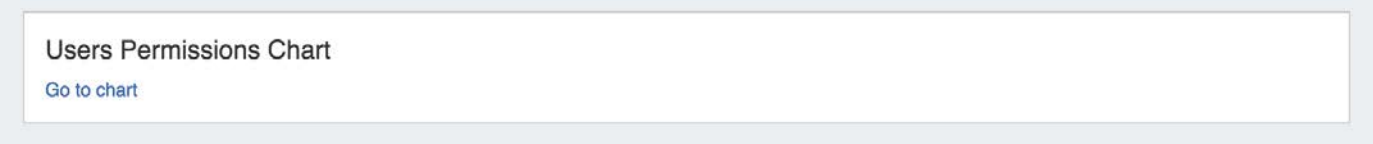
To verify the permissions of a certain user, simply select their user account from the dropdown menu, the resource you are checking against, and click on the green "Query" button.

The resulting output will display the detailed permissions for that user and resource combination, as well as list groups effecting the user.

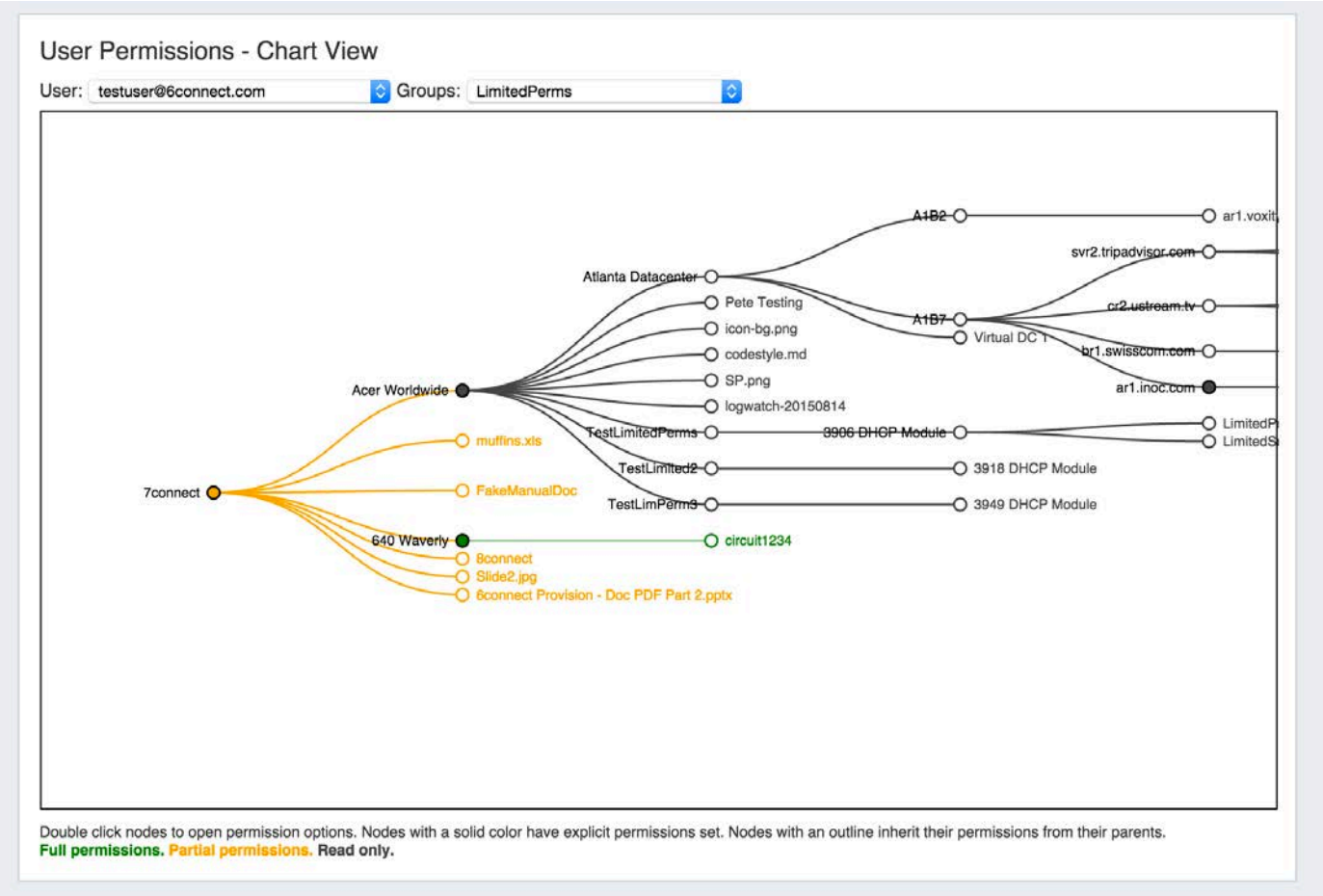


User Permissions Chart

The User Permissions Chart is accessed from the bottom of the **Users** tab page. Click "Go to Chart" and the chart page will open.

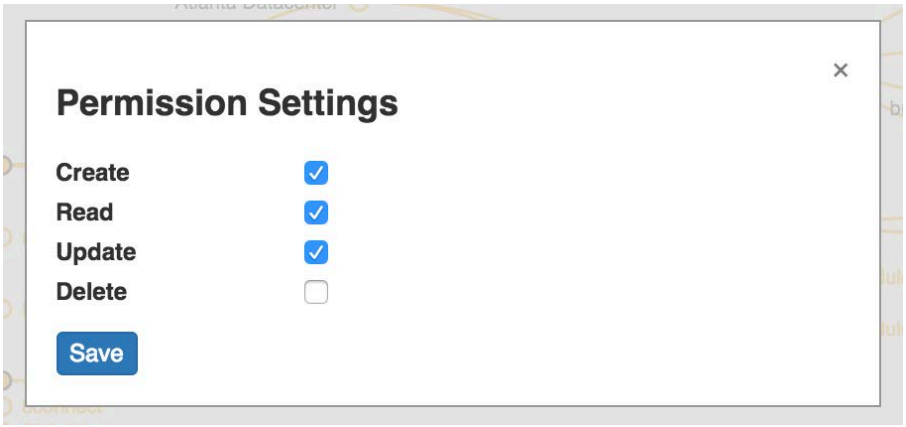


You can graphically view the permissions level of a user by selecting their user name and group from the dropdowns at the top of the chart. The resulting chart will be color coded depending on permissions level.



Single clicking on a filled circle resource node will expand and collapse the node.

Double clicking on a resource will open up the Edit Permissions box, where CRUD permissions may be updated for allowed Resources. Check / Uncheck the desired permissions level, and click save.



Additional Information

For more information on Users and Groups, see the following areas:

Users & Permissions

Global Permissions

Resource Permissions

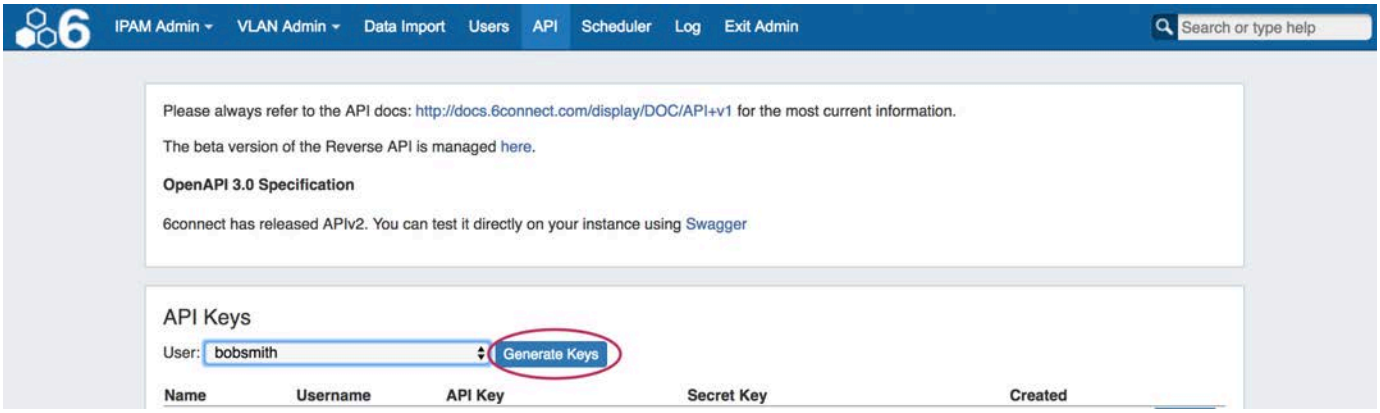
API Tab

API Tab

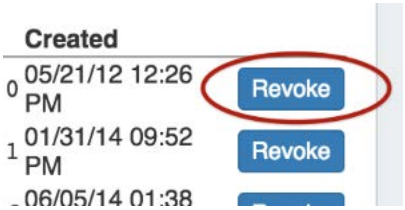
The **API** tab allows you to create and manager API keys for users. It also provides links to APIv1 / APIv2 documentation and beta Reverse API management.

Create / Revoke a Key

To create a key, simply select the user, and click on "Generate Keys". The Name, Username, API Key, Secret Key, and Created date information will be added to the list below.



To revoke a user's key, click "Revoke" at the end of their entry.



For detailed information on working with API features, please refer to ProVision Developer Tools and API v1.

Table of Contents:

- Reverse API 1

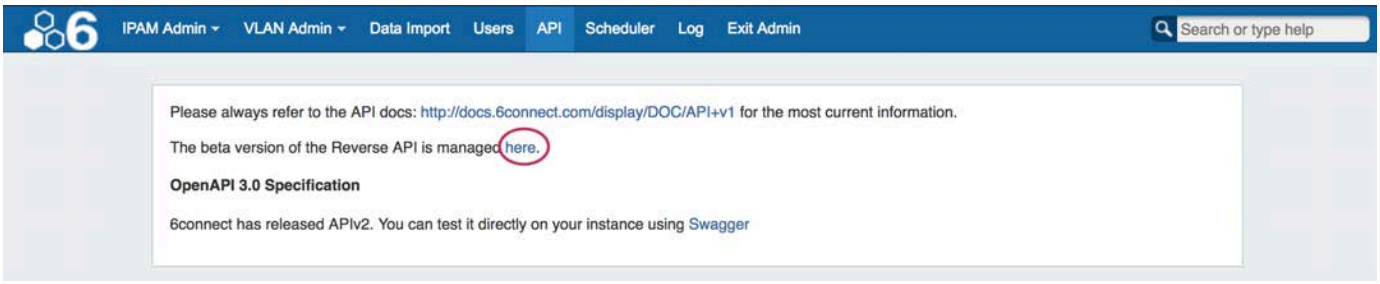
Reverse API 1

Reverse API User Interface

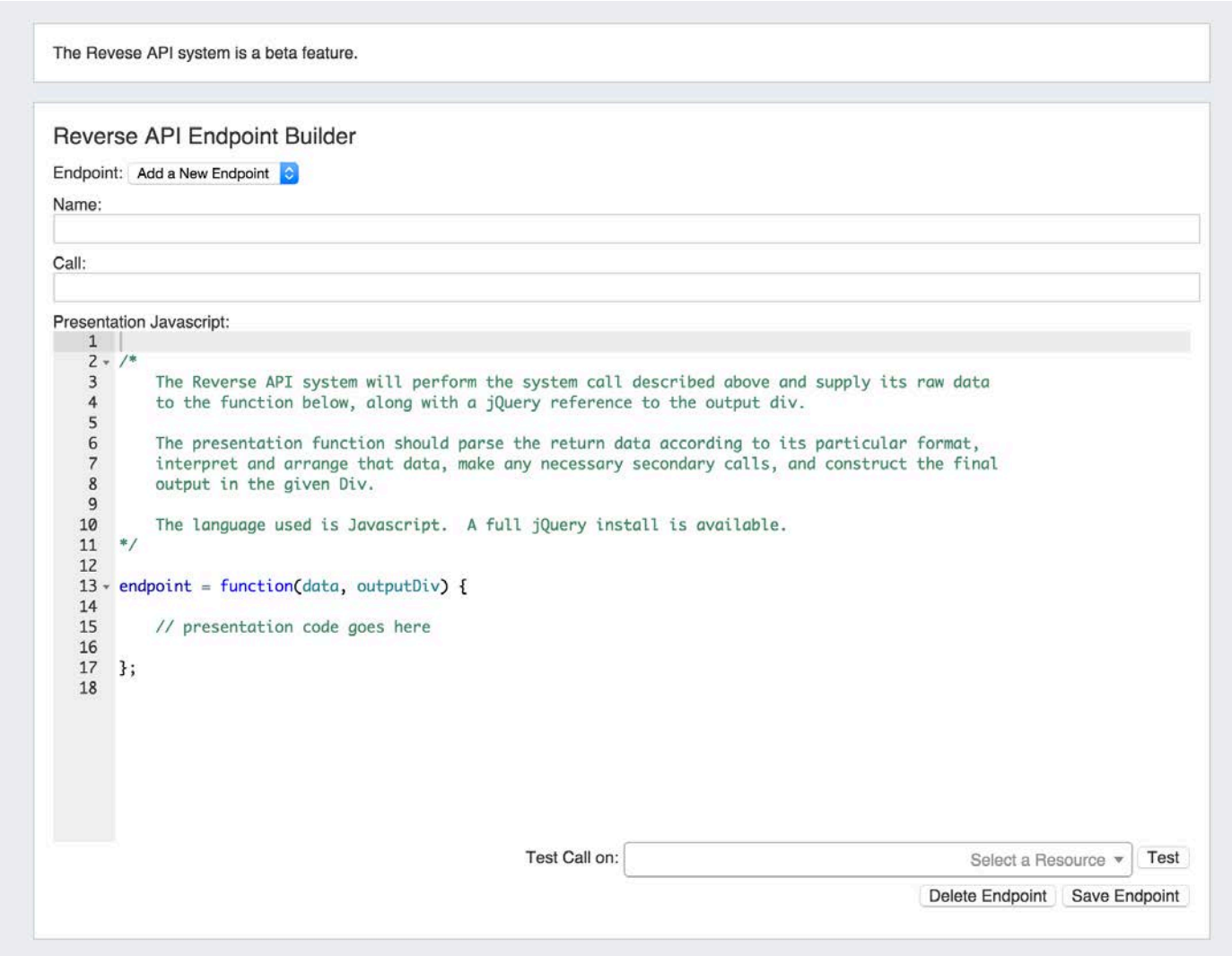
This is a beta feature

- Reverse API User Interface
  - Add a New Endpoint
  - Delete an Endpoint
  - Edit an Endpoint
  - Displaying the Endpoint

The Reverse API endpoint builder is accessed from the Admin section of ProVision, under the **API** tab.



Clicking on the link circled above will take you to the Reverse API Endpoint Builder interface.



Add a New Endpoint

To add a new endpoint

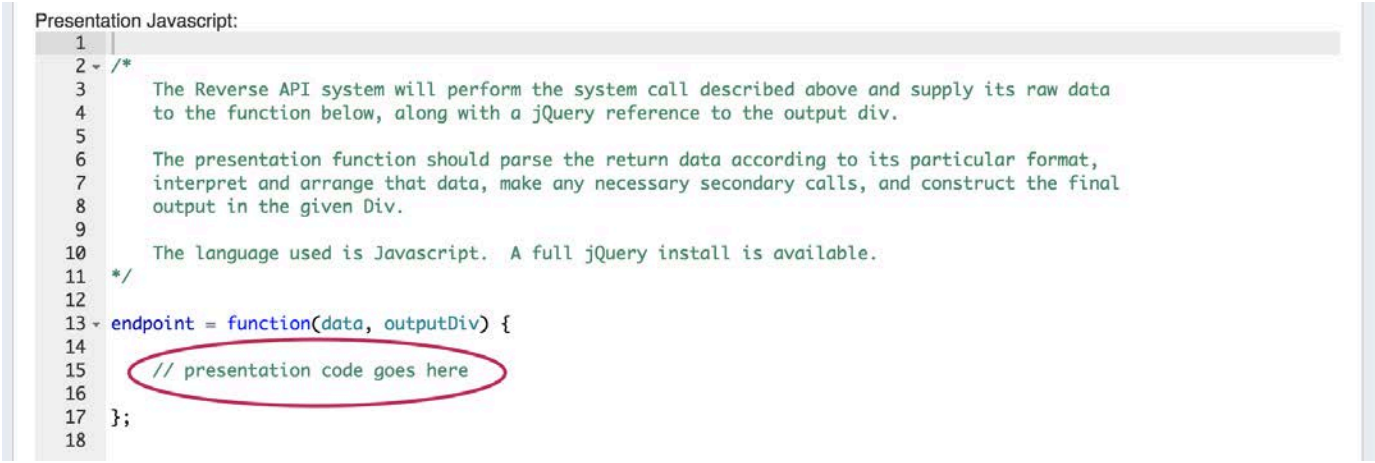
1) Select "Add a New Endpoint" next to "Endpoint".



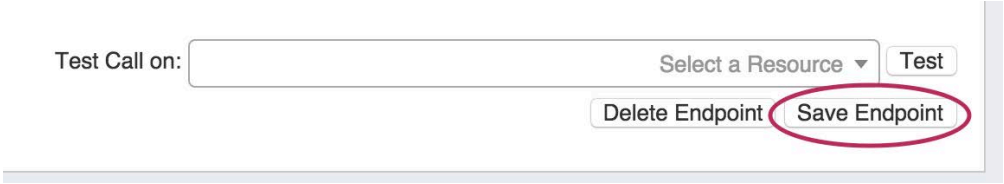
2) Type in the desired endpoint name and call



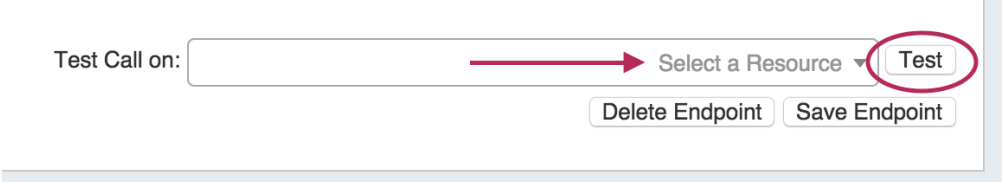
3) Review / Edit the presentation Javascript as desired for your output



4) Save your new endpoint



5) Test the call on a selected Resource by selecting a Resource from the dropdown, then hitting the "Test" button.



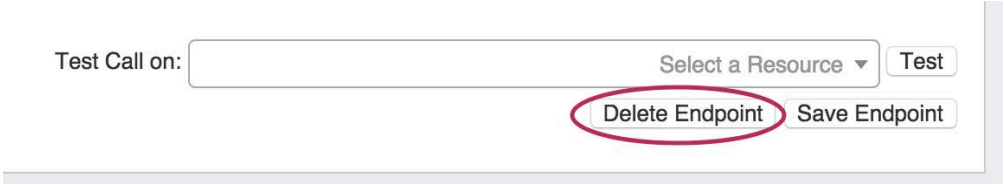
Note: The selected Resource must have a Reverse API gadget enabled section! See Customizing Sections and Gadgets for information on enabling gadgets for a section.

Delete an Endpoint

1) Select the endpoint name next to "Endpoint".



2) At the bottom of the screen, click "Delete Endpoint"



Edit an Endpoint

1) Select the endpoint name next to "Endpoint".



4) Save the endpoint



Displaying the Endpoint

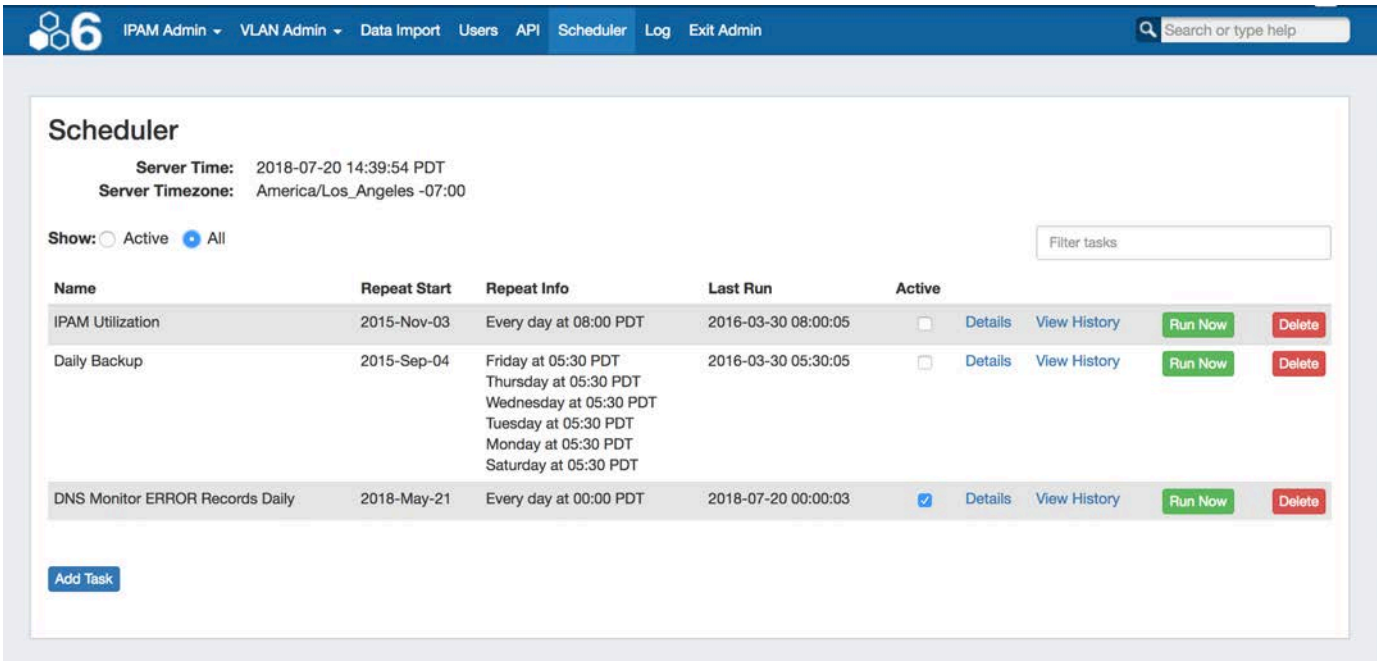
The created Reverse API endpoint may be added to the Reverse API Console Gadget, and shown on a Resource's Entry Page.

To add the Reverse API Gadget to a Section, see Customizing Sections.

For information on working in the Gadget, see Gadgets.

Scheduler

Scheduler



The Scheduler tab allows you to manage and schedule repeating tasks in ProVision.

- Scheduler
  - Available Task Types
    - Enter Task Details
    - Add Repeat Settings
    - Verify Repeat Settings and Save
  - Managing Tasks
    - Activate / Deactivate Tasks
    - Edit a Task
    - View Task Log
    - Delete a Task

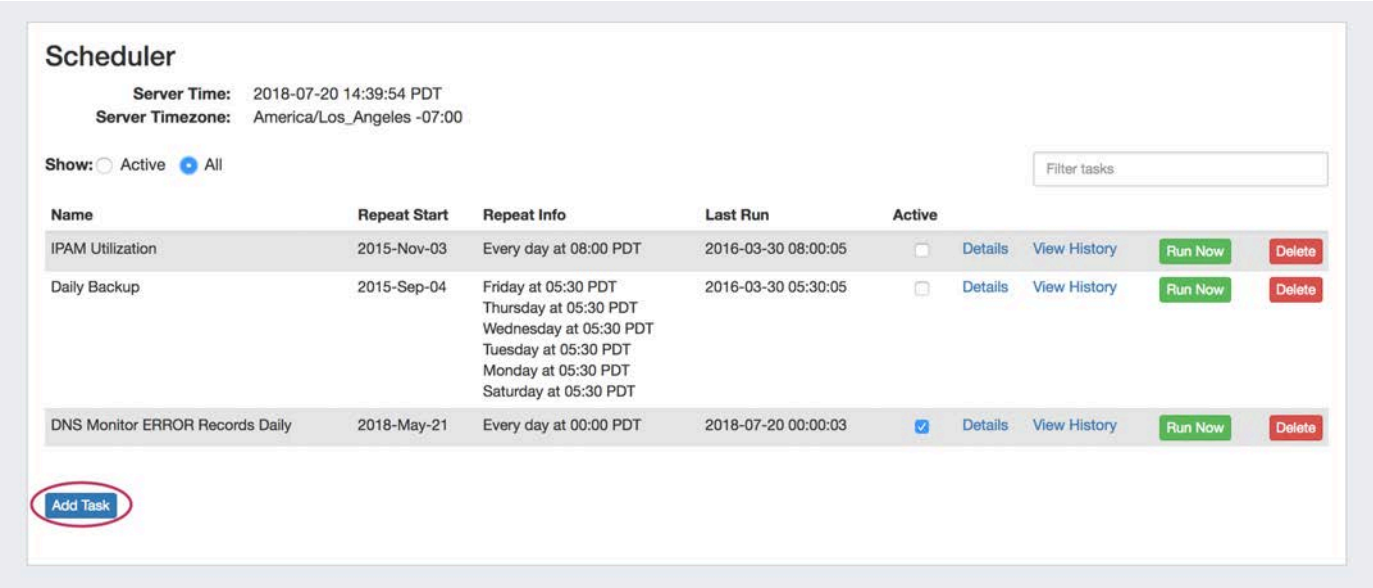
Available Task Types

Eight predefined task types are available through the scheduler:

- **IPAM Alerts** - Set a time / frequency for IPAM aggregate alert emails to be sent to alert recipients. Alert recipient email and available space threshold percentage is set through the action menu for individual aggregates under the IPAM Tab.
- **DHCP Server Push** - Pushes configuration updates to the selected DHCP server(s). Select push to a single server, or all DHCP servers
- **DNS Resource Monitor** - Checks Records/Zones periodically to monitor DNS server reponse.
- **Test Task** - Tests the scheduler and notification system - it simply sends a notification email to the provided address at the specified run time.
- **DNS Zone Transfer** - Pushes zone updates to the DNS servers. Select pushes to apply to all zones in a DNS Group, all zones on one server, or one particular zone. Requires the intended Group / Server / Zone to already exist in ProVision
- **DNS Zone Error Monitor** - Checks the DNS records in theError state to determine if they have been fixed. Default setup is once a day, and you can set it to run more often if necessary. This task may replace or supplement the manual DNS Zone / Record "Check" button, depending on set frequency.
- **Process Holding Tank** - Removes IP Blocks from the Holding Tank and returns them to the available pool.
- **Backup** - Performs a data backup to the 6connect cloud servers, or to a selected resource server existing in ProVision.

Add Task

To add a new scheduled task, hit the "Add Task" button below the Scheduler task list. You will then see the Task Detail Settings area.



Enter Task Details

Under "Task Detail", fill in the following fields:

Name: Create a name for your task

Task: Select a pre-created task from the dropdown menu.

Process Holding Tank Settings: When selecting the Process Holding Tank task, you will have the following setting options:

> **Email From / Email To**: Enter an email address to send task notifications from, and the desired recipient. This is an optional setting.

DNS Zone Transfer Settings: When selecting the DNS Zone Transfer task, you will have the following setting options:

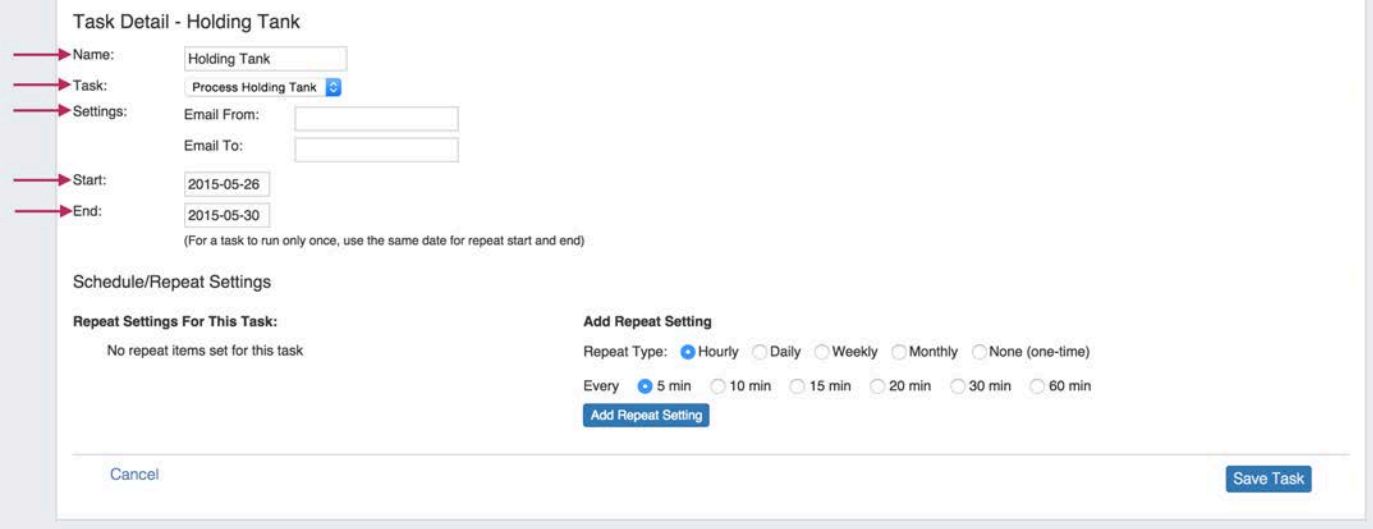
> **Select Action Type / Select Server:** When selecting the DNS zone transfer task, you can choose the action type to a) push all zones to all servers b) push all zones on a single server (then select server) or c) push one zone (then select zone).

Backup the Database Settings: When selecting the Backup task, you will have the following setting options:

> **Email From / Email To**: Enter an email address to send task notifications from, and the desired recipient. This is an optional setting.
>
> **Location:** Choose "Cloud" to save the backup to the 6connect cloud, or "Server" to save to a specified server resource already defined in ProVision. If "Server" is selected, choose the server resource from the dropdown list.

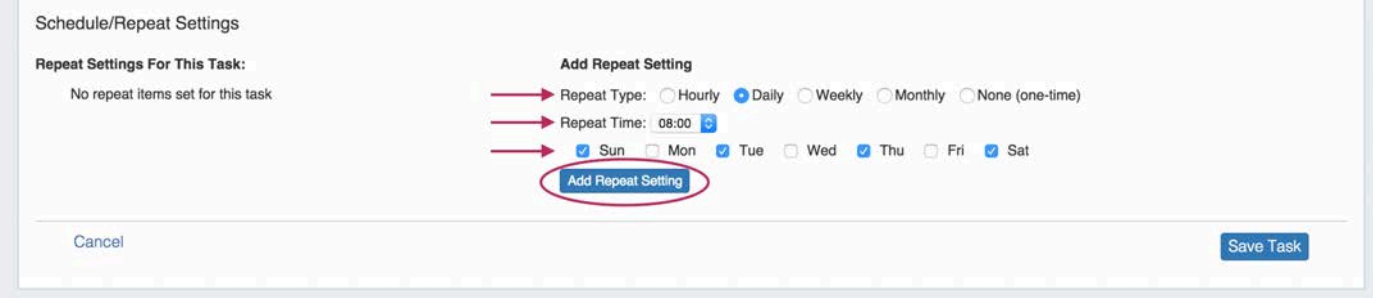
Start: The start date you wish the task to begin repeatedly occurring. By default, the current day is entered.

End: The date you wish repeat settings to end. For one-time tasks, enter the next calendar day after the start (run) date.



Add Repeat Settings

Add repeat settings for your tasks. First, select your repeat type - hourly, daily, weekly, monthly, or none (one time). The type that you select will prompt different additional setting options.



Repeat Type options and details:

Hourly: Selecting the hourly repeat type will then prompt for the number of repeat times per hour - every 5,10, 15, 20, 30, or 60 minutes. All minute selections start from the top of the hour. For example, if every 20 minutes is selected, the task will run at :00, :20, :40 during each hour.

Daily: Selecting the daily repeat type will prompt for the specific time to repeat the task, and which days you wish the task to occur. The Repeat Time is based on a 24 hour clock set at UTC time. To assist in any time zone conversions, the current UTC time is shown at the top of the scheduler page.

Weekly: Selecting the weekly repeat type prompts for the specific time to repeat the task as well as frequency by week. Weekly tasks occur on the day selected as the start date. Using the weekly option, you may choose from a range of weekly (every 1 week) to once every two months (every 8 weeks).

Monthly: Selecting the monthly repeat type prompts for a repeat time and a day. For the day option, you can either choose a specific day (ie, the 15th of each month) or a relative day (the 1st Friday of each month).

For Monthly repeat settings, verify that the calendar day you select the task to occur exists! For example, if you select the 5th Saturday of each month, the task will repeat only in months which have 5 Saturdays and skip all other months.

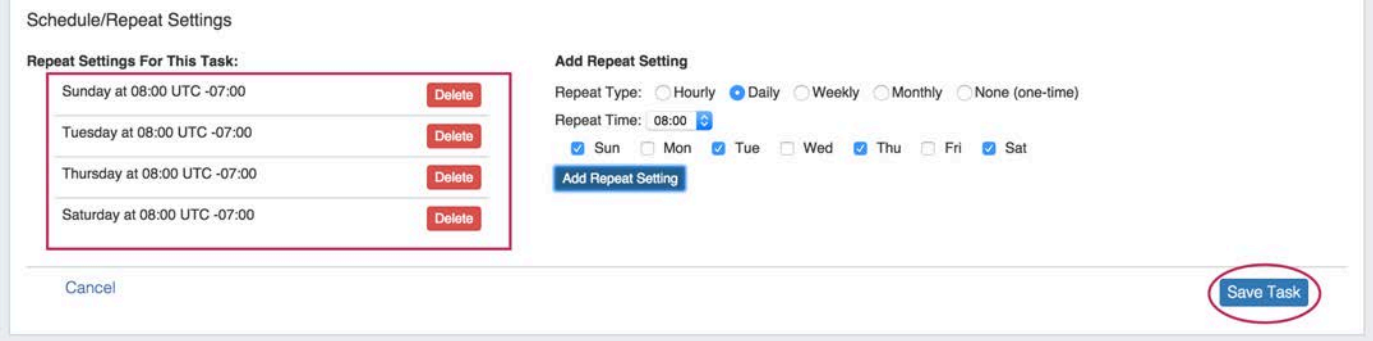
None (one-time): Select the time and day that you wish the single occurrence task to run. (Ensure that "End Date" in task details is set to the day after the desired run date).

After you have selected your Repeat Setting options, click the "Add Repeat Setting" button.

Verify Repeat Settings and Save

When you add repeat settings, they will appear under the "Repeat Settings For This Task" area. Verify your settings, and if desired, you may delete individual repeat settings by hitting the "Delete" button.

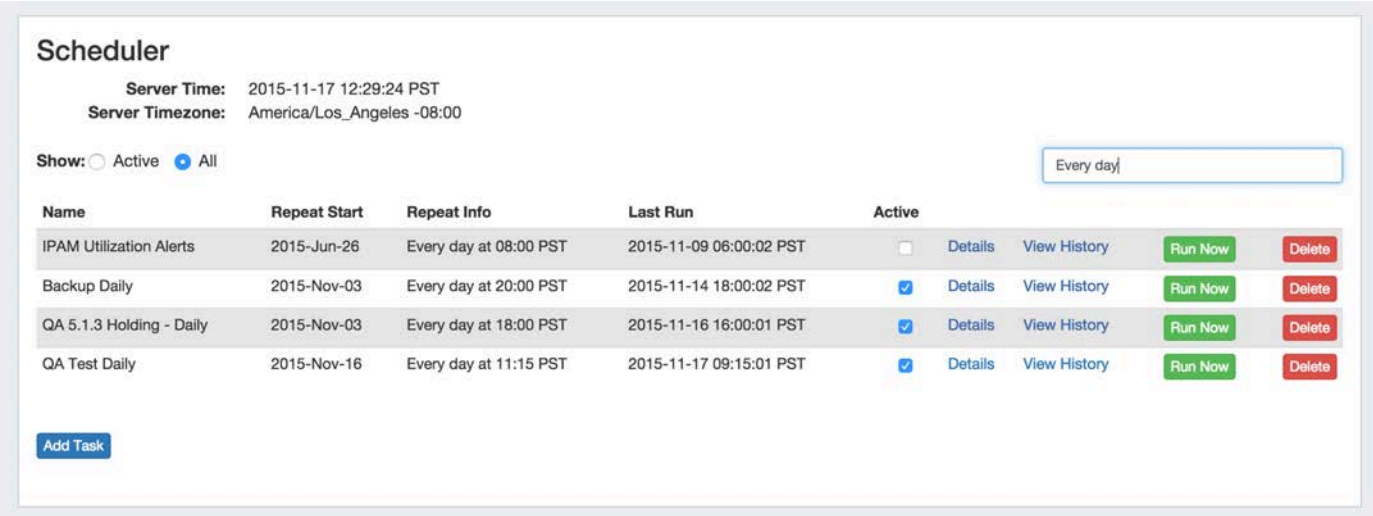
When complete, hit the "Save Task" button to finalize your changes and add your new task to the task list.



Managing Tasks

After tasks have been added, they will show in the task list along with their basic settings. The list may be filtered to show all tasks or active tasks only by selecting the radio buttons next to "Show:". Further filtering may be done by typing a Name or Repeat Info keyword into the "Filter Tasks" input box above the task list.

From the task list itself, you can enable/disable tasks, view details, view history, run tasks, and delete tasks.



Filters:

Show: Select "Active" to show only active tasks (checkbox enabled under "Active" for the task), or "All" to show all tasks

Filter Tasks (Text box): To further filter the task list, type a Name or Repeat Info keyword (ie, "Backup" or "Every day") into the Filter Tasks text box.

Task List:

Name: The task name assigned during the "Add Task" creation process.

Repeat Start: The date selected for the repeat settings to start

Repeat Info: The repeat settings chosen for the task.

Last Run: The date and time the task was last run, if applicable.

Active: To activate tasks, simply click the check box under "Active" in the task list. To deactivate a task, uncheck it. By default, tasks are checked as active once created.

Details: This link will bring up the task details and repeat settings, which then may be edited and re-saved if needed.

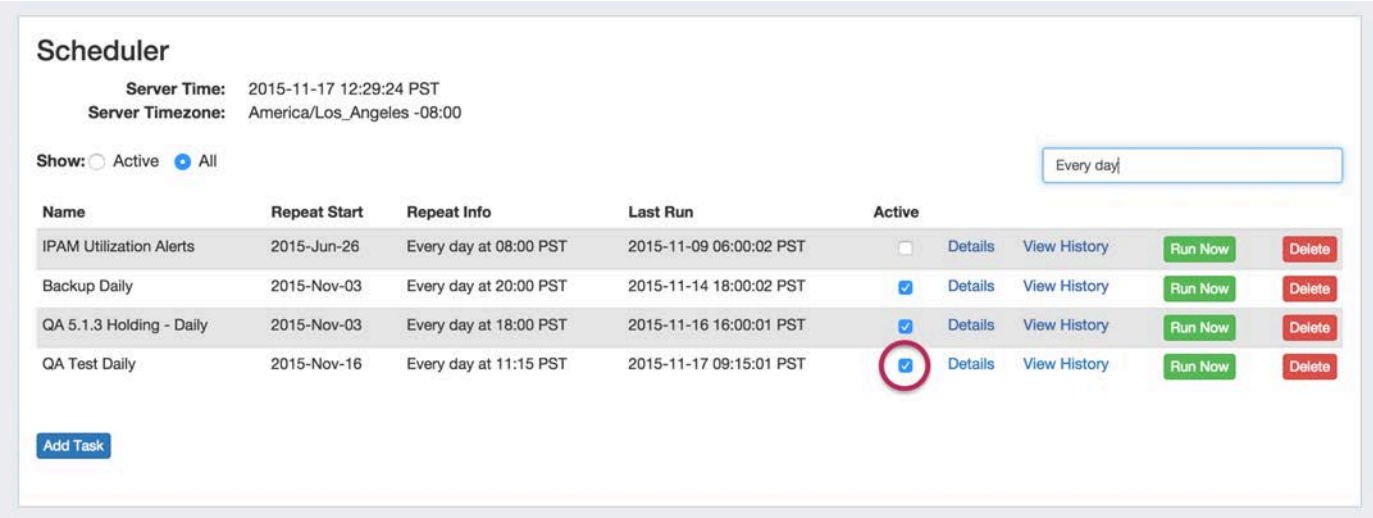
View History: This link will show the log for task actions.

Run Now: The "Run Now" button will run the scheduled task when pressed, regardless of the scheduled repeat settings.

Delete: Deletes the task.

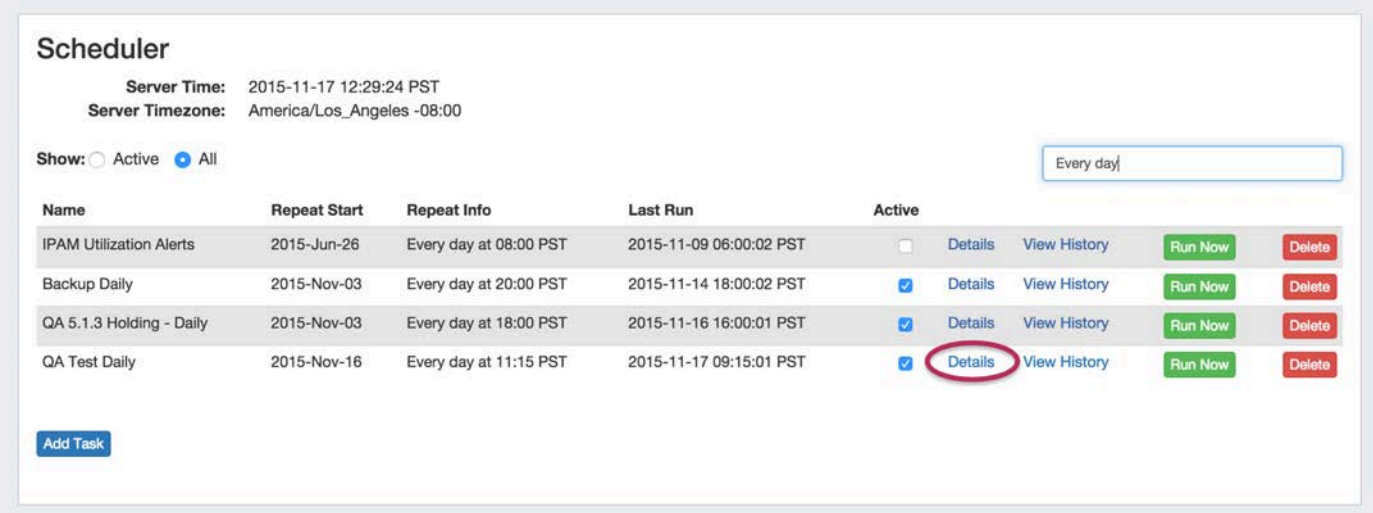
Activate / Deactivate Tasks

Click on the check box under "Active" for the task

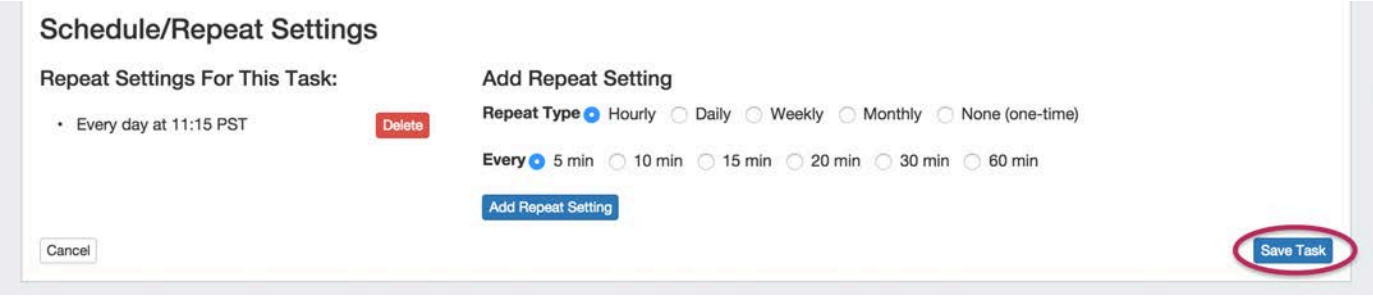


Edit a Task

Click on the "Details" link for the task, then edit the Task Detail and Repeat Settings that appear below as needed.

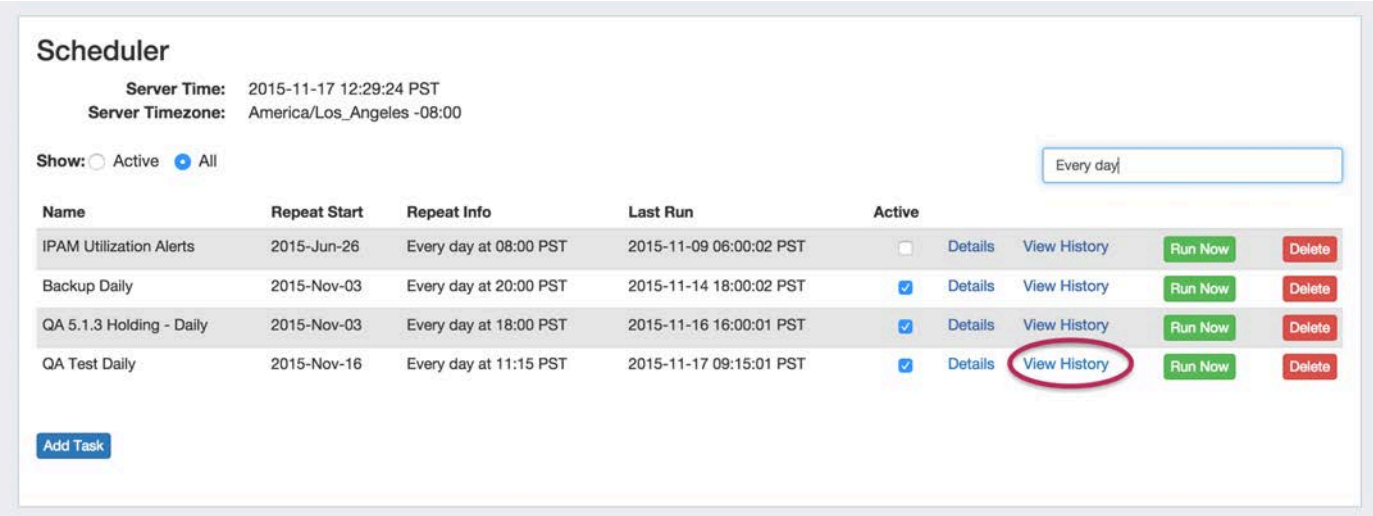


Hit "Save Task" when finished to save your changes, or select "Cancel" to close the view without saving.



View Task Log

Click on the "History" link for the task.

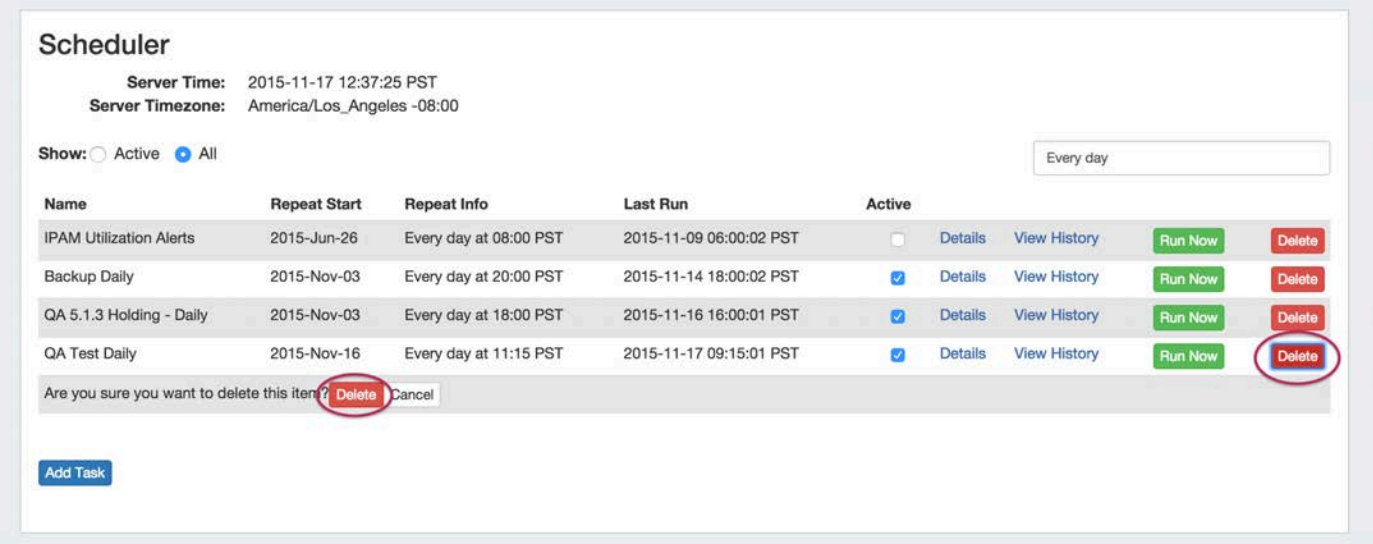


The log details for the task will be shown below. When done, click on the "Close" button to exit the view.



Delete a Task

To delete a task, hit the "Delete" button at the end of the row in the Task List. You will be presented with message asking if you are sure you wish to delete the task. Click on the "Delete" button next to the message to verify the deletion, or hit "Cancel" to exit without deleting.



Enabling the Scheduler

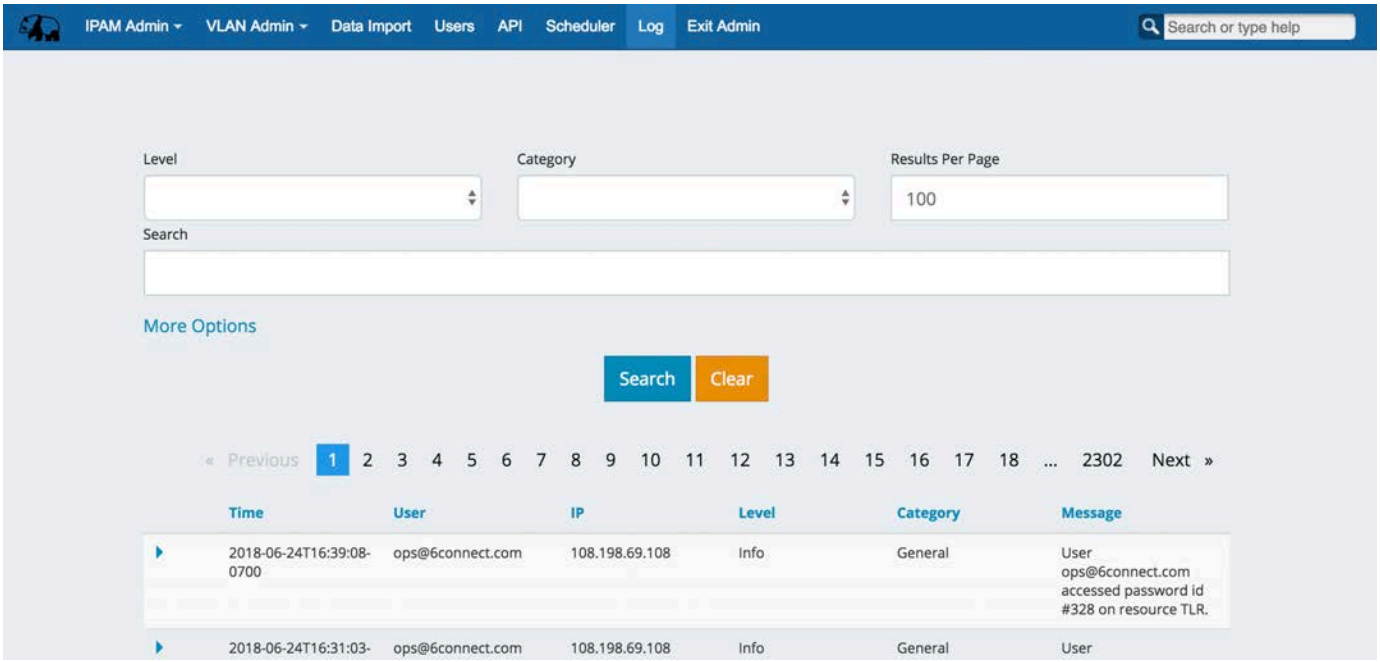
To enable the scheduler in a local instance, add the following to cron:
* * * * * /path/to/php /path/to/ProVision/scheduler/task-runner.php

Log

Log

The 6connect ProVision log provides detailed information on actions performed in ProVision. The Log is only available to Admin users.

To access the Log, either navigate to the Admin area of ProVision, then click the **Log** Tab, or follow an area-specific Log link.



Log Features

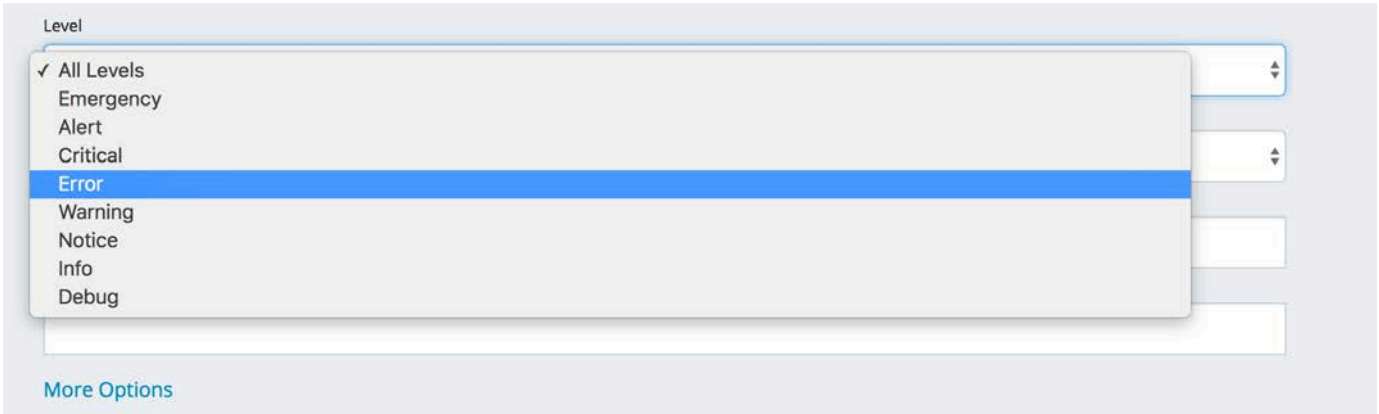
Filters and search options:

Filter the log list by selecting (or typing) the desired filter value at the top of the Log page, then click on the "Search" button.

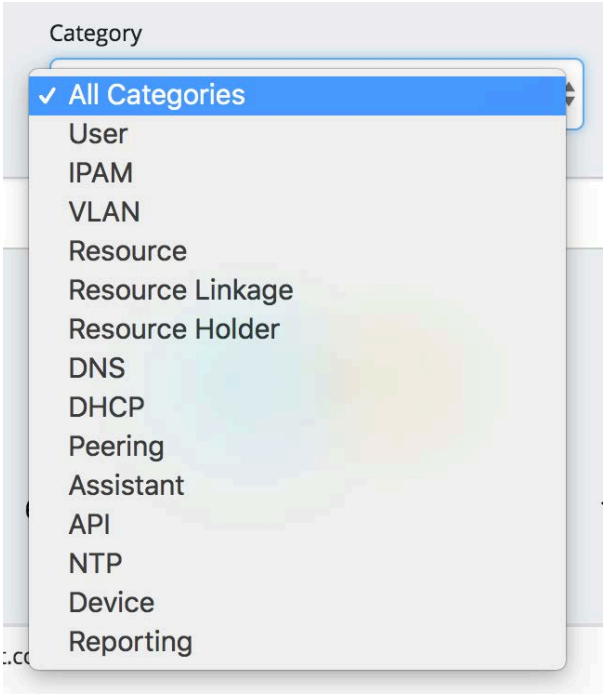
Additional options are made visible by clicking on the "More Options" link below "Search".

The following filters and options are available:

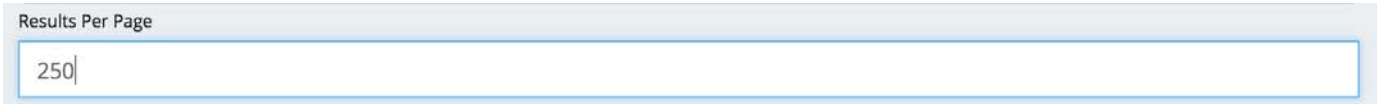
Level: Under the "Level" dropdown box, select "All Levels", "Emergency", "Alert", "Critical", "Error", "Warning", "Notice", "Info", or "Debug".



Category: Under the "Category" dropdown box, select "All Categories", "User", "IPAM", "VLAN", "Resource", "Resource Linkage", "Resource Holder", "DNS", "Peering", "Assistant", "API", "NTP", "Device", or "Reporting".



Results per page: In the "Results per page" text box, type the desired number of log entries to see per page. By default, this value is set to 100.



Search: Type a search string, such as the name of a resource, then click the "Search" button.

More Options:

Additional detailed filter and search options are available under the "More Options" link.

Log ID: Retrieve a specific log entry from its log_id parameter.

IP: Search by IP of the machine that created the event in the log.

Username: Filter logs by ProVision username, or by "Unknown" user.

Time: Filter log entries by exact time created. Must be in datetime format (2016-08-14 16:41:18).

Time Minimum: The earliest day and time of log entries to show in results. Must be in datetime format (2016-08-14 16:41:18).

Time Maximum: The latest day and time of log entries to show in results. Must be in datetime format (2016-08-14 16:41:18).

Tip

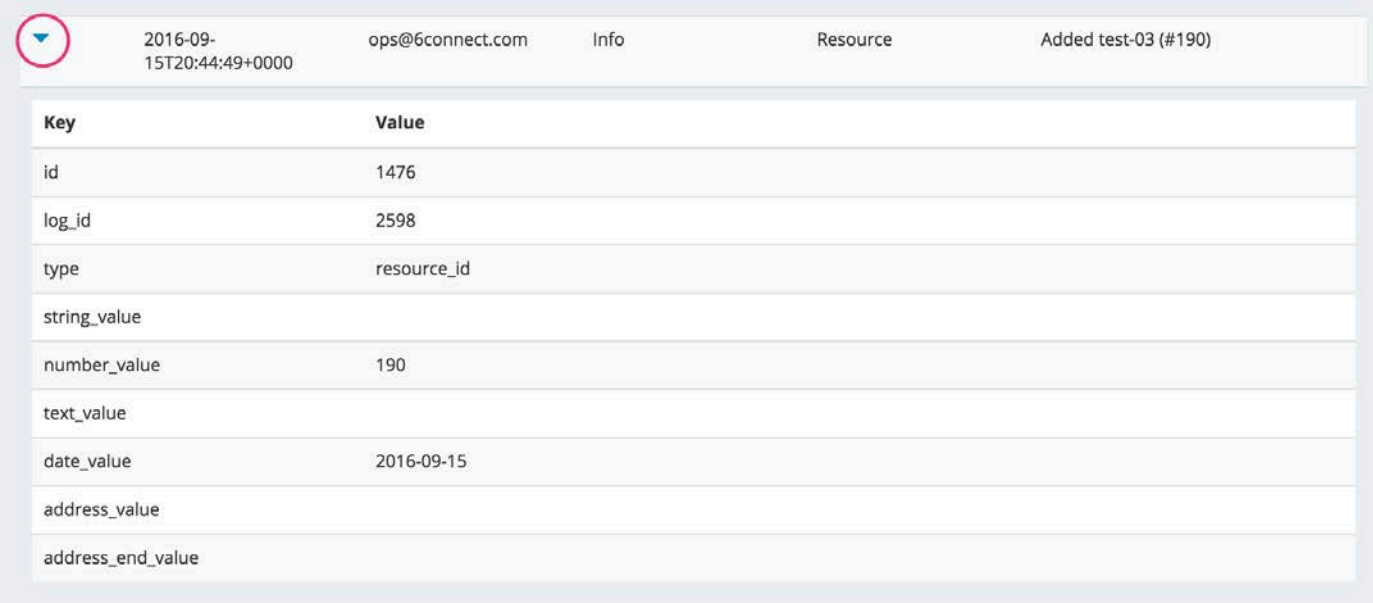
Use "Time Minimum" and "Time Maximum" together to specify a specific date / time range, such as one 24 hour period:



Log Details:

Clicking on the blue arrow on the left side of each log entry expands the entry to show additional details pertaining to that entry.

Not all fields may be populated, and not all log types have applicable details.



Detail fields may include:

id: The id of the log detail record.

log_id: The id of the parent log entry (the entry that was clicked to reveal details).

type: Log category value.

string_value: String details.

number_value: For resource type log entries, the number_value is the associated resource id.

text_value: API details for API type log entries.

date_value: Day of the log entry.

address_value: For IPAM type log entries, address_value is the starting integer value of the IP Block.

address_end_value: For IPAM type log entries, address_value is the ending integer value of the IP Block.

RIR Integration: SWIP/RPSL

Overview

ProVision supports updating SWIP/RPSL functions for ARIN and RIPE blocks through simple reassigns using the RIR Integration action.

- Overview
  - RIR Integration
    - When do I use RIR Integration and Why?
    - What is Simple Re-assign?
  - Workflow
    - 1) LIR Setup
    - 2) Assign IP blocks to the ARIN or RIPE RIRs
    - 3) RIR Integration
      - ARIN / Update SWIP
      - RIPE / Update RPSL

RIR Integration

When do I use RIR Integration and Why?

From ARIN.net:

Organizations that receive space allocations from ARIN, either directly or as a downstream customer, must provide reassignment information back to ARIN. This information must be sent within seven days of the reassignment so that the WHOIS database may be maintained. ARIN also uses utilization history, projected requirements, and other information in order to make future space allocations.

From RIPE.net:

All RIPE assignments and allocations must be registered in the RIPE Database. This is necessary to ensure uniqueness and to support network operations.
Only allocations and assignments registered in the RIPE Database are considered valid. Registration of objects in the database is the final step in making an allocation or assignment. Registration data (range, contact information, status etc.) must be correct at all times (i.e. they have to be maintained).The RIPE community's policies require LIRs to register an inetnum object in the RIPE Database for their own infrastructure and customers' networks. LIRs must ensure registration information is correct and up to date at all times.

RIPE Integration

You must be using an MD5 password to update RIPE objects from ProVision! Please refer to RIPE's authentication page for additional details:

https://www.ripe.net/manage-ips-and-asns/db/support/security/protecting-data#solving-authentication-problems

What is Simple Re-assign?

From ARIN.net:

Used to subdelegate IP addresses to a customer that does not need to:

- subdelegate the addresses to their own customers
- maintain their own in-addr.arpa delegation
- display their own point of contact (POC) information.

It can also be used to change the customer name and address information (but not the range) on an existing simple reassignment and to remove simple reassignments. It is submitted by an ARIN Online user account linked to the parent organization's Admin or Tech POC, or the Tech POC for the resource.

Workflow

1) LIR Setup

ProVision supports multiple LIRs (Local Internet Registries) in a single instance. This means that you have the ability to update SWIP/RPSL functions for a given allocation with the LIR information that you wish. LIRs are set up and managed from the IPAM Admin area of ProVision, and thus require Admin level permissions to set up.

While setting up the LIR, the POCs / Contact fields that are filled in will later be used for the RIR Integration.

For detailed step by step instructions, see:

LIR Management and Use

ARIN LIR Setup and Use

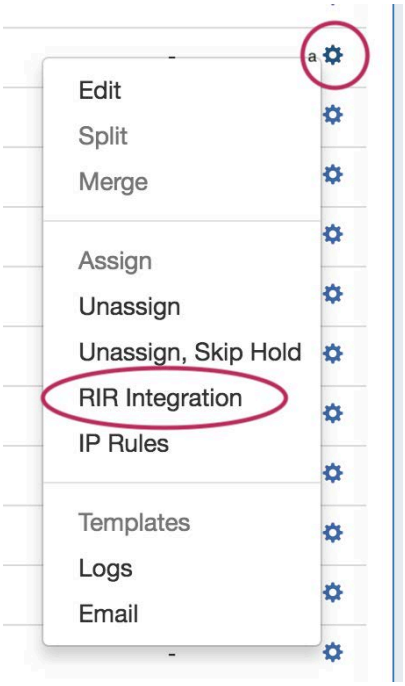
RIPE LIR Setup and Use

2) Assign IP blocks to the ARIN or RIPE RIRs

Assign an IP block to a Resource using the IPAM Gadget or the Assign function from the IPAM Manage screen. See Working with IP Blocks for additional detail.

3) RIR Integration

Once LIRs have been configured, and blocks assigned under the applicable RIR, you will be able to use the **RIR integration** feature from the Action Menu on the IPAM Manage screen or IPAM Gadget.

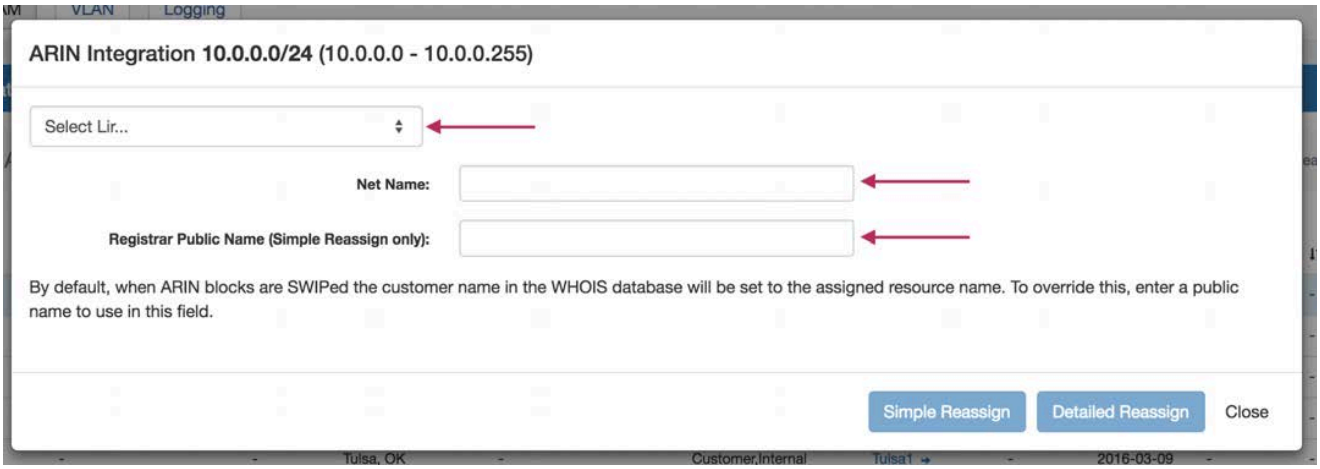


Depending on the RIR associated with the block, either an ARIN or RIPE Integration box will pop up.

ARIN / Update SWIP

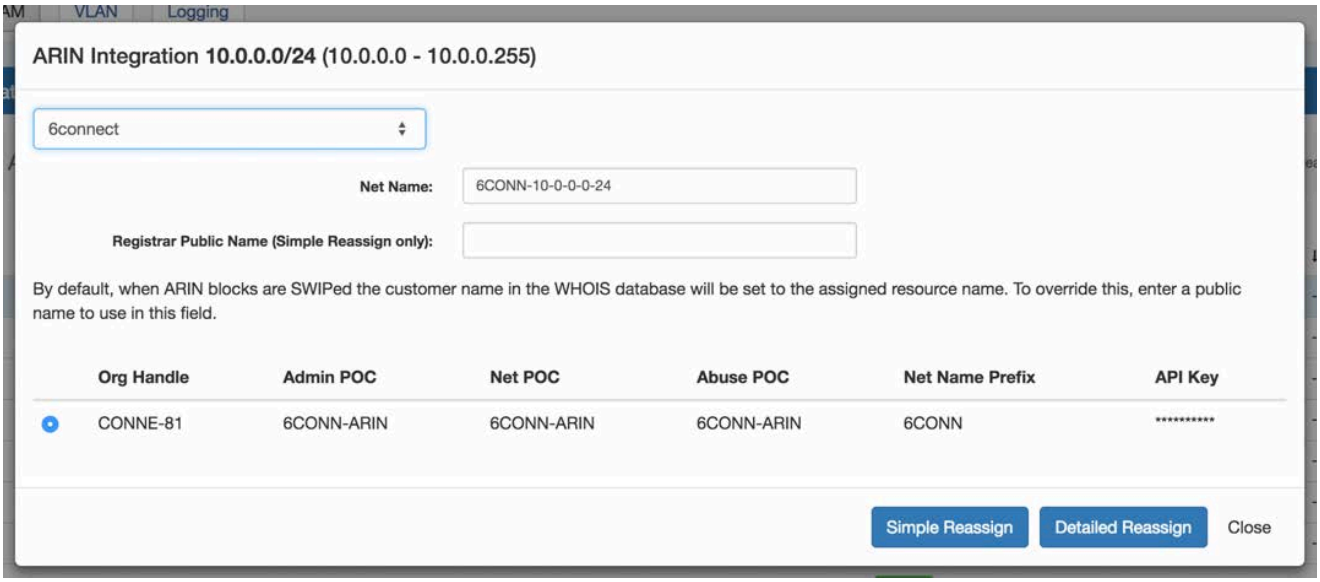
Select LIR:

Select the desired LIR, and verify the Net Name / Public Name if desired.



Reassign:

Once the LIR and Handle is selected, you can click either "Simple Reassign", "Detailed Reassign" or "Close" to exit.

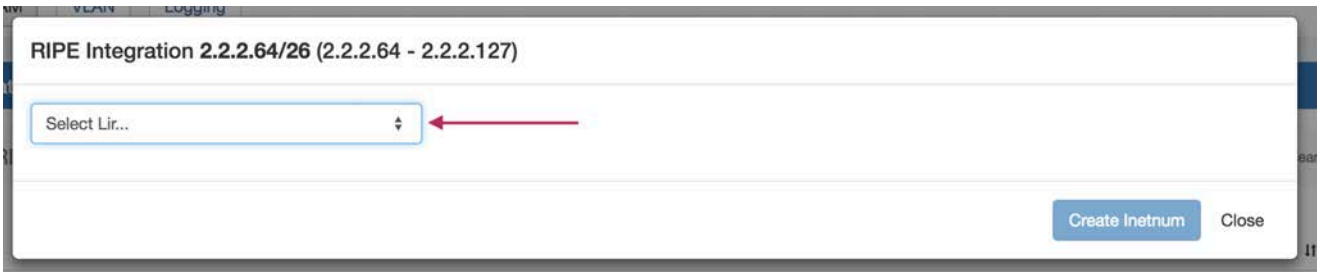


For detailed step by step instructions, see:

ARIN LIR Setup and Use

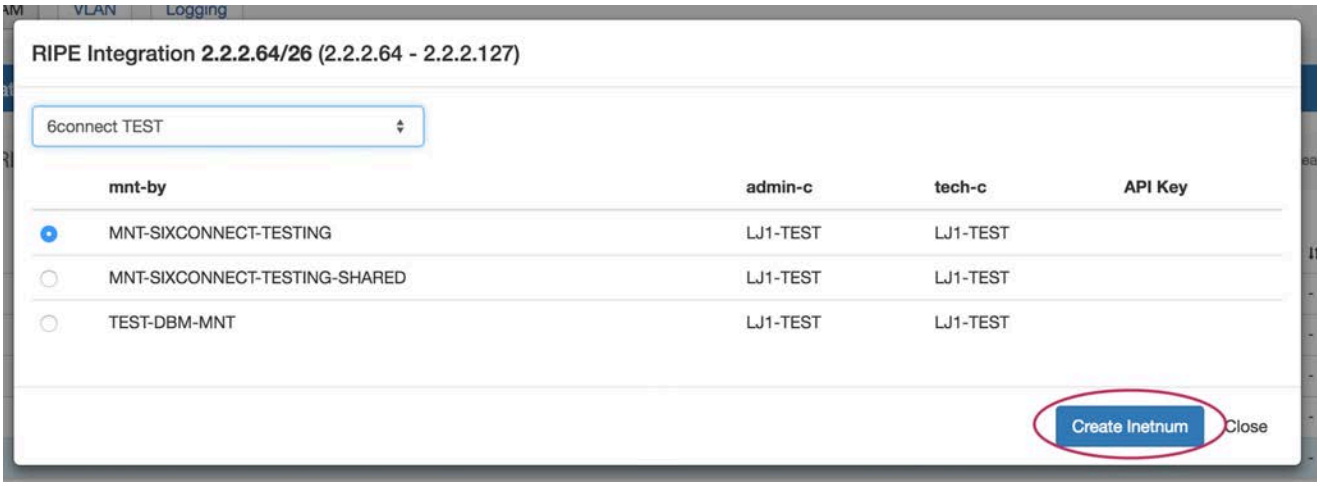
RIPE / Update RPSL

Select LIR:



Select Data Set / Create Inetnum

Identify which LIR data you want to use for the netnum update, and select either "Create Inetnum" or "Close" to exit.



For detailed step by step instructions, see:

RIPE LIR Setup and Use

ProVision Developer Tools

Developer Tools

6connect ProVision can integrate with your existing tools and workflow through use of the API and CLI. The 6connect API allows you to access the data and functions of the 6connect web tools to run advanced commands in ProVision, and supports a wide variety of update and deletion conditions not available in the UI.

To use the API, you will need a basic understanding of object oriented programming in PHP and the right tools installed on your system.

Table of Contents

- API v1
- CLI (Alpha)
- Toolkit
- Resource Concepts 1
- Portable Gadgets

API v1

API v1

- 1 - Overview
- 2 - Making API Requests
- 3 - SDK
- API Module - Admin and Audit
- API Module - DHCP
- API Module - DNS
- API Module - IPAM
- API Module - LIR
- API Module - Peering
- API Module - Resource
- API Module - VLAN
- How Do I...
- Reverse API

1 - Overview

6connect API - Overview

The 6Connect API is a RESTful API to access your data in the 6Connect tools. ReST relies on stateless, client-server communication, and is usually always implemented using the HTTP protocol (the 6Connect API uses HTTPS). It is a simple and lightweight alternative to Web Services and can implemented in nearly any language. The 6Connect API operates similarly to other popular ReST APIs you may have worked with, such as Facebook or Twitter. You simply create an HTTP GET or POST request according to our standard, send it to the server, and receive data back.

To learn more about request formatting, making requests, and the tools available, visit Making API Requests.
You can also get the PHP SDK for PHP libraries and sample code.

Here are some important details about our ReST implementation:

- The API only comes with the full 6Connect IPAM product. If you would like to upgrade to the full version, contact sales@6Connect.com.
- All transactions are over HTTPS (SSL - port 443) only. Any transaction not using SSL will be rejected, and you will have potentially exposed sensitive data.
- All API results are formatted in JSON. XML support is coming soon.
- All requests are either HTTP GET or POST requests. We suggest using POST if the length of data in the request is over 8KB.
- You can use any language you would like to query the API. We currently have an SDK for PHP. Looking at the sample code would probably help you implement it in any language though.

2 - Making API Requests

6connect API - Making API Requests

API requests can be generated within the web UI by the API Request Generator, or generated programmatically in any language.

An API request looks like this:
https://cloud.6connect.com/ex/api/v1/api.php?target=ipam&action=get&type=IP&mask=24

An API response is a JSON-encoded text string, and looks like this:


```
{"success":1, "message":"1 blocks found", "data":[{"id":"7539","oct1":"1","oct2":"2","oct3":"3","
oct4":"0","mask":"24","child1":null,"child2":null,"is_assigned":"0","is_swipped":"0", "is_aggregate":"
1","custid":"holding","last_updated_time":"2012-03-20 09:49:00","description":null,"parent":null,"
rir":"ARIN","notes":"2012-03-20 09:49:00","generic_code":null,"region":null,"vlan":null,"arin_net_id":
null,"arin_cust_id":null,"arin_swip_time":"0000-00-00 00:00:00","assigned_time":"2012-03-20 09:45:
12"}]}
```


Instructions on decoding this return data can be found in the API endpoint documentation pages.

Using API Keys:
When using the API without pre-established authentication to ProVision, you must include both your API Key and a specially-prepared query hash parameter, like so:

> https://cloud.6connect.com/ex/api/v1/api.php?target=ipam&action=get&type=IP&mask=24&apiKey=116-MX15LUYY78ZZTW5&hash=8jxj4IApYmgb5IZ0wBY4tFv+WilXb5JuVpjrwupyXQo=

API Keys can be generated from your ProVision instance by navigating to the Admin panel by using the gear icon in the upper right hand corner, then navigating to the API tab. The API tab will present the API authentication information in the following format:

> API Key: 38-TMHQV8CV2XZYC2ZS
>
> Secret Key: 6e04e5822ce90feaa8947ded46c46878

The secret key serves as an API password and is used in the creation of the API Authentication hash. The formula for creating a API query hash from an API query and a Secret Key is the following:

> Hash = Base64Encode(Sha256HMACHash (QueryString, SecretKey))

In PHP, this would be performed with the following line of code:

> $hash = *base64_encode(hash_hmac('sha256', $_SERVER['QUERY_STRING'], $secretKey, TRUE));*

Because the hash function is computed based on the query string, you must calculate a unique hash for every API request!
Example

Lets say you wanted to create a hash for the following API request:

> https://cloud.6connect.com/6c_375/api/v1/api.php?target=ipam&action=get&type=IP&mask=24

And that your API Key and Secret Key are as follows:

> API Key: 32-5DAYTJQY2TZHOFOB
>
> Secret Key: 48b278ec873bda4738923dbc467f8669

The first step is to append your API Key to the URL. The API Key indicates which user is executing the API query.

> https://cloud.6connect.com/6c_375/api/v1/api.php?target=ipam&action=get&type=IP&mask=24&apiKey=32-5DAYTJQY2TZHOFOB

The next step is to isolate the Query String from the request URL. The Query String is everything which follows the question mark. So,

> Query String: target=ipam&action=get&type=IP&mask=24&apiKey=32-5DAYTJQY2TZHOFOB

The next step is to calculate the SHA256 hash of this string with your Secret Key. In PHP, this would be:

> $sha256 = hash_hmac('sha256', "target=ipam&action=get&type=IP&mask=24&apiKey=32-5DAYTJQY2TZHOFOB", "48b278ec873bda4738923dbc467f8669", TRUE);

As this value has been 256-bit hashed, it will contain many unprintable characters. The solution to this is to encode it in base 64 for transport. Again, in PHP:

```
$hash = base64_encode($sha256);
```

Calculating it out yields the completed hash:

```
$hash = yneSFMyxPPe+3W4IOkVp50K3VStatBcRRak+2ygDUWQ=
```

The calculated hash can then be appended to the full API Query URL to form a completed request:

https://cloud.6connect.com/6c_375/api/v1/api.php?target=ipam&action=get&type=IP&mask=24&apiKey=32-5DAYTJQY2TZHOFOB&hash=yneSFMyxPPe+3W4IOkVp50K3VStatBcRRak+2ygDUWQ=

A Note on False Positives

ProVision utilizes several possible authentication schemes of which key-based API authentication is only one. Session-based, username /password authentication is used for the majority of user interaction with the ProVision front end. Because session information is stored in browsers cookies, a browser can be authenticated to execute API commands as long as the session is active.

Unfortunately, this can lead to confusion when using a machine-based API as the user might use an authenticated browser session to test API-Key based API queries. These queries will always succeed regardless of whether the API Query Hash was calculated correctly as the system defaults to Session-based authentication when it is available.

To ensure that session-based authentication is not polluting your API-Key based testing, always use a separate browser which is not logged in to your ProVision instance to test API queries.

Other Languages

The 6Connect API can be used in just about any scripting or programming language. We have a PHP SDK that provides example code, and several useful functions for interacting with the API. Even if you don't want to use PHP, the samples will help you create code in other languages.

3 - SDK

6connect API - Getting Started with the SDK

The 6connect API allows you to access to data and functions of the 6connect web tools. The SDK for PHP or Python will help you get this setup quickly by outlining the requirements, prerequisites and provide sample code.

- 6connect API - Getting Started with the SDK
  - SDK for PHP
    - Prerequisites
    - Requirements
    - Install the SDK
  - SDK for Python
    - Install the SDK
  - Need More Information?

SDK for PHP

Prerequisites

The API only comes with a licensed 6connect ProVision application. If you would like access to a ProVision license please contact sales@6connect.com.

Create Your API Credentials

To use the 6connect SDK for PHP, you will need a 6connect API Key and Secret Key.

To create your API Key and Secret Key:

- Log into your 6connect instance (hosted or local)
- Click on the Admin icon, and go into the Administration section.
- Click on the "API" tab.
- Select the user from the drop down you want to enable API access for, and click "Generate Keys".
- The API Key and the Secret Key will now appear directly below that.

*Note that generating a new API will automatically revoke an older API Key.

6connect recommends that each user accessing the API have their own API key configured. However, you can alternatively setup API users by functionality or roles. While the platform is flexible, you should follow your organizations security policies.

Important!

Your Secret Key is a secret! Only you and 6connect should ever know this information. It is important to keep it confidential to protect the privacy of your data. Store it securely and never share this key with other users or place it on other systems. Never include the secret key in requests to 6connect, support requests to 6connect, and never e-mail it to anyone. Do not share it outside your organization. No one who legitimately represents 6connect will ever ask you for your Secret Key.

Requirements

Aside from following the prerequisites, you will need a basic understanding of object oriented programming in PHP and the right tools installed on your system to use the API.

Minimum Requirements

- PHP 5.5 or newer.
- PHP JSON and PCRE extensions (XML will be coming soon).
- Curl PHP extension compiled with OpenSSL libraries. Click here for more information on curl.

If you aren't sure what is running on your system, you can create a php page on your system and call phpinfo() and view this page in a browser, or run php -i on the command line.

Install the SDK

Download the file 6connect-PHP-SDKv2.tar.gz

Configure the SDK Security Credentials

- Extract the zipped tar file to a directory.
- Open the api-config.php located in the downloaded SDK files.

- Read through the file and place in your instance name (or path for local installs), API Key and Secret Key information as specified.
- Make sure all files are in the same directory (the core class looks for a config file in the same directory by default).
- Run the sample code api-examples.php!

Important!

You must setup user API access before running the sample. See the previous section "Create Your API Credentials" for more information.

SDK for Python

In Progress

This section is in progress. More information will be added as we improve and refine our new Python SDK!

Install the SDK

Download the file 6connect-Python-SDK.zip

Configure the SDK Security Credentials

- Extract the zipped tar file to a directory.
- Open the apiclient.py located in the downloaded SDK files.
- Read through the file and place in your instance name (or path for local installs), API Key and Secret Key information as specified.
- Make sure all files are in the same directory (the core class looks for a config file in the same directory by default).
- Run the sample code api-examples.php!

Important!

You must setup user API access before running the sample. See the previous section "Create Your API Credentials" for more information.

Need More Information?

If you need more general information on the API, try the API Overview.
If you need information on methods available via the API, look at the API Reference.
The SDK also contains a README file with other useful information particular to php.

API Module - Admin and Audit

Admin and Audit

This section covers the functions found under the Admin section of ProVision.

Table of Contents

- Authentication Testing
- Backup
- Log Management
- Scheduler 1
- Zone Templates

- Authentication Testing
  - testSSH
  - testLDAP
  - testSecure64

Authentication Testing

testSSH	
URL	/api/v1/api.php?target=auth&action=testSSH
Description	Returns success or failure of a connection to an external server via SSH.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Success!"} ERROR: *{'success':0, 'message':'error message'}*
Required Parameters	**Name** | **Type** | **Example** | **Description** SSHServer | STRING | totally.awesome.com | IP or FQDN of server. SSHPort | NUMBER | 22 | Port ssh is running on.
Optional Parameters	**Name** | **Type** | **Example** | **Description** username | STRING | jsmith | Username on target server. password | STRING | password123 | Password for user. directory | STRING | /tmp | Directory to attempt to access after successful login.
Example URL	/api/v1/api.php?target=auth&action=testSSH&username=jsmith&password=password123

testLDAP	
URL	/api/v1/api.php?target=auth&action=testLDAP
Description	Test basic connectivity to an LDAP server. Does not test actual authenticaion against server.
Returns	**Examples:** SUCCESSFUL: *{'success':1, 'id':'12345'}* ERROR: *{'success':0, 'message':'unable to add block'}>*
Required Parameters	

Name	Type	Example	Description
ldapServer	STRING	ldap.awesome.com	IP or FQDN of the LDAP server.
ldapPort	NUMBER	389	User-defined block code as defined in Admin-IPAM settings: Generic Code Per Block Name
ldapMode	STRING	SSL	Options are: SSL, TLS, or None.

Optional Parameters	None
Example URL	/api/v1/api.php?target=auth&action=testLDAP&ldapPort=389&ldapServer=ldap.awesome.com&ldapMode=None

testSecure64

URL	/api/v1/api.php?target=auth&action=testSecure64
Description	Returns success or failure of a connection to an Secure64 DNS appliance.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Success!"} ERROR: *{'success':0, 'message':'error message'}*

Required Parameters

Name	Type	Example	Description
SSHServer	STRING	totally.awesome.com	IP or FQDN of server.
SSHPort	NUMBER	22	Port ssh is running on.

Optional Parameters

Name	Type	Example	Description
username	STRING	jsmith	Username on target server.
password	STRING	password123	Password for user.
directory	STRING	/tmp	Directory to attempt to access after successful login.

Example URL	/api/v1/api.php?target=auth&action=testSecure64&username=jsmith&password=passwor

Backup

- Backup
  - backup now

Backup

backup now	
URL	/api/v1/api.php?target=backup&action=now
Description	Performs a manual backup to the designated location.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Backup via curl complete: "} ERROR: *{"success":0, "message":"Error Message"}>*
Required Parameters	**Name** / **Type** / **Example** / **Description** type / STRING / curl / File transfer type. Acceptable values are 'curl' or 'scp'
Optional Parameters	**Name** / **Type** / **Example** / **Description** resource / INTEGER / 2213 / The server resource id to backup to. If left blank, backs up to the 6connect cloud.
Example URL	**Backup to designated server:** /api/v1/api.php?target=backup&action=now&type=scp&resource_id=2213 **Backup to 6connect cloud:** /api/v1/api.php?target=backup&action=now&type=curl

When specifying a backup server resource ID, that server must have the Hostname, Username, and Password fields correctly provided in ProVision.

- Log Management
  - Get

Log Management

Get	
URL	/api/v1/api.php?target=log&action=get
Description	Returns a list of log entries. Use optional parameters to filter the list.
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":" Search Successful.","data":[{"logId":"31568","time": "2012-05-07 17:44:43", "logLevel":" INFO","userId":"39","userName": "user@6connect.com","logCategory": "User","message": "User Doe (user@6connect.com) logged in via local authentication","ip":"107.111.0.228"}]}* ERROR: *{'success':0, 'message':'error message'}* **Data Detail** **Name** / **Type** / **Example** / **Description** log_id / INTEGER / 24 / Unique log entry id. time / DATETIME / 2012-05-07 22:10:07 / Date and time year to second. log_level / STRING / NOTICE / Standard syslog log levels in verbose format (EMERG, ALERT, CRIT, ERR, WARNING, NOTICE, INFO, DEBUG). user_id / INTEGER / 11 / The unique user id associated with the log entry. username / STRING / user@6connect.com / The unique user name associated with the log entry.

log_category	STRING	IPAM	The 6connect category for the log entry (User, IPAM, Resource Holder, DNS, Peering, Assistant, NTP, Reporting).
message	STRING	Created new children from 1.0.0.0/24	The detailed log message.
ip	STRING	107.111.0.228	The remote IP address of the user who took the action being logged.

Required Parameters

None*

*There are no required parameters, but at least one optional parameter must be provided for the call to succeed.

Optional Parameters

Name	Type	Example	Description
block_id	INTEGER	310	Id of the IPAM netblocks to which the logs belong
get_attributes	BOOLEAN	1	Display the log attributes along with the log into the client's response. (The attributes for each log record). Valid values are 1 (true) and 0 (false).
log_id	INTEGER	24	Unique log entry id.
time_min	DATETIME	2015-05-07 [21:00:00]	Retrieve logs starting at this Date and optional time year to second.
time_max	DATETIME	2015-05-07 [22:00:00]	Retrieve logs ending at this Date and optional time year to second.
limit	INTEGER	100	Total log entries to retrieve. Default limit is 1000 records.

orderby	STRING	log_id	Order results by log_id, time, log_level
order	STRING	ASC	Order by ascending / descending (ASC / DESC).
offset	INTEGER	50	Offset from 0 to retrieve log entries
username	STRING	user@6connect.com	The unique user name associated with the log entry.
log_category	STRING	IPAM	The 6connect category for the log entry (User, IPAM, Resource Holder, DNS, Peering, Assistant, NTP, Reporting).
log_level	STRING	NOTICE	Standard syslog log levels in verbose format (EMERG, ALERT, CRIT, ERR, WARNING, NOTICE, INFO, DEBUG).
ip	STRING	1.2.3.4	The remote IP address of the user whose action was logged
search	STRING	Aggregate Added	Search for a string in the logs. It searches in 'message', 'username', 'time', 'ip' and 'log_category'
time	DATETIME	2015-05-07 [21:00:00]	Search logs from a specific time.

Example URL	/api/v1/api.php?target=log&action=get&block_id=310&order_by=log_id&order=DESC

Scheduler 1

This API is in beta and subject to change.

- Scheduler
  - addTask
  - updateTask
  - deleteTask
  - executeTask
  - getTaskHistory
  - getTasks

Scheduler

addTask

URL	/api/v1/api.php?target=scheduler&action=addTask
Description	Add a new scheduled task. Request is POST-only. Params are a single JSON object:
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Message "} ERROR: *{"success":0, "message":"Error Message"}>*
Required Parameters	

Name	Type	Example	Description
user_id	INTEGER	100	User ID
name	STRING	Process Holding Tank	Name given to the new
class_name	STRING	\scheduler\Tasks\ProcessHoldingTask	PHP namespaced name for the task class. Currently available: \scheduler\Tasks\Pro \scheduler\Tasks\Bac \scheduler\Tasks\DN
data		{}	
active	BOOL	true	
repeat_items			

Optional Parameters

Name	Type	Example	Description
repeat_start	STRING	2015-05-06	Date to start repeating task. Date must be in YYYY-MM-DD format.
repeat_end	STRING	null	Date to end repeating task or null. Date must be in YYYY-MM-DD format.

repeat_hour	INTEGER	8	Hour of day to repeat, from 0-23
repeat_minute	INTEGER	0	Minute of that hour to repeat , from 0-59
repeat_day	INTEGER	1	Day of month, from 1-31
repeat_week	INTEGER	1	Week of month, from 1-5
repeat_weekday	INTEGER	1	Days to repeat. Sun-Sat are from 0-6 respectively

updateTask

URL	/api/v1/api.php?target=scheduler&action=updateTask
Description	Update a scheduler task
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"*Message*"} ERROR: *{"success":0, "message":"Error Message"}>*

Required Parameters

Name	Type	Example	Description
id	INTEGER	4	ID of the task to be updated
user_id	INTEGER	100	User ID
name	STRING	Process Holding Tank	Name given to the new
class_name	STRING	\scheduler\Tasks\ProcessHoldingTask	namespaced name for the task class. Currently available: \scheduler\Tasks\Pro \scheduler\Tasks\Bac \scheduler\Tasks\DN
data		{}	
active	BOOL	true	
repeat_items			

Optional Parameters

Name	Type	Example	Description
repeat_start	STRING	2015-05-06	Date to start repeating task. Date

			must be in YYYY-MM-DD format.
repeat_end	STRING	null	Date to end repeating task or null. Date must be in YYYY-MM-DD format.
repeat_hour	INTEGER	8	Hour of day to repeat, from 0-23
repeat_minute	INTEGER	0	Minute of that hour to repeat, from 0-59
repeat_day	INTEGER	1	Day of month, from 1-31
repeat_week	INTEGER	1	Week of month, from 1-5
repeat_weekday	INTEGER	1	Days to repeat. Sun-Sat are from 0-6 respectively

deleteTask

URL	/api/v1/api.php?target=scheduler&action=deleteTask
Description	Delete a scheduler task
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Task \"New Task\" (2) deleted","data":null} ERROR: *{"success":0, "message":"Error Message"}>*
Required Parameters	**Name** / **Type** / **Example** / **Description** taskId / INTEGER / 4 / ID of the task to delete
Optional Parameters	None
Example URL	/api/v1/api.php?target=scheduler&action=deleteTask&taskId=2

executeTask

URL	/api/v1/api.php?target=scheduler&action=executeTask
Description	Execute a task immediately
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Holding tank processed. 0 IPv4 and 0 IPv6 blocks moved to the available pool. ","data":null} ERROR: *{"success":0, "message":"Error Message"}>*

Required Parameters	<table><tr><th>Name</th><th>Type</th><th>Example</th><th>Description</th></tr><tr><td>taskId</td><td>INTEGER</td><td>4</td><td>ID of the task to run</td></tr></table>
Optional Parameters	None
Example URL	/api/v1/api.php?target=scheduler&action=executeTask&taskId=1

getTaskHistory

URL	/api/v1/api.php?target=scheduler&action=getTaskHistory
Description	Get history for a specific task
Returns	**Examples:** SUCCESSFUL: { "success": 1, "data": [{ "log_id": "1005037", "time": "2015-05-07 12:29:45", "log_level": "6", "user_id": "31", "username": "usernamehere", "log_category": "System", "message": "task_id=1 task=Process Holding Tank action=finished message=Holding tank processed. 0 IPv4 and 0 IPv6 blocks moved to the available pool.", "ip": null }, { "log_id": "1005033", "time": "2015-05-07 12:28:50", "log_level": "6", "user_id": "31", "username": "usernamehere", "log_category": "System", "message": "task_id=1 task=Process Holding Tank action=finished message=Holding tank processed. 1 IPv4 and 0 IPv6 blocks moved to the available pool.", "ip": null }, { "log_id": "1005030", "time": "2015-05-07 12:28:14", "log_level": "6", "user_id": "31", "username": "usernamehere", "log_category": "System", "message": "Task \"Process Holding Tank\" (1) added", "ip": null }] } ERROR: *{"success":0, "message":"Error Message"}>*
Required Parameters	<table><tr><th>Name</th><th>Type</th><th>Example</th><th>Description</th></tr><tr><td>taskId</td><td>INTEGER</td><td>1</td><td>ID of the task to view</td></tr></table>
Optional Parameters	None
Example URL	/api/v1/api.php?target=scheduler&action=getTaskHistory&taskId=1

getTasks

URL	/api/v1/api.php?target=scheduler&action=getTasks
Description	Gets a list of scheduled tasks and their repeat settings
Returns	**Examples:** SUCCESSFUL: { "success": 1, "data": [{ "id": "1", "user_id": "31", "name": "Process Holding Tank", "class_name": "\\scheduler\\Tasks\\ProcessHoldingTask", "active": true, "data": null, "last_run": "2015-05-07 12:29:45", "repeat_items": [{ "id": "1", "task_id": "1", "repeat_start": "2015-05-06", "repeat_end": null, "repeat_interval": null, "repeat_year": "*", "repeat_month": "*", "repeat_day": "*", "repeat_week": "*", "repeat_weekday": "4", "repeat_hour": "20", "repeat_minute": "0" }, { "id": "2", "task_id": "1", "repeat_start": "2015-05-06", "repeat_end": null, "repeat_interval": null, "repeat_year": "*", "repeat_month": "*", "repeat_day": "*", "repeat_week": "*", "repeat_weekday": "5", "repeat_hour": "20", "repeat_minute": "0" }, { "id": "3", "task_id": "1", "repeat_start": "2015-05-06", "repeat_end": null, "repeat_interval": null, "repeat_year": "*",

	"repeat_month": "*", "repeat_day": "*", "repeat_week": "*", "repeat_weekday": "6", "repeat_hour": "20", "repeat_minute": "0" }, { "id": "4", "task_id": "1", "repeat_start": "2015-05-06", "repeat_end": null, "repeat_interval": null, "repeat_year": "*", "repeat_month": "*", "repeat_day": "*", "repeat_week": "*", "repeat_weekday": "0", "repeat_hour": "20", "repeat_minute": "0" }] }, { "id": "2", "user_id": "31", "name": "New Task", "class_name": "\\scheduler\\Tasks\\TestTask", "active": true, "data": null, "last_run": "2015-05-07 12:35:41", "repeat_items": [{ "id": "5", "task_id": "2", "repeat_start": "2015-05-06", "repeat_end": null, "repeat_interval": null, "repeat_year": "*", "repeat_month": "*", "repeat_day": "*", "repeat_week": "*", "repeat_weekday": "1", "repeat_hour": "21", "repeat_minute": "0" }, { "id": "6", "task_id": "2", "repeat_start": "2015-05-06", "repeat_end": null, "repeat_interval": null, "repeat_year": "*", "repeat_month": "*", "repeat_day": "*", "repeat_week": "*", "repeat_weekday": "2", "repeat_hour": "21", "repeat_minute": "0" }, { "id": "7", "task_id": "2", "repeat_start": "2015-05-06", "repeat_end": null, "repeat_interval": null, "repeat_year": "*", "repeat_month": "*", "repeat_day": "*", "repeat_week": "*", "repeat_weekday": "3", "repeat_hour": "21", "repeat_minute": "0" }, { "id": "8", "task_id": "2", "repeat_start": "2015-05-06", "repeat_end": null, "repeat_interval": null, "repeat_year": "*", "repeat_month": "*", "repeat_day": "*", "repeat_week": "*", "repeat_weekday": "4", "repeat_hour": "21", "repeat_minute": "0" }, { "id": "9", "task_id": "2", "repeat_start": "2015-05-06", "repeat_end": null, "repeat_interval": null, "repeat_year": "*", "repeat_month": "*", "repeat_day": "*", "repeat_week": "*", "repeat_weekday": "5", "repeat_hour": "21", "repeat_minute": "0" }] }] } ERROR: *{"success":0, "message":"Error Message"}>*
Required Parameters	None
Optional Parameters	None
Example URL	/api/v1/api.php?target=scheduler&action=getTasks

Zone Templates

- Zone Templates
  - Get
  - Update
  - Delete

Zone Templates

Get	
URL	/api/v1/api.php?target=zoneTemplate&action=get
Description	Returns success or failure of a connection to an external server via SSH.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Found 1 records for template \" Awesome Template\".","data": {"templateId":"1011","name":" Awesome Template"," created":"2013-07-31 14:01: 24","modified":"2013-07-31 14: 01:24","userId":"112","soa": null,"refresh":"14400","retry":" 3600","expire":"604800"," minimum":null,"ttl":null," userName":"joe@smith.com"," records":[{"templateRecordId":" 4","templateId":"1011","host":" www","type":"A","ttl":"3600"," value":"1.2.3.4","ordering":" 0"}]}} ERROR: *{'success':0, 'message':'error message'}*
Required Parameters	None
Optional Parameters	**Name** / **Type** / **Example** / **Description** templateId / NUMBER / 3 / ID of the specific template to get.
Example URL	/api/v1/api.php?target=zoneTemplate&action=get

Update	
URL	/api/v1/api.php?target=zoneTemplate&action=update
Description	Create a new template or update an existing template.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Template updated","data": {"templateId":"1011","name":"Awesome Template","created":"2013-08-05 23:15:52","modified":"2013-08-05 23:15:52","userId":"112","

soa":"ns1.test.net hostmaster.ns1.test.net","refresh":"14400","retry":"3600","expire":"604800","minimum":null,"ttl":false,"userName":"anna@6connect.com","records":"1"}}
ERROR: *{'success':0, 'message':'Error updating template: error details'}>*

Required Parameters

Name	Type	Example	Description
name	STRING	Test Template	The name of the template to be created or updated.

Optional Parameters

Name	Type	Example	Description
soa	STRING	ns1.test.net hostmaster. ns1.test.net	A valid SOA for the template in for format
ttl	INTEGER	86400	The TTL for the zone template, which is the default expiration time for all records without their own TTL.
refresh	INTEGER	14400	The time period for slaves to refresh the zone.
retry	INTEGER	3600	Time that a slave should retry refreshing the zone in case of incident.
expire	INTEGER	604800	Time for a slave to expire a zone.
mininum	INTEGER	3600	The maximum caching time in the event of failed lookups.
count_records	INTEGER	5	Number of host records submitted with the update. All the following parameters names should be followed with their position in the count. In this example, the first record would have all the parameters

				for the first record followed by _1, the second record _2, and so on. This will be the order all records in the template follow.
	host_1	STRING		The DNS record value.
	ttl_1	INTEGER	3600	TTL of the specific host record.
	type_1	STRING	A	A valid DNS record type.
	value_1	IP	1.2.3.4	A valid IPv4 or IPv6 address.

Example URL	api/v1/api.php? target=zoneTemplate&action=update&templateId=1011&count_records=1 &refresh=14400&retry=3600&expire=604800&minimum=3600&value_0=u

Delete

URL	/api/v1/api.php?target=zoneTemplate&action=delete
Description	Deletes a DNS template.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Template \"Test Template\" delete."} ERROR: {"success":0,"message":"No template found for templateId \"1005\"."}
Required Parameters	**Name** / **Type** / **Example** / **Description** templateId / INTEGER / 3 / ID of the template to delete.
Optional Parameters	None.
Example URL	/api/v1/api.php? target=zoneTemplate&action=delete&templateId=1005

API Module - DHCP

DHCP Management Version 2

- DHCP Management Version 2
  - Overview
  - API Updates
  - DHCP API How-To
    - Relate with Resources
    - Create DHCP IP Aggregates
    - Subnets and Hosts
    - Linking Subnets and Hosts with DHCP Servers
    - Pushing Configurations
  - Detailed API Specification

Overview

DHCP Management Version 2 integrates DHCP management with ProVision's resource and permissions hierarchy, as well as the IP Management system. Individual DHCP servers can be assigned via Resource Permissions to different internal user groups, to be managed by only the appropriate parties.

Under DHCPv2 there is no distinct “DHCP Server” type or section – instead there is a “DHCP Module” which, when attached as a child to an existing resource, transforms it into a DHCP-enabled device. The most common use is to take the generic “Server” Section and turn it into a DHCP Server by attaching the DHCP Module as a child. This configuration allows users to add functionality to a basic resource and provides a cleaner management interface.

API Updates

The DHCPv1 API operated via calls to the DHCPServer and the DHCPEntry endpoint families. However, now that DHCPv2 is contained entirely within the resource system, most of the API calls to manipulate DHCP data do so using the Resource family of API endpoints to modify specific Resource attributes reserved for DHCP functionality.

DHCP API How-To

Relate with Resources

The DHCPv2 system builds upon the ProVision Resource API. With the exception of a few configuration commands all DHCPv2 API commands use the Resource family of API endpoints.

How to attach the DHCP Module as a child

As described above, DHCPv2 functionality is enabled on a particular resource by attaching a DHCP Module as a child. A command to do this is as follows:

```
            [ProVision root]/api/v1/api.php?target=resource&action=add

            data:
meta[type]: dhcp_module
meta[name]: [parent resource id] DHCP Module
meta[parent_id]: [parent resource id]
```

The special resource type “dhcp_module” indicates to ProVision that the DHCP system is enabled for the parent object. The attributes associated with the “dhcp_module” resource govern the DHCP system's behavior.

Updating the attributes of a DHCP Server uses a Resource Update command:

```
[ProVision root]/api/v1/api.php?target=resource&action=update&meta[id]=2178 &meta[type]
=dhcp_module&fields[_dhcp_attributes][]={"type":"ISC","notes":"notes go here","username":"
username","port":"port","config_test":"/etc/init.d/dhcpd configtest","server_stop":"/etc/init.d
```

```
/dhcpd stop","server_start":"/etc/init.d/dhcpd start","config_path":"/tmp/dhcpd.conf","
option_routers":"192.168.0.0","option_domain_name_servers":"ns1.6connect.com","
option_domain_name":"6connect.com","authoritative":"1","default_lease_time":"600","
max_lease_time":"7200","local_port":"67","log_facility":"local7","password":"password","
server_ip":"192.168.0.1","freeLines":3,"freeLine1":"free line 1","freeLine2":"free line 2","
freeLine3":"free line 3"}
```

This command appears rather complicated, but can be broken apart into reasonable pieces. The first section:

```
target=resource&action=update&meta[id]=2178&meta[type]=dhcp_module
```

is familiar from other parts of ProVision. We are updating a resource of type “dhcp_module” whose resource id is 2178. The second section of the command details the update values, starting with

```
 fields[_dhcp_attributes][]=
```

which contains a JSON-encoded string of all the fields specific to a DHCP server's function. When expanded into its full object form it is substantially easier to digest:

```
{
            "type":"ISC",
            "notes":"notes go here",
            "username":"username",
            "port":"port",
            "config_test":"/etc/init.d/dhcpd configtest",
            "server_stop":"/etc/init.d/dhcpd stop",
            "server_start":"/etc/init.d/dhcpd start",
            "config_path":"/tmp/dhcpd.conf",
            "option_routers":"192.168.0.0",
            "option_domain_name_servers":"ns1.6connect.com",
            "option_domain_name":"6connect.com",
            "authoritative":"1",
            "default_lease_time":"600",
            "max_lease_time":"7200",
            "local_port":"67",
            "log_facility":"local7",
            "password":"password",
            "server_ip":"192.168.0.1",
            "freeLines":3,
            "freeLine1":"free line 1",
            "freeLine2":"free line 2",
            "freeLine3":"free line 3"
}
```

This object describes all the most common DHCP server configuration options. For a full explanation of each of the fields, see the Detailed API Specification later in this document.

Please note that the object above must be passed to the DHCP system as a JSON-encoded string. It must be passed into the special “_dhcp_attributes” attribute for it to be functional, as in the example URL.

Create DHCP IP Aggregates

For details on how to manage IP aggregates using ProVision's IPAM API, see API Module - IPAM.

Of particular interest to DHCP management is the addition of DHCP aggregates, which are sections of IP space marked as available for use by the DHCPv2 system.

How to add a DHCP Aggregate

An example command to add a DHCP Aggregate is:

```
[ProVision root]/api/v1/api.php?target=ipam&action=add&block=192.168.0.0/24&rir=
1918&vlan=&tags=&region=&resourceId=1282&allowSubAssignments=true
```

The important part to note is that the IP block is being assigned to resourceId 1282, which corresponds to the DHCP Available resource. The DHCP Available resource is a system-level resource which is used to hold all unassigned DHCP IP addresses. Every instance has its own DHCP Available resource, whose id can be found with the following command:

```
[ProVision root]/api/v1/api.php?target=resource&action=get&slug=dhcp-available
```

New DHCP subnets and hosts draw their IPs from this pool. If there are no IPs in the DHCP Available pool new subnets and hosts will not be able to be created.

DHCP IP aggregates are fetched, updated, split, and deleted using the standard IPAM management API endpoints. Please see the IPAM API Documentation. for details.

Subnets and Hosts

Every DHCP configuration file consists primarily of Subnet and Host declarations, mapping out what IP addresses are available for what purpose. In DHCPv2, DHCP Pools are reusable components that can be attached to several DHCP Servers in order to build flexible, responsive DHCP configurations.

In ProVision DHCPv2 all DHCP Pools regardless of whether they span Subnets or individual Hosts require that a “dhcp_pool” resource be created to govern them.

How to create DHCP Pools

Similar to how the “dhcp_module” resource was created above, the command to create a DHCP Pool is as follows:

```
[ProVision root]/api/v1/api.php?target=resource&action=add&meta[type]=dhcp_pool &meta[name]=New
Subnet&fields[_dhcp_type][]=subnet&fields[_dhcp_pool_attributes][]={"mac":"","rangeStart":"","
rangeEnd":"","freeLines":3,"freeLine1":"Free Line 1","freeLine2":"Free Line 2","freeLine3":"Free
Line 3"}
```

The first half of this command is relatively straightforward:

```
target=resource&action=add&meta[type]=dhcp_pool&meta[name]=New Subnet
```

This section informs the API that we wish to create a new, empty “dhcp_pool” resource whose name is “New Subnet.”

```
fields[_dhcp_type][]=subnet&fields[_dhcp_pool_attributes][]={"mac":"","rangeStart":"",
"rangeEnd":"","freeLines":3,"freeLine1":"Free Line 1","freeLine2":"Free Line 2","freeLine3":"Free
Line 3"}
```

The second half of the command behaves in a similar manner to the “dhcp_module.” The “_dhcp_pool_attributes” field holds a JSON-encoded string which describes the dhcp_pool resource. When expanded, the JSON string becomes the following object:

```
{
            "mac":"",
            "rangeStart":"",
            "rangeEnd":"",
            "freeLines":3,
            "freeLine1":"Free Line 1",
            "freeLine2":"Free Line 2",
            "freeLine3":"Free Line 3"
}
```

For a full explanation of each of the fields, see the Detailed API Specification.

Please note that the object above must be passed to the DHCP system as a JSON-encoded string. It must be passed into the “_dhcp_pool_attributes” attribute for it to be functional, as in the example URL.

Once a dhcp_pool resource is in the system it can be updated with IP data obtained from the IP Management system. Under DHCPv2, the DHCP system uses all the standard IPAM API endpoints and can make use of both the smartAssign and the directAssign methods. Please see the IPAM API documentation for details.

How to smart-assign a DHCP IP range from the DHCP Available resource to a dhcp_pool resource

An example command for smart-assigning a DHCP IP range from the DHCP Available resource to a newly-created dhcp_pool resource is as follows:

```
[ProVision root]/api/v1/api.php?target=ipam&action=smartAssign&resourceId=2180&
type=ipv4&mask=31&rir=1918&assignedResourceId=1282
```

In this example we are using the IPAM API endpoint to smart-assign an IPv4 /31 from the DHCP Available resource (resource id 1282) to the newly-created dhcp_pool object (resource id 2180). This action removes this IP range from the available pool and prevents it from being used by other parts of ProVision.

Once an IP block is assigned to a dhcp_pool it should be updated with the proper range start and range end. A Resource Update command is used for this.

```
[ProVision root]/api/v1/api.php?target=resource&action=update&meta[type]=dhcp_pool& meta[name]
=Another Test&fields[_dhcp_type][]=subnet&fields[_dhcp_pool_attributes][]={"mac":"","rangeStart":"
10.10.10.4","rangeEnd":"10.10.10.5","freeLines":3,"freeLine1":"example1","freeLine2":"example2","
freeLine3":"example3"}&fields[_dhcp_ip_id][]=92430&meta[id]=2180
```

The key information here is that the “rangeStart” and the “rangeEnd” fields in the JSON-encoded '_dhcp_pool_attributes' attribute have been populated with the beginning and end of the IP range assigned by smart-assign. Also note that a new field is being populated as '_dhcp_ip_id', which contains the IPAM id of the newly-assigned IP block.

When assigning dhcp_pools covering a single host the steps are much the same, but the 'mac' field in the '_dhcp_pool_attributes' object must be populated with the MAC address of the host in question.

Linking Subnets and Hosts with DHCP Servers

DHCP Pools exist as re-usable components which can be individually assigned to any number of DHCP Servers in order to assemble flexible DHCP Configurations. Once created, a DHCP Pool is not attached to any DHCP Server in the system. DHCP Pools must be linked to a server for the pool to be included in DHCP configuration pushes.

How to link a dhcp_pool and a DHCP Server

An example of building a link between a dhcp_pool and a DHCP Server is:

```
[ProVision root]/api/v1/api.php?target=resource&action=addLink&resource_id1=2178&
resource_id2=1452&relation=dhcpPoolLink
```

The Resource Linkage system controls which DHCP Pools are associated with a given DHCP Server. In the case of linking a DHCP Pool to a DHCP Server, the relation used is “dhcpPoolLink”. This is a directional link, so it is important that resource_id1 and resource_id2 do not get confused.

```
relation:  "dhcpPoolLink"
resource_id1:  the id of the dhcp_module this pool is being linked to
resource_id2:  the id of the dhcp_pool being linked
```

It is very important that resource_id1 not be confused with resource_id2. The link will not function with the values reversed.

To undo the above and break a DHCP Pool link, use the same command but substitute “deleteLink” for the action “addLink”.

```
[ProVision root]/api/v1/api.php?target=resource&action=deleteLink&resource_id1=2178&
resource_id2=2179&relation=dhcpPoolLink
```

Pushing Configurations

Pushing configuration files and restarting a DHCP server is a fairly straightforward process.

How to push configuration files

Once the server has been configured according to the previous sections, hitting the following API endpoint will trigger a DHCP push:

```
[ProVision root]/api/v1/api.php?target=dhcp&action=push&id=2178
```

The “id” in the above string is the id of the dhcp_module resource attached to the server you whose configuration is to be pushed. The API return payload will contain success or failure codes, as well as a description of any errors which might have occurred.

When a DHCP configuration file is pushed an SSH connection is opened to the configured server using the user, password, and port supplied to the '_dhcp_attributes' attribute on the dhcp_module resource. If the system successfully connects, it will assemble a DHCP configuration from the information given to the dhcp_module's '_dhcp_attribute' attribute and then parse and add in all linked dhcp_pool resources.

After the assembled file has been transferred to the DHCP server it will be placed in the location given by 'config_path' on the dhcp_module, and then the command described in 'config_test' will be run to determine whether or not this new file parses correctly. If 'config_test' is blank or omitted, this step is skipped.

If the file parses correctly the DHCP will be stopped and restarted according to the 'server_stop' and 'server_start' commands on the DHCP module. If there are errors at any point the system backs out, replaces old config files, and reports the errors via the 'message' return field of the API call.

Detailed API Specification

A detailed listing of API endpoints related to DHCP Servers, Pools, and Links can be found here:

- API Module - DHCPv2

API Module - DHCPv2

- DHCPv2 Module
  - get all DHCP-enabled resources
  - create a new DHCP-enabled resource
  - update a DHCP-enabled resource with new configuration info
  - remove DHCP functionality from a resource
  - get all DHCP Pools
  - create a new DHCP Pool resource
  - update a DHCP Pool
  - delete a DHCP Pool
  - assigning an IP address or blocks to a DHCP Pool
  - get all DHCP Pool linkages
  - add a new DHCP Pool linkage
  - delete DHCP Pool linkages
  - push a DHCP config
  - DHCP search
- Data Attributes
  - _dhcp_attributes
  - _dhcp_pool_attributes

DHCPv2 Module

The DHCPv2 system is built upon the Resource API, so actions relating to DHCP tasks are largely expressed in terms of Resource actions.

This section describes common DHCP tasks and how they are accomplished via the DHCPv2 system.

get all DHCP-enabled resources	
Description	Finds all resources from section 'dhcp_module,' which indicates that their parents are DHCP-enabled. Adding in other Resource-Get API parameters can filter this list further.
URL	/api/v1/api.php?target=resource&action=get&type=dhcp_module
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Search successful","data": [{"id":"1432", "name":"1392 DHCP Module" ,"slug":"1392-dhcp-module", "type": "dhcp_module", "parent_id":" 1392", "category_id":null, "attr": {"_dhcp_attributes":"{\"type\":\" ISC\", \"notes\":\"\", \" username\":\"\", \"port\":\"\", \" config_test\":\"\/etc\/init.d\ /dhcpd configtest\", \" server_stop\":\"\/etc\/init.d\ /dhcpd stop\", \"server_start\":\" \/etc\/init.d\/dhcpd start\", \" config_path\":\"\", \" option_routers\":\"\", \" option_domain_name_servers\":\"\", \" \"1\", \"default_lease_time\":\" 600\", \"max_lease_time\":\" 7200\", \"local_port\":\"67\", \" log_facility\":\"local7\", \" password\":\"\", \"server_ip\":\" 10.0.0.0\", \"freeLines\":0}", "_dhcp_config_id":"33"}}], "result_count":1, "found_count":1} ERROR: {"success":0, "message":"error

	message"}

Return Detail:

Name	Type	Description
id	INTEGER	ID of the dhcp_module resource
name	STRING	The name of the dhcp_module
slug	STRING	The unique reference string for this resource
type	STRING	Always 'dhcp_module
parent_id	INTEGER	The resource to which the dhcp_module is attached
category_id	INTEGER	The category to which this dhcp_module is associated
result_count	INTEGER	How many dhcp_modules are returned in this search.
found_count	INTEGER	How many dhcp_modules were found in this query, without pagination.

Attributes:

Key	Type	Description
_dhcp_attributes	JSON	A JSON-encoded string containing all the specific configuration parameters which govern this DHCP server. An expansion of the JSON object is given below in the Data Attributes section.
_dhcp_config_id	INTEGER	A reference to the DHCP Config file written within the system. This field is maintained by the DHCPv2 system itself and should not be set externally.

create a new DHCP-enabled resource

Description	A resource becomes a DHCP-enabled by adding a special "dhcp_module" resource as a child. This action is identical to a normal Resource Create command.
URL	/api/v1/api.php?target=resource&action=add&meta[type]=dhcp_module&meta[name]=2163 DHCP Module&meta[parent_id]=2163
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Resource added","data":{"id": 2165,"name" :"2163 DHCP Module","slug": "2163-dhcp-module-2","type" :" dhcp_module","parent_id": 2163,"category_id": null,"attr": []}} ERROR: {"success":0, "message":"error message"} **Return Detail:**

Name	Type	Description
id	INTEGER	ID of the newly created dhcp_module
name	STRING	The name of the dhcp_module
slug	STRING	The unique reference string for this resource
type	STRING	Always 'dhcp_module'
parent_id	INTEGER	The resource to which the dhcp_module is attached
category_id	INTEGER	The category to which this dhcp_module is associated

update a DHCP-enabled resource with new configuration info

Description	Modifying an existing dhcp_module uses the identical commands as all other Resource-Update actions. An example of configuring a DHCP server is given below.
URL	/api/v1/api.php?target=resource&action=update&meta[id]=2178&meta[type]=dhcp_module&fields[_dhcp_attributes][]={"type":"ISC","notes":"notes go here","username":"username","port":"port","config_test":"/etc/init.d/dhcpd configtest","server_stop":"/etc/init.d/dhcpd stop","server_start":"/etc/init.d/dhcpd start","config_path":"/tmp/dhcpd.conf","option_routers":"192.168.0.0","option_domain_name_servers":"ns1.6connect.com","option_domain_name":"6connect.com","authoritative":"1","default_lease_time":"600","max_lease_time":"7200","local_port":"67","log_facility":"local7","password":"password","server_ip":"192.168.0.1","freeLines":3,"freeLine1":"free line 1","freeLine2":"free line 2","freeLine3":"free line 3"}
Returns	**Examples:**

SUCCESSFUL:	{"success": 1,"message": "Resource Updated","data": {"id":"2166" ,"name":"2163 DHCP Module", "slug":"2163-dhcp-module-3" ,"type":" dhcp_module", "parent_id":" 2163", "category_id":null ,"attr": {"_dhcp_attributes": "{\"type\":\" ISC\", \"notes\":\"notes go here\", \"username\":\" username\", \"port\":\"port\", \" config_test\":\"\\\/etc\\\/init.d\\\ /dhcpd configtest\" ,\" server_stop\":\"\\\/etc\\\/init.d\\\ /dhcpd stop\" ,\"server_start\":\" \\\/etc\\\/init.d\\\/dhcpd start\",\" config_path\":\"\\\/tmp\\\/dhcpd. conf\", \"option_routers\":\" 192.168.0.0\", \" option_domain_name_servers\":\"ns1.
ERROR:	{"success":0, "message":"error message"}

Return Detail:

Name	Type	Description
id	INTEGER	ID of the newly created dhcp_module
name	STRING	The name of the dhcp_module
slug	STRING	The unique reference string for this resource
type	STRING	Always 'dhcp_module'
parent_id	INTEGER	The resource to which the dhcp_module is attached
category_id	INTEGER	The category to which this dhcp_module is associated

Attributes:

Key	Type	Description
_dhcp_attributes	JSON	A JSON-encoded string containing all the specific configuration parameters which govern this DHCP server. An expansion of the JSON object is given below in the Data Attributes section.

remove DHCP functionality from a resource

Description	To remove DHCP functionality, delete the dhcp_module child resource. This operation uses general Resource->Delete functionality.

URL	/api/v1/api.php?target=resource&action=delete&id=2166
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Deleted 2163-dhcp-module-3 (#2166)"} ERROR: {"success":0, "message":"error message"}

get all DHCP Pools

Description	As with the dhcp_module commands, the API endpoints governing DHCP IP Pools use the general Resource system. All the modifiers that can be applied to a Resource-Get can be used to filter this query.
URL	/api/v1/api.php?target=resource&action=get&type=dhcp_pool
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Search successful","data": [{"id":"1482","name":"Blah","slug":"blah","type":"dhcp_pool","parent_id":"1","category_id":null,"attr":{"_dhcp_type":"subnet","_dhcp_pool_attributes":"{\"mac\":\"\",\"rangeStart\":\"13.0.0.0\",\"rangeEnd\":\"13.0.0.255\",\"freeLines\":0}","_dhcp_ip_id":"80902"}}],"result_count":1,"found_count":1} ERROR: {"success":0, "message":"error message"} **Return Detail:** **Name** / **Type** / **Description** id / INTEGER / ID of the dhcp_pool resource name / STRING / The name of the dhcp_pool slug / STRING / The unique reference string for this resource type / STRING / Always 'dhcp_pool' parent_id / INTEGER / The resource to which the dhcp_pool is attached category_id / INTEGER / The category to which this dhcp_pool is associated result_count / INTEGER / How many

		dhcp_pools are returned in this search.
found_count	INTEGER	How many dhcp_pools were found in this query, without pagination.

Attributes:

Key	Type	Description
_dhcp_type	STRING	Either 'subnet' or 'host'. Determines whether this DHCP Pool is describing a Subnet or a Host.
_dhcp_pool_attributes	JSON	A JSON-encoded string containing all the specific configuration parameters which govern this DHCP Pool. An expansion of the JSON object is given below in the Data Attributes section.
_dhcp_ip_id	INTEGER	The id of the IPAM subnet or host which is assigned to this DHCP Pool

create a new DHCP Pool resource

Description	Uses the general Resource-Add endpoint to create a DHCP Pool resource.
URL	/api/v1/api.php?target=resource& action=add& meta[type] =dhcp_pool& meta[name]=New Subnet& fields[_dhcp_type][]=host& fields[_dhcp_pool_attributes][]={"mac":"aa:bb:cc:dd:ee:ff", "rangeStart":"", "rangeEnd":"", "freeLines":3, "freeLine1":"Free Line 1", "freeLine2":"Free Line 2", "freeLine3":"Free Line 3"}
Returns	**Examples:**

SUCCESSFUL:	{"success":1,"message":" Resource added","data":{"id": 2167,"name":"New Subnet"," slug":"new-subnet","type":" dhcp_pool ","parent_id":1," category_id":null,"attr":[]}}
ERROR:	{"success":0, "message":"error message"}

Return Detail:

Name	Type	Description
id	INTEGER	ID of the newly created dhcp_pool

name	STRING	The name of the dhcp_pool
slug	STRING	The unique reference string for this resource
type	STRING	Always 'dhcp_pool'
parent_id	INTEGER	The parent resource; by default the TLR.
category_id	INTEGER	The category to which this dhcp_pool is associated

update a DHCP Pool

Description	Modifying an existing dhcp_pool uses the identical commands as all other Resource-Update actions.
URL	/api/v1/api.php?target=resource& action=update& meta[type] =dhcp_pool& meta[name]=Another Test& fields[_dhcp_type][] =subnet& fields[_dhcp_pool_attributes][]={"mac":"", "rangeStart":" 10.10.10.4", "rangeEnd":"10.10.10.5", "freeLines":3, "freeLine1":" example1", "freeLine2":"example2", "freeLine3":"example3"}&fields [_dhcp_ip_id][]=92430&meta[id]=2165
Returns	**Examples:**

SUCCESSFUL:	{"success":1, "message":" Resource Updated", "data": {"id":"2165", "name":"Another Test", "slug":"2163-dhcp-module-2", "type":" dhcp_module", "parent_id":" 2163", "category_id":null,"attr": {"_dhcp_type":"subnet" ," _dhcp_pool_attributes":"{\" mac\":\"\", \"rangeStart\":\" 10.10.10.4\", \"rangeEnd\":\" 10.10.10.5\", \"freeLines\":3, \" freeLine1\":\"example1\", \" freeLine2\":\"example2\", \" freeLine3\":\"example3\"}", "_dhcp_ip_id":"92430"}}}
ERROR:	{"success":0, "message":"error message"}

Return Detail:

Name	Type	Description
id	INTEGER	ID of the newly created dhcp_module
name	STRING	The name of the dhcp_module
slug	STRING	The unique reference string for this resource
type	STRING	Always 'dhcp_module'
parent_id	INTEGER	The resource to which the dhcp_module is attached
category_id	INTEGER	The category to which this dhcp_module is associated

Attributes:

Key	Type	Description
_dhcp_type	STRING	Either 'subnet' or 'host'. Determines whether this DHCP Pool is describing a Subnet or a Host.
_dhcp_pool_attributes	JSON	A JSON-encoded string containing all the specific configuration parameters which govern this DHCP Pool. An expansion of the JSON object is given below in the Data Attributes section.
_dhcp_ip_id	INTEGER	The id of the IPAM subnet or host which is assigned to this DHCP Pool

delete a DHCP Pool

Description	To delete a DHCP Pool, use the standard Resource-Delete functionality
URL	/api/v1/api.php?target=resource&action=delete&id=2165
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Deleted 2165-another-subnet-3 (#2165)"} ERROR: {"success":0, "message":"error message"}

assigning an IP address or blocks to a DHCP Pool

Description	Assigning IP addresses or blocks to a DHCP Pool resource removes them from the available pool so they cannot be assigned out again. This procedure uses all the standard IPAM assignment functions, so long as the resource assigned **from** is the DHCP Available resource.
URL	/api/v1/api.php?target=ipam&action=smartAssign&resourceId=2162&type=ipv4&mask=31
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Assigned 10.8.1.4\/31 to

	208.39.104.106 (2162) via Smart Assign","id":94468," data":{"id":94468,"type":"ipv4," top_aggregate":44616,"cidr":" 10.8.1.4\/31","formatted_ip":" 10.8.1.4\/31","address":" 168296708","end_address":" 168296709","mask":31," netmask":"255.255.255.254"," child1":null,"child2":null," is_assigned":1,"is_swipped":0," is_aggregate":1,"custid":2162," resource_id":2162," resource_name":null," last_updated_time":"2015-03-15 20:17:32","description": null,"parent":80882,"rir":" 1918","lir_id":null,"notes":null," generic_code":null,"code":null," region":"Quito"," region_name":"Quito","vlan": null,"arin_net_id":null," arin_cust_id":null,"org_id":null," arin_swip_time":null," assigned_time":"2015-03-15 20:17:32","asn":null," allowSubAssignments":false," permissions":[],"range":" 10.8.1.4 - 10.8.1.5","tags": ["DHCP"]}}
ERROR:	{"success":0, "message":"error message"}

Return Detail:

For a detailed breakdown of this endpoint's return data, please see the IPAM documentation.

get all DHCP Pool linkages

Description	The association between DHCP Pools and DHCP Modules belongs to the Resource Linkage family of endpoints. The 'relation' field should be set to the 'dhcpPoolLink' type to pull only DHCP Pool linkage information.
URL	/api/v1/api.php? target=resource&action=getLink&relation=dhcpPoolLink
Returns	**Examples:**

SUCCESSFUL:	{"success":1 ,"message":" Search successful", "data": {"meta":{"totalRecords":"3", "retrieved":3}, "0":{"id":"22", "resource_id1":"1292", "resource_id2":"1302", "relation":"dhcpPoolLink"}, "1": {"id":"2", "resource_id1":" 1292", "resource_id2":"1452", "relation":"dhcpPoolLink"}, "2": {"id":"12", "resource_id1":" 1422", "resource_id2":"1482", "relation":"dhcpPoolLink"}}}
ERROR:	{"success":0, "message":"error message"}

Return Detail:

Name	Type	Description
id	INTEGER	Id of the pool-module linkage
resource_id1	INTEGER	The id of the dhcp_module resource
resource_id2	INTEGER	The id of the dhcp_pool resource
relation	STRING	The relation type. Always 'dhcpPoolLink'

Meta Attributes:

Name	Type	Description
totalRecords	INTEGER	How many records were found by this query, without pagination.
retrieved	INTEGER	How many records were returned by this query, with pagination.

Optional Attributes:

Name	Type	Description
resultsPerPage	INTEGER	How many records to include per page display.*
page	INTEGER	Which page to display, when used with "resultsPerPage"*
*Example pagination: api.php?target=resource&action=getLink&relation=dhcpPoolLink&resultsPerPag		

add a new DHCP Pool linkage

Description	Adds a new link between a DHCP Pool and a dhcp_module resource. A single pool can be linked to many dhcp_modules, and a single dhcp_module can have any number of linked pools.
URL	/api/v1/api.php?target=resource&action=addLink&resource_id1=1292&resource_id2=216
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Resource link added"} ERROR: {"success":0, "message":"error message"} **Data Detail:**

Name	Type	Description
resource_id1	INTEGER	The id of the dhcp_module resource

resource_id2	INTEGER	The id of the dhcp_pool resource
relation	STRING	The relation type being added. Always 'dhcpPoolLink'

delete DHCP Pool linkages

Description	Deletes a link between a dhcp_module and a dhcp_pool. Uses the standard Resource Linkage endpoints.
URL	/api/v1/api.php?target=resource&action=deleteLink&id=22
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Resource link(s) deleted."} ERROR: {"success":0, "message":"error message"}

push a DHCP config

Description	Builds a DHCP configuration from the attributes assigned to a dhcp_module and all of the linked dhcp_pools. Pushes that config to the configured DHCP server, tests it against the config parsing function, then restarts the server with the new configuration.
URL	/api/v1/api.php?target=dhcp&action=push&id=1292
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Pushes Attempted.","data":[[1," 1292","381 DHCP Module"," Configuration successfully pushed."]]} ERROR: {"success":0, "message":"error message"} **Data Detail** **Name** / **Type** / **Description** id / INTEGER / The id of the dhcp_module resource whose configuration is to be pushed.

DHCP search

Description	Searches DHCP information by name, mac, or IP.
URL	/api/v1/api.php?target=dhcp&action=search&searchType=name&searchValue=Blah

/api/v1/api.php?
target=dhcp&action=search&searchType=mac&searchValue=22:

/api/v1/api.php?
target=dhcp&action=search&searchType=ip&searchValue=13.
0.0.255

Returns

Examples:

SUCCESSFUL:	{"success":1,"message":"Search Successful","data":[{"id":"1482","name":"BlahBlah","slug":"blah","type":"dhcp_pool","parent_id":"1","category_id":null,"attr":{"_dhcp_type":"subnet","_dhcp_pool_attributes":"{\"mac\":\"\",\"rangeStart\":\"13.0.0.0\",\"rangeEnd\":\"13.0.0.255\",\"freeLines\":0}","_dhcp_ip_id":"80902"},"dhcp_links":["1422","3673"]}]}
ERROR:	{"success":0, "message":"error message"}

Return Detail

Name	Type	Description
id	INTEGER	ID of the dhcp_module resource
name	STRING	The name of the dhcp_module
slug	STRING	The unique reference string for this resource
type	STRING	Always 'dhcp_pool'
parent_id	INTEGER	The resource to which the dhcp_module is attached
category_id	INTEGER	The category to which this dhcp_module is associated

Attributes

Key	Type	Description
_dhcp_type	STRING	Either 'subnet' or 'host'. Determines whether this DHCP Pool is describing a Subnet or a Host.
_dhcp_pool_attributes	JSON	A JSON-encoded string containing all the specific configuration parameters which govern this DHCP Pool. An expansion of the JSON object is given below in the Data Attributes section.
_dhcp_ip_id	INTEGER	The id of the IPAM subnet or host which is assigned to this DHCP Pool.
_dhcp_links	INTEGER	The id of dhcp links.

Required Parameters

Name	Type	Example	Description
searchType	STRING	name	Type of search performing. Valid values are 'name', 'mac', and 'ip'.
searchValue	STRING	Blah	If searchType is 'IP': searchValue must be a valid IP address or IP Block. if searchType" is 'mac': searchValue must be a full or partial mac address. Not required if searchType is 'name', but is recommended to filte

Optional Parameters

Name	Type	Example	Description
selectOffset	INTEGER	10	Start number for the first result to show on the page to set pagination. For example, a selectOffset=15 with
selectCount	INTEGER	5	The number of results to return on the page, if return results exceed the number of selectCount. Default value is 10
linkedTo	INTEGER	1422	The ID of the resource's DHCP link(s)

Data Attributes

_dhcp_attributes

Description	The _dhcp_attributes data attribute holds the specific settings used to generate a DHCP configuration file, place it on a server via SCP, and restart that server via a SSH session.
Example:	{"type":"ISC", "notes":"notes here", "username":"username", "port":"22", "config_test":"/etc/init.d/dhcpd configtest", "server_stop":"/etc/init.d/dhcpd stop", "server_start":"/etc/init.d/dhcpd start", "config_path":"/tmp/dhcpd.conf", "option_routers":"", "option_domain_name_servers":"", "option_domain_name":"", "authoritative":"1", "default_lease_time":"600", "max_lease_time":"7200", "local_port":"67", "log_facility":"local7", "password":"", "server_ip":"10.0.0.0", "freeLines":0}

Data Description

Name	Type	Description
type	STRING	The type of DHCP server being administered. Currently only 'ISC' is supported.
notes	STRING	Notes associated with this DHCP server
server_ip	STRING	The IP address of the DHCP server
username	STRING	The SSH username employed when transferring the DHCP configuration file to the server.
password	STRING	The SSH password employed when transferring the DHCP configuration file to the server.
port	INTEGER	The SSH port employed when transferring the DHCP configuration file to the server.
config_test	STRING	The command to test if a configuration file parses correctly. ex: /etc/init.d/dhcpd configtest
server_stop	STRING	The command to stop the DHCP server. ex: /etc/init.d/dhcpd stop
server_start	STRING	The command to start the DHCP server. ex: /etc/init.d/dhcpd start
config_path	STRING	Where to place the configuration file on the server.
authoritative	BOOL	Whether or not this

		DHCP server is authoritative.
default_lease_time	INTEGER	The default lease time for IPs distributed by this DHCP server.
max_lease_time	INTEGER	The max lease time for IPs distributed by this DHCP server.
local_port	INTEGER	The port on which this DHCP server listens
option_routers	STRING	The information which populates the "routers" option in the DHCP configuration
option_domain_name_servers	STRING	The information which populates the "domain_name_servers" o
option_domain_name	STRING	The information which populates the "domain_name" option in the DHCP configuration
log_facility	STRING	The log facility to which this DHCP Server sends its logging information
freeLines	INTEGER	As this system cannot hope to support all the thousands of different DHCP configurations, ProVision's DHCPv2 system includes a mechanism for adding "free lines" to the end of certain DHCP config sections so that administrators can customize their DHCP config file to their needs. The "freeLines" field indicates how many of these lines exist to be inserted after the general server definition section but before the subnets and hosts are enumerated.
freeLine#	STRING	Free line data to be inserted after the general server definition section but before the subnets and hosts

		are enumerated. There can be multiple instances of this attribute, numbered appropriately. ex: "freeLine1", "freeLine2", "freeLine3", etc. The number of freeLine# entries must match the number in the "freeLines" attribute.

_dhcp_pool_attributes

Description	A JSON-encoded string containing all the specific configuration parameters which govern this DHCP Pool.
Example:	{"mac":"ab:cc:de:ff:aa:bc","rangeStart":"13.0.0.0","rangeEnd":"13.0.0.255","freeLines":1,"freeLines1":"free line"}

Data Description

Name	Type	Description
mac	STRING	Only used when setting up a DHCP Host-type Pool. Holds the MAC address of the system to which the IP will be associated.
rangeStart	STRING	Only used when setting up a DHCP Subnet-type Pool. Holds the beginning of the Subnet range being allocated.
rangeEnd	STRING	Only used when setting up a DHCP Subnet-type Pool. Holds the end of the Subnet range being allocated.
freeLines	INTEGER	As this system cannot hope to support all the thousands of different DHCP configurations, ProVision's DHCPv2 system includes a mechanism for adding "free lines" to the end of certain DHCP config sections so that administrators

		can customize their DHCP config file to their needs. The "freeLines" field indicates how many of these lines exist to be inserted within the DHCP Pool declaration.
freeLine#	STRING	Free line data to be inserted after the general server definition section but before the subnets and hosts are enumerated. There can be multiple instances of this attribute, numbered appropriately. ex: "freeLine1", "freeLine2", "freeLine3", etc. The number of freeLine# entries must match the number in the "freeLines" attribute.

API Module - DNS

- DNS Server Control
  - get
  - add
  - delete
  - update
  - transferServer
  - transferSingle
- DNS Zone Control
  - get
  - search
  - update
  - add
  - delete
  - getRecordTypes
  - getFile
  - getDSFile
  - checkZone
  - getArchivedZone
- DNS Record Control
  - get
  - update
  - add
  - delete
  - switch
- Server-Zone Linkage
  - get
  - add
  - delete
- Name Server Control
  - get
  - add
  - delete
  - setDefault
  - orderUp
  - orderDown

DNS Server Control

get	
URL	/api/v1/api.php?target=dnsServer&action=get
Description	If provided with an id, fetches that DNS Server from the database. If not, fetches a list of all stored DNS Servers
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Fetch Sucessful.","data":[{"id":" 10","server":"mrbomasm-dns-4.onnet.net", "username":" user", "password":"vwvddp"," port":"2600","customer_name": null,"transfer_type": "SCP"," remote_directory":"zones", "named_conf_path":"\/etc\ /zones","active":"1"," post_command":null," pre_command":null," dyn_DNSSEC_contact":null, "powerdns_backend":"Bind"," db_username":null," db_password":"","db_port": null,"db_name":null, "server_type":"slave","SOA": null,"master_id":null,"options":

	"{\"customer_name\":\"\",\" server_type\":\"slave\" ,\" SOA\":\"\",\"remote_directory\": \"zones\", \" named_conf_path\":\"\\\/etc\\\ /zones\", \" dyn_DNSSEC_contact\" :\"\",\" post_command\":\"\" ,\" pre_command\":\"\",\" powerdns_backend\":\"Bind\", \"db_username\":\"\",\" db_password\":\"(\",\"db_port\": \"\",\"db_name\":\"\",\" enable_views\":\"1\"}", "testID":"963","zoneCount":" 8","views":"[{\"id\":\"1\", \" server_id\":\"10\",\"name\":\" _6connectDefault\",\"extras\":\" []\",\"description\":\"\",\" timestamp\":\"1371789181\"}, {\"id\":\"3\",\"server_id\":\"10\",\" name\":\"internal\",\"extras\":\"[] \",\"description\":null,\" timestamp\":\"1374686650\"}]"}]}
ERROR:	{"success":0, "message":"error message"}

Data Detail:

Name	Type	Description
id	INTEGER	Server ID
server	STRING	Server Name
username	STRING	Login Name
password	CRYPT	Login Password
port	INTEGER	Port the Server listens on
zoneCount	INTEGER	The number of zones attached to this server.
options	JSON	The options entry is a JSON-encoded string containing a variety of server-specific configuration options. This string will vary widely by server type and configuration. The following are a selection of common settings.
transfer_type	STRING	Protocol used for transfer of DNS zones and records. Valid settings include SCP, PowerDNS, Secure64, Secure64Signer

	server_type | STRING | Whether this server is a master or a slave server SOA | STRING | The SOA entry to be used for zones on this server remote_directory | STRING | The directory where SCP will place the zone files. named_conf_path | STRING | The path to the zone files used within the named.conf file. pre_command | STRING | The command executed on the server before the zones are transferred post_command | STRING | The command executed on the server after the transfer is complete enable_views | INTEGER | Whether or not Views are enabled views | JSON | The views entry is a JSON-encoded string containing all the information about the Views attached to this server, if any exist. id | INTEGER | The View ID server_id | INTEGER | The ID of the server the View is attached to name | STRING | The name of the View description | STRING | A description of the View timestamp | INTEGER | The UNIX timestamp of when the view was created. extras | JSON | A JSON-encoded array of the extra attributes printed out in the view definition in the config file.
Required Parameters	None
Optional Parameters	**Name** | **Type** | **Example** | **Description** id | INTEGER | 15 | The server id to fetch.
Example URL	/api/v1/api.php?target=dnsServer&action=get&id=15

add

URL	/api/v1/api.php?target=dnsServer&action=add
Description	Adds a new DNS Server
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Add Successful."} ERROR: {"success":0, "message":"error message"}

Required Parameters

Name	Type	Example	Description
server	STRING	dns.yourdomain.com	IP or FQDN of the DNS Server
password	STRING	password1	Login password for Server
transferType	STRING	SCP	Protocol used for transfer of DNS zones and records. Valid settings include SCP, PowerDNS, Secure64, Secure64Signer
serverType	STRING	Master	Values are 'Master' or 'Slave' only
displayName	STRING	Primary NS	The name displayed representing the DNS server, can be the same as server or different
SOA	STRING	ns1.6connect.com. hostmaster.6connect.com.	Server of Authority record for DNS server

Optional Parameters

These optional parameters vary according to what type of server is being configured.

Name	Type	Example	Description
customerName	STRING	/tmp/zones	Customer Name
remoteDirectory	STRING	/tmp/zones	Zone Directory on Server
port	INTEGER	22	Port for ssh or scp access to server
namedConfPath	STRING	/tmp	The path to the zone files used within the named.conf file.

	preCommand STRING /path/to/stuff /precommand Command to execute before zone transfer postCommand STRING /path/to/stuff /postcommand Command to execute after zone transfer DNSSECContact STRING joeuser For use with Dyn dns service username STRING bobuser Login name for Server active INTEGER 0 Values 0 or 1 only, sets the server to inactive on 0 value masterId INTEGER 53 Master server ID. If a server is a slave, masterId points to its master. powerDNSBackend STRING Bind or MySQL pDNS server backend type dbDatabaseName STRING pdns_1 DB name for pDNS servers with MySQL powerDNSBackend t dbPort INTEGER 3306 Port for for pDNS servers with MySQL powerDNSBackend t dbUsername STRING someuser DB username for pDNS servers with MySQL powerDNSBackend t dbPassword STRING somepass DB password for pDNS servers with MySQL powerDNSBackend t
Example URL	/api/v1/api.php?target=dnsServer&action=add&server=dns.yourdomain.com&transferType=Secure64&displayName=PrimaryNS&serverType=master&password=password1&SOA=ns1.6connect.com.+hostmaster.6connect.com.

delete

URL	/api/v1/api.php?target=dnsServer&action=delete
Description	Deletes a DNS Server
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Delete Successful."} ERROR: {"success":0, "message":"error message"}

Required Parameters	**Name** / **Type** / **Example** / **Description** id / INTEGER / 5 / ID of server to delete
Optional Parameters	None
Example URL	/api/v1/api.php?target=dnsServer&action=delete&id=5

update

URL	/api/v1/api.php?target=dnsServer&action=update
Description	Updates an existing DNS Server with new information.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Update Successful."} ERROR: {"success":0, "message":"error message"}

Required Parameters

Name	Type	Example	Description
id	INTEGER	5	ID of server
server	STRING	dns.yourdomain.com	IP or FQDN of the DNS Server
SOA	STRING	ns1.6connect.com. hostmaster.6connect.com.	Server of Authority record for DNS server
transferType	STRING	SCP	Protocol used for transfer of DNS zones and records. Valid settings include SCP, PowerDNS, Secure64, Secure64Signer

Optional Parameters

These optional parameters vary according to what type of server is being configured.

Name	Type	Example	Description
active	INTEGER	0	Values 0 or 1 only, sets the server to inactive on 0 value
customerName	STRING	/tmp/zones	Customer Name
dbDatabaseName	STRING	pdns_1	DB name for pDNS servers with MySQL powerDNSBackend t
dbPassword	STRING	somepass	DB password for pDNS servers with MySQL powerDNSBackend t

dbPort	INTEGER	3306	Port for for pDNS servers with MySQL powerDNSBackend t
dbUsername	STRING	someuser	DB username for pDNS servers with MySQL powerDNSBackend t
displayName	STRING	Primary NS	The name displayed representing the DNS server, can be the same as server or different
DNSSECContact	STRING	joeuser	For use with Dyn dns service
enable_views	INTEGER	1	Whether or not Views are enabled. Valid values are '1' for enable or '0' for do not enable
masterId	INTEGER	53	Master server ID. If a server is a slave, masterId points to its master.
namedConfPath	STRING	/tmp	The path to the zone files used within the named. conf file.
password	STRING	password1	Login password for Server
port	INTEGER	22	Port for ssh or scp access to server
powerDNSBackend	STRING	Bind or MySQL	pDNS server backend type
postCommand	STRING	/path/to/stuff /postcommand	Command to execute after zone transfer
preCommand	STRING	/path/to/stuff /precommand	Command to execute before zone transfer
remoteDirectory	STRING	/tmp/zones	Zone Directory on Server
serverType	STRING	Master	Values are 'Master' or 'Slave' only
username	STRING	bobuser	Login name for Server

Example URL	/api.php?target=dnsServer&action=update&id=74&transferType=SCP&server=dns.yourdomain.com&SOA=ns1.6connect.com.+hostmaster.6connect.com.

transferServer

URL	/api/v1/api.php?target=dnsServer&action=transferServer
Description	Performs a full zone push on a DNS Server, executing pre and post commands, transferring files, and restarting services.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Transfer Successful."} ERROR: {"success":0, "message":"error message"}
Required Parameters	**Name** | **Type** | **Example** | **Description** push | INTEGER | 1 | The ID of the server to push zones to
Optional Parameters	None
Example URL	/api/v1/api.php?target=dnsServer&action=transferServer&push=1

transferSingle

URL	/api/v1/api.php?target=dnsServer&action=transferSingle
Description	Transfers a single Zone file to all its associated DNS Servers, along with updated server configurations. Performs pre and post commands on the target servers, transfers the zone file(s), and restarts services.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Updated Zone: $name. zone on $server via SCP"} ERROR: {"success":0, "message":"error message"}
Required Parameters	**Name** | **Type** | **Example** | **Description** zoneId | INTEGER | 35 | The ID of the zone to push
Optional Parameters	None
Example URL	/api/v1/api.php?target=dnsServer&action=transferSingle&zoneId=35

DNS Zone Control

get

URL	/api/v1/api.php?target=zone&action=get

Description	Accepts search criteria to retrieve a list of all matching DNS Zones and associated Records. Search can be performed on any combination of Zone and Record attributes.
Returns	**Examples:**

SUCCESSFUL:	{"success":1,"message":" Search Successful.","data":[{"zoneId":"932","zoneName":" 185.160.209.in-addr.arpa"," zoneResourceId":"81", "zoneSerial":"2013040302"," zoneRefresh":"28800"," zoneRetry":"7200"," zoneExpire":"604800"," zoneMinimum":"86400"," zoneSOA":null, "zoneTTL":" 28800","zoneAutoCheck":"1"," zoneEnableDNSSEC":null," recordId":"154110"," recordZoneId":"932", "recordHost": "185.160.209. inaddr.arpa.","recordType":" NS","recordValue":"auth01. veroxity.net."," recordDescription":null," recordTTL":"28800", "recordOrdering":"1", "recordErrors":null,"assetId":" 0","userCanCreate":0," userCanDelete":1," userCanUpdate":1}]}
ERROR:	{"success":0, "message":"error message"}

Data Detail:

Name	Type	Description
zoneId	INTEGER	The Id of the Zone entry. A single Zone entry might have multiple Records.
zoneName	STRING	The Zone name.
zoneResourceId	INTEGER	The resource Id associated with this Zone.
zoneSerial	INTEGER	Zone Serial.
zoneRefresh	INTEGER	Zone Refresh.
zoneRetry	INTEGER	Zone Retry.
zoneExpire	INTEGER	Zone Expire.
zoneMinimum	INTEGER	Zone Minimum.
zoneSOA	STRING	Zone SOA.
zoneTTL	STRING	Zone TTL.
zoneEnableDNSSEC	BOOL	Whether or not DNSSEC is enabled for this Zone.

zoneAutoCheck	BOOL	Whether or not this zone is configured to be automatically validated on load /edit.
recordId	INTEGER	The Id of this Record Entry. It is always included with its parent Zone.
recordHost	STRING	The Hostname of this Record.
recordType	STRING	The Record Type (MX,NS,A,PTR,etc)
recordValue	STRING	The Value of this Record.
recordDescription	STRING	A short description of this Record.
recordTTL	STRING	The TTL of this Record.
recordOrdering	INTEGER	The numerical order in which the record appears in the zone.
recordErrors	STRING	A string containing any detected problems with this record
userCanCreate	BOOL	Whether or not the user has DNS CREATE permissions on this zone's resource
userCanUpdate	BOOL	Whether or not the user has DNS UPDATE permissions on this zone's resource
userCanDelete	BOOl	Whether or not the user has DNS DELETE permissions on this zone's resource
unpagedRows	INTEGER	If pagination is used, this value will contain a total count of records had the pagination not been used.

Required Parameters: None

Optional Parameters

Name	Type	Example	Description
likeFlag	BOOL	1	When 1, string searches are done via LIKE with wildcards at both ends. When 0, strict

			comparison is used.
generalFlag	BOOL	1	When 1, searches over the provided parameters using OR. If 0 or omitted, uses AND.
selectCount	INTEGER	30	When supplied only returns the first X entries

Name	Type	Example	Description
zoneId	INTEGER	123	The Zone Id to search for.
zoneName	STRING	foo	The Zone Name to search for.
zoneResourceId	INTEGER	5	The Resource Id to search for.
zoneSerial	INTEGER	2012033001	The Zone Serial to search for.
zoneRefresh	INTEGER	36000	The Zone Refresh to search for.
zoneRetry	INTEGER	800	The Zone Retry to search for.
zoneExpire	INTEGER	6090000	The Zone Expire to search for.
zoneMinimum	INTEGER	10	The Zone Minimum to search for.
zoneSOA	STRING	200	The Zone SOA to search for.
zoneTTL	INTEGER	3600	The Zone TTL to search for.
zoneEnableDNSSEC	INTEGER	1	Search based on DNSSEC settings.
recordId	INTEGER	123	The Record Id to search for.
recordZoneId	INTEGER	123	The parent Zone to search for.
recordHost	STRING	@	The Record Host to search for.
recordType	STRING	NS	The Record Type to search for.
recordValue	STRING	ns1.dns.6connect.	The Record Value to

		com.	search for.
recordDescription	STRING	Description	Search based on Record Description.
recordTTL	STRING	3600	The Record TTL to search for.

Example URL	/api/v1/api.php?target=zone&action=get&zoneId=123

search

URL	/api/v1/api.php?target=zone&action=search
Description	Accepts search criteria to retrieve a list of all matching DNS Zones but NO associated Records. Search can be performed on any combination of Zone and Record attributes.
Returns	**Examples:**

SUCCESSFUL:	{"success":1,"message":" Search Successful.","data": [{"zoneId":"123","zoneName":" foobs.net","zoneResourceId":" 483","zoneIpver":null," zoneMask":null,"zoneSerial":" 2012121803","zoneRefresh": null,"zoneRetry":null," zoneExpire":null," zoneMinimum":null," zoneSOA":null,"zoneTTL":" 3600","zoneEnableDNSSEC":" 1","zoneLocalSigning":"1"," assetId":"0","recordCount":"1"," unpagedRows":"215"}]}
ERROR:	{"success":0, "message":"error message"}

Data Detail:

Name	Type	Description
zoneId	INTEGER	The Id of the Zone entry. A single Zone entry might have multiple Records.
zoneName	STRING	The Zone name.
zoneResourceId	INTEGER	The resource Id associated with this Zone.
zoneSerial	INTEGER	Zone Serial.
zoneRefresh	INTEGER	Zone Refresh.
zoneRetry	INTEGER	Zone Retry.
zoneExpire	INTEGER	Zone Expire.
zoneMinimum	INTEGER	Zone Minimum.
zoneSOA	STRING	Zone SOA.
zoneTTL	STRING	Zone TTL.

	zoneEnableDNSSEC: BOOL — Whether or not DNSSEC is enabled for this Zone. zoneAutoCheck: BOOL — Whether or not this zone is configured to be automatically validated on load /edit. recordCount: INTEGER — How many records are associated with this zone. userCanCreate: BOOL — Whether or not the user has DNS CREATE permissions on this zone's resource userCanUpdate: BOOL — Whether or not the user has DNS UPDATE permissions on this zone's resource userCanDelete: BOOl — Whether or not the user has DNS DELETE permissions on this zone's resource unpagedRows: INTEGER — If pagination is used, this value will contain a total count of records had the pagination not been used.
Required Parameters	None
Optional Parameters	**Name** / **Type** / **Example** / **Description** likeFlag: BOOL: 1 — When 1, string searches are done via LIKE with wildcards at both ends. When 0, strict comparison is used. generalFlag: BOOL: 1 — When 1, searches over the provided paramenters using OR. If 0 or omitted, uses AND. selectCount: INTEGER: 30 — When supplied only returns the first X entries selectOffset: INTEGER: 10 — When supplied, only returns entries after record X

sortArray	JSON	{"zoneName":"desc","zoneMask":"asc"}	A JSON-encoded object containing a list of columns to sort on and the direction in which to sort. Any API variable may be used for sorting. Valid sort directions are ASC and DESC.

Name	Type	Example	Description
zoneId	INTEGER	123	The Zone Id to search for.
zoneName	STRING	foo	The Zone Name to search for.
zoneResourceId	INTEGER	5	The Resource Id to search for.
zoneSerial	INTEGER	2012033001	The Zone Serial to search for.
zoneRefresh	INTEGER	36000	The Zone Refresh to search for.
zoneRetry	INTEGER	800	The Zone Retry to search for.
zoneExpire	INTEGER	6090000	The Zone Expire to search for.
zoneMinimum	INTEGER	10	The Zone Minimum to search for.
zoneSOA	STRING	200	The Zone SOA to search for.
zoneTTL	INTEGER	3600	The Zone TTL to search for.
zoneEnableDNSSEC	INTEGER	1	Search based on DNSSEC settings.
recordId	INTEGER	123	The Record Id to search for.
recordZoneId	INTEGER	123	The parent Zone to search for.
recordHost	STRING	@	The Record Host to search for.
recordType	STRING	NS	The Record Type to search for.
recordValue	STRING	ns1.dns.6connect.	The Record Value to

		com.	search for.
recordDescription	STRING	Description	Search based on Record Description.
recordTTL	STRING	3600	The Record TTL to search for.

Example URL	/api/v1/api.php?target=zone&action=search&zoneId=123

update

URL	/api/v1/api.php?target=zone&action=update
Description	First performs a search based on the submitted Zone and Record criteria, then performs an Update across those entries based on new values.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Update Successful."} ERROR: {"success":0, "message":"error message"}
Required Parameters	None
Optional Parameters	

Name	Type	Example	Description
likeFlag	BOOL	1	When 1, string searches are done via LIKE with wildcards at both ends. When 0, strict comparison is used.
generalFlag	BOOL	1	When 1, searches over the provided parameters using OR. If 0 or omitted, uses AND.

Name	Type	Example	Description
searchZoneId	INTEGER	123	The Zone Id to search for.
searchZoneName	STRING	foo	The Zone Name to search for.
searchZoneResourceId	INTEGER	5	The Resource Id to search for.
searchZoneSerial	INTEGER	2012033001	The Zone Serial to search for.
searchZoneRefresh	INTEGER	36000	The Zone Refresh to

			search for.
searchZoneRetry	INTEGER	800	The Zone Retry to search for.
searchZoneExpire	INTEGER	6090000	The Zone Expire to search for.
searchZoneMinimum	INTEGER	10	The Zone Minimum to search for.
searchZoneSOA	STRING	200	The Zone SOA to search for.
searchZoneTTL	INTEGER	3600	The Zone TTL to search for.
searchZoneEnableDNSSEC	INTEGER	1	Search based on DNSSEC settings.
searchRecordId	INTEGER	123	The Record Id to search for.
searchRecordHost	STRING	@	The Record Host to search for.
searchRecordType	STRING	NS	The Record Type to search for.
searchRecordValue	STRING	ns1.dns.6connect.com.	The Record Value to search for.
searchRecordDescription	STRING	Description	Search based on Record Description.
searchRecordTTL	STRING	3600	The Record TTL to search for.

Name	Type	Example	Description
updateZoneName	STRING	foo	The Zone name to replace into the searched rows.
updateZoneSerial	INTEGER	2012033001	The Zone Serial to replace into the searched rows.
updateZoneRefresh	INTEGER	36000	The Zone Refresh to replace into the searched rows.
updateZoneRetry	INTEGER	800	The Zone Retry to

			replace into the searched rows..
updateZoneExpire	INTEGER	6090000	The Zone Expire to replace into the searched rows.
updateZoneMinimum	INTEGER	10	The Zone Minimum to replace into the searched rows.
updateZoneSOA	STRING	200	The Zone SOA to replace into the searched rows.
updateZoneTTL	INTEGER	3600	The Zone TTL to replace into the searched rows.
updateZoneEnableDNSSEC	INTEGER	1	Update DNSSEC Settings.
updateRecordHost	STRING	@	The Record Host to replace into the searched rows.
updateRecordType	STRING	NS	The Record Type to replace into the searched rows.
updateRecordValue	STRING	ns1.dns.6connect.com.	The Record Value to replace into the searched rows.
updateRecordDescription	STRING	Description	Update Record Descriptions.
updateRecordTTL	STRING	3600	The Record TTL to replace into the searched rows.
updateZoneAutoCheck	BOOL	1	Whether or not this zone is configured to be automatically validated on load/edit.

Name	Type	Example	Description
recordZoneId	INTEGER	123	The parent zone ID

Example URL	/api/v1/api.php?target=zone&action=update&searchZoneId=123&updateZoneExpire=6090000

add

URL	/api/v1/api.php?target=zone&action=add
Description	Adds a new DNS Zone.

Returns

Examples:

SUCCESSFUL:	{"success":1,"message":"Add Successful.","data":123}
ERROR:	{"success":0, "message":"error message"}

Data Detail:

Name	Type	Description
data	INTEGER	The Id of the new Zone entry.

Required Parameters

Name	Type	Example	Description
zoneName	STRING	254.221.67.in-addr.arpa	The name for the new Zone.

Optional Parameters

Name	Type	Example	Description
likeFlag	BOOL	1	When 1, string searches are done via LIKE with wildcards at both ends. When 0, strict comparison is used.
zoneIpver	STRING	IPv6	The IP Version.
zoneLocalSigning	BOOL	1	Whether or not this zone should be signed by the ProVision server when DNSSEC is enabled. If set to false, ProVision will deliver the zone unsigned to the DNS server and the signing / updating process should be triggered by the post-push command
zoneSerial	INTEGER	2012033001	Serial for the new Zone.
zoneRefresh	INTEGER	36000	Refresh for the new Zone.

zoneRetry	INTEGER	800	Retry for the new Zone.
zoneExpire	INTEGER	6090000	Expire for the new Zone.
zoneMinimum	INTEGER	10	Minimum for the new Zone.
zoneSOA	STRING	200	SOA for the new Zone.
zoneTTL	STRING	3600	TTL for the new Zone.
zoneEnableDNSSEC	INTEGER	1	Whether or not this new zone uses DNSSEC.

Example URL	/api/v1/api.php?target=zone&action=add&zoneName=254.221.67.in-addr.arpa&zoneSerial=2012033001

delete

URL	/api/v1/api.php?target=zone&action=delete
Description	Performs a search over the Zones and Records dataset and deletes all found Zones, plus all associated Records of those Zones.
Returns	**Examples:**

SUCCESSFUL:	{"success":1,"message":"Zones and Associated Records Deleted."}
ERROR:	{"success":0, "message":"error message"}

Required Parameters	No specific parameter is required, however, one or more optional parameters must be used for a successful return
Optional Parameters	

Name	Type	Example	Description
deleteZoneId	INTEGER	123	The Zone Id to search for.
deleteZoneName	STRING	foo	The Zone Name to search for.
deleteZoneResourceId	INTEGER	5	The Resource Id to search for.
deleteZoneSerial	INTEGER	2012033001	The Zone Serial to search for.
deleteZoneRefresh	INTEGER	36000	The Zone Refresh to search for.
deleteZoneRetry	INTEGER	800	The Zone Retry to search for.
deleteZoneExpire	INTEGER	6090000	The Zone Expire to search for.
deleteZoneMinimum	INTEGER	10	The Zone

			Minimum to search for.
deleteZoneSOA	STRING	200	The Zone SOA to search for.
deleteZoneTTL	INTEGER	3600	The Zone TTL to search for.
deleteZoneEnableDNSSEC	INTEGER	1	Search based on DNSSEC settings.
deleteRecordId	INTEGER	123	The Record Id to search for.
deleteRecordHost	STRING	@	The Record Host to search for.
deleteRecordType	STRING	NS	The Record Type to search for.
deleteRecordValue	STRING	ns1.dns.6connect.com.	The Record Value to search for.
deleteRecordDescription	STRING	Description	Search based on Record Description.
deleteRecordTTL	STRING	3600	The Record TTL to search for.
deleteRecordZoneId	INTEGER	123	The parent zone ID

Example URL	/api/v1/api.php?target=zone&action=delete&deleteZoneId=123

getRecordTypes

URL	/api/v1/api.php?target=zone&action=getRecordTypes
Description	Returns a list of all Record Types allowed by the system.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Search Successful.","data": [{"recordType":"A"}, {"recordType":"AAAA"}, {"recordType":"MX"}, {"recordType":"CNAME"}, {"recordType":"PTRG"}, {"recordType":"NS"}, {"recordType":"TXT"}, {"recordType":"DNSKEY"}, {"recordType":"SRV"}, {"recordType":"DS"}, {"recordType":"TEST"}]} ERROR: {"success":0, "message":"error message"} **Data Detail:**

	<table><tr><th>Name</th><th>Type</th><th>Description</th></tr><tr><td>recordType</td><td>STRING</td><td>A Record Type</td></tr></table>
Required Parameters	None
Optional Parameters	None
Example URL	/api/v1/api.php?target=zone&action=getRecordTypes

getFile

URL	/api/v1/api.php?target=zone&action=getFile&zoneId=50
Description	Returns a fully written zone file. If one does not exist, returns false.
Returns	A Zone File
Required Parameters	<table><tr><th>Name</th><th>Type</th><th>Example</th><th>Description</th></tr><tr><td>zoneId</td><td>INTEGER</td><td>50</td><td>The Id of the zone to retrieve.</td></tr><tr><td>format</td><td>ENUMERATED</td><td>'html' or ''</td><td>If html, the zone file will be formatted for display via a web browser. If blank or omitted, the zone file will be formatted for display in a file system.</td></tr><tr><td>unsigned</td><td>BOOL</td><td>1</td><td>For a DNSSEC-enabled zone, determines whether or not the system retrieves the signed or unsigned zone file. Ignored for non-DNSSEC zones.</td></tr></table>
Optional Parameters	None
Example URL	/api/v1/api.php?target=zone&action=getFile&zoneId=50&zoneId=50&format=html&unsigned=1

getDSFile

URL	/api/v1/api.php?target=zone&action=getDSFile
Description	Returns a fully written zone DS key file. If one does not exist, returns false.
Returns	A Zone DS Key File
Required Parameters	

Name	Type	Example	Description
zoneId	INTEGER	50	The Id of the zone whose DS keys are to be retrieved.

Optional Parameters	None
Example URL	/api/v1/api.php?target=zone&action=getDSFile&zoneId=50

checkZone

URL	/api/v1/api.php?target=zone&action=checkZone
Description	Runs a zone file through Named checkzone
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"No errors found."} ERROR: {"success":0,"message":"21: ignoring out-of-zone data (veggie.com) 22: ignoring out-of-zone data (veggie.com) dns_rdata_fromtext: 23: near '2001:db8:': bad IPv6 address dns_rdata_fromtext: 24: near '1.2.3.': bad dotted quad dns_rdata_fromtext: 25: near '2001::db8::\/32': bad IPv6 address "}
Required Parameters	**Name** / **Type** / **Example** / **Description** zoneId / INTEGER / 50 / The Id of the zone to check.
Optional Parameters	None
Example URL	/api/v1/api.php?target=zone&action=checkZone&zoneId=50

getArchivedZone

URL	/api/v1/api.php?target=zone&action=getArchivedZone
Description	Searches for all archived versions of the a zone. Zones are archived every time changes are pushed to their DNS Server.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Search Successful.","data": [{"zoneArchiveId" :"2768"," zoneId":"1227", "zoneArchiveTimestamp":" 1375298692", "zoneArchiveFingerprint":" d060e59d69606326d80b2e55b50f0bc]} ERROR: {"success":0, "message":"error message"} **Data Detail:** **Name** / **Type** / **Description**

	zoneId INTEGER The Id of the Zone entry to find archived versions of. zoneArchiveId INTEGER The ID of the Archive Entry zoneArchiveTimestamp INTEGER A timestamp marking when this zone was archived. zoneArchiveFingerprint STRING A hash value identifying this zone. Used for comparing versions. zoneName INTEGER Zone Name. zoneSerial INTEGER Zone Serial. zoneRefresh INTEGER Zone Refresh. zoneRetry INTEGER Zone Retry. zoneExpire INTEGER Zone Expire. zoneMinimum INTEGER Zone Minimum. zoneSOA STRING Zone SOA. zoneTTL INTEGER Zone TTL. zoneEnableDNSSEC STRING Whether or not this version had DNSSEC enabled. zoneResourceId STRING Zone Resource ID zonePreviousViewLinkage JSON A JSON-encoded array of views this zone was linked to.
Required Parameters	None
Optional Parameters	**Name** / **Type** / **Example** / **Description** zoneId INTEGER 123 The Zone Id to search for. zoneArchiveId INTEGER 123 The Zone Archive Id zoneArchiveTimestamp INTEGER 2012033001 The Zone Archive Timestamp fetchArchiveFile BOOL 1 Whether or not to return the full Zone file with the result set..
Example URL	/api/v1/api.php?target=zone&action=getArchivedZone&zoneId=123

DNS Record Control

get

URL	/api/v1/api.php?target=record&action=get
Description	Accepts search criteria to retrieve a list of all matching DNS Records. Search can be performed on any combination of Zone and Record attributes.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Search Successful.","data": [{"recordId":"30894"," recordZoneId":"229"," recordHost":"@"," recordType":"NS"," recordValue":"ns1.domain. com.","recordDescription":""," recordTTL":""}]} ERROR: {"success":0, "message":"error message"} **Data Detail:** **Name** / **Type** / **Description** recordId / INTEGER / The ID of this Record Entry. It is always included with its parent Zone. recordZoneId / INTEGER / The ID of this Record's parent Zone. recordHost / STRING / The Hostname of this Record. recordType / STRING / The Record Type (MX,NS,A,PTR,etc) recordValue / STRING / The Value of this Record. recordDescription / STRING / A short description of this Record. recordTTL / STRING / The TTL of this Record.
Required Parameters	None
Optional Parameters	**Name** / **Type** / **Example** / **Description** likeFlag / BOOL / 1 / When 1, string searches are done via LIKE with wildcards at both ends. When 0, strict comparison is used. selectCount / INTEGER / 30 / When supplied only returns the first X entries selectOffset / INTEGER / 10 / When supplied, only

			returns entries after record X

Name	Type	Example	Description
recordId	INTEGER	123	The Record ID to search for.
recordZoneId	INTEGER	123	The parent Zone to search for.
recordHost	STRING	@	The Record Host to search for.
recordType	STRING	NS	The Record Type to search for.
recordValue	STRING	ns1.dns.6connect.com.	The Record Value to search for.
recordDescription	STRING	Description	Search based on Record Description.
recordTTL	STRING	3600	The Record TTL to search for.

Name	Type	Example	Description
zoneId	INTEGER	123	The Zone Id to search for.
zoneName	STRING	foo	The Zone Name to search for.
zoneResourceId	INTEGER	5	The Resource Id to search for.
zoneCustName	STRING	foo	The Customer Name to search for.
zoneIpver	STRING	IPv6	The IP Version to search for.
zoneSerial	INTEGER	2012033001	The Zone Serial to search for.
zoneRefresh	INTEGER	36000	The Zone Refresh to search for.
zoneRetry	INTEGER	800	The Zone Retry to search for.
zoneExpire	INTEGER	6090000	The Zone Expire to search for.
zoneMinimum	INTEGER	10	The Zone Minimum to search for.
zoneSOA	STRING	200	The Zone SOA to search for.
zoneTTL	INTEGER	3600	The Zone TTL to search for.

zoneEnableDNSSEC	INTEGER	1	Search based on DNSSEC settings.

Example URL	/api/v1/api.php?target=record&action=get&selectCount=30&zoneId=123

update

URL	/api/v1/api.php?target=record&action=update
Description	First performs a search based on the submitted Zone and Record criteria, then performs an Update across those entries based on new values.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Update Successful."} ERROR: {"success":0, "message":"error message"}
Required Parameters	None
Optional Parameters	

Name	Type	Example	Description
likeFlag	BOOL	1	When 1, string searches are done via LIKE with wildcards at both ends. When 0, strict comparison is used.
generalFlag	BOOL	1	When 1, searches over the provided parameters using OR. If 0 or omitted, uses AND.

Name	Type	Example	Description
searchZoneId	INTEGER	123	The Zone ID to search for.
searchZoneName	STRING	foo	The Zone Name to search for.
searchZoneCustId	INTEGER	5	The Customer ID to search for.
searchZoneSerial	INTEGER	2012033001	The Zone Serial to search for.
searchZoneRefresh	INTEGER	36000	The Zone Refresh to search for.
searchZoneRetry	INTEGER	800	The Zone Retry to search for.

searchZoneExpire	INTEGER	6090000	The Zone Expire to search for.
searchZoneMinimum	INTEGER	10	The Zone Minimum to search for.
searchZoneSOA	STRING	200	The Zone SOA to search for.
searchZoneTTL	INTEGER	3600	The Zone TTL to search for.
searchZoneEnableDNSSEC	INTEGER	1	Search based on DNSSEC settings.
searchRecordId	INTEGER	123	The Record ID to search for.
searchRecordHost	STRING	@	The Record Host to search for.
searchRecordType	STRING	NS	The Record Type to search for.
searchRecordValue	STRING	ns1.dns.6connect.com.	The Record Value to search for.
searchRecordDescription	STRING	Description	Search based on Record Description.
searchRecordTTL	STRING	3600	The Record TTL to search for.
searchZoneResourceId	INTEGER	5	The Resource Id to search for.
searchRecordZoneId	INTEGER	123	The Zone ID of the Record to search for.

Name	Type	Example	Description
updateZoneName	STRING	foo	The Zone name to replace into the searched rows.
updateZoneSerial	INTEGER	2012033001	The Zone Serial to replace into the searched rows.
updateZoneRefresh	INTEGER	36000	The Zone Refresh to replace into the searched rows.

updateZoneRetry	INTEGER	800	The Zone Retry to replace into the searched rows..
updateZoneExpire	INTEGER	6090000	The Zone Expire to replace into the searched rows.
updateZoneMinimum	INTEGER	10	The Zone Minimum to replace into the searched rows.
updateZoneSOA	STRING	200	The Zone SOA to replace into the searched rows.
updateZoneTTL	INTEGER	3600	The Zone TTL to replace into the searched rows.
updateZoneEnableDNSSEC	INTEGER	1	Update DNSSEC Settings.
updateRecordHost	STRING	@	The Record Host to replace into the searched rows.
updateRecordType	STRING	NS	The Record Type to replace into the searched rows.
updateRecordValue	STRING	ns1.dns.6connect.com.	The Record Value to replace into the searched rows.
updateRecordDescription	STRING	Description	Update Record Descriptions.
updateRecordTTL	STRING	3600	The Record TTL to replace into the searched rows.
updateZoneResourceId	INTEGER	5	The Resource Id to replace into the searched rows.
updateZoneAutoCheck	BOOL	1	Whether or not this zone is configured to be automatically validated on load/edit.

Example URL	/api/v1/api.php?target=record&action=update&searchZoneId=123&updateZoneTTL=3600

add

URL	/api/v1/api.php?target=record&action=add
Description	Adds a new Record to a supplied Zone.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Add Successful.","data":123} ERROR: {"success":0, "message":"error message"} **Data Detail:** Name / Type / Description data / INTEGER / The ID of the new Record entry.
Required Parameters	Name / Type / Example / Description newRecordZoneId / INTEGER / 123 / The Zone ID of the new Record. newRecordHost / STRING / @ / New Host Name. newRecordType / STRING / PTR / New Record Type. newRecordValue* / STRING / 123 / New Record Value. *newRecordValue required only for certain Record Types
Optional Parameters	Name / Type / Example / Description likeFlag / BOOL / 1 / When 1, string searches are done via LIKE with wildcards at both ends. When 0, strict comparison is used. newRecordDescription / STRING / Description. / Notes for the Record. newRecordTTL / INTEGER / 3600 / Record TTL.
Example URL	/api/v1/api.php?target=record&action=add&newRecordZoneId=123&newRecordHost=@host&newRecordType=PTR&newRecordTTL=3600

delete

URL	/api/v1/api.php?target=record&action=delete
Description	Performs a search over the Zones and Records dataset and deletes all found Records, but leaves their parent Zones intact.
Returns	**Examples:**

SUCCESSFUL:	{"success":1,"message":"Deletion Successful."}
ERROR:	{"success":0, "message":"error message"}

Required Parameters	None
Optional Parameters	

Name	Type	Example	Description
deleteZoneId	INTEGER	123	The Zone ID to search for.
deleteZoneName	STRING	foo	The Zone Name to search for.
deleteZoneCustId	INTEGER	5	The Customer ID to search for.
deleteZoneIpver	STRING	IPv6	The IP Version to search for.
deleteZoneSerial	INTEGER	2012033001	The Zone Serial to search for.
deleteZoneRefresh	INTEGER	36000	The Zone Refresh to search for.
deleteZoneRetry	INTEGER	800	The Zone Retry to search for.
deleteZoneExpire	INTEGER	6090000	The Zone Expire to search for.
deleteZoneMinimum	INTEGER	10	The Zone Minimum to search for.
deleteZoneSOA	STRING	200	The Zone SOA to search for.
deleteZoneTTL	INTEGER	3600	The Zone TTL to search for.
deleteZoneEnableDNSSEC	INTEGER	1	Search based on DNSSEC settings.
deleteRecordId	INTEGER	123	The Record ID to search for.
deleteRecordHost	STRING	@	The Record Host to search for.
deleteRecordType	STRING	NS	The Record Type to search for.
deleteRecordValue	STRING	ns1.dns.6connect.com.	The Record Value to search for.

	deleteRecordDescription	STRING	Description	Search based on Record Description.
	deleteRecordTTL	STRING	3600	The Record TTL to search for.
	deleteZoneResourceId	INTEGER	5	The Resource Id to search for.
	deleteZoneCustName	STRING	foo	The Customer Name to search for.
Example URL	/api/v1/api.php?target=record&action=delete&deleteZoneName=foo			

switch	
URL	/api/v1/api.php?target=record&action=switch
Description	Switches the order of two record entries.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Record Moved."} ERROR: {"success":0, "message":"error message"}
Required Parameters	**Name** / **Type** / **Example** / **Description** moveWhichId / INTEGER / 123 / The Record Id to be moved. moveAfterId / INTEGER / 42 / The Id of the Record the first Record is to be moved after.
Optional Parameters	None
Example URL	/api/v1/api.php?target=record&action=switch&moveWhichId=123&moveAfterId=42

Server-Zone Linkage

get	
URL	/api/v1/api.php?target=zoneLinkage&action=get
Description	Searches for Server-Zone Linkages. If no search parameters are supplied, all linkages are returned.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"2 rows retrieved.","data":[{"id":"285","zoneId":"64","serverId":"1","serverName":"173.164.182.169","serverType":"SCP","serverMasterType":"master","zoneName":"bind.com","

	resourceId":"483"},{"id":"287","zoneId":"371","serverId":"1","serverName":"173.164.182.169","serverType":"SCP","serverMasterType":"master","zoneName":"132.235.198.in-addr.arpa","resourceId":"577"}}
ERROR:	{"success":0, "message":"error message"}

Data Detail:

Name	Type	Description
id	INTEGER	The Linkage Id.
zoneId	INTEGER	The ZoneId involved in this link.
serverId	INTEGER	The ServerId involved in this link.
serverName	STRING	The server name
serverType	STRING	The server transfer type
serverMasterType	STRING	Whether this server is a master or a slave.
zoneName	STRING	The zone name
resourceId	INTEGER	The Resource Id the Zone is attached to.

Required Parameters	None
Optional Parameters	

Name	Type	Example	Description
id	INTEGER	15	Fetches the linkage with the matching id.
serverId	INTEGER	15	Fetches all linkages with the matching serverId.
zoneId	INTEGER	15	Fetches all linkages with the matching zoneId.

Example URL	/api/v1/api.php?target=zoneLinkage&action=get&id=15

add

URL	/api/v1/api.php?target=zoneLinkage&action=add
Description	Adds a new link between a DNS Server and a Zone
Returns	**Examples:**

SUCCESSFUL:	{"success":1,"message":"Link Added."}
ERROR:	{"success":0, "message":"error message"}

Required Parameters

Name	Type	Example	Description
serverId	INTEGER	16	The DNS Server Id.
zoneId	INTEGER	105	The Zone Id.

serverSlave	BOOL	1	Whether or not this zone is a master or a slave on the linked server. Values are: 1 for slave, 0 for master.

Optional Parameters	None
Example URL	/api/v1/api.php?target=zoneLinkage&action=add&serverId=16&zoneId=105&serverSlave=0

delete

URL	/api/v1/api.php?target=zoneLinkage&action=delete
Description	Deletes a link between a DNS Server and a Zone
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Link Deleted."} ERROR: {"success":0, "message":"error message"}
Required Parameters	None
Optional Parameters	**Name** / **Type** / **Example** / **Description** id / INTEGER / 15 / Fetches the linkage with the matching id. serverId / INTEGER / 15 / Fetches all linkages with the matching serverId. zoneId / INTEGER / 15 / Fetches all linkages with the matching zoneId.
Example URL	/api/v1/api.php?target=zoneLinkage&action=delete

Name Server Control

get

URL	/api/v1/api.php?target=nameServer&action=get
Description	Fetches a list of all stored Name Servers
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":" Fetch Sucessful.","data":[{"id":"1","nameserver":"ns1.dns.6connect.net","add_to_zones_default":"1","ordering":"10","uses":"34"},{"id":"10","nameserver":"ns2.dns.6connect.net","add_to_zones_default":"1","ordering":"11","uses":"46"},{"id":"9","nameserver":"ns4.dns.6connect.net","add_to_zones_default":"1","ordering":"14","uses":"12"},{"id":"3","nameserver":"ns3.*

	dns.6connect.net"," add_to_zones_default":"1"," ordering":"15","uses":"46"}]}
ERROR:	*{"success":0, "message":"error message"}*

Data Detail:

Name	Type	Description
id	INTEGER	Server ID
nameserver	STRING	Server Name
add_to_zones_default	BOOL	Whether or not this is a default server.
ordering	INTEGER	Display order
uses	INTEGER	How many zones have been assigned to this NameServer

Required Parameters	None
Optional Parameters	

Name	Type	Example	Description
default	INTEGER	1	Set server as default

Example URL	/api/v1/api.php?target=nameServer&action=get&default=1

add

URL	/api/v1/api.php?target=nameServer&action=add
Description	Adds a new DNS Server
Returns	**Examples:**

SUCCESSFUL:	*{"success":1,"message":"Add Successful."}*
ERROR:	*{"success":0, "message":"error message"}*

Required Parameters

Name	Type	Example	Description
newServer	STRING	ns.yourdomain.com	Name of the NameServer

Optional Parameters	None
Example URL	/api/v1/api.php?target=nameServer&action=add&newServer=ns.yourdomain.com

delete

URL	/api/v1/api.php?target=nameServer&action=delete
Description	Deletes a NameServer
Returns	**Examples:**

SUCCESSFUL:	*{"success":1,"message":"Server Deleted."}*
ERROR:	*{"success":0, "message":"error message"}*

Required Parameters	**Name** / **Type** / **Example** / **Description** id / INTEGER / 5 / ID of server to delete.
Optional Parameters	None
Example URL	/api/v1/api.php?target=nameServer&action=delete&id=5

setDefault

URL	/api/v1/api.php?target=nameServer&action=setDefault
Description	Default NameServers have all new zones added to them as they are created. Multiple NameServers can be classified as Default.
Returns	**Examples:** SUCCESSFUL: *{"success":1, "message":"Success."}* ERROR: *{"success":0, "message":"error message"}*
Required Parameters	**Name** / **Type** / **Example** / **Description** id / INTEGER / 5 / ID of server to modify. value / INTEGER / 1 / 1 = Default, 0 = Normal
Optional Parameters	None
Example URL	/api/v1/api.php?target=nameServer&action=setDefault&id=3&value=1

orderUp

URL	/api/v1/api.php?target=nameServer&action=orderUp
Description	Swaps the index order of the targeted NameServer with that of the one above it.
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":"Reordering Successful."}* ERROR: *{"success":0, "message":"error message"}*
Required Parameters	**Name** / **Type** / **Example** / **Description** id / INTEGER / 5 / ID of server to modify.
Optional Parameters	None
Example URL	/api/v1/api.php?target=nameServer&action=orderUp&id=3

orderDown

URL	/api/v1/api.php?target=nameServer&action=orderDown
Description	Swaps the index order of the targeted NameServer with that of the one below it.

<table>
<tr><td>Returns</td><td>Examples:<table><tr><td>SUCCESSFUL:</td><td>{"success":1,"message":"Reordering Successful."}</td></tr><tr><td>ERROR:</td><td>{"success":0, "message":"error message"}</td></tr></table></td></tr>
<tr><td>Required Parameters</td><td><table><tr><th>Name</th><th>Type</th><th>Example</th><th>Description</th></tr><tr><td>id</td><td>INTEGER</td><td>5</td><td>ID of server to activate.</td></tr></table></td></tr>
<tr><td>Optional Parameters</td><td>None</td></tr>
<tr><td>Example URL</td><td>/api/v1/api.php?target=nameServer&action=orderDown&id=5</td></tr>
</table>

API Module - IPAM

- IP Address Management (IPv4 and IPv6)
  - IP Block Management
    - Get
    - Add
    - Update
    - Delete
    - Add Tag
    - Delete Tag
    - Smart Assign
    - Direct Assign
    - Unassign
    - Aggregate
    - Split
    - Scan Block
    - Get Scan Results
    - Get Options
    - Get VLAN
    - Get Resource Hierarchy
  - IP Subnets List
    - Get Subnet List
  - IP Tag List
    - Get Tags List
    - Add Tag To List
    - Delete Tag From List
  - IP Regions List
    - Get Regions List
    - Add Region To List
    - Get Utilization
    - Get Host Utilization
  - Holding Tank
    - Override Holding
    - Process Holding Tank
  - Reports
    - Mask Report
- IPAM SWIP Calls:
  - Deassign
  - Get RIR List
  - Simple Reassign
- IPAM API Calls Subject to Change:
  - Get Attribute List

IP Address Management (IPv4 and IPv6)

IP Block Management

Get	
URL	/api/v1/api.php?target=ipam&action=get
Description	Returns a list of IP blocks. Use optional parameters to filter the list. If multiple parameters are specified, only blocks matching all parameters will be returned.
Returns	**Examples:** SUCCESSFUL: *{ "success": 1, "message": "1 blocks found. ", "data": [{ "id": 5890, "type": "ipv4", "top_aggregate": null, "cidr": "192.168.0.0\/24", "formatted_ip": "192.168.0.0\/24", "address": "3232235520", "end_address": "3232235775", "mask": 24, "child1": null, "child2": null,*

	"is_assigned": 0, "is_swipped": 0, "is_aggregate": 1, "custid": 81, "resource_id": 81, "resource_name": "Available", "last_updated_time": null, "description": null, "parent": null, "rir": "1918", "lir_id": null, "notes": null, "generic_code": null, "code": null, "region": "SFO", "vlan": 100, "arin_net_id": null, "arin_cust_id": null, "org_id": null, "arin_swip_time": null, "assigned_time": null, "asn": null, "allowSubAssignments": false, "permissions": { "permissionIPAMRead": "1", "permissionIPAMUpdate": "1", "permissionIPAMCreate": "1", "permissionSWIP": "1", "permissionAdmin": "1" }, "range": "192.168.0.0 - 192.168.0.255", "tags": ["Customer", "PTP"] }] }
ERROR	*{'success':0, 'message':'error message'}*

Required Parameters: None

Optional Parameters

Name	Type	Example	Description
address	INTEGER	1125449728	IP address of the block in decimal format
asn	INTEGER	1000	Filters blocks based on their ASN
allowSubAssignments	BOOL	true	Filters blocks based on wether they allow sub-assignments or not. Acceptable values: "true" or "false"
block	STRING	213.37.29.0 /24	CIDR block description
code	STRING	Code X	User-defined block code as defined in Admin-IPAM settings: Generic Code Per Block Name
endAddress	INTEGER	1125453823	End IP address of the block in decimal format
id	INTEGER	1234	The ID of the block
includeAttributes	BOOL	true	Allows loading and

			display of the attributes for all of the blocks in the collection.
isAggregate	BOOL	true	Indicates if the block has been split into children or not. A value of 'true' will return blocks with no children.
isAssigned	BOOL	true	Acceptable values: "true" or "false"
isSwipped	BOOL	true	Acceptable values: "true" or "false"
lastUpdateTime	DATETIME	=2015-8-19 21:08:54	SQL Datetime format, prefaced by an "=" for exact time updated, "=>" for blocks updated after the given time, or "=<" for blocks updated before the given time.
lirId	INTEGER	101	The numeric ID of an LIR resource the block should be linked to
mask	INTEGER	24	Integer bitmask
notes	STRING	note123	Returns blocks with exact matches in the notes field against the provided "notes" string.
region	STRING	SFO	The value from the list of name /value pairs which make up the list of available regions
resourceHolderId	STRING	`cust-001`	(**Deprecated**: Use resourceQuery instea A custom ID which can be used to link resources in the 6Connect

			database back to your organization.
resourceId	INTEGER	1234	The ID of the resource the block is assigned to
resourceQuery	JSON	{"parent_id": 15}	A JSON object representing a valid resource query. Any parameters that can be used for a Resource GET API call can be used. Use of the resourceQuery param
rir	STRING	ARIN	Acceptable values: ARIN, RIPE, APNIC, AfriNIC, LACNIC, 1918
search	STRING	192.168	If a search term is provided, all IPAM fields including assigned Resource Holder name will be checked with a LIKE comparison to find matching blocks
selectCount	INTEGER	50	# of blocks to get
selectOffset	INTEGER	25	Offset for results set; useful for paging (e.g. selectCount = 50, selectOffset = 100 would return the 3rd page of 50 results)
sortField	STRING	cidr	Attribute to sort blocks by. Accepable values: cidr, mask, rir, vlan, code, updateTime
sortOrder	INTEGER	ASC	ASC or DESC
tags	STRING	customer,vpn	Comma-separated list of tags to filter by. If

				used in conjunction with 'search', performs the search operation and then filters results by the provided tag. Use with tagsMode to specify filter approach.
	tagsMode	STRING	"strict"," exclude", "intersection", or "union".	Denotes how the "tags" parameter is handled: "strict" - matches only blocks that have the exact set of tags of specified. "exclude" - matches only blocks which are note tagged with any of the blocks specified. "intersection" - matches any blocks which has all of the tags. "union" - matches all blocks which has any one of the tags. If not otherwise specified, tagsMode defaults to "intersection".
	topAggregateId	INTEGER	1234	The ID of the aggregate block to which the block belongs
	type	STRING	"ipv4" or "ipv6"	IP type
	vlan	INTEGER	123	VLAN for the block
Example URL	/api/v1/api.php?target=ipam&action=get&rir=ARIN&tags=customer,vpn			

Add	
URL	/api/v1/api.php?target=ipam&action=add

Description	Adds an IPv4 or IPv6 block

Returns

Examples:

SUCCESSFUL	*{"success":1,"message":"Block 192.168.0.0/24 (12345) added", "id":12345, "data":{ "id":12345, "cidr":192.168.0.0 /24", ...} }*
ERROR	*{ "success":0, "message":" error message" }*

Required Parameters

Name	Type	Example	Description
block	STRING	213.37.29.0 /24	CIDR block description
rir	STRING	ARIN	Acceptable values: ARIN, RIPE, APNIC, AfriNIC, LACNIC, 1918

Optional Parameters

Name	Type	Example	Description
allowDuplicate	BOOL	true	Allow the creation of duplicate blocks. The default behavior is to reject duplicates.
allowSubAssignments	BOOL	true	Does the block allow sub-assignments? If the block is assigned and allowSubAssignments Acceptable values: "true" or "false"
asn	INTEGER	1000	ASN for the block
code	STRING	Code X	User-defined block code as defined in Admin-IPAM settings: Generic Code Per Block Name
region	STRING	SFO	The value from the list of name /value pairs which make up the list of available regions
resourceId	INTEGER	1234	The ID of the resource the block is assigned to

tags	STRING	customer,vpn	Comma-separated list of tags
vlan	INTEGER	123	VLAN for the block

Example URL: /api/v1/api.php?target=ipam&action=add&block=213.37.29.0/24&rir=ARIN

Update

URL: /api/v1/api.php?target=ipam&action=update

Description: Updates detail data about an IP block.

Returns

Examples:

SUCCESSFUL	SINGLE BLOCK	*{"success":1," message":"Block 192.168.0.0/24 (12345) updated", "data":{ "id":12345, "cidr":192.168.0.0 /24", ...} }*
SUCCESSFUL	MULTIPLE BLOCKS	*{"success":1," message":"3 blocks updated", "data":[{ "id": 12345, "cidr": 192.168.0.0/24", ...}, {"id":12346, "cidr", "192.168.0.1 /32", ...}] }*
ERROR		*{ "success":0, "message":"error message" }*

Required Parameters

Name	Type	Example	Allow Multiple	Description
id*	INTEGER	125	Yes	ID of the IP block. Multiple block IDs can be specified in a comma-separated list.
block*	STRING	192.0.0.0 /24	Yes	CIDR or the block. Multiple CIDRs can be specified in a comma-separated list.
*Either block or id can be used, but only one must be provided				

Optional Parameters

Name	Type	Example	Description
allowSubAssignments	BOOL	true	Does the block allow sub-assignments? If the block is assigned and allowSubAssignments Acceptable values: "true" or "false"
asn	INTEGER	1000	ASN for the block
code	STRING	Code X	Arbitrary user-defined block code
lirId	INTEGER	101	The numeric ID of an LIR resource the block should be linked to
notes	STRING	Words	Misc. Notes
region	STRING	Chicago, IL	The region this IP block is assigned to.
propagate	BOOL	true	Propagates all attribute values to any smaller child blocks of the block being updated. ***Available in version 5.1.0***
rir	STRING	ARIN	Acceptable values: ARIN, RIPE, APNIC, AfriNIC, LACNIC, 1918
tags	STRING	Customer, vpn	Comma-separated list of tags
tags_action	STRING	replace	What action to take on the supplied tags. This action must be taken in conjunction with the tags parameter. Valid sett
vlan	NUMERIC	123	VLAN for the block

Example URL	/api/v1/api.php?target=ipam&action=update&block=192.0.0.0/24&notes=Notes_here

Delete

URL	/api/v1/api.php?target=ipam&action=delete

Description	Deletes an aggregate block
Returns	Examples SUCCESSFUL: *{"success":1,"message":" Aggregate deleted: 192.168.0.0/24", "data":{ "id": 12345, "cidr":192.168.0.0/24", ...} }* ERROR: *{ "success":0, "message":" error message" }*
Required Parameters	**Name** / **Type** / **Example** / **Description** block* / STRING / 213.37.29.0/24 / CIDR block description id* / INTEGER / 125 / ID of the IP block *Either block or id can be used, but only one must be provided
Optional Parameters	**Name** / **Type** / **Example** / **Description** force / BOOL / true / Forces the aggregate block to be deleted even if the block is split or contains sub blocks which are assigned. The default behavior is to reject attempts to delete blocks which have been split or are assigned.
Example URL	/api/v1/api.php?target=ipam&action=delete&block=213.37.29.0/24

Add Tag

URL	/api/v1/api.php?target=ipam&action=addTag
Description	Adds a tag to an IP block.
Returns	**Examples:** SUCCESSFUL: *{ "success":1,"message":"Tag Added.", "data":{ "id":12345, "cidr":192.168.0.0/24", ...} }* ERROR: *{ "success":0, "message":" error message" }*
Required Parameters	**Name** / **Type** / **Example** / **Description** id* / INTEGER / 125 / ID of the block block* / STRING / 192.0.0.0/24 / CIDR of the block *Either block or id can be used, but only one must be provided

	tag | STRING | Customer | The tag to add
Optional Parameters	None
Example URL	/api/v1/api.php?target=ipam&action=addTag&id=125&tag=Customer

Delete Tag

URL	/api/v1/api.php?target=ipam&action=deleteTag
Description	Removes a tag from an IP block.
Returns	**Examples:** SUCCESSFUL: *{ "success":1,"message":"Tag Removed.", "data":{ "id": 12345, "cidr":192.168.0.0/24", ...} }* ERROR: *{ "success":0, "message":" error message" }*
Required Parameters	**Name** | **Type** | **Example** | **Description** id* | INTEGER | 125 | ID of the block block* | STRING | 192.0.0.0/24 | CIDR of the block *Either block or id can be used, but only one must be provided tag | STRING | Customer | The tag to delete
Optional Parameters	None
Example URL	/api/v1/api.php? target=ipam&action=deleteTag&id=125&tag=Customer

Smart Assign

URL	/api/v1/api.php?target=ipam&action=smartAssign
Description	Selects a block based on supplied parameters (rir, tags, mask size, etc.) and assigns it to a Resource Holder.
Returns	**Examples:** SUCCESSFUL: *{ "success":1,"message":" Assigned 192.168.0.0/24 to Resource (1234) via Smart Assign", "id":12345, "data":{ "id":12345, "cidr":192.168.0.0 /24", ...} }* ERROR: *{ "success":0, "message":" error message" }*
Required Parameters	**Name** | **Type** | **Example** | **Description** mask | INTEGER | 24 | The size of the block to be assigned rir | STRING | ARIN | Acceptable values: ARIN, RIPE, APNIC,

			AfriNIC, LACNIC, 1918
resourceId*	INTEGER	1234	Integer ID of the resource to assign the block to
resourceQuery*	JSON	`{"custom_id":"cust-001"}`	A JSON object representing a valid resource query. Any parameters that can be used for a Resource GET API call can be used. Use of the resourceQuery param
*Either resourceId or resourceQuery can be used, but only one must be provided			
resourceHolderId	STRING	`cust-001`	(**Deprecated**: Use resourceQuery instea A custom ID which can be used to link resources in the 6Connect database back to your organization.
type	STRING	"IPv4" or "IPv6"	The type of block to assign

Optional Parameters*

Name	Type	Example	Description
assignedResourceId	STRING	"123" or "ignore"	The ID of the resource the block is assigned to, or the string "ignore". If assignedResourceId
code	STRING	Code X	Arbitrary user-defined block code
lirId	INTEGER	101	The ID of an LIR resource
region	STRING	Ashburn	Region to assign from
tags	STRING	customer,vpn	Comma separated string of tags. Matches blocks which have at least the set of tag specified by this parameter
tagsMode	STRING	"strict","	

		exclude", "intersection", or "union".	Denotes how the "tags" parameter is handled: "strict" - matches only blocks that have the exact set of tags of specified. "exclude" - matches only blocks which are note tagged with any of the blocks specified. "intersection" - matches any blocks which has all of the tags. "union" - matches all blocks which has any one of the tags. If not otherwise specified, tagsMode defaults to "intersection".
vlan	INTEGER	1023	VLAN designated to a given block

*Additional or fewer "optional" parameters may be required in order to result in a successful assignment, depending on the attributes of available blocks.

Example URL	/api/v1/api.php?target=ipam&action=smartAssign&mask=24&type=IPv4&resourceId=250

Direct Assign

URL	/api/v1/api.php?target=ipam&action=directAssign
Description	Assigns a block to an Resource Holder
Returns	**Examples:**

SUCCESSFUL	SINGLE BLOCK	*{ "success":1," message":" Assigned 192.168.0.0/24 to Resource (1234)", "id":12345, "data":{ "id":12345, "cidr": 192.168.0.0/24", ...} }*
SUCCESSFUL	MULTIPLE	*{ "success":1,"*

	BLOCKS	*message":" Assigned 5 blocks to Resource (1234) via Direct Assign", "data":{ "ids": [12345, 12346, 12347, ...] } }*
ERROR		*{ "success":0, "message":"error message" }*

Required Parameters

Name	Type	Example	Description
block*	STRING	213.37.29.0 /24	CIDR block description
id*	INTEGER	125	ID of the IP block, comma separated list of ids, or json encoded array of ids
*Either block or id can be used, but only one must be provided			
resourceHolderId**	STRING	`cust-001`	(**Deprecated**: Use resourceQuery instea A custom ID which can be used to link resources in the 6Connect database back to your organization.
resourceId**	INTEGER	1234	Integer ID of the resource to assign the block to
resourceQuery**	JSON	`{"custom_id":"cust-001"}`	A JSON object representing a valid resource query. Any parameters that can be used for a Resource GET API call can be used. Use of the resourceQuery param
**Either resourceId, resourceQuery, or resourceHolderId can be used, but only one must be provided			

Optional Parameters*

Name	Type	Example	Description
code	STRING	Code X	Arbitrary user-defined block code
lirId	INTEGER	101	The ID of an LIR resource

region	STRING	Ashburn	Region to assign from
rir	STRING	ARIN	Acceptable values: ARIN, RIPE, APNIC, AfriNIC, LACNIC, 1918
tags	STRING	customer,vpn	Comma separated string of tags. Matches blocks which have at least the set of tag specified by this parameter
tagsMode	STRING	"strict"," exclude", "intersection", or "union".	Denotes how the "tags" parameter is handled: "strict" - matches only blocks that have the exact set of tags of specified. "exclude" - matches only blocks which are note tagged with any of the blocks specified. "intersection" - matches any blocks which has all of the tags. "union" - matches all blocks which has any one of the tags. If not otherwise specified, tagsMode defaults to "intersection".
vlan	INTEGER	1023	VLAN designated to a given block

*Additional or fewer "optional" parameters may be required in order to result in a successful assignment, depending on the attributes of available blocks.

Example URL	/api/v1/api.php?target=ipam&action=directAssign&block=213.37.29.0/24&resourceId=1234

Unassign

URL	/api/v1/api.php?target=ipam&action=unassign
Description	Reclaims the specified block to be reassigned in the future
Returns	**Examples:** SUCCESSFUL: *{ "success":1,"message":" 192.168.0.0/24 unassigned", "id":12345, "data":{ "id": 12345, "cidr":192.168.0.0/24", ...} }* ERROR: *{ "success":0, "message":" error message" }*
Required Parameters	**Name** / **Type** / **Example** / **Description** block* / STRING / 213.37.29.0/24 / CIDR block description id* / INTEGER / 125 / ID of the IP block *Either block or id can be used, but only one must be provided
Optional Parameters	**Name** / **Type** / **Example** / **Description** skipHolding / BOOL / true / If set to true (skipHolding=true) th Acceptable values: "true" or "false"
Example URL	/api/v1/api.php?target=ipam&action=unassign&block=213.37.29.0/24

Aggregate

URL	/api/v1/api.php?target=ipam&action=aggregate
Description	Aggregates a selected block to the mask specified. If no mask specified, re-aggregates blocks to next parent. IE. calling aggregate on a /25 will aggregate both children back to the parent /24. All child blocks must be Available for aggregation to succeed.
Returns	**Examples:** SUCCESSFUL: `{"success":1,"message":"10.2.0.128\/25 aggregated into 10.2.0.0\/24","id":16326}` ERROR: *{'success':0, 'message':'error message'}*
Required Parameters	**Name** / **Type** / **Example** / **Description** id* / INTEGER / 125 / ID of the IP block. block* / STRING / 213.37.29.0/24 / CIDR block. *Either block or id can be used, but only one must be provided

Optional Parameters	

Name	Type	Example	Description
autoAggregateToMask	INTEGER	24	All blocks and IPs smaller than this netmask will be aggregated.
ignoreAssignments	BOOL	TRUE	If the ignoreAssignment fla

Example URL	/api/v1/api.php?target=ipam&action=aggregate&id=125&autoAggregateToMask=24

Split

URL	/api/v1/api.php?target=ipam&action=split
Description	Splits a selected block to the mask specified. If no mask specified, it split blocks to next child. IE. calling aggregate on a /24 will split both parent to the child /25s. All parent blocks must be Available, or have Allow Sub Assignments on for a split to succeed.
Returns	**Examples:**

SUCCESSFUL	`{"success":1,"message":"10.1.0.0\/24 split into 10.1.0.0\/25 and 10.1.0.128\/25","data":{"child1":23441,"child2":23451}}`
ERROR	*{'success':0, 'message':'error message'}*

Required Parameters

Name	Type	Example	Description
id*	INTEGER	125	ID of the IP block.
block*	STRING	213.37.29.0/24	CIDR block.
*Either block or id can be used, but only one must be provided			

Optional Parameters

Name	Type	Example	Description
autoSplitToMask	INTEGER	24	Auto aggregate the block back to this mask size. Note all blocks up this mask size must be Available or call will fail.
autoSplitLimit	INTEGER	4	A number the power of 2 (^2).

Example URL	/api/v1/api.php?target=ipam&action=split&block=213.37.29.0

	/24&autoSplitLimit=4

Scan Block

URL	/api/v1/api.php?target=ipam&action=scanBlock
Description	Initiates an asynchronous ping (ICMP) scan of the target block specified. Results of the scan can be checked with get.
Returns	**Examples:** SUCCESSFUL: `{"success":1,"message":"Ping scan started for 8.8.8.0\/27"}` ERROR: *{'success':0, 'message':'error message'}*
Required Parameters	**Name** / **Type** / **Example** / **Description** id* / INTEGER / 125 / ID of the IP block. block* / STRING / 213.37.29.0/24 / CIDR block. *Either block or id can be used, but only one must be provided
Optional Parameters	None
Example	/api/v1/api.php?target=ipam&action=scanBlock&block=213.37.29.0/24

Get Scan Results

URL	/api/v1/api.php?target=ipam&action=getScanResults
Description	Initiates an asynchronous ping (ICMP) scan of the target block specified. Results of the scan can be checked with get
Returns	**Examples:** SUCCESSFUL: `{"success":1,"data":{"block":"8.8.8.0\/27","date":"07\/14\/2014 11:07:10","data":[{"address":"8.8.8.8","host":"google-public-dns-a.google.com","status":"Up"}],"status":"completed","hostsScanned":4,"activeHosts":4}}` ERROR: *{'success':0, 'message':'error message'}*
Required Parameters	**Name** / **Type** / **Example** / **Description** block / STRING / 213.37.29.0/24 / CIDR block.
Optional Parameters	None
Example	/api/v1/api.php?target=ipam&action=getScanResults&block=213.37.29.0/24

Get Options

URL	/api/v1/api.php?target=ipam&action=getOptions
Description	Returns a list of options available for the block
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Options for 14.0.0.0\/25 (125)","options":{"actions": ["aggregate"],"templates": [{"name":"Auto Split","masks": [26,27,28,29,30,31,32]}]}} ERROR: *{'success':0, 'message':'error message'}*
Required Parameters	**Name** / **Type** / **Example** / **Description** id / INTEGER / 125 / ID of the IP block
Optional Parameters	None
Example URL	/api/v1/api.php?target=ipam&action=getOptions&id=125

Get VLAN

URL	/api/v1/api.php?target=ipam&action=getVlan
Description	Returns the VLAN for the block
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":" Found VLAN 1002 (14.0.0.0\ /25)","data":{"id":125,"type":" ipv4","top_aggregate":81532," cidr":"14.0.0.0\/25"," formatted_ip":"14.0.0.0\/25"," address":"234881024"," end_address":"234881151"," mask":25,"netmask":" 255.255.255.128","child1": null,"child2":null,"is_assigned": 0,"is_swipped":0," is_aggregate":1,"custid":402," resource_id":402," resource_name":"6connect Labz","last_updated_time":" 2015-01-22 12:30:37"," description":null,"parent": 81532,"rir":"ARIN","lir_id":" 1062","notes":"Test Notes"," generic_code":"Datacenter2"," code":"Datacenter2","region":" ATL","region_name":"Atlanta, GA","vlan":1002,"arin_net_id": null,"arin_cust_id":null,"org_id": null,"arin_swip_time":null," assigned_time":"2015-01-14 10:30:31","asn":"143"," allowSubAssignments":true," permissions": {"permissionIPAMRead":"1"," permissionIPAMUpdate":"1"," permissionIPAMCreate":"1"," permissionIPAMDelete":"1"," permissionSWIP":"1","

	permissionAdmin":"1"},"range":"14.0.0.0 - 14.0.0.127","tags":["Customer"]}}
ERROR	*{'success':0, 'message':'error message'}*

Required Parameters

Name	Type	Example	Description
id*	INTEGER	125	ID of the IP block
block*	STRING	213.37.29.0/24	CIDR block.
*Either block or id can be used, but only one must be provided			

Optional Parameters	None
Example URL	/api/v1/api.php?target=ipam&action=getVlan&id=125

Get Resource Hierarchy

URL	/api/v1/api.php?target=ipam&action=getResourceHierarchy&id=79124
Description	Returns the Parent Resource and Parent Resource id for the provided block.
Returns	**Examples:**

SUCCESSFUL	{"success":"1","data":[{"id":"4208","name":"a6connectEntry"}]}
ERROR	*{'success':0, 'message':'error message'}*

Required Parameters

Name	Type	Example	Description
id	INTEGER	125	ID of the IP block

Optional Parameters	None
Example URL	/api/v1/api.php?target=ipam&action=getResourceHierarchy&id=79124

IP Subnets List

Get Subnet List

URL	/api/v1/api.php?target=ipam&action=getSizes
Description	Returns the IPv4 or IPv6 Subnet Lists
Returns	**Examples:**

SUCCESSFUL	*{"success":1,"message":"Lookup was successful","data":["36","48","56","64","126","127","128"]}*
ERROR	*{'success':0, 'message':'error message'}*

Required Parameters

Name	Type	Example	Description
size	INTEGER	"4" or "6"	The IP Subnet list to return - "4" for IPv4 subnets, or "6" for IPv6 subnets.

Optional Parameters	None
Example URL	/api/v1/api.php?target=ipam&action=getSizes&size=6

IP Tag List

Get Tags List

URL	/api/v1/api.php?target=ipam&action=getTagList
Description	Returns a list of all valid IP Tags in the database.
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":"Tags Retrieved. ","data":["ActiveE DIA","Colo DIA","Customer","DSL","Fiber DIA","GPON DIA","Infra","Inhouse","LAN","LBIntf","POP","PTP","Prod","VPN","WEBHOST","Cable","Internal"]}* ERROR: *{'success':0, 'message':'error message'}*

Add Tag To List

URL	/api/v1/api.php?target=ipam&action=addTagToList
Description	Adds a tag to the IPAM tag list
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":"Tag Added."}* ERROR: *{'success':0, 'message':'error message'}*
Required Parameters	
Optional Parameters	None
Example URL	/api/v1/api.php?target=ipam&action=addTagToList&newTag=Loopback C

Name	Type	Example	Description
newTag	STRING	Loopback C	The value to add to the list of name /value pairs which make up the list of available regions

Delete Tag From List

URL	/api/v1/api.php?target=ipam&action=deleteTagFromList
Description	Deletes a tag from the IPAM tag list
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":"Tag Deleted."}* ERROR: *{'success':0, 'message':'error message'}*
Required Parameters	**Name** / **Type** / **Example** / **Description** tag / STRING / tag123 / The name of the tag to be deleted.
Optional Parameters	None
Example URL	/api/v1/api.php?target=ipam&action=deleteTagFromList&tag=tag123

IP Regions List

Get Regions List

URL	/api/v1/api.php?target=ipam&action=getRegionList
Description	Returns a list of all valid Regions in the database.
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":"Regions Retrieved.","data":[{"value":"ANY","name":"Any Region"},{"value":"ASH1","name":"Ashburn, VA"},{"value":"BOS","name":"Boston, MA"},{"value":"CHI","name":"Chicago, IL"},{"value":"DAL","name":"Dallas, TX"},{"value":"DEN","name":"Denver, CO"},{"value":"FRKT","name":"Frankfurt, DE"},{"value":"LON1","name":"London, UK"},{"value":"MIA","name":"Miami, FL"},{"value":"PAR","name":"Paris, FR"},{"value":"SFO","name":"San Francisco, CA"},{"value":"SEA","name":"Seattle, WA"},{"value":"Tokyo","name":"Tokyo"},{"value":"Singapore","name":"Singapore"},{"value":"Jakarta","name":"Jakarta"}]}* ERROR: *{'success':0, 'message':'error message'}*

Add Region To List

URL	/api/v1/api.php?target=ipam&action=addRegionToList
Description	Adds a region to the IPAM region list.
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":"Region Added."}*

	ERROR *{'success':0, 'message':'error message'}*
Required Parameters	**Name** / **Type** / **Example** / **Description** newRegion / STRING / SFO / The value to add to the list of name /value pairs which make up the list of available regions
Optional Parameters	None
Example URL	/api/v1/api.php?target=ipam&action=addRegionToList&newRegion=SFO

Utilization

Get Utilization	
URL	/api/v1/api.php?target=ipam&action=utilization
Description	Gets the utilization percentages for a specific ip block or ip block and mask combination.

Returns	**Examples:**
	SUCCESSFUL
	{
	"success": 1,
	"totalBlocks": 1,
	"totalHosts": "256",
	"hostsAssigned": 0,
	"hostsAllocated": "256",
	"hostsAvailable": "256",
	"hostsInHolding": 0,
	"availablePercentage": "100.00",
	"assignedPercentage": "0.00",
	"allocatedPercentage": "100.00",
	"inHoldingPercentage": "0.00",
	"resources": [{
	"id": 351,
	"name": "Customer 1",
	"type": "entry",
	"hosts": "256",
	"blocks": "1",
	"percentage": "100.00"

	}], "blocksAssigned":0, "blocksAllocated": 1, "blocksAvailable": "1", "blocksInHolding": null, "blocksAssignedPercentage": "0.00", "blocksAllocatedPercentage": "100.00", "blocksAvailablePercentage": "100.00", "blocksInHoldingPercentage": "0.00" }
ERROR	*{'success':0, 'message':'error message'}*

Required Parameters

Name	Type	Example	Description
block*	STRING	213.37.29.0/24	CIDR block description
id*	INTEGER	125	ID of the IP block
*Either block or id can be used, but only one must be provided			

Optional Parameters

Name	Type	Example	Description
mask	INTEGER	24	The specific mask size to retrieve utilization for. If using this parameter, the id parameter should be the id of the aggregate.

Example URL	/api/v1/api.php?target=ipam&action=utilization&id=125

Get Host Utilization

URL	/api/v1/api.php?target=ipam&action=getHostUtilization
Description	Gets the host utilization statistics with support for filters.
Returns	**Examples:**

SUCCESSFUL	{ "success": 1, "totalHosts": "256",

	"hostsAssigned": 0,
	"hostsAllocated": "256",
	"hostsAvailable": "256",
	"hostsInHolding": 0,
	"availablePercentage": "100.00",
	"assignedPercentage": "0.00",
	"allocatedPercentage": "100.00",
	"inHoldingPercentage": "0.00",
	"resources": [{
	"id": 351,
	"name": "Customer 1",
	"type": "entry",
	"hosts": "256",
	"blocks": "1",
	"percentage": "100.00"
	}]
	}

ERROR	*{'success':0, 'message':'error message'}*

Required Parameters

Name	Type	Example	Description
type	STRING	"ipv4" or "ipv6"	IP type

Optional Parameters

Name	Type	Example	Multiple Values	Description
code	STRING	"code-1"	Yes	User-defined block code as defined in Admin-IPAM settings: Generic Code Per Block Name
region	STRING	"SFO"	Yes	Region to assign from
rir	STRING	ARIN	No	Acceptable values
tags	STRING	"Customer"	Yes	Comma separated string of tags
vlan	INTEGER	1000	Yes	VLAN

	designated to a gi NOTE: to filter using multiple values, pass the values as a JSON-encoded string representation of an array. For example, to get utilization data for multiple tags, you could use the following URL: */api/v1/api.php?target=ipam&action=getHostUtilization&type=ipv4&tags=["Customer","PTP"]*
Example URL	/api/v1/api.php?target=ipam&action=getHostUtilization&type=ipv4&tags=["Customer","PTP"]&region=SMF

Holding Tank

Override Holding	
URL	/api/v1/api.php?target=ipam&action=processHoldingTank
Description	Overrides holding for a specific block, returning the block to available status
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":"15.15.15.128\/30 unassigned and returned to resource a6connectEntry","id":79134,"data":{"id":79134,"type":"ipv4","top_aggregate":79121,"cidr":"15.15.15.128\/30","formatted_ip":"15.15.15.128\/30","address":"252645248","end_address":"252645251","mask":30,"netmask":"255.255.255.252","child1":null,"child2":null,"is_assigned":0,"is_swipped":0,"is_aggregate":1,"custid":4208,"resource_id":4208,"resource_name":"a6connectEntry","last_updated_time":"2017-11-20 12:35:08","description":null,"parent":79132,"rir":"ARIN","lir_id":null,"notes":null,"generic_code":null,"code":null,"region":"STL","region_name":"St. Louis","vlan":null,"arin_net_id":null,"arin_cust_id":null,"org_id":null,"arin_swip_time":null,"assigned_time":"2017-11-20 12:35:08","asn":null,"allowSubAssignments":true,"permissions":{"permissionIPAMRead":"1","permissionIPAMUpdate":"1","permissionIPAMCreate":"1","permissionIPAMDelete":"1","permissionSWIP":"1","permissionAdmin":"1"},"range":"15.15.15.128 - 15.15.15.131","tags":[],"attributes":{"custody_chain":"[]"}}}* ERROR: *{'success':0, 'message':'error*

	message'}
Required Parameters	**Name** / **Type** / **Example** / **Description** block* / STRING / 213.37.29.0/24 / CIDR block description id* / INTEGER / 125 / ID of the IP block *Either block or id can be used, but only one must be provided
Optional Parameters	None.
Example URL	/api/v1/api.php?target=ipam&action=overrideHolding&block=213.37.29.0/24&preview=true

Process Holding Tank

URL	/api/v1/api.php?target=ipam&action=processHoldingTank
Description	Processes the Holding Tank, returning held blocks to available status
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"1 IPv4 and 0 IPv6 blocks would be moved to the available pool. ","data":[{"id":77712,"type":"ipv4","top_aggregate":77552,"cidr":"23.92.0.64\/26","formatted_ip":"23.92.0.64\/26","address":"391905344","end_address":"391905407","mask":26,"netmask":"255.255.255.192","child1":null,"child2":null,"is_assigned":0,"is_swipped":0,"is_aggregate":1,"custid":188,"resource_id":188,"resource_name":"6connect holding","last_updated_time":"2014-10-29 11:25:41","description":null,"parent":77682,"rir":"ARIN","lir_id":"451","notes":null,"generic_code":null,"code":null,"region":"PHX","region_name":"Phoenix, AZ","vlan":null,"arin_net_id":null,"arin_cust_id":null,"org_id":null,"arin_swip_time":null,"assigned_time":"2014-10-29 11:20:34","asn":null,"allowSubAssignments":false,"permissions":{"permissionIPAMRead":"1","permissionIPAMUpdate":"1","permissionIPAMCreate":"1","permissionIPAMDelete":"1","permissionSWIP":"1","permissionAdmin":"1"},"range":"23.92.0.64 - 23.92.0.127","tags":["Customer","DSL"]}]} ERROR: *{'success':0, 'message':'error message'}*
Required Parameters	None

Optional Parameters	<table><tr><th>Name</th><th>Type</th><th>Example</th><th>Description</th></tr><tr><td>preview</td><td>BOOL</td><td>true</td><td>Acceptable values: "true" or "false" If set to "true", returns a list of blocks that would be removed from the holding tank, but does not complete the process holding tank action. If set to "false", processes the holding tank and returns a list of blocks returned to available status.</td></tr></table>
Example URL	/api/v1/api.php? target=ipam&action=processHoldingTank&preview=true

Reports

Mask Report	
URL	/api/v1/api.php?target=ipam&action=maskReport
Description	Returns a JSON report breakdown of by-mask use statistics
Returns	**Examples:** <table><tr><td>SUCCESSFUL</td><td>{"success":1,"data":{"IPv4": {"total":576,"available":21," assigned":555,"holding":0," allocated":446,"prefixes":{"32": {"assigned_percentage": 91.666666," holding_percentage":0," available_percentage": 8.333333," allocated_percentage":0,"total": 12,"available":1,"assigned":11," holding":0,"allocated":0},"31": {"assigned_percentage":100," holding_percentage":0," available_percentage":0," allocated_percentage":30," total":20,"available":0," assigned":20,"holding":0," allocated":6},"30": {"assigned_percentage":50," holding_percentage":0," available_percentage":50," allocated_percentage":0,"total": 8,"available":4,"assigned":4," holding":0,"allocated":0},"29": {"assigned_percentage":100,"</td></tr></table>

	holding_percentage":0," available_percentage":0," allocated_percentage":20," total":40,"available":0," assigned":40,"holding":0," allocated":8},"28": {"assigned_percentage": 66.666666," holding_percentage":0," available_percentage": 33.333333," allocated_percentage": 33.333333,"total":48," available":16,"assigned":32," holding":0,"allocated":16},"27": {"assigned_percentage":100," holding_percentage":0," available_percentage":0," allocated_percentage":50," total":64,"available":0," assigned":64,"holding":0," allocated":32},"26": {"assigned_percentage":100," holding_percentage":0," available_percentage":0," allocated_percentage":100," total":128,"available":0," assigned":128,"holding":0," allocated":128},"25": {"assigned_percentage":100," holding_percentage":0," available_percentage":0," allocated_percentage":100," total":256,"available":0," assigned":256,"holding":0," allocated":256}}," assigned_percentage": 96.354166," holding_percentage":0," available_percentage": 3.645833," allocated_percentage": 77.430555},"IPv6":{"total": 16777216,"available":0," assigned":16777216,"holding": 0,"allocated":0,"prefixes":{"41": {"assigned_percentage":100," holding_percentage":0," available_percentage":0," allocated_percentage":0,"total": 16777216,"available":0," assigned":16777216,"holding": 0,"allocated":0}}," assigned_percentage":100," holding_percentage":0," available_percentage":0," allocated_percentage":0}}}
ERROR	*{ "success":0, "message":" error message" }*

Required Parameters	None
Optional Parameters	

Name	Type	Example	Description
resource_id	INTEGER	1234	Integer value of a Resource ID. If provided, the system will display an

				IPAM Mask Usage report limited to blocks associated with a single resource.
	tags	STRING	"Customer"	Comma separated list of tags. If provided, limits the IPAM Mask Usage report to certain tags (standard tag mode).
	regions	STRING	"DEN"	Comma separated list of regions. If provided, limits the IPAM Mask Usage Report to blocks associated with the provided region(s).
Example URL	/api/v1/api.php?target=ipam&action=maskReport&regions=DEN			

IPAM SWIP Calls:

Deassign

URL	/api/v1/api.php?target=ipam&action=deassign
Description	Performs a SWIP deassignment for the indicated IPAM block.
Returns	**Examples:** SUCCESSFUL: *{ "success":1,"message":"success message" }* ERROR: *{ "success":0, "message":"error message" }*

Required Parameters

Name	Type	Example	Description
blockId	INTEGER	1234	ID of the block to deassign
block	STRING	67.221.241.0/24	the CIDR of the block being de-assigned
resourceId	INTEGER	1234	ID of resource representing the customer to deassign
lirId	INTEGER	1234	The ProVision id

				of the LIR which is performing the de-assignment
	entityHandle	STRING	CONNE-81	The Org ID for the LIR.

Optional Parameters

Name	Type	Example	Description
netName	STRING	6CONN-67-221-241-0-24	Optional name for the network to override the default. The default net name will be created using the Net Name Prefix and IP address for the block.

Example URL: /api/v1/api.php?target=ipam&action=deassign&resourceId=1234&blockId=1234&lirId=123

Get RIR List

URL: /api/v1/api.php?target=ipam&action=getRIRList

Description: Returns a list of all valid RIRs in the database.

Returns

Examples:

SUCCESSFUL	*{"success":1,"message":"RIRs Retrieved.","data":[{"value":"ARIN","name":"ARIN"},{"value":"1918","name":"1918"},{"value":"AfriNIC","name":"AfriNIC"},{"value":"APNIC","name":"APNIC"},{"value":"LACNIC","name":"LACNIC"},{"value":"RIPE","name":"RIPE"},{"value":"1918-SJC","name":"1918-SJC"}]}*
ERROR	*{'success':0, 'message':'error message'}*

Simple Reassign

URL: /api/v1/api.php?target=ipam&action=simpleReassign

Description: ARIN SWIP - simple reassign. Creates an ARIN customer record for the assigned resource and reassigns the block to the ARIN customer record.

Returns

Examples:

SUCCESSFUL	*{ "success":1,"message":"Sent ARIN SWIP with action simpleReassign for 67.221.244.0/28 for Acme, Message: Success" }*
ERROR	*{ "success":0, "message":"*

	error message" }
Required Parameters	**Name** / **Type** / **Example** / **Description** blockId / INTEGER / 1234 / ID of the block to reassign resourceId / INTEGER / 1234 / ID of resource representing the customer to reassign to lirId / INTEGER / 1234 / The ProVision id of the LIR to use for reassignment entityHandle / STRING / CONNE-81 / The Org ID for the LIR.
Optional Parameters	**Name** / **Type** / **Example** / **Description** netName / STRING / NET-ACME-67-221-244-0-28 / Optional name for the network to override the default. The default net name will be created using the Net Name Prefix and IP address for the block.
Example URL	/api/v1/api.php?target=ipam&action=simpleReassign&resourceId=121&blockId=31559&lir

IPAM API Calls Subject to Change:

Calls below this point are subject to change, and are not recommended for use in production code.

Get Attribute List	
URL	/api/v1/api.php?target=ipam&action=getAttributeLists
Description	Returns a list of attributes
Returns	**Examples:** SUCCESSFUL: {"asns":[],"masks":["24"],"rirs":["1918"],"lirs":[],"tags":["DHCP"],"codes":[],"vlans":[],"regions":["Quito"],"resources":[{"id":"1302","name":"Quito Lab 1","slug":"quito-lab-1","type":"dhcp_pool","parent_id":"1","category_id":null,"attr":{"_dhcp_type":"subnet","_dhcp_pool_attributes":"{\"mac\":\"\",\"rangeStart\":\"10.8.0.0\",\"rangeEnd\":\"10.8.0.255\",\"freeLines\":0}","_dhcp_ip_id":"125"}}]} ERROR: *{'success':0, 'message':'error*

	message'}
Required Parameters	**Name** | **Type** | **Example** | **Description** id | INTEGER | 125 | ID of the IP block
Optional Parameters	None
Example URL	/api/v1/api.php?target=ipam&action=getAttributeLists&id=125

API Module - LIR

- LIR Management
  - Get
  - Delete
  - Add
  - Update

LIR Management

Get	
URL	/api/v1/api.php?target=lir&action=get
Description	Returns a list of LIRs

Returns	**Examples:**
	SUCCESSFUL
	{
	"success": 1,
	"message": "2 objects found",
	"data": [
	{
	"id": "100",
	"name": "RIPE Test LIR",
	"slug": "ripe-test-lir",
	"entities": [
	{
	"mnt_by": "mntner@email.com"
	"mnt_by_password": "password",
	"admin_c": "test-admin-c",
	"tech_c": "test-tech-c",
	"api_key": null
	}
	],
	"rir": "RIPE"
	},
	{
	"id": "101",
	"name": "ARIN Test LIR",
	"slug": "arin-test-lir",
	"entities": [
	{
	"org_handle": "TEST-10",
	"admin_poc": "TEST-ARIN",
	"net_poc": "TEST-ARIN",

<table>
<tr><td></td><td><table>
<tr><td></td><td><pre>"abuse_poc": "",

"net_name_prefix":
"PRFX",

"api_key": "API-XXXX-
YYYY-ZZZZ-1234"
                }
            ],
            "rir":
"ARIN",
            "asn":
"1000"
        }
    ]
}</pre></td></tr>
<tr><td>ERROR</td><td><pre>{
    "success":0,
    "message":"error
message"
}</pre></td></tr>
</table></td></tr>
<tr><td>Example URL</td><td>/api/v1/api.php?target=lir&action=get</td></tr>
</table>

Delete

<table>
<tr><td>URL</td><td>/api/v1/api.php?target=lir&action=delete&id=<ID></td></tr>
<tr><td>Description</td><td>Deletes an LIR</td></tr>
<tr><td>Returns</td><td>Examples:<table>
<tr><td>SUCCESSFUL</td><td><pre>{
    "success": 1,
    "message": "LIR deleted."
}</pre></td></tr>
<tr><td>ERROR</td><td><pre>{
    "success":0,
    "message":"error message"
}</pre></td></tr>
</table></td></tr>
<tr><td>Example URL</td><td>/api/v1/api.php?target=lir&action=delete&id=100</td></tr>
</table>

Add

<table>
<tr><td>URL</td><td>/api/v1/api.php?target=lir&action=add</td></tr>
<tr><td>Description</td><td>Adds a new LIR.

LIR creation utilizes the Resource system - see API Module - Resource - Add for details on adding a resource via the API.</td></tr>
<tr><td>Returns</td><td>Examples:<table>
<tr><td>SUCCESSFUL</td><td><pre>{
    "success":1,

 "message":"Resource
added",

 "data":</pre></td></tr>
</table></td></tr>
</table>

```
 {
 "id":14063,
 "name":"TestLIR",
 "slug":"testlir",
 "type":"entry",
 "parent_id":1,
 "category_id":null,
 "date":1517513319,
 "modified":1517513319,
 "attr":
 {
 "_section":"94",
 "rir":"ARIN",
 "asn":"20202",
 "org-handle":"org1",
 "admin-poc":"admin1",
 "net-poc":"tech1",
 "abuse-poc":"abuse1",
 "net-name-prefix":"
test-",
 "api-key":10,
 "mnt-by":"",
 "admin-c":"",
 "tech-c":"",
 "_gadget_data":false
 },
 "section":
 {
 "id":"94",
 "name":"LIR",
 "slug":"lir",
            "type":"
section",

"parent_id":"1",

"category_id":null,
            "date":
1375846025,
            "modified":
1375846025,
            "attr":[]
         },
         "gadgets":
false
    }
```

	}

ERROR	{
	"success":0,
	"message":"error message"
	}

Required Parameters

(meta[ ] =)

Name	Type	Example	Notes
meta[name]	STRING	TestLIR	Name of the new LIR resource.
meta[type]	STRING	entry	Type of resource - Adding a LIR will **always** be "entry".
meta[section]	STRING	lir	Section of the resource object. - Adding a LIR will **always** have "lir" as the section.

Fields Information

Detail on the "fields" array:		

fields[]	ARRAY	Entry field values (for fields that have already been assigned to the section) can be populated when the entry is created.
		
		The format is field [field-slug][field-instance]. If the field instance is left blank, it will simply be the next value in the instance array. For example:
		
		fields[network-fqdn][]=example.com &fields[network-fqdn][]=test.com
		
		would be written in JSON as
		
		`var fields = {`
		`"network-fqdn": [`
		`"example.com",`
		`"test.com"`
		`]`
		`}`

			Please note that the Field Slug might differ from the Field Name! To find the correct slug to use in adding resources with field values go the Section of the Resource you are adding, click 'Edit', then click the name of the Fields you will be populating. An Edit Field box will pop up which displays the Field's slug. A field can be added to a section multiple times. The field instance is used to keep track of which field occurrence we are referring. In this example, the network-fqdn field had been added twice to the section so we were able to store two values for it.

Required Parameters

(fields[ ][] =)

Name	Type	Example	Notes
fields[rir][]	STRING	ARIN	The RIR for the LIR. Accepted values are "ARIN", "RIPE", "LACNIC", "AfriNIC", "APNIC", and "1918".
fields[asn][]	STRING	20202	The ASN (Autonomous System Number) for the LIR.

Optional Parameters

(fields[ ][] =)

For LIRs with RIR = "ARIN":

Name	Type	Example	Notes
fields[org-handle][]	STRING	org1	Organization Handle. **NOTE:** When providing field information, Organization handle is **required** in order for the remaining

			fields to populate.
fields[admin-poc][]	STRING	admin1	The admin point of contact object name.
fields[net-poc][]	STRING	tech1	The network point of contact for this object.
fields[abuse-poc][]	STRING	abuse1	The abuse point of contact object name.
fields[net-name-prefix][]	STRING	test-	The network name prefix.
fields[api-key][]	STRING	akakakakakaka	API key used to interface with RIR services

For LIRs with RIR = "RIPE", "LACNIC", "AfriNIC", "APNIC", and "1918". :

Name	Type	Example	Notes
fields[mnt-by][]	STRING	maint1	The name of the maintainer object. **NOTE:** When providing field information, Maintainer name is **required** in order for the remaining fields to populate.
fields[mnt-by-password][]	STRING	passwordabcd	The maintainer password.
fields[admin-c][]	STRING	admin1	Administrative contac
fields[tech-c][]	STRING	tech1	Technical contact for this object

Example URL

ARIN example with only minimum required information:

/api/v1/api.php?target=lir&action=add&meta[name]=TestLIR1&meta[type]=entry&meta[section]=lir&fields[rir][]=ARIN&fields[asn][]=20202

ARIN example with full Organization field details:

/api/v1/api.php?target=lir&action=add&meta[name]=TestLIR2&meta[type]=entry&meta[section]=lir&fields[rir][]=ARIN&fields[asn][]

	=20202&fields[org-handle][]=org1&fields[admin-poc][]=admin1&fields[net-poc][]=tech1&fields[abuse-poc][]=abuse1&fields[net-name-prefix][]=test-&fields[api-key][]=akakakakakaka **Non-ARIN example with only minimum required information:** /api/v1/api.php?target=lir&action=add&meta[name]=TestLIR3&meta[type]=entry&meta[section]=lir&fields[rir][]=RIPE&fields[asn][]=20202 **Non-ARIN example with full Organization field details:** /api/v1/api.php?target=lir&action=add&meta[name]=TestLIR4&meta[type]=entry&meta[section]=lir&fields[rir][]=1918&fields[asn][]=20202&fields[mnt-by][]=maint1&fields[mnt-by-password][]=passwordabcd&fields[admin-c][]=admin1&fields[tech-c][]=tech1

Update	
URL	/api/v1/api.php?target=lir&action=update
Description	Updates a LIR. LIR updates utilize the Resource system - see API Module - Resource - Update for details on editing a resource via the API.
Returns	**Examples:** SUCCESSFUL { "success":1, "message":"Resource Updated", "data": { "id":"206", "name":"TestName", "slug":"dstl-reserved", "type":"entry", "parent_id":"191", "category_id":null, "date":1470755920, "modified":1517513903, "attr": { "mnt-by-password":11, "rir":"RIPE", "asn":"30303", "mnt-by":"Maint2", "admin-c":"admin4", "tech-c":"Tech5", "_section":"162", "_custom_id":"", "_gadget_data":[] },

<table>
<tr><td></td><td><table>
<tr><td></td><td><pre>"section":
{
"id":"162",
"name":"4BCH",
"slug":"4bch",
"type":"section",
"parent_id":"1",
"category_id":null,
"date":1468356788,
"modified":1468356788,
"attr":[]
},
"gadgets":[]
}
}</pre></td></tr>
<tr><td>ERROR</td><td><pre>{
"success":0,
"message":"error message"
}</pre></td></tr>
</table></td></tr>
<tr><td>Required Parameters
(meta[ ] =)</td><td><table>
<tr><th>Name</th><th>Type</th><th>Example</th><th>Notes</th></tr>
<tr><td>meta [id]</td><td>INTEGER</td><td>1234</td><td>ID of the LIR resource to update.</td></tr>
<tr><td>meta [type]</td><td>STRING</td><td>entry</td><td>Type of resource - Updating a LIR will always be "entry".</td></tr>
</table></td></tr>
<tr><td>Optional Parameters
(meta[ ] =)</td><td><table>
<tr><th>Name</th><th>Type</th><th>Example</th><th>Notes</th></tr>
<tr><td>meta[name]</td><td>STRING</td><td>TestLIR</td><td>Name of the LIR resource.</td></tr>
</table></td></tr>
<tr><td></td><td></td></tr>
<tr><td>Required Parameters
(fields[ ][] =)</td><td>None.</td></tr>
<tr><td>Optional Parameters
(fields[ ][] =)</td><td><table>
<tr><td colspan="4">For LIRs with RIR = "ARIN":</td></tr>
<tr><th>Name</th><th>Type</th><th>Example</th><th>Notes</th></tr>
<tr><td>fields[rir][]</td><td>STRING</td><td>ARIN</td><td>The RIR for the LIR. Accepted values are "ARIN",</td></tr>
</table></td></tr>
</table>

			"RIPE", "LACNIC", "AfriNIC", "APNIC", and "1918".
fields[asn][]	STRING	20202	The ASN (Autonomous System Number) for the LIR.
fields[org-handle][]	STRING	org1	Organization Handle. **NOTE:** When providing field information, Organization handle is **required** in order for the remaining Org. object fields to populate.
fields[admin-poc][]	STRING	admin1	The admin point of contact object name.
fields[net-poc][]	STRING	tech1	The network point of contact for this object.
fields[abuse-poc][]	STRING	abuse1	The abuse point of contact object name.
fields[net-name-prefix][]	STRING	test-	The network name prefix.
fields[api-key][]	STRING	akakakakakaka	API key used to interface with RIR services

For LIRs with RIR = "RIPE", "LACNIC", "AfriNIC", "APNIC", and "1918". :

Name	Type	Example	Notes
fields[rir][]	STRING	RIPE	The RIR for the LIR. Accepted values are "ARIN", "RIPE", "LACNIC", "AfriNIC", "APNIC", and "1918".
fields[asn][]	STRING	20202	The ASN (Autonomous System Number) for the LIR.

	fields[mnt-by][]	STRING	maint1	The name of the maintainer object. **NOTE:** When providing field information, Maintainer name is **required** in order for the remaining maintainer object fields to populate.
	fields[mnt-by-password][]	STRING	passwordabcd	The maintainer password.
	fields[admin-c][]	STRING	admin1	Administrative contac
	fields[tech-c][]	STRING	tech1	Technical contact for this object
Example URL	**Update example with only the minimum required information:** This sends only the meta data required for a success response, but it does not actually change any values. /api/v1/api.php?target=lir&action=update&meta[id]=14068&meta[type]=entry **Update example with changing RIR, ASN, and updating all field information:** This example changes a non-ARIN LIR to an ARIN LIR, and provides all updated information for the new RIR type. /api/v1/api.php?target=lir&action=update&meta[id]=14068&meta[type]=entry&meta[name]=TestLIR4b&fields[rir][]=ARIN&fields[asn][]=30303&fields[org-handle][]=OrgABC&fields[admin-poc][]=AdminABC&fields[tech-c][]=TechABC&fields[abuse-poc][]=AbuseABC&fields[net-name-prefix][]=TestABC-&fields[api-key][]=abcdabcd			

API Module - Peering

- Peering
  - getCommunications
  - getPeers
  - getRequests
  - getSessions
  - addSession
  - configureSession
  - deleteSession
  - updateSession
  - resestPeerStatus
  - sendRequest
  - sendEmail
  - updatePeer

Peering

getCommunications	
Base URL	/api/v1/api.php?target=peering&action=getCommunications
Description	Returns all communication data on peers at a particular exchange.
Returns	**Examples:**

SUCCESSFUL	{"success":1,"message":"8 records found.","data":[{"name":"1&1 Internet","asn":"8560","request_status":null,"qualified":null,"is_peer":false,"id":"262","log_data":[]},{"name":"Akamai Technologies","asn":"20940","request_status":null,"qualified":null,"is_peer":false,"id":"2","log_data":[]},{"name":"Amazon.com","asn":"16509","request_status":null,"qualified":null,"is_peer":false,"id":"1418","log_data":[]},{"name":"Atrato IP Networks","asn":"5580","request_status":null,"qualified":null,"is_peer":false,"id":"2283","log_data":[]},{"name":"OVH","asn":"16276","request_status":null,"qualified":null,"is_peer":false,"id":"1264","log_data":[]},{"name":"Verisign","asn":"7342","request_status":null,"qualified":null,"is_peer":false,"id":"873","log_data":[]},{"name":"Walmart.com","asn":"17374","request_status":null,"qualified":null,"is_peer":false,"id":"627","log_data":[]},{"name":"Windstream Communications","asn":"7029","request_status":null,"qualified":null,"is_peer":false,"id":"1820","log_data":[]},{"name":"Yahoo!","asn":"10310","request_status":null,"qualified":null,"is_peer":false,"id":"27","log_data":[]},{"name":"Zayo","asn":"19092","request_status":null,"qualified":null,"is_peer":false,"id":"3611","

	log_data":[]},{"name":"Zayo (Abovenet Communications Inc.)","asn":"6461","request_status":null,"qualified":null,"is_peer":false,"id":"541","log_data":[]},{"name":"tw telecom","asn":"4323","request_status":null,"qualified":null,"is_peer":false,"id":"540","log_data":[]}]}
ERROR	*{'success':0, 'message':'error message'}*

Required Parameters

Name	Type	Example	Description
public_id	INTEGER	1	The unique numerical identifier of the exchange to retrieve peering communicaiton recor

Optional Parameters	None
Example URL	/api/v1/api.php?target=peering&action=getCommunications&public_id=1

getPeers

URL	/api/v1/api.php?target=peering&action=getPeers
Description	Returns a list of all peers available at an exchange
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"184 peers found.","data":[{"id":"262","public_id":"1","asn":"8560","name":"1&1 Internet","qualified":true,"is_peer":0,"request_status":"sent","info_prefixes":"150","public_ips":[],"contacts":[],"log_data":[]},"id":"286","public_id":null,"asn":"3856","name":"Packet Clearing House","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":"100","public_ips":[],"contacts":[],"log_data":[]},{"id":"890","public_id":null,"asn":"13768","name":"PEER 1 Hosting","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":"3000","public_ips":[],"contacts":[],"log_data":[]},{"id":"1676","public_id":null,"asn":"3737","name":"PenTeleData","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":"80","public_ips":[],"contacts":[],"log_data":[]},{"id":"1560","public_id":null,"asn":"23265","name":"Pocketinet Communications, Inc.","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":"200","public_ips":[],"contacts":[],"log_data":[]},{"id":"576","public_id":null,"asn":"32787","name":"Prolexic","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":"500","public_ips":[],"contacts":[],"log_data":[]},"id":"353","public_id":null,"asn":"27524","name":"Xeex Communications","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":"3000","public_ips":[],"contacts":[],"log_data":[]},{"id":"27","public_id":null,"asn":"10310","name":"Yahoo!","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":"500","public_ips":[],"contacts":[],"log_data":[]},{"id":"1751","public_id":null,"asn":"13238","name":"Yandex LLC","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":"60","public_ips":[],"contacts":[],"log_data":[]},{"id":"4078","public_id":null,"asn":"19468","name":"YieldBuild Inc","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":"5","public_ips":[],"contacts":[],"log_data":[]},{"id":"541","public_id":null,"asn":"6461","name":"Zayo (Abovenet Communications Inc.)","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":"20000","public_ips":[],"contacts":[],"log_data":[]},{"id":"2351","public_id":null,"asn":"14824","name":"Zynga Game Network, Inc.","qualified":true,"is_peer":0,"request_status":null,"info_prefixes":null,"public_ips":[],"contacts":[],"

log_data":[]}}
ERROR: {"success":1,"message":"No peers found."}

Required Parameters: None

Optional Parameters

Name	Type	Example	Description
public_id	INTEGER	1	The unique numerical identifier of the exchange to retrieve peering communication record
id	INT	1	The unique numerical identifier of the peer in peeringDB.
asn	INT	4436	
name	STRING	GTT	
aka	STRING	nLayer	
website	STRING	http://www.gt-t.net	
notes_public	STRING		
notes_private	STRING		
irr_as_set	STRING	AS-NLAYER	
info_traffic	ENUM	1 Tbps+	enum('Not Disclosed','0-20 Mbps','20-100Mbps','100-1000M
info_ratio	ENUM	Mostly Outbound	enum('Not Disclosed','Heavy Ou
info_scope	ENUM	Global	enum('Not Disclosed','Regional',
info_type	ENUM	NSP	enum('Not Disclosed','NSP','Con
info_prefixes	INT	10000	
info_lookingglass	STRING	http://lg.nlayer.net/	
info_routeserver	STRING	telnet://route-server.nlayer.net	
info_unicast	CHAR	1	
info_multicast	CHAR		
info_ipv6	CHAR	1	
policy_url	STRING	http://www.gt-t.net/Peering_policies.aspx	
policy_general	ENUM	Selective	enum ('Open','Selective','Re
policy_locations	ENUM	Required - International	enum('Not Required','Preferred',
policy_ratio	ENUM	No	enum ('Yes','No')

			DEFAULT NULL
policy_contracts	ENUM	Not Required	enum('Not Required','Private On
policy_nopublic	ENUM	N	enum('Y','N') NOT NULL DEFAULT 'N'
policy_noprivate	ENUM	N	enum('Y','N') NOT NULL DEFAULT 'N'
date_created	DATETIME	2013-03-21 15:36:42	Date the peeringdb entry was created
date_lastupdated	DATETIME	2013-03-21 15:36:42	Date the peeringdb entry was last updated
include_public_ips	BOOL	TRUE	Returns a list of all public facing IPs
include_contacts	BOOL	TRUE	Returns a list of all contacts associated with peer(s)
include_log_data	BOOL	TRUE	Returns a list of all log data associated with the peer (s) (use with care)

Example URL	/api/v1/api.php?target=peering&action=getPeers&public_id=1

getRequests

URL	/api/v1/api.php?target=peering&action=getRequests
Description	Returns a list of all peering requests issued
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"1 request found.","data":[{"id":"131","public_id":"5","source_participant_id":"2335","source_asn":"8038","peer_participant_id":"1418","peer_asn":"16509","peer_name":"Amazon.com","email_type":"text","email_from":"ops@6connect.com","email_to":"nalinmk@gmail.com","subject":"Peering request from 6connect, Inc.","body":"Peering, \r\n\r\n6connect, Inc., 8038, would like to peer with Amazon.com at our common locations.\r\n\r\nFacility, IP Address\r\nEquinix Ashburn - 206.126.236.68\r\nEquinix Palo Alto - 198.32.176.36\r\nEquinix Ashburn - 206.126.236.35\r\nEquinix San Jose - 206.223.116.177 \r\nLINX Juniper LAN - 195.66.225.175\r\n\r\nSincerely, \r\nOperations\r\nops@6connect.com\r\n\r\n\r\n\r\n6connect, Inc. information:\r\nEquinix Palo Alto, 2001:504:d::33\r\nEquinix Palo Alto, 198.32.176.51\r\n\r\nPeeringDB: http:\/\/as8038.peeringdb.com\/\r\n","status":null,"created":"2014-04-23 10:31:33","modified":"2014-04-23 10:31:33"}]} ERROR: {"success":1,"message":"No request found.","data":[]}
Required Parameters	None
Optional Parameters	

Name	Type	Example	Description
peer_participant_id		1	The numerical id of

	INTEGER		the peer

Example URL	/api/v1/api.php?target=peering&action=getRequests& peer_participant_id=1

getSessions

URL	/api/v1/api.php?target=peering&action=getSessions
Description	Returns a list of all bgp peering sessions
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"1 sessions found.","data":[{"id":"51","source_asn":"32787","source_ipaddr":"1.2.3.4","resource_id":"422","resource_name":"ar1.inoc.com","peer_asn":"20940","peer_name":"Akamai Technologies","peer_participant_id":"2","peer_ipaddr":"206.126.236.102","peer_hostname":null,"peer_group":"group b","public_id":"1","public_name":"Equinix Ashburn","ip_type":"ipv4","type":"Peer","state":"not configured","prfx_max":"20","prfx_received":null,"password":"0","note":null},]} ERROR: {"success":1,"message":"No peers found."}

Required Parameters

Name	Type	Example	Description
public_id	INTEGER	1	The unique numerical identifier of the exchange to retrieve peering communicaiton recor

Optional Parameters

Name	Type	Example	Description
id	INTEGER	41	
public_id	INTEGER		
source_asn	INTEGER		
source_ipaddr	STRING		
resource_id	INTEGER		
peer_asn	INTEGER		
peer_name	STRING		
peer_participant_id	INTEGER		
peer_ipaddr	STRING		
peer_hostname	STRING		
peer_group	STRING		
password	INTEGER		
type	STRING		
state	STRING		
prfx_max	INTEGER		
prfx_received	INTEGER		
ip_type	ENUM		enum('ipv4','ipv6') NOT NULL DEFAULT 'ipv4'
note	STRING		

created	TIMESTAMP		
modified	TIMESTAMP		
deleted	INTEGER		
public_id	INTEGER		

Example URL: /api/v1/api.php?target=peering&action=getPeers&public_id=1

addSession

URL: /api/v1/api.php?target=peering&action=addSession

Description: Adds a bgp session

Returns:

Examples:
SUCCESSFUL: {"success":1,"message":"Session added: Amazon.com (AS8038\/1.2.3.5) - (AS16509\/206.126.236.68)","data":{"id":111,"source_asn":"8038","source_ipaddr":"1.2.3.5","resource_id":"422","resource_name":null,"peer_asn":"16509","peer_name":"Amazon.com","peer_participant_id":"1418","peer_ipaddr":"206.126.236.68","peer_hostname":null,"peer_group":"LAME-PEERS","public_id":"1","public_name":null,"ip_type":"ipv4","type":"Peer","state":"not configured","prfx_max":"200","prfx_received":null,"password":"ace12345","note":"I'm a fancy note."}}

ERROR: {"success":1,"message":"No request found.","data":[]}

Required Parameters: None

Optional Parameters:

Name	Type	Example	Description
source_asn	INTEGER	1	The numerical id of the peer
source_ipaddr	STRING		
resource_id	INTEGER		
peer_asn	INTEGER		
peer_name	STRING		
peer_participant_id	INTEGER		
peer_ipaddr	STRING		
peer_hostname	STRING		
peer_group	STRING		
public_id			
type	STRING		
ip_type	ENUM		enum('ipv4','ipv6')
state	STRING		
prfx_max	INTEGER		
note	STRING		

Example URL: /api/v1/api.php?target=peering&action=getRequests&peer_participant_id=1

configureSession

URL: /api/v1/api.php?target=peering&action=configureSession

Description	Configure a BGP session on the router
Returns	**Examples:** SUCCESSFUL: ERROR: {"success":0,"message":"Unable to authenticate "}
Required Parameters	

Name	Type	Example	Description
session_id	INTEGER	1	

Optional Parameters

Name	Type	Example	Description
session_id	INTEGER	1	The numerical id of the peer
source_ipaddr	STRING		
resource_id	INTEGER		
peer_asn	INTEGER		
peer_name	STRING		
peer_participant_id	INTEGER		
peer_ipaddr	STRING		
peer_hostname	STRING		
peer_group	STRING		
public_id			
type	STRING		
ip_type	ENUM		enum('ipv4','ipv6')
state	STRING		
prfx_max	INTEGER		
note	STRING		

Example URL	/api/v1/api.php? target=peering&action=configureSession&session_id=51&username=&co

deleteSession

URL	/api/v1/api.php?target=peering&action=deleteSession
Description	Delete sessions matching criteria
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"1 sessions deleted."} ERROR: {"success":0,"message":"No sessions found to delete."}
Required Parameters	None
Optional Parameters	

Name	Type	Example	Description
id	INTEGER	41	
public_id	INTEGER		
source_asn	INTEGER		
source_ipaddr	STRING		
resource_id	INTEGER		

	peer_asn	INTEGER		
	peer_name	STRING		
	peer_participant_id	INTEGER		
	peer_ipaddr	STRING		
	peer_hostname	STRING		
	peer_group	STRING		
	password	INTEGER		
	type	STRING		
	state	STRING		
	prfx_max	INTEGER		
	prfx_received	INTEGER		
	ip_type	ENUM		enum('ipv4','ipv6') NOT NULL DEFAULT 'ipv4'
	note	STRING		
	created	TIMESTAMP		
	modified	TIMESTAMP		
	deleted	INTEGER		
	public_id	INTEGER		
Example URL	/api/v1/api.php?target=peering&action=deleteSession&id=171			

updateSession

URL	/api/v1/api.php?target=peering&action=updateSession
Description	Updates session values with any new values specified
Returns	**Examples:** SUCCESSFUL:{"success":1,"message":"Session updated: 123.net (AS32787\/1.2.3.4) - (AS12129\/206.126.236.70)","data":{"id":"41","source_asn":"32787","source_ipaddr":"1.2.3.4","resource_id":"422","resource_name":"ar1.inoc.com","peer_asn":"12129","peer_name":"123.net","peer_participant_id":"3899","peer_ipaddr":"206.126.236.70","peer_hostname":null,"peer_group":"group a","public_id":"1","public_name":"Equinix Ashburn","ip_type":"ipv4","type":"Peer","state":"not configured","prfx_max":"10","prfx_received":null,"password":"0","note":"Adding an awesome note."}} ERROR:
Required Parameters	None
Optional Parameters	

Name	Type	Example	Description
id	INTEGER	41	
public_id	INTEGER		
source_asn	INTEGER		
source_ipaddr	STRING		
resource_id	INTEGER		
peer_asn	INTEGER		
peer_name	STRING		

	peer_participant_id	INTEGER		
	peer_ipaddr	STRING		
	peer_hostname	STRING		
	peer_group	STRING		
	password	INTEGER		
	type	STRING		
	state	STRING		
	prfx_max	INTEGER		
	prfx_received	INTEGER		
	ip_type	ENUM		enum('ipv4','ipv6') NOT NULL DEFAULT 'ipv4'
	note	STRING		
	created	TIMESTAMP		
	modified	TIMESTAMP		
	deleted	INTEGER		
	public_id	INTEGER		

Example URL	/api/v1/api.php?target=peering&action=updateSession&note=Adding+an+awesome+note

resestPeerStatus

URL	/api/v1/api.php?target=peering&action=resetPeerStatus
Description	
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"1&1 Internet status reset","data":{"id":"262","public_id":"1","asn":"8560","name":"1&1 Internet","qualified":true,"is_peer":0,"request_status":"none","info_prefixes":null,"public_ips":[],"contacts":[],"log_data":[{"message":"Peer status reset","time":"2014-05-22 23:14:54","request_id":null,"session_id":null,"public_id":"1"},{"message":"Peer status reset","time":"2014-05-22 23:14:18","request_id":null,"session_id":null,"public_id":"1"},{"message":"Session deleted: 1&1 Internet (AS32787\/1.2.3.4) - (AS8560\/206.126.236.200)","time":"2014-05-22 22:39:43","request_id":null,"session_id":"71","public_id":"1"},{"message":"Request sent: ","time":"2014-04-12 13:24:43","request_id":"121","session_id":null,"public_id":"1"},{"message":"Session added: 1&1 Internet (AS32787\/1.2.3.4) - (AS8560\/206.126.236.200)","time":"2014-04-07 11:32:37","request_id":null,"session_id":"71","public_id":"1"}]}} ERROR: {"success":0,"message":"Could not find peer matching parameters"}
Required Parameters	**Name** / **Type** / **Example** / **Description** participant_id / INTEGER / 262 / The id of the peer in from the peeringDB peerParticipants table public_id / INTEGER / 1 / The id of the exchange point from the peeringDB

				mgmtPublics table.

Optional Parameters	None
Example URL	/api/v1/api.php?target=peering&action=resetPeerStatus&participant_id=262&public_id=1

sendRequest

URL	/api/v1/api.php?target=peering&action=sendRequest
Description	Send a peering request (email) to a prospective peer. This will be deprecated in the next version for a simpler call, strongly suggest against using.
Returns	**Examples:** SUCCESSFUL: {"success":1,"message":"Request sent","data":{"id":"922","public_id":"1","asn":"10933","name":"ATX Communications, Inc.","qualified":true,"is_peer":0,"request_status":"sent","info_prefixes":null,"public_ips":[],"contacts":[],"log_data":[{"message":"Request sent to ","time":"2014-05-27 16:59:01","request_id":"181","session_id":null,"public_id":"1"},{"message":"Request sent to ","time":"2014-05-27 16:49:30","request_id":"171","session_id":null,"public_id":"1"}]}} ERROR: {"success":0,"message":"Internal error"}

Required Parameters

Name	Type	Example	Description
public_id	INTEGER		
peer_participant_id	INTEGER		
source_participant_id	INTEGER		
peer_name	STRING		
peer_asn	INTEGER		
email_from	STRING	262	
email_to	STRING	1	
subject	STRING		
body	STRING		
type	ENUM	html	enum('text','html')
status	ENUM	sent	enum ('sent','accepted','rejecte

Optional Parameters	None
Example URL	https://ops.6connect.com/peering-demo/api/v1/api.php?target=peering&action=sendRequest&peer_participant_id=1909&peer_as&public_id=1&type=text&email_from=ops%406connect.com&email_to=operations%40as28929.net&cc=&bcc=&subject=Peering+request+from+6connect%2C+Inc.&body=%0D%0APeering%2C%0D%0A%0D%0A6connect%2C+Inc.%2C+8038%2C+would+like+to+peer+with+ASDASD+srl+at+our+common+locations.%

sendEmail

URL	/api/v1/api.php?target=peering&action=sendEmail
Description	Send a peering request (email) to a prospective peer. This will be deprecated in the next version for a simpler call, strongly suggest

	against using.
Returns	**Examples:** SUCCESSFUL: ERROR:
Required Parameters	<table><tr><th>Name</th><th>Type</th><th>Example</th><th>Description</th></tr><tr><td>participant_id</td><td>INTEGER</td><td>262</td><td></td></tr><tr><td>public_id</td><td>INTEGER</td><td>1</td><td></td></tr></table>
Optional Parameters	None
Example URL	

updatePeer	
URL	/api/v1/api.php?target=peering&action=updatePeer
Description	
Returns	**Examples:** SUCCESSFUL: ERROR:
Required Parameters	<table><tr><th>Name</th><th>Type</th><th>Example</th><th>Description</th></tr><tr><td>participant_id</td><td>INTEGER</td><td>262</td><td></td></tr><tr><td>public_id</td><td>INTEGER</td><td>1</td><td></td></tr></table>
Optional Parameters	None
Example URL	

API Module - Resource

- Resources
  - get
  - add
  - update
  - delete
  - get resource link
  - get resource search

Resources

get	
URL	/api/v1/api.php?target=resource&action=get
Description	Get a resource or resources
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":" Search successful","data":[{"id":"57","name":" 2nd Email","slug":"6c-contact-email2","type":"field","parent_id":"1","category_id":null,"attr":[]}]}* ERROR: *{"success":0,"message":"Search failed"}*
Optional Parameters	

Name	Type	Notes/Example
name	STRING	Name of the resource. Example: 6Connect, Inc.
slug	STRING	The unique URL friendly name of the resource. Example: 6connect-inc
type	STRING	Type of resource (eg. *entry*, *field*, *category*)

At most, one of the following:

Name	Type	Notes /Example
id	INTEGER	Get the resource which has this ID
resource__in	ARRAY	Get any resource which has any of these IDs Syntax: & resource__in []=1771& resource__in []=14238 (Each resource id you wish to

		search over gets its own phrase.)
resource__not_in	ARRAY	Get all the resources which do not have any of these IDs Syntax: &resource__not_in[]=1771&resource__not_in[]=14238 (Each resource id you wish to exclude gets its own phrase.)

At most, one of the following:

Name	Type	Notes /Example
parent_id	INTEGER	Get the resources whose parent has this ID
parent__in	ARRAY	Get any resource whose parents have any of these IDs. Syntax: &parent__in[]=162&parent__in[]=299 (Each parent id you wish to search over gets its own phrase.)
parent__not_in	ARRAY	Get all resources whose parents do not have any of these IDs Syntax: &parent__not_in[]=1771&parent__not_in[]=14238 (Each parent id you wish to exclude

		gets its own phrase.)

At most, one of the following:

Name	Type	Notes /Example
category_id	INTEGER	Get the resources of the category that has this ID
category__in	ARRAY	Get any resources whose categories have any of these IDs. Syntax: &category__in[]=11002&category__in[]=11003 (Each category id you wish to search over gets its own phrase.)
category__not_in	ARRAY	Get the resources of all the categories that do not have any of these IDs Syntax: &parent__not_in[]=11002&parent__not_in[]=11003 (Each category id you wish to exclude gets its own phrase.)

You can set the order of the results by setting the STRING value of the parameter **orderby** to one of the following :

- none
- id
- name *(default)*
- slug
- type
- parent_id
- date
- resource__in *(preserve order given in the resource__in array)*

You can set the direction of the ordering of the results by setting the STRING value of the parameter **order** to one of the following :

- ASC *(default)*
- DESC

You can further limit the results based on attributes the resources may have:

Name	Type	Notes /Example
attr_key	STRING	The name of the attribute. Example: network-fqdn
attr_value	STRING	The value of any attribute, or if attr_key is specified, the value of the attribute defined in attr_key.
attr_compare	STRING	If both attr_key and attr_value are given, the results are by default compared based on the value given as attr_value being equal to the value stored in the database. You can optionally change this by setting the STRING value of attr_compare to one of the following: • = *(default)* • != • > • >= • < • <=

		• LIKE • NOT LIKE • IN • NOT IN • BETWEEN • NOT BETWEEN

When attr_compare is set to IN, NOT IN, BETWEEN, NOT BETWEEN, then attr_value must either be an array or a comma separated string.

You can search on multiple attributes by including an array of attribute options:

Name	Type	Notes /Example

attributes	ARRAY	`var data = {`
		
		`"type: "entry",`
		`"attributes": [`
		`{`
		`"attr_key": "_section",`
		`"attr_value": "105",`
		`},`
		`{`
		`"attr_key": "address-mail-state",`
		`"attr_value": "CA",`
		`}`
		`],`
		
		`"resources_per_page: 10`
		`}`

You can restrict the range of the resources returned.

Name	Type	Notes /Example
resources_per_page	INTEGER	How many resources to return.
offset	INTEGER	How many resources to offset (the initial resource is 0, not 1).

	paged	INTEGER	The page to return (starts at 1, not 0). This parameter is provided for convenience and is used to calculate the offset where: offset=(pag

Example URL	/api/v1/api.php?target=resource&action=get&id=7

add

URL	/api/v1/api.php?target=resource&action=add
Description	Add a resource.
Returns	**Examples:** */api/v1/api.php?target=resource&action=add&meta[name]=apitest&meta[type]=entry&meta[section]=firewall&fields[network-fqdn][]=www.example.com* SUCCESSFUL: {"success":1,"message":"Resource added","data":{"id":1077,"name":"apitest","slug":"apitest","type":"entry","parent_id":1,"category_id":"NULL","attr":{"_section":"70","network-fqdn":"www.example.com"},"section":{"id":"70","name":"Firewall","slug":"firewall","type":"section","parent_id":"1","category_id":null,"attr":{}}}} */api/v1/api.php?target=resource&action=add&meta[name]=apitest&meta[type]=entry&fields[network-fqdn][]=www.example.com* ERROR:{"success":0,"message":"Entries must be assigned to a section"}
Required Parameters	**Name** / **Type** / **Notes/Example** meta [name] / STRING / Name of the resource meta[type] / STRING / Type of resource (entry, section, field, etc.)
Optional Parameters	**Name** / **Type** / **Notes/Example** meta[parent_id] / INTEGER / ID of the parent resource meta[category_id] / INTEGER / ID of the category
Required Parameters (meta[type] = entry)	One of the following: **Name** / **Type** / **Notes/Example** meta [section_id] / INTEGER / ID of the section that the entry will be assigned to meta [section] / STRING / Slug of the section that the entry will be assigned to
Optional Parameters (meta[type] = entry)	**Name** / **Type** / **Notes/Example** fields[] / ARRAY / Entry field values (for fields that have

		already been assigned to the section) can be populated when the entry is created.
		
		The format is field [field-slug][field-instance]. If the field instance is left blank, it will simply be the next value in the instance array. For example:
		
		fields[network-fqdn][]=example.com&fields[network-fqdn][]=test.com
		
		would be written in JSON as
		
		`var fields = {`
		`"network-fqdn": [`
		`"example.com",`
		`"test.com"`
		`]`
		`}`
		
		Please note that the Field Slug might differ from the Field Name! To find the correct slug to use in adding resources with field values go the Section of the Resource you are adding, click 'Edit', then click the name of the Fields you will be populating. An Edit Field box will pop up which displays the Field's slug.
		
		A field can be added to a section multiple times. The field instance is used to keep track of which field occurrence we are referring. In this example, the network-fqdn field had been added twice to the section so we were able to store two values for it.

meta[custom_id]	STRING	A custom ID for the entry. In the past

			this has been called the Resource Holder ID or Customer ID. Most recently it was implemented as a text field with the slug "6c-resourceholder-id." Now it is a fundamental part the entry type resources.

Required Parameters

(meta[type] = field)

Name	Type	Notes/Example
meta[field_type]	STRING	Type of field • text • textarea • radios • checkboxes • choicebox

Optional Parameters

(meta[type] = field)

Name	Type	Notes/Example
meta[help_block]	STRING	Fields can have a line of text under them with instructions
meta[options]	ARRAY	Fields of type radios, checkboxes, or choicebox can have multiple options. This could be multiple radio buttons or a choicebox (dropdown) with several options. For example: meta[type] =field&meta[name] =Colors&meta [field_type] =choicebox&meta [options][] =Blue&meta [options][]=Green Will create a choicebox with dropdown options of Blue and Green.

Required Parameters

(meta[type] = gadgets)

Name	Type	Notes/Example
gadgets[x][uuid]	INTEGER	x: The nth gadget being described in the call ('0' for the first gadget, '1' for the second, and so on).

		uuid: User-generated ID of the gadget to be created.
gadgets[x][code]	STRING	x: The nth gadget being described in the call ('0' for the first gadget, '1' for the second, and so on). code: Slug of the gadget code to be created. List of valid Gadget codes: • Contact Info: "_contact_info" • Contacts: "_contacts" • DHCP Server: "_dhcp_server" • DNS: "_dns" • Document Storage: "_document_storage" • IPAM: "_ipam" • Peer Groups: "_peering_peer_groups" • Peering Sessions: "_peering_sessions" • VRFs: "_peering_vrfs" • Resource Linkage: "_resource_linkage" • Resource View: "_resource_view" • Reverse API Console: "_reverse_api" • Tech Info: "_tech_info"

Example - Adding the IPAM Gadget to a Section:

api.php?target=resource&action=add&meta[type]=section&meta[name]=TestSection_1&meta[parent_id]=1&gadgets[0][uuid]=uuid-586dbd260d6ef&gadgets[0][code]=_ipam

update	
URL	/api/v1/api.php?target=resource&action=update
Description	Update a resource.
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":"Resource Updated","data":{"id":"1055","name":"87-child-1","slug":"87-child-1","type":"entry","parent_id":"87","category_id":"65","attr":{"_section":"70"},"section":{"id":"70","name":"Firewall","slug":"firewall","type":"section","parent_id":"1","category_id":null,"attr":{}}}}* ERROR: *{"success":0,"message":"No resource found with ID: 1079"}*
Required Parameters	

Name	Type	Notes/Example

meta[id]	INTEGER	ID of resource
meta [type]	STRING	Type of resource (entry, section, field, ect)

Optional Parameters

(meta[type] = entry)

Name	Type	Notes/Example
fields[]	ARRAY	See "add" documentation

Optional Parameters

(meta[type] = section)

Name	Type	Notes/Example
fields[]	ARRAY	The fields value should be all the fields that are assigned to the section. Giving an empty array as the fields value will remove all fields from the section. The format is: fields[position][key] The position value is the position that the field will appear in (0 is first). The position value must always be included. An example field format for an existing field could be: fields[0][id]=2 fields[0][slug]=asset-serial-number fields[0][help_block]=something fields[0][new]=false ■ Either the id or the slug is required, not both. ■ When the "new" parameter is not included, FALSE is assumed If you want to create a new field and assign it to the section, use a format like this: fields[10][name]=TextArea fields[10][field_type]=textarea fields[10][new]=true

delete

URL	/api/v1/api.php?target=resource&action=delete
Description	Delete a resource.
Returns	**Examples:** SUCCESSFUL: *{"success":1,"message":"Resource deleted."}* ERROR: *{"success":0,"message":"No resource found with ID: 57"}*
Required Parameters	**Name** | **Type** | **Notes/Example** id | INTEGER | ID of the resource
Optional Parameters	**Name** | **Type** | **Notes/Example** recursive | BOOL | When 1, deletes parent and child entries for the resource A recursive delete will delete all resources, which are permitted to be deleted, from the bottom up. Imagine the following hierarchy: A B1 B2 C11 C12 C21 C22 If a recursive delete is performed on A, but C21 is not deletable, the following resources would still be deleted: (B1, C11, C12, C22). B2 would not be deleted because it depends on C21 and A would not be deleted because it depends on B2.
Example URL	/api/v1/api.php?target=resource&action=delete&id=57

get resource link

Description	Get available resource links. If no resource links exist for the given resource, an empty object is returned.
URL	/api/v1/api.php?target=resource&action=getLink
Returns	**Examples:** SUCCESSFUL: {"success":1 ,"message":" Search successful", "data": {"meta":{"totalRecords":"3", "retrieved":3}, "0":{"id":"22", "resource_id1":"1292", "resource_id2":"1302", "relation":"dhcpPoolLink"}, "1":

	{"id":"2", "resource_id1":"1292", "resource_id2":"1452", "relation":"dhcpPoolLink"}, "2":{"id":"12", "resource_id1":"1422", "resource_id2":"1482", "relation":"dhcpPoolLink"}}}
ERROR:	{"success":0, "message":"error message"}

Return Detail:

Name	Type	Description
id	INTEGER	Id of the resource linkage
resource_id1	INTEGER	The id of the parent resource
resource_id2	INTEGER	The id of the linked resource
relation	STRING	The relation type. Relation types include: contact, dhcpPoolLink, dns dnsZoneMaster, dnsZoneServer, dnsZoneV

Meta Attributes:

Name	Type	Description
totalRecords	INTEGER	How many records were found by this query, without pagination.
retrieved	INTEGER	How many records were returned by this query, with pagination.

Optional Attributes:

Name	Type	Description
resultsPerPage	INTEGER	How many records to include per page display.*
page	INTEGER	Which page to display, when used with "resultsPerPage"*
*Example pagination: api.php?target=resource&action=getLink&relation=dhcpPoolLink&resultsPerPag		

get resource search

Description	Search the resource system for the provided term. Performs a "LIKE" search to return similar results.
URL	/api/v1/api.php?target=resource&action=get&search=
Returns	**Examples:**

SUCCESSFUL:	{"success":1,"message":" Search successful","data": [{"id":"11011","name":" a6connectchildentry","slug":" a6connectchildentry","type":" entry","parent_id":"4210"," category_id":null,"date": 1499106555,"modified": 1499106555,"attr":{"_section":" 4214"},"section":{"id":"4214"," name":"aQA Section","slug":" aqa-section","type":"section"," parent_id":"1","category_id": null,"date":1498775688," modified":1499106630,"attr": []},"gadgets":[]}],"result_count": 1,"found_count":1}
ERROR:	{"success":0, "message":"error message"}

Return Detail:

Name	Type	Description
id	INTEGER	Id of the resource linkage
name	STRING	The resource name
slug	STRING	The resource slug
type	STRING	The resource type.
parent_id	INTEGER	ID of the parent resource
category_id	INTEGER	ID of the resource category type
date	INTEGER	Resouce creation date
modified	INTEGER	Resource last modified date
attr	JSON	A JSON list of resource attributes

Required Attributes:

Name	Type	Description
search	STRING	The search term

Example URL	/api/v1/api.php?target=resource&action=get&search=6connect

API Module - VLAN

- getById
- getDomains
- enable
- addDomain
- getAvailable
- get
- deleteDomain
- updateDomain
- update
- removeTags
- addTags
- smartAssign

getById

URL	api/v1/api.php?target=vlan&action=getById
Description	get information of a vlan
Returns	Examples SUCCESSFUL: {"success":1,"message":"1 VLAN found. ","data":[{"id":"11190","vlan":"176","domain_id":"1","resource_id":null,"name":"","domain_name":"DefaultDomain","tags":[],"tagsString":[]}]} ERROR: *{ "success":0, "message":" error message" }*
Required Parameters	**Name** / **Type** / **Example** / **Description** id / INTEGER / 11190 / ID of the VLAN
Optional Parameters	None
Example URL	api/v1/api.php?target=vlan&action=getById&id=11190

getDomains

URL	api/v1/api.php?target=vlan&action=getDomains
Description	get list and info of all domains or a specific domain. When specifying the domain the results will include tag info
Returns	Examples SUCCESSFUL: {"success":1,"message":"6 Domains found. ","data":[{"id":"1","domain":"DefaultDomain","attr_6c":{"range":"S","restricted":[1,1002,1003,1004,1005]},"vlans":[{"id":"22866","vlan":"10","name":""},{"id":"1","vlan":"101","name":""},{"id":"3","vlan":"102","name":""},{"id":"22865","vlan":"104","name":""},

	{"id":"4","vlan":"105","name":"NewName"},{"id":"5","vlan":"106","name":"ThisIsAName"},{"id":"2","vlan":"107","name":""},{"id":"11189","vlan":"175","name":"SomeWhoCallMeTim"},{"id":"11190","vlan":"176","name":""},{"id":"11191","vlan":"177","name":""},{"id":"11192","vlan":"178","name":""},{"id":"7","vlan":"996","name":""},{"id":"8","vlan":"997","name":""},{"id":"9","vlan":"998","name":""},{"id":"10","vlan":"999","name":"CORP-DMZ"},{"id":"6","vlan":"1021","name":""}],"vlans_unnamed":null,"type":"domain","util":{"S":{"used":15,"total":1000}}},{"id":"2","domain":"6connect.com","attr_6c":{"range":"S","restricted":[1,1002,1003,1004,1005]},"vlans":[{"id":"19","vlan":"10","name":""},{"id":"20","vlan":"11","name":""}, ... {"id":"12755","vlan":"70","name":""},{"id":"12756","vlan":"71","name":""},{"id":"12757","vlan":"72","name":""},{"id":"12758","vlan":"73","name":""},{"id":"12759","vlan":"74","name":""},{"id":"12760","vlan":"75","name":""}],"vlans_unnamed":null,"type":"domain","util":{"S":{"used":38,"total":1000},"E":{"used":0,"total":3089}}}]}
ERROR	*{ "success":0, "message":" error message" }*

Required Parameters	None
Optional Parameters	**Name** / **Type** / **Example** / **Description**: id / INTEGER / 2 / ID of the domain
Example URL	api/v1/api.php?target=vlan&action=getDomains&id=2

enable

URL	api/v1/api.php?target=vlan&action=enable
Description	enable vlans in a domain
Returns	Examples

SUCCESSFUL	{"success":1,"message":" Success. ","data":"3 in request (0 were found already named\ /available and 3 found as

	unavailable). 3 VLANs have been updated to unnamed available entries."}
ERROR	*{ "success":0, "message":" error message" }*

Required Parameters	

Name	Type	Example	Description
domainId	INTEGER	1	the domain ID in the domain table
vlan	INTEGER or STRING	18,19,20	vlan for this domain that we want enabled

Optional Parameters	None
Example URL	api/v1/api.php?target=vlan&action=enable&domainId=1&vlan=18,19,20

addDomain

URL	api/v1/api.php?target=vlan&action=addDomain
Description	add a domain
Returns	Examples

SUCCESSFUL	{"success":1,"message":" domain 10 added","id":10," data":[{"id":"10","domain":" sdfasfaf","attr_6c":{"range":" S","restricted": [1,1002,1003,1004,1005]}," vlans":[],"vlans_unnamed":{"1": null,"2":null,"3":null,"4":null,"5": null,"6":null,"7":null,"8":null,"9": null,"10":null,"11":null,"12": null,"13":null,"14":null,"15": null,"16":null,"17":null,"18": null,"19":null,"20":null,"21": null,"22":null,"23":null,"24": null,"25":null,"26":null,"27": null,"28":null,"29":null,"30": null,"31":null,"32":null,"33": null,"34":null,"35":null,"36": null,"37":null,"38":null,"39": null,"40":null,"41":null,"42": null,"43":null,"44":null,"45": null,"46":null,"47":null,"48": null,"49":null,"50":null,"51": null,"52":null,"53":null,"54": null,"55":null,"56":null,"57": null,"58":null,"59":null,"60": null,"61":null,"62":null,"63": null,"64":null,"65":null,"66": null,"67":null,"68":null,"69": null,"70":null,"71":null,"72": null,"73":null,"74":null,"75": null,"76":null,"77":null,"78": null,"79":null,"80":null,"81": null,"82":null,"83":null,"84": null,"85":null,"86":null,"87": null,"88":null,"89":null,"90": null,"91":null,"92":null,"93": null,"94":null,"95":null,"96": null,"97":null,"98":null,"99": null,"100":null,"101":null,"102": null,"103":null,"104":null,"105": null,"106":null,"107":null,"108":

null,"109":null,"110":null,"111":
null,"112":null,"113":null,"114":
null,"115":null,"116":null,"117":
null,"118":null,"119":null,"120":
null,"121":null,"122":null,"123":
null,"124":null,"125":null,"126":
null,"127":null,"128":null,"129":
null,"130":null,"131":null,"132":
null,"133":null,"134":null,"135":
null,"136":null,"137":null,"138":
null,"139":null,"140":null,"141":
null,"142":null,"143":null,"144":
null,"145":null,"146":null,"147":
null,"148":null,"149":null,"150":
null,"151":null,"152":null,"153":
null,"154":null,"155":null,"156":
null,"157":null,"158":null,"159":
null,"160":null,"161":null,"162":
null,"163":null,"164":null,"165":
null,"166":null,"167":null,"168":
null,"169":null,"170":null,"171":
null,"172":null,"173":null,"174":
null,"175":null,"176":null,"177":
null,"178":null,"179":null,"180":
null,"181":null,"182":null,"183":
null,"184":null,"185":null,"186":
null,"187":null,"188":null,"189":
null,"190":null,"191":null,"192":
null,"193":null,"194":null,"195":
null,"196":null,"197":null,"198":
null,"199":null,"200":null,"201":
null,"202":null,"203":null,"204":
null,"205":null,"206":null,"207":
null,"208":null,"209":null,"210":
null,"211":null,"212":null,"213":
null,"214":null,"215":null,"216":
null,"217":null,"218":null,"219":
null,"220":null,"221":null,"222":
null,"223":null,"224":null,"225":
null,"226":null,"227":null,"228":
null,"229":null,"230":null,"231":
null,"232":null,"233":null,"234":
null,"235":null,"236":null,"237":
null,"238":null,"239":null,"240":
null,"241":null,"242":null,"243":
null,"244":null,"245":null,"246":
null,"247":null,"248":null,"249":
null,"250":null,"251":null,"252":
null,"253":null,"254":null,"255":
null,"256":null,"257":null,"258":
null,"259":null,"260":null,"261":
null,"262":null,"263":null,"264":
null,"265":null,"266":null,"267":
null,"268":null,"269":null,"270":
null,"271":null,"272":null,"273":
null,"274":null,"275":null,"276":
null,"277":null,"278":null,"279":
null,"280":null,"281":null,"282":
null,"283":null,"284":null,"285":
null,"286":null,"287":null,"288":
null,"289":null,"290":null,"291":
null,"292":null,"293":null,"294":
null,"295":null,"296":null,"297":
null,"298":null,"299":null,"300":
null,"301":null,"302":null,"303":
null,"304":null,"305":null,"306":
null,"307":null,"308":null,"309":
null,"310":null,"311":null,"312":
null,"313":null,"314":null,"315":
null,"316":null,"317":null,"318":
null,"319":null,"320":null,"321":
null,"322":null,"323":null,"324":
null,"325":null,"326":null,"327":
null,"328":null,"329":null,"330":
null,"331":null,"332":null,"333":

null,"334":null,"335":null,"336":
null,"337":null,"338":null,"339":
null,"340":null,"341":null,"342":
null,"343":null,"344":null,"345":
null,"346":null,"347":null,"348":
null,"349":null,"350":null,"351":
null,"352":null,"353":null,"354":
null,"355":null,"356":null,"357":
null,"358":null,"359":null,"360":
null,"361":null,"362":null,"363":
null,"364":null,"365":null,"366":
null,"367":null,"368":null,"369":
null,"370":null,"371":null,"372":
null,"373":null,"374":null,"375":
null,"376":null,"377":null,"378":
null,"379":null,"380":null,"381":
null,"382":null,"383":null,"384":
null,"385":null,"386":null,"387":
null,"388":null,"389":null,"390":
null,"391":null,"392":null,"393":
null,"394":null,"395":null,"396":
null,"397":null,"398":null,"399":
null,"400":null,"401":null,"402":
null,"403":null,"404":null,"405":
null,"406":null,"407":null,"408":
null,"409":null,"410":null,"411":
null,"412":null,"413":null,"414":
null,"415":null,"416":null,"417":
null,"418":null,"419":null,"420":
null,"421":null,"422":null,"423":
null,"424":null,"425":null,"426":
null,"427":null,"428":null,"429":
null,"430":null,"431":null,"432":
null,"433":null,"434":null,"435":
null,"436":null,"437":null,"438":
null,"439":null,"440":null,"441":
null,"442":null,"443":null,"444":
null,"445":null,"446":null,"447":
null,"448":null,"449":null,"450":
null,"451":null,"452":null,"453":
null,"454":null,"455":null,"456":
null,"457":null,"458":null,"459":
null,"460":null,"461":null,"462":
null,"463":null,"464":null,"465":
null,"466":null,"467":null,"468":
null,"469":null,"470":null,"471":
null,"472":null,"473":null,"474":
null,"475":null,"476":null,"477":
null,"478":null,"479":null,"480":
null,"481":null,"482":null,"483":
null,"484":null,"485":null,"486":
null,"487":null,"488":null,"489":
null,"490":null,"491":null,"492":
null,"493":null,"494":null,"495":
null,"496":null,"497":null,"498":
null,"499":null,"500":null,"501":
null,"502":null,"503":null,"504":
null,"505":null,"506":null,"507":
null,"508":null,"509":null,"510":
null,"511":null,"512":null,"513":
null,"514":null,"515":null,"516":
null,"517":null,"518":null,"519":
null,"520":null,"521":null,"522":
null,"523":null,"524":null,"525":
null,"526":null,"527":null,"528":
null,"529":null,"530":null,"531":
null,"532":null,"533":null,"534":
null,"535":null,"536":null,"537":
null,"538":null,"539":null,"540":
null,"541":null,"542":null,"543":
null,"544":null,"545":null,"546":
null,"547":null,"548":null,"549":
null,"550":null,"551":null,"552":
null,"553":null,"554":null,"555":
null,"556":null,"557":null,"558":

null,"559":null,"560":null,"561":
null,"562":null,"563":null,"564":
null,"565":null,"566":null,"567":
null,"568":null,"569":null,"570":
null,"571":null,"572":null,"573":
null,"574":null,"575":null,"576":
null,"577":null,"578":null,"579":
null,"580":null,"581":null,"582":
null,"583":null,"584":null,"585":
null,"586":null,"587":null,"588":
null,"589":null,"590":null,"591":
null,"592":null,"593":null,"594":
null,"595":null,"596":null,"597":
null,"598":null,"599":null,"600":
null,"601":null,"602":null,"603":
null,"604":null,"605":null,"606":
null,"607":null,"608":null,"609":
null,"610":null,"611":null,"612":
null,"613":null,"614":null,"615":
null,"616":null,"617":null,"618":
null,"619":null,"620":null,"621":
null,"622":null,"623":null,"624":
null,"625":null,"626":null,"627":
null,"628":null,"629":null,"630":
null,"631":null,"632":null,"633":
null,"634":null,"635":null,"636":
null,"637":null,"638":null,"639":
null,"640":null,"641":null,"642":
null,"643":null,"644":null,"645":
null,"646":null,"647":null,"648":
null,"649":null,"650":null,"651":
null,"652":null,"653":null,"654":
null,"655":null,"656":null,"657":
null,"658":null,"659":null,"660":
null,"661":null,"662":null,"663":
null,"664":null,"665":null,"666":
null,"667":null,"668":null,"669":
null,"670":null,"671":null,"672":
null,"673":null,"674":null,"675":
null,"676":null,"677":null,"678":
null,"679":null,"680":null,"681":
null,"682":null,"683":null,"684":
null,"685":null,"686":null,"687":
null,"688":null,"689":null,"690":
null,"691":null,"692":null,"693":
null,"694":null,"695":null,"696":
null,"697":null,"698":null,"699":
null,"700":null,"701":null,"702":
null,"703":null,"704":null,"705":
null,"706":null,"707":null,"708":
null,"709":null,"710":null,"711":
null,"712":null,"713":null,"714":
null,"715":null,"716":null,"717":
null,"718":null,"719":null,"720":
null,"721":null,"722":null,"723":
null,"724":null,"725":null,"726":
null,"727":null,"728":null,"729":
null,"730":null,"731":null,"732":
null,"733":null,"734":null,"735":
null,"736":null,"737":null,"738":
null,"739":null,"740":null,"741":
null,"742":null,"743":null,"744":
null,"745":null,"746":null,"747":
null,"748":null,"749":null,"750":
null,"751":null,"752":null,"753":
null,"754":null,"755":null,"756":
null,"757":null,"758":null,"759":
null,"760":null,"761":null,"762":
null,"763":null,"764":null,"765":
null,"766":null,"767":null,"768":
null,"769":null,"770":null,"771":
null,"772":null,"773":null,"774":
null,"775":null,"776":null,"777":
null,"778":null,"779":null,"780":
null,"781":null,"782":null,"783":

null,"784":null,"785":null,"786":
null,"787":null,"788":null,"789":
null,"790":null,"791":null,"792":
null,"793":null,"794":null,"795":
null,"796":null,"797":null,"798":
null,"799":null,"800":null,"801":
null,"802":null,"803":null,"804":
null,"805":null,"806":null,"807":
null,"808":null,"809":null,"810":
null,"811":null,"812":null,"813":
null,"814":null,"815":null,"816":
null,"817":null,"818":null,"819":
null,"820":null,"821":null,"822":
null,"823":null,"824":null,"825":
null,"826":null,"827":null,"828":
null,"829":null,"830":null,"831":
null,"832":null,"833":null,"834":
null,"835":null,"836":null,"837":
null,"838":null,"839":null,"840":
null,"841":null,"842":null,"843":
null,"844":null,"845":null,"846":
null,"847":null,"848":null,"849":
null,"850":null,"851":null,"852":
null,"853":null,"854":null,"855":
null,"856":null,"857":null,"858":
null,"859":null,"860":null,"861":
null,"862":null,"863":null,"864":
null,"865":null,"866":null,"867":
null,"868":null,"869":null,"870":
null,"871":null,"872":null,"873":
null,"874":null,"875":null,"876":
null,"877":null,"878":null,"879":
null,"880":null,"881":null,"882":
null,"883":null,"884":null,"885":
null,"886":null,"887":null,"888":
null,"889":null,"890":null,"891":
null,"892":null,"893":null,"894":
null,"895":null,"896":null,"897":
null,"898":null,"899":null,"900":
null,"901":null,"902":null,"903":
null,"904":null,"905":null,"906":
null,"907":null,"908":null,"909":
null,"910":null,"911":null,"912":
null,"913":null,"914":null,"915":
null,"916":null,"917":null,"918":
null,"919":null,"920":null,"921":
null,"922":null,"923":null,"924":
null,"925":null,"926":null,"927":
null,"928":null,"929":null,"930":
null,"931":null,"932":null,"933":
null,"934":null,"935":null,"936":
null,"937":null,"938":null,"939":
null,"940":null,"941":null,"942":
null,"943":null,"944":null,"945":
null,"946":null,"947":null,"948":
null,"949":null,"950":null,"951":
null,"952":null,"953":null,"954":
null,"955":null,"956":null,"957":
null,"958":null,"959":null,"960":
null,"961":null,"962":null,"963":
null,"964":null,"965":null,"966":
null,"967":null,"968":null,"969":
null,"970":null,"971":null,"972":
null,"973":null,"974":null,"975":
null,"976":null,"977":null,"978":
null,"979":null,"980":null,"981":
null,"982":null,"983":null,"984":
null,"985":null,"986":null,"987":
null,"988":null,"989":null,"990":
null,"991":null,"992":null,"993":
null,"994":null,"995":null,"996":
null,"997":null,"998":null,"999":
null,"1000":null,"1001":null,"
1002":null,"1003":null,"1004":
null,"1005":null},"type":"

	domain","util":{"S":{"used":0,"total":1000}}}]}
ERROR	*{ "success":0, "message":" error message" }*

Required Parameters

Name	Type	Example	Description
domain	STRING	6connect.com	Name to call new domain
range	STRING	S	'S' or 'E', depending on VLAN range: extended or standard

Optional Parameters	None
Example URL	api/v1/api.php?target=vlan&action=addDomain&domain=6connect.com&range=S

getAvailable

URL	api/v1/api.php?target=vlan&action=getAvailable
Description	get VLANs in a domain that have not been named yet (even names that are '', or just uses their number as a name)

Returns

Examples

SUCCESSFUL	{"success":1,"message":"16 VLANs found. ","data":{"1":{"2":null,"3":null,"4":null,"5":null,"6":null,"7":null,"8":null,"9":null,"12":null,"13":null,"15":null,"16":null,"17":null,"21":null,"22":null,"23":null}}}
ERROR	*{ "success":0, "message":" error message" }*

Required Parameters

Name	Type	Example	Description
domainId	INTEGER or STRING	'1,2'	ID of the domains to search for

Optional Parameters

Name	Type	Example	Description
tags	INTEGER or STRING	'111,112'	ID of the tags to search for
min	INTEGER	24	VLAN to start search
max	INTEGER	200	VLAN to end search

Example URL	api/v1/api.php?target=vlan&action=getAvailable&domainId=1&tags=111,112&min=2&max=23

get

URL	api/v1/api.php?target=vlan&action=get
Description	Searches enabled VLANs. When not given options, returns all enabled VLANs. Note: an optional parameter, unavailable, is used to get available (un-enabled) VLANs as well

Returns	Examples SUCCESSFUL: {"success":1,"message":"2 domains found. ","data":{"1":{"id":"1","domain":"DefaultDomain","attr_6c":null,"attr_custom":null,"vlans":[{"id":"4","vlan":"105","name":"NewName"}]},"2":{"id":"2","domain":"6connect.com","attr_6c":null,"attr_custom":null,"vlans":[{"id":"20","vlan":"11","name":""}]}}} ERROR: *{ "success":0, "message":" error message" }*
Required Parameters	None
Optional Parameters	**Name** / **Type** / **Example** / **Description** unavailable / STRING / true / Will return available, instead of enabled, VLANs domainId / INTEGER / 1 / domain ID id / INTEGER / 21 / VLAN ID in the vlan table
Example URL	api/v1/api.php?target=vlan&action=get&id=4&unavailable=true

deleteDomain

URL	api/v1/api.php?target=vlan&action=deleteDomain
Description	Deletes a VLAN domain.
Returns	Examples SUCCESSFUL: *{"success":1,"message":"Domain #7 successfully deleted."}* ERROR: *{ "success":0, "message":"error message" }*
Required Parameters	**Name** / **Type** / **Example** / **Description** domainId / INTEGER / 7 / domain ID
Optional Parameters	None
Example URL	api/v1/api.php?target=vlan&action=deleteDomain&domainId=7

updateDomain

URL	api/v1/api.php?target=vlan&action=updateDomain
Description	Updates a VLAN domain.
Returns	Examples SUCCESSFUL: {"success":1,"message":"Domain 2 updated","id":"2","data":[{"id":"2","domain":"44.6 connect.com","attr_6c":{"range":"S","restricted":

	[1,1002,1003,1004,1005]},"vlans":[{"id":"19","vlan":"10","name":"","tags":["105"]},{"id":"20","vlan":"11","name":"","tags":["105"]},{"id":"21","vlan":"12","name":"","tags":["105"]},{"id":"22","vlan":"13","name":"","tags":["105"]},{"id":"23","vlan":"14","name":"","tags":["105"]},{"id":"24","vlan":"15","name":"","tags":["105"]},{"id":"25","vlan":"16","name":"","tags":["105"]},{"id":"26","vlan":"17","name":"","tags":["105"]},{"id":"27","vlan":"18","name":"","tags":["105"]},{"id":"29","vlan":"20","name":"","tags":["105"]},{"id":"30","vlan":"21","name":"","tags":["105"]},{"id":"3286","vlan":"3281","name":"Bob","tags":["105"]},{"id":"3287","vlan":"3282","name":""},{"id":"3288","vlan":"3283","name":""},{"id":"3289","vlan":"3284","name":""},{"id":"3290","vlan":"3285","name":""},{"id":"3291","vlan":"3286","name":""},{"id":"3292","vlan":"3287","name":""},{"id":"3293","vlan":"3288","name":""},{"id":"3294","vlan":"3289","name":""}],"vlans_unnamed":{"1":null,"2":{"tags":["105"]},"3":{"tags":["105"]}, ... "1002":null,"1003":null,"1004":null,"1005":null},"type":"domain","util":{"S":{"used":11,"total":1000}}}]}
ERROR	*{ "success":0, "message":" error message" }*

Required Parameters

Name	Type	Example	Description
domain	STRING	6connect.com	Name of VLAN Domain
range	STRING	E	VLAN range. E for Extended or S for Standard

Optional Parameters: None.

Example URL: api/v1/api.php?target=vlan&action=updateDomain&domain=DefaultDomainnn&range=S

update

URL	api/v1/api.php?target=vlan&action=update
Description	Update properties of a VLAN: name, tags, or both.
Returns	Examples
SUCCESSFUL	*{"success":1,"message":" VLAN (table ID 11735) updated","id":"11735","data":[{"id":"11735","vlan":"50","domain_id":"7","resource_id":*

	null,"name":"","domain_name":"lax.com","tags":[],"tagsString":[]}}}
ERROR	*{ "success":0, "message":" error message" }*

Required Parameters

Name	Type	Example	Description
id	INTEGER	11735	VLAN record ID (which is “id” of the VLANs in “get” action results)

Optional Parameters

Name	Type	Example	Description
name	STRING or empty string	HQ	name of the VLAN
tags	INTEGER or STRING or empty string	111,112	ID of tags to be assigned to this VLAN. Multiple tags IDs are assignable with a comma delimiter. Empty parameter will result in no tags being assigned.
enabled	BOOLEAN	false	Set the VLAN as enabled (true) or disabled (false).

Example URL

api/v1/api.php?target=vlan&action=update&id=11735&name=OReilly&tags=156,159

api/v1/api.php?target=vlan&action=update&id=11735&name=OReilly

api/v1/api.php?target=vlan&action=update&id=11735&tags=156,159

api/v1/api.php?target=vlan&action=update&id=11735&name=

api/v1/api.php?target=vlan&action=update&id=11735&tags=

removeTags

URL	api/v1/api.php?target=vlan&action=removeTags
Description	Remove tags from a domain's VLAN(s).

Returns

Examples

SUCCESSFUL	*{"success":1,"message":" VLAN tags for VLANs updated","id":"1","data":[]}*
ERROR	*{ "success":0, "message":" error message" }*

<table>
<tr><td>Required Parameters</td><td>
<table>
<tr><th>Name</th><th>Type</th><th>Example</th><th>Description</th></tr>
<tr><td>domainId</td><td>INTEGER</td><td>1</td><td>domain ID for domain whose VLANs will be un-tagged</td></tr>
<tr><td>vlan</td><td>INTEGER or STRING</td><td>500

500-510</td><td>a single VLAN (not the record ID from the get action results, but actual VLAN ID), OR

range of VLANs (using hyphen for minimum through maximum), OR comma-delimited list of VLANs and /or VLAN ranges</td></tr>
<tr><td>tagId</td><td>INTEGER or STRING</td><td>150,99</td><td>a single tag ID or comma-delimited list of tag ID's</td></tr>
</table>
</td></tr>
<tr><td>Optional Parameters</td><td>None</td></tr>
<tr><td>Example URL</td><td>api/v1/api.php?target=vlan&action=removeTags&domainId=1&vlan=500-510&tagId=159

api/v1/api.php?target=vlan&action=removeTags&domainId=1&vlan=400,406,500-510&tagId=159,160</td></tr>
</table>

addTags

<table>
<tr><td>URL</td><td>api/v1/api.php?target=vlan&action=addTags</td></tr>
<tr><td>Description</td><td>Add tags to a domain's VLAN(s).</td></tr>
<tr><td>Returns</td><td>Examples
<table>
<tr><td>SUCCESSFUL</td><td>{"success":1,"message":" VLAN tags for VLANs updated","id":"1","data":[]}</td></tr>
<tr><td>ERROR</td><td>{ "success":0, "message":" error message" }</td></tr>
</table>
</td></tr>
<tr><td>Required Parameters</td><td>
<table>
<tr><th>Name</th><th>Type</th><th>Example</th><th>Description</th></tr>
<tr><td>domainId</td><td>INTEGER</td><td>1</td><td>domain ID for domain whose VLANs will be un-tagged</td></tr>
<tr><td>vlan</td><td>INTEGER or STRING</td><td>515

500-510</td><td>a single VLAN (not the record ID</td></tr>
</table>
</td></tr>
</table>

			from the get action results, but actual VLAN ID), OR range of VLANs (using hyphen for minimum through maximum), OR comma-delimited list of VLANs and /or VLAN ranges
tagId	INTEGER or STRING	150,99	a single tag ID or comma-delimited list of tag ID's

Optional Parameters	None
Example URL	api/v1/api.php?target=vlan&action=addTags&domainId=1&vlan=500-510&tagId=159 api/v1/api.php?target=vlan&action=addTags&domainId=1&vlan=400,406,500-510&tagId=159,160

smartAssign

URL	api.php?target=vlan&action=smartAssign
Description	Finds and enables an unassigned VLAN matching the provided search parameters and optionally renames it.

Returns

Examples

SUCCESSFUL	Return Data: the full domain description, plus the chosen VLAN with its new name (optional) *{"success":1,"message":" VLAN enabled. ","data":{"21":{"id":"21","domain":"Swisscom Demo","attr_6c":"{\"range\":\" E\"}","attr_custom":null,"vlans":[{"id":"46183","vlan":"1808","name":"RESERVED_DEV"}]}}}*
ERROR	*{ "success":0, "message":" error message" }*

Required Parameters

Name	Type	Example	Description
domainId	INTEGER	15	The Id number of the domain you would like to Smart Assign from.

Optional Parameters

Name	Type	Example	Description
tags	INTEGER	5,7,12	A comma-separated list

				of the Tag Ids being used for the Smart Assign.
	name	STRING	CustomerName_IT3, RESERVED_DEV	The name that will be assigned to the VLAN matching the given search parameters.
Example URL	api.php?target=vlan&action=smartAssign&domainId=21&name=RESERVED_DEV&tags=5,7,12			

How Do I...

How Do I... (Use Cases)

If you want to get a jumpstart on common API use cases, you came to the right place! Expand the text areas below for walkthroughs and code samples of API calls...

-
- How Do I... (Use Cases)
    - IP Blocks - Update Fields
    - IP Blocks - Assign / Subassign
    - DNS
    - DHCP
  - Python SDK:
    - IP Blocks

IP Blocks - Update Fields

Context: How do I update the notes field of an IP block using the API in PHP?

Click here to expand...

1) Start with providing instance information, API key, Secret Key, and DNS Server IP; set up the connection

```
<?php
//
// This file walks through an example of how to look up a block id number
// in ProVision, and then use it to attach a notes field
//
// supply the URL of your ProVision instance, your API key and your Secret key.
$proVisionURL = "https://ops.6connect.com/qa-4.2.2";
$apiKey = "32-5DAYTJEE2TZHOFOB";
$apiSecretKey = "48b278ec873bda473a323dbc467f8669";
// this example uses 6connect's PHP APIClient
require_once("APIClient.php");
// set up the connection
$apiClient = new APIClient($proVisionURL, $apiKey, $apiSecretKey);
```

2) Split the metadata you want to have showing in the notes, and find the block with which it should associate

```
// lets imagine we have some metadata in the following format:
//
$string = "10.1.245.5||DFW7|HP a5820x|its-erp.dfw7.us.corp||";
//
// And we want to insert the Colo, Server type, and hostname into the Notes field of the IP block

// first we split everything up
$pieces = explode("|", $string);
$ip = $pieces[0];
$colo = $pieces[2];
$type = $pieces[3];
$host = $pieces[4];

// then we pull the IP block using the API.
$params = array();
$params['block'] = "$ip/32";                                    // the IP block we're looking for,
with netmask
// make the call to the IPAM-GET endpoint
$apiResponse = $apiClient->sendRequest('ipam', 'get', $params);
if ($apiResponse->status != 1) {
        echo "Could not pull information for block: $ip/32 !\n";
        die();
```

```
}
if (trim($apiResponse->message) == "No blocks found.") {
        echo "IP block $ip/32 not found in ProVison!\n";
    die();
}

// we now have the ipObject associated with this IP block.  Lets get its block id.
$blockId = $apiResponse->data[0]['id'];
echo "IP block id: $blockId \n";
```

3) Update the block with the notes

```
// it is time to update the block with the new notes.
$notes = "$colo,$type,$host";
$params = array();
$params['id'] = $blockId;
$params['notes'] = $notes;
// make the call to the IPAM-UPDATE endpoint
$apiResponse = $apiClient->sendRequest('ipam', 'update', $params);

// and done!
echo $apiResponse->message . "\n";
```

IP Blocks - Assign / Subassign

Context: I unassigned an IP address and now it's in the Holding Tank. Now I want to assign an IP from the Holding Tank. I don't want to unassign an IP randomly, in case it is allocated to a Resource. What are my options?

Click here to expand...

There are 3 options:

1) If you know the specific IP, you can use use the ipam-get api call to determine if it is in Holding:

```
/api/v1/api.php?target=ipam&action=get&cidr=1.2.3.4/32

{
  id:1234,
  cidr:"1.2.3.4",
  ...
  resource_name:"Holding"
}
```

2) If you want to show all blocks/IPs in Holding, you can use the following ipam-get API call:

```
/api/v1/api.php?target=ipam&action=get&resourceQuery={"name":"Holding"}
```

3) If you know the block is in Holding, you can issue another ipam-unassign API call to move it from Holding to Available:

```
/api/v1/api.php?target=ipam&action=unassign&block=1.2.3.4/32
```

Context: I need to create a Resource Holder, assign them an IP block, then subassign some IPs out of that block to two new Resource Holders. What does this look like in Python?

Click here to expand...

We broke this up in a few steps so it's easier to link together.

1) Let's create a Resource Holder called "Ned"

```
query_string    = 'target=resource&action=add&meta[type]=entry&meta[section]=resource-holder&meta
[name]=Ned'
query_string    += '&apiKey=' + api_key
hash            = base64.b64encode( hmac.new(api_secret_key, query_string, hashlib.sha256).digest()
)
url                = base_url + '?' + query_string + '&hash=' + hash
print 'Create Ned resource holder'
```

```
print url, "\n"
data = json.load(urllib2.urlopen(url))
ned_resource_id = data['data']['id']
```

2) Now let's add the 213.29.27.0/24 IP block

```
query_string =    'target=ipam&action=add&rir=RIPE&block=213.29.27.0/24'
query_string    += '&apiKey=' + api_key
hash            = base64.b64encode( hmac.new(api_secret_key, query_string, hashlib.sha256).digest()
)
url               = base_url + '?' + query_string + '&hash=' + hash
print 'Create 213.29.27.0/24 block'
print url, "\n"
data = json.load(urllib2.urlopen(url))
```

3) With the block in the system, we can assign 213.29.27.0/24 to "Ned" the Resource Holder

```
query_string =    "target=ipam&action=directAssign&block=213.29.27.0/24&resourceId=%d" %
(ned_resource_id)
query_string    += '&apiKey=' + api_key
hash            = base64.b64encode( hmac.new(api_secret_key, query_string, hashlib.sha256).digest()
)
url               = base_url + '?' + query_string + '&hash=' + hash
print 'Assign 213.29.27.0/24 block to Ned'
print url, "\n"
data = json.load(urllib2.urlopen(url))
```

4) Since we plan on assigning IPs out of this block, we should enable subassignments for 213.29.27.0/24

```
query_string =    'target=ipam&action=update&block=213.29.27.0/24&allowSubAssignments=true'
query_string    += '&apiKey=' + api_key
hash            = base64.b64encode( hmac.new(api_secret_key, query_string, hashlib.sha256).digest()
)
url               = base_url + '?' + query_string + '&hash=' + hash
print 'Update 213.29.27.0/24 to allow sub assignments'
print url, "\n"
data = json.load(urllib2.urlopen(url))
```

5) Now let's create a Resource Holder "Tara"

```
query_string =    "target=resource&action=add&meta[type]=entry&meta[section]=resource-holder&meta
[name]=Tara&meta[parent_id]=%d" % (ned_resource_id)
query_string    += '&apiKey=' + api_key
hash            = base64.b64encode( hmac.new(api_secret_key, query_string, hashlib.sha256).digest()
)
url               = base_url + '?' + query_string + '&hash=' + hash
print 'Create Tara resource holder'
print url, "\n"
data = json.load(urllib2.urlopen(url))
tara_resource_id = data['data']['id']
```

6) To keep it interesting, let's create another Resource Holder "Una"

```
query_string =    "target=resource&action=add&meta[type]=entry&meta[section]=resource-holder&meta
[name]=Una&meta[parent_id]=%d" % (ned_resource_id)
query_string    += '&apiKey=' + api_key
hash            = base64.b64encode( hmac.new(api_secret_key, query_string, hashlib.sha256).digest()
)
url               = base_url + '?' + query_string + '&hash=' + hash
print 'Create Una resource holder'
print url, "\n"
data = json.load(urllib2.urlopen(url))
una_resource_id = data['data']['id']
```

7) Assign a /28 block from Ned's 213.29.27.0/24 to Tara

```
query_string =    "target=ipam&action=smartAssign&type=ipv4&rir=RIPE&mask=28&&resourceId=%
d&assignedResourceId=%d" % (tara_resource_id, ned_resource_id)
query_string    += '&apiKey=' + api_key
hash            = base64.b64encode( hmac.new(api_secret_key, query_string, hashlib.sha256).digest()
)
url               = base_url + '?' + query_string + '&hash=' + hash
print 'Assign block from Ned\'s 213.29.27.0/24 to Tara'
print url, "\n"
data = json.load(urllib2.urlopen(url))
```

8) Then assign another /28 block from Ned's 213.29.27.0/24 to Una

```
query_string =    "target=ipam&action=smartAssign&type=ipv4&rir=RIPE&mask=28&&resourceId=%
d&assignedResourceId=%d" % (una_resource_id, ned_resource_id)
query_string    += '&apiKey=' + api_key
hash            = base64.b64encode( hmac.new(api_secret_key, query_string, hashlib.sha256).digest()
)
url               = base_url + '?' + query_string + '&hash=' + hash
print 'Assign block from Ned\'s 213.29.27.0/24 to Una'
print url, "\n"
data = json.load(urllib2.urlopen(url))
```

DNS

Context: I need to set up a DNS server using ProVision's API in PHP, create a zone with a few simple records, and push it to the server.

Click here to expand...

1) Start with providing instance information, API key, Secret Key, and DNS Server IP

```
<?php
//
//
// supply the URL of your ProVision instance, your API key and your Secret key.
$proVisionURL = "https://ops.6connect.com/qa-4.2.2";
$apiKey = "Nnvz8xKZDQUWke6gDxb";
$apiSecretKey = "2YojRbrHnToPZ7cDeFBzcTAvcfMbPVmX";
// this example uses 6connect's PHP APIClient
require_once("APIClient.php");
// set up the connection
$apiClient = new APIClient($proVisionURL, $apiKey, $apiSecretKey);

// save this.  IP of the DNS Server we're creating.
$serverIp = "208.39.106.184";
```

2) Add a DNS server

```
// begin making api calls.  We begin by adding a simple DNS server.
$params = array();
$params['displayName'] = "Example Server";
// the pretty name of the DNS server
$params['server'] =
"208.39.106.184";                                                    // the IP of the
DNS Server
$params['active'] =
1;
// whether or not this server is currently enabled
$params['transferType'] = "SCP";                                     // we are using an ISC Bind
server which we will communicate with via SCP
$params['username'] =
"6connect";                                                            // the username
used to SCP zones to this server
$params['password'] =
"password";                                                            // the password
used to SCP zones to this server
$params['port'] = 22;                                                // the port used to SCP zones
to this server
$params['serverType'] =
```

```
"master";                                                                    // whether this
server is a master or a slave
$params['SOA'] = "ns1.dns.6connect.net. hostmaster.6connect.net.";         // the default SOA
$params['remoteDirectory'] = "/tmp
/";                                                                  // where to place the zone
files on the server
$params['namedConfPath'] = "/tmp
/";                                                                  // the path to the
zones within the configuration file.  Usually the same as 'remoteDirectory'
$params['postCommand'] = "touch /tmp/allFinished";                                    // the
command to execute on the server after the transfer is complete.
// add the server
$apiResponse = $apiClient->sendRequest('dnsServer', 'add', $params);
if ($apiResponse->status == 1) {
        echo "Successfully added DNS Server '" . $params['displayName'] . "'\n";
} else {
        echo "Could not add DNS Server '" . $params['displayName'] . "' !\n";
        die();
}

// now we fetch the id of our newly created server
$params = array();
$apiResponse = $apiClient->sendRequest('dnsServer', 'get', $params);
$data = $apiResponse->data;
for ($i = 0; $i < count($data); $i++) {
        if ($data[$i]['server'] == $serverIp) {
                        // we save the id for later.
                        $serverId = $data[$i]['id'];
                        break;
        }
}
echo "Server Id is: $serverId \n";
```

3) Create a zone

```
// okay, DNS server is set up -- time to create a zone.
$params = array();
$params['zoneName'] = "atestzone.com";                                  // zone name
$params['zoneResourceId'] = 1;                                                 // the owner of the
zone; 1 is default
$apiResponse = $apiClient->sendRequest('zone', 'add', $params);
if ($apiResponse->status == 1) {
    echo "Successfully added DNS Zone '" . $params['zoneName'] . "'\n";
} else {
    echo "Could not add DNS Zone '" . $params['zoneName'] . "' !\n";
    die();
}
// snag the zoneId for later.
$zoneId = $apiResponse->data;
```

4) Add Zone records

```
// Lets add some records to our new zone!
$params = array();
$params['newRecordZoneId'] = $zoneId;                           // parent zone id
$params['newRecordType'] = 'A';                                       // record type
$params['newRecordHost'] = "www";                       // the host field of the record
$params['newRecordValue'] = "1.2.3.4";                  // the value field of the record
$params['newRecordTTL'] = "3600";                       // the value of the TTL field
$apiResponse = $apiClient->sendRequest('record', 'add', $params);
if ($apiResponse->status == 1) {
    echo "Successfully added Record to zone #$zoneId\n";
} else {
    echo "Could not add Record to zone #$zoneId!\n";
    die();
}

$params = array();
$params['newRecordZoneId'] = $zoneId;                    // parent zone id
```

```
$params['newRecordType'] = 'A';                          // record type
$params['newRecordHost'] = "dev";                        // the host field of the record
$params['newRecordValue'] = "2.3.4.5";                       // the value field of the record
$params['newRecordTTL'] = "3600";                                    // the value of the TTL
field
$apiResponse = $apiClient->sendRequest('record', 'add', $params);
if ($apiResponse->status == 1) {
    echo "Successfully added Record to zone #$zoneId\n";
} else {
    echo "Could not add Record to zone #$zoneId!\n";
    die();
}

$params = array();
$params['newRecordZoneId'] = $zoneId;                  // parent zone id
$params['newRecordType'] = 'A';                        // record type
$params['newRecordHost'] = "cloud";                    // the host field of the record
$params['newRecordValue'] = "3.4.5.6";                 // the value field of the record
$params['newRecordTTL'] = "3600";                      // the value of the TTL field
$apiResponse = $apiClient->sendRequest('record', 'add', $params);
if ($apiResponse->status == 1) {
    echo "Successfully added Record to zone #$zoneId\n";
} else {
    echo "Could not add Record to zone #$zoneId!\n";
    die();
}
```

4) Link the Zone to the new DNS server and push

```
// Okay, we have some zones with records.  Time to link this zone to the new DNS Server
$params = array();
$params['serverId'] = $serverId;                                                  // the server id
$params['zoneId'] = $zoneId;                                                         // the zone
id
$params['serverSlave'] = 0;                                                                  // not
a slave zone
$apiResponse = $apiClient->sendRequest('zoneLinkage', 'add', $params);
if ($apiResponse->status == 1) {
    echo "Successfully linked Zone #$zoneId to server #serverId\n";
} else {
    echo "Could not link Zone #$zoneId to server #serverId!\n";
    die();
}
// now we can push the zone to the server
$params = array();
$params['zoneId'] = $zoneId;                           // the zone id to push
$apiResponse = $apiClient->sendRequest('dnsServer', 'transferSingle', $params);
if ($apiResponse->status == 1) {
    echo "Zone pushed!\n";
} else {
    echo "Could not push zone!\n";
    die();
}
?>
```

DHCP

Context: I need to attach the DHCP module as a child

Click here to expand...

DHCPv2 functionality is enabled on a particular resource by attaching a DHCP Module as a child. A command to do this is as follows:

```
            [ProVision root]/api/v1/api.php?target=resource&action=add

            data:
meta[type]: dhcp_module
```

```
meta[name]: [parent resource id] DHCP Module
meta[parent_id]: [parent resource id]
```

The special resource type “dhcp_module” indicates to ProVision that the DHCP system is enabled for the parent object. The attributes associated with the “dhcp_module” resource govern the DHCP system's behavior.

Updating the attributes of a DHCP Server uses a Resource Update command:

```
[ProVision root]/api/v1/api.php?target=resource&action=update&meta[id]=2178 &meta[type]
=dhcp_module&fields[_dhcp_attributes][]={"type":"ISC","notes":"notes go here","username":"
username","port":"port","config_test":"/etc/init.d/dhcpd configtest","server_stop":"/etc/init.d
/dhcpd stop","server_start":"/etc/init.d/dhcpd start","config_path":"/tmp/dhcpd.conf","
option_routers":"192.168.0.0","option_domain_name_servers":"ns1.6connect.com","
option_domain_name":"6connect.com","authoritative":"1","default_lease_time":"600","
max_lease_time":"7200","local_port":"67","log_facility":"local7","password":"password","
server_ip":"192.168.0.1","freeLines":3,"freeLine1":"free line 1","freeLine2":"free line 2","
freeLine3":"free line 3"}
```

This command appears rather complicated, but can be broken apart into reasonable pieces. The first section:

```
target=resource&action=update&meta[id]=2178&meta[type]=dhcp_module
```

is familiar from other parts of ProVision. We are updating a resource of type “dhcp_module” whose resource id is 2178. The second section of the command details the update values, starting with

```
 fields[_dhcp_attributes][]=
```

which contains a JSON-encoded string of all the fields specific to a DHCP server's function. When expanded into its full object form it is substantially easier to digest:

```
{
            "type":"ISC",
            "notes":"notes go here",
            "username":"username",
            "port":"port",
            "config_test":"/etc/init.d/dhcpd configtest",
            "server_stop":"/etc/init.d/dhcpd stop",
            "server_start":"/etc/init.d/dhcpd start",
            "config_path":"/tmp/dhcpd.conf",
            "option_routers":"192.168.0.0",
            "option_domain_name_servers":"ns1.6connect.com",
            "option_domain_name":"6connect.com",
            "authoritative":"1",
            "default_lease_time":"600",
            "max_lease_time":"7200",
            "local_port":"67",
            "log_facility":"local7",
            "password":"password",
            "server_ip":"192.168.0.1",
            "freeLines":3,
            "freeLine1":"free line 1",
```

```
            "freeLine2":"free line 2",
            "freeLine3":"free line 3"
}
```

This object describes all the most common DHCP server configuration options. For a full explanation of each of the fields, see the Detailed API Specification later in this document.

Please note that the object above must be passed to the DHCP system as a JSON-encoded string. It must be passed into the special "_dhcp_attributes" attribute for it to be functional, as in the example URL.

Context: I need to add a DHCP aggregate

Click here to expand...

An example command to add a DHCP Aggregate is:

```
[ProVision root]/api/v1/api.php?target=ipam&action=add&block=192.168.0.0/24&rir=
1918&vlan=&tags=&region=&resourceId=1282&allowSubAssignments=true
```

The important part to note is that the IP block is being assigned to resourceId 1282, which corresponds to the DHCP Available resource. The DHCP Available resource is a system-level resource which is used to hold all unassigned DHCP IP addresses. Every instance has its own DHCP Available resource, whose id can be found with the following command:

```
[ProVision root]/api/v1/api.php?target=resource&action=get&slug=dhcp-available
```

New DHCP subnets and hosts draw their IPs from this pool. If there are no IPs in the DHCP Available pool new subnets and hosts will not be able to be created.

DHCP IP aggregates are fetched, updated, split, and deleted using the standard IPAM management API endpoints. Please see the IPAM API Documentation. for details.

Context: I need to add a DHCP Pool

Click here to expand...

Similar to how the "dhcp_module" resource was created above, the command to create a DHCP Pool is as follows:

```
[ProVision root]/api/v1/api.php?target=resource&action=add&meta[type]=dhcp_pool &meta[name]=New
Subnet&fields[_dhcp_type][]=subnet&fields[_dhcp_pool_attributes][]={"mac":"","rangeStart":"","
rangeEnd":"","freeLines":3,"freeLine1":"Free Line 1","freeLine2":"Free Line 2","freeLine3":"Free
Line 3"}
```

The first half of this command is relatively straightforward:

```
target=resource&action=add&meta[type]=dhcp_pool&meta[name]=New Subnet
```

This section informs the API that we wish to create a new, empty "dhcp_pool" resource whose name is "New Subnet."

```
fields[_dhcp_type][]=subnet&fields[_dhcp_pool_attributes][]={"mac":"","rangeStart":"",
"rangeEnd":"","freeLines":3,"freeLine1":"Free Line 1","freeLine2":"Free Line 2","freeLine3":"Free
Line 3"}
```

The second half of the command behaves in a similar manner to the “dhcp_module.” The “_dhcp_pool_attributes” field holds a JSON-encoded string which describes the dhcp_pool resource. When expanded, the JSON string becomes the following object:

```
{
            "mac":"",
            "rangeStart":"",
            "rangeEnd":"",
            "freeLines":3,
            "freeLine1":"Free Line 1",
            "freeLine2":"Free Line 2",
            "freeLine3":"Free Line 3"
}
```

For a full explanation of each of the fields, see the Detailed API Specification.

Please note that the object above must be passed to the DHCP system as a JSON-encoded string. It must be passed into the “_dhcp_pool_attributes” attribute for it to be functional, as in the example URL.

Once a dhcp_pool resource is in the system it can be updated with IP data obtained from the IP Management system. Under DHCPv2, the DHCP system uses all the standard IPAM API endpoints and can make use of both the smartAssign and the directAssign methods. Please see the IPAM API documentation for details.

Context: I need to link a DHCP pool to a DHCP server

Click here to expand...

An example of building a link between a dhcp_pool and a DHCP Server is:

```
[ProVision root]/api/v1/api.php?target=resource&action=addLink&resource_id1=2178&
resource_id2=1452&relation=dhcpPoolLink
```

The Resource Linkage system controls which DHCP Pools are associated with a given DHCP Server. In the case of linking a DHCP Pool to a DHCP Server, the relation used is “dhcpPoolLink”. This is a directional link, so it is important that resource_id1 and resource_id2 do not get confused.

```
relation:  "dhcpPoolLink"
resource_id1:  the id of the dhcp_module this pool is being linked to
resource_id2:  the id of the dhcp_pool being linked
```

It is very important that resource_id1 not be confused with resource_id2. The link will not function with the values reversed.

To undo the above and break a DHCP Pool link, use the same command but substitute “deleteLink” for the action “addLink”.

```
[ProVision root]/api/v1/api.php?target=resource&action=deleteLink&resource_id1=2178&
resource_id2=2179&relation=dhcpPoolLink
```

Context: I need to push a DHCP config file

Click here to expand...

Once the server has been configured according to the previous sections, hitting the following API endpoint will trigger a DHCP push:

```
[ProVision root]/api/v1/api.php?target=dhcp&action=push&id=2178
```

The “id” in the above string is the id of the dhcp_module resource attached to the server you whose configuration is to be pushed. The API return payload will contain success or failure codes, as well as a description of any errors which might have occurred.

When a DHCP configuration file is pushed an SSH connection is opened to the configured server using the user, password, and port supplied to the '_dhcp_attributes' attribute on the dhcp_module resource. If the system successfully connects, it will assemble a DHCP configuration from the information given to the dhcp_module's '_dhcp_attribute' attribute and then parse and add in all linked dhcp_pool resources.

After the assembled file has been transferred to the DHCP server it will be placed in the location given by 'config_path' on the dhcp_module, and then the command described in 'config_test' will be run to determine whether or not this new file parses correctly. If 'config_test' is blank or omitted, this step is skipped.

If the file parses correctly the DHCP will be stopped and restarted according to the 'server_stop' and 'server_start' commands on the DHCP module. If there are errors at any point the system backs out, replaces old config files, and reports the errors via the 'message' return field of the API call.

Python SDK:

IP Blocks

Context: How do I create aggregates, get block information, and delete aggregates using the API / python SDK?

Click here to expand...

```
#!/usr/bin/python
from apiclient import APIClient, APIResponse

# REPLACE WITH CORRECT VALUES FOR YOUR INSTANCE

base_url                = 'https://<ProVision Instance URL>'

api_key                         = '00-ABCDEFGHIJ123456'

api_secret_key        = '0123456789abcdef0123456789abcdef'


# create the APIClient

client = APIClient(base_url, api_key, api_secret_key)



# create aggregate 1.2.3.0/24

target = 'ipam'

action = 'add'

params = {'block': '1.2.3.0/24', 'rir':'ARIN'}

url = client.get_request_url(target, action, params)

print url

response = client.make_api_call(target, action, params)

print response
```

```
# get block 1.2.3.0/24

target = 'ipam'

action = 'get'

params = {'block': '1.2.3.0/24'}

url = client.get_request_url(target, action, params)

print url

response = client.make_api_call(target, action, params)

print response



# delete aggregate 1.2.3.0/24

target = 'ipam'

action = 'delete'

params = {'block': '1.2.3.0/24'}

url = client.get_request_url(target, action, params)

print url

response = client.make_api_call(target, action, params)

print response
```

Reverse API

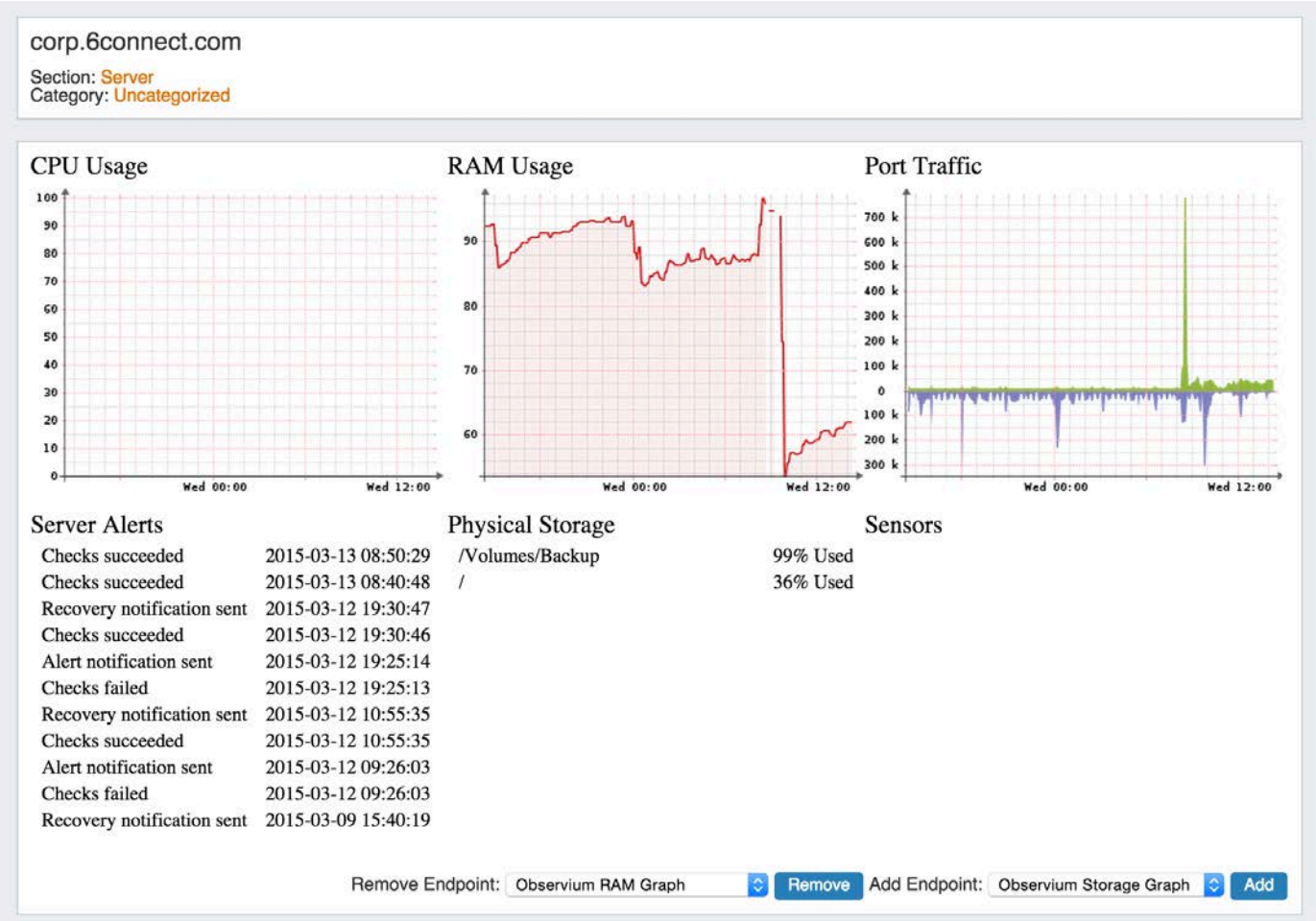
Reverse API

- Reverse API
  - Overview
  - API Call Formatting
  - Reverse API Detail:

Overview

Reverse API Tools - Beta

ProVision's Reverse API calls and UI elements allow for integration with outside APIs to improve workflow and create custom display content. In the ProVision user interface, the Reverse API page allows for endpoints to be built and provides a text editor to create presentation JavaScript commands. This JavaScript presentation code is then displayed in the Reverse API Gadget.



Using the ProVision Reverse API (rAPI), you can perform these same actions and customize to meet your specific needs outside of the ProVision UI.

API Call Formatting

Reverse API (rAPI) calls are made to hit a user-defined URL or command line program. If the rAPI type is set to 'external,' the system makes a HTTP request. If the rAPI type is set to 'local', it executes the call on the command line of the local machine. In both cases the call is first customized with the attributes of the resource supplied to the execute function. This allows a single rAPI endpoint to serve a wide array of individual resources, fetching only information relevant to that particular resource without having to store anything locally.

rAPI calls are formatted thusly:

```
http://observium.tcp0.com/graph.php?
height=200&width=265&type=device_bits&legend=no&username=api&password=password&device={observium-id}
```

The interesting part about this URL is the bit in curly-braces: {observium-id}. When this call is made, the system decodes the URL by searching for everything within curly-braces and replaces it with data pulled from a resource. A rAPI call may have as many or as a few curly-brace replacement targets as is needed. Each curly-brace target will be replaced with exactly one resource attribute. All targets must be successfully replace for the command to succeed.

For example, when this rAPI call is decoded with information from the 1-dev resource, the decoded call is as follows:

```
http://observium.tcp0.com/graph.php?
height=200&width=265&type=device_bits&legend=no&username=api&password=password&device=21
```

The rAPI service then pulls the data from that URL, pairs it with its presentation code, and returns it via the execute endpoint. If a user attempted to execute the above call on a resource which did not have the “observium-id” property, the execute endpoint would return the following error: "Reverse API Call references token 'observium-id', which does not exist in resource."

All curly-brace calls reference a resource attribute by their unique resource slugs. The list of available resource attributes can be customized using the existing resource system endpoints.
Curly-brace targets may take the form {parent.observium-id}. The “parent.” prefix indicates that in decoding this particular brace, the attributes of the resource's parent are to be used.

Reverse API Detail:

For detail on the Reverse API calls and parameters, proceed to Reverse API - Detail.

Reverse API - Detail

- Reverse API Module
  - reverseAPI_add
  - reverseAPI_get
  - reverseAPI_delete
  - reverseAPI_update
  - reverseAPI_execute

Reverse API Module

reverseAPI_add

URL	/api/v1/api.php?target=reverseAPI&action=add
Description	Add a Reverse API Endpoint
Returns	**Examples:** SUCCESSFUL: performs ResourceAPI->Get operation to return the newly-created entry, thus will appear similar to a "Get" return: {"success":1,"message":" Reverse API endpoint added","data":{"id":"1","name":" TestPoint2","type":"local","created_by":"user","last_modified":"2015-04-01 12:24:57","call":"http:\/\/www.test.com?api.php","presentation":null}} ERROR: {"success":0, "message":"error message"}
Required Parameters	**Name** / **Type** / **Example** / **Description** name / STRING / TestPoint2 / The name of the rAPI object type / STRING / local / The type of the rAPI object. Valid settings are 'local' and 'external'. call / STRING / http://www.test.com?api.php&server={serverId} / The URL or the system path which is first decoded against a resource and then hit when this rAPI endpoint is invoked.
Optional Parameters	

Name	Type	Example	Description
presentation	STRING	endpoint = function(data, outputDiv) {console.log (outputDiv);};	The Javascript code used to display this endpoint on a resource page.

Example URL	/api/v1/api.php?target=reverseAPI&action=add&name=TestPoint2&type=local&call=http://www.test.com?api.php&server={serverId}

reverseAPI_get

URL	/api/v1/api.php?target=reverseAPI&action=get
Description	Fetches one or more Reverse API endpoints
Returns	**Examples:**

SUCCESSFUL	An array of ReverseAPI objects: {"id":"535","name":"Observium RAM Graph","type":"external","created_by":"user","last_modified":"2015-03-18 09:16:49","call":"http:\/\/observium.tcp0.com\/graph.php?height=200&width=265&type=device_
ERROR	{"success":0, "message":"error message"}

Return Detail:

Name	Type	Description
id	INTEGER	The id of the rAPI objectSt
name	STRING	The name of the rAPI object
type	STRING	The type of the rAPI object
created_by	STRING	What user created the rAPI object
last_modified	DATETIME	When this rAPI object was last changed
call	STRING	The URL or the system path which is first decoded against a resource and then hit when this rAPI endpoint is invoked.
presentation	STRING	The Javascript

			code used to display this endpoint on a resource page.

Required Parameters	None
Optional Parameters	

Name	Type	Example	Description
id	INTEGER	535	The id of the rAPI objectSt
name	STRING	TestPoint2	The name of the rAPI object
type	STRING	external	The type of the rAPI object. Valid settings are 'local' and 'external'.
created_by	STRING	user	The user who created the rAPI object

Example URL	/api/v1/api.php?target=reverseAPI&action=get&name=TestPoint2

reverseAPI_delete

URL	/api/v1/api.php?target=reverseAPI&action=delete
Description	Delete a rAPI endpoint
Returns	**Examples:**

SUCCESSFUL	{"success":1,"message":"Reverse API endpoint deleted"}
ERROR	{"success":0, "message":"error message"}

Required Parameters	

Name	Type	Example	Description
id	INTEGER	535	The id of the rAPI objectSt
name	STRING	TestPoint2	The name of the rAPI object

Optional Parameters	None
Example URL	/api/v1/api.php?target=reverseAPI&action=delete&id=535&name=TestPoint2

reverseAPI_update

URL	/api/v1/api.php?target=reverseAPI&action=update
Description	Update an existing rAPI endpoint
Returns	**Examples:** SUCCESSFUL: ResourceAPI->Get updated object, thus will appear the same as a "Get" return: {"id":"535","name":"Observium RAM Graph","type":"external","created_by":"user","last_modified":"2015-03-18 09:16:49","call":"http:\/\/observium.tcp0.com\/graph.php?height=200&width=265&type=device_ ERROR: {"success":0, "message":"error message"}

Required Parameters

Name	Type	Example	Description
id	INTEGER	535	The id of the rAPI objectSt

Optional Parameters

Name	Type	Example	Description
name	STRING	TestPoint2	The name of the rAPI object
type	STRING	external	The type of the rAPI object. Valid settings are 'local' and 'external'.
call	STRING	http://www.test.com?api.php&server={serverId}&action=12	The URL or the system path which is first decoded against a resource and then hit when this rAPI endpoint is invoked.
presentation	STRING	endpoint = function(data, outputDiv) {console.log(outputDiv);};	The Javascript code used to display this endpoint on a resource page.

Example URL	/api/v1/api.php?target=reverseAPI&action=update&id=535&name=TestPoint2

reverseAPI_execute

<table>
<tr><td>URL</td><td>/api/v1/api.php?target=reverseAPI&action=execute</td></tr>
<tr><td>Description</td><td>Decodes a rAPI call against a Resource, executes the resulting string, returns the result along with the presentation code</td></tr>
<tr><td>Returns</td><td>Examples:

<table>
<tr><td>SUCCESSFUL</td><td>The result of a rAPI call customized for the supplied resource

{“data”:”\u0017\u00fbD","presentation":"endpoint = function(data, outputDiv) {\n\n\toutputDiv.css('font-family','Helvetica, Arial, sans-serif;');\n\n\toutputDiv.append(\" Port Traffic<\/div>\");\n\n\tif (data == \"\") {\n\t\toutputDiv.append(\" No data returned<\/div>\");\n\t\treturn;\n\t}\n\n\toutputDiv.append('<div style=\"margin-left:-15px; margin-top:-3px;\"><\/div>');\n};"}}</td></tr>
<tr><td>ERROR</td><td>{"success":0, "message":"error message"}</td></tr>
</table>
Return Detail:

<table>
<tr><th>Name</th><th>Type</th><th>Description</th></tr>
<tr><td>data</td><td>MIXED</td><td>The result of the reverse API call. Could be binary data, could be JSON.</td></tr>
<tr><td>presentation</td><td>STRING</td><td>The presentation javascript which pairs with this rAPI call</td></tr>
</table></td></tr>
<tr><td>Required Parameters</td><td>
<table>
<tr><th>Name</th><th>Type</th><th>Example</th><th>Description</th></tr>
<tr><td>id</td><td>INTEGER</td><td>537</td><td>The id of the rAPI objectSt</td></tr>
<tr><td>resource_id</td><td>INTEGER</td><td>2245</td><td>ID of the resource</td></tr>
</table></td></tr>
<tr><td>Optional Parameters</td><td>None</td></tr>
<tr><td>Example URL</td><td>/api/v1/api.php?target=reverseAPI&action=execute&id=537&resource_id=2245</td></tr>
</table>

CLI (Alpha)

Command Line Interface - ALPHA

- Command Line Interface - ALPHA
  - Overview
  - CLI Commands (ALPHA)

Overview

The command line interface for ProVision is a beta feature that has been release for feedback.

How to Access the CLI from your browser

When logged into ProVision via a web browser, use the key combination "**Control+Shift+S**" or "**Control+Shift+~**" to access/close the CLI

CLI Commands (ALPHA)

CLI Help

When in the CLI, type:

```
ipam man
```

for sample commands and syntax

Currently, the CLI supports the following commands:

```
ipam <command> [-t] [<cidr>] [<resource name>] [<args>]

    show:     show details for a block. Examples:

                - "ipam show 10.0.0.0/8" will show details for the block 10.0.0.0/8

                - "ipam show holding" will show details for all blocks in the Holding Tank

                - "ipam show "<resource name>"" will show details for all blocks assigned to
<resource name>


    add:      add a block. ex: "ipam add 192.168.0.0/24"


    update:   update attributes for a block. ex: ipam update 192.168.0.0/24 --vlan=100 tags=VM,Dev


    assign:   assign a block to a resource. ex: ipam assign 192.168.0.0/24 "<resource name>"


    assign:   smart assign a block to a resource. ex: ipam assign --mask=24 --rir=ARIN --type=ipv4
"<resource name>"


    unassign: reclaims a block from a resource and places it in the Holding Tank. If the block is
already in the holding tank, reclaims it an makes it available.
```

Toolkit

Toolkit Modules

6connect provides customers with additional CLI modules for ProVision power users in the /tools directory. These tools are outlined below.

- Toolkit Modules
  - Users / Permissions
    - Add User
    - Rebuild Permissions
  - DNS
    - Audit Forward DNS
    - Audit Reverse DNS
  - Database
    - Database Analyzer
    - Database Reset
  - Import
    - IPPlan Importer:
    - PHPIPAM Importer:
  - Internal 6connect Tools
  - Additional Information:

Users / Permissions

Add User

File: add-user.php

Description: Adds a new user to 6connect Provision. Especially useful if an admin has locked themselves out and has access to the local server, or for adding additional credentials to a new local installation. The tool walks through the required information.

Command: “php add-user.php"

Rebuild Permissions

File: rebuild-permissions.php

Description: Will rebuild the cached permissions for the specified resource id. Should not be used outside of a db reset, or a corruption of the permissions cache.

Command: “php rebuild-permissions.php"

DNS

Audit Forward DNS

File: audit_forward_dns.php

Description: Audits the forward DNS for a specified IP block, comparing the DNS records in 6connect to a publicly facing server. Will provide detail output showing resolved records, if records conflict and why.

Command: “php audit_forward_dns.php”

Options:

-v Sets verbose mode to print debug information

-a <message key> run as asycronous request (note: you should not use this, it is for the GUI)

-b <ip block> IP block in cird notation to audit dns for.

-n <name server> FQDN or IP address of specific name server to resolve records against. Defaults to 8.8.8.8.

-h print help

Audit Reverse DNS

File: audit_reverse_dns.php

Description: Audits the reverse DNS for a specified IP block, comparing the DNS records in 6connect to a publicly facing server. Will provide detail output showing resolved records, if records conflict and why.

Command: "php audit_reverse_dns.php"

Options:

-v Sets verbose mode to print debug information

-a <message key> run as asycronous request (note: you should not use this, it is for the GUI)

-b <ip block> IP block in cird notation to audit dns for.

-n <name server> FQDN or IP address of specific name server to resolve records against. Defaults to 8.8.8.8.

-h print help

Database

Database Analyzer

File: databaseAnalyzer.php

Description: Takes two MySQL dump files for input, one thought to be the “good” database, and one the “bad” to compare against each other for differences. Is used internally to audit database changes after an upgrade to ensure all changes for the upgrade were committed by comparing against the template database for that release. Can be useful for diagnosing missed upgrades, or other database issues. Could be used on any set of databases, and is not 6connect specific.

Command: "php databaseAnalyzer.php"

Options:

-v Sets verbose mode to print debug information

-a <database name> Required - the 'good' database to comapre against.

-b <database name> Required - the 'bad' database to check for incorrect information.

-g <globals file> Optional - The script will automatically look to the globals.php file for database connection information. This option can be used to specify and alternate file, or file location.

-h print help

Database Reset

File: dbReset.php

Description: This tool resets portions or all of the database to factory defaults. Can reset only IP, DNS, or Resource information or all all three. If erasing resource information, you need to enter a user to assume global admin permission to log back in with. Even if this isn’t specified, the add-user tool can be used to add a user back after. We recommend creating a backup prior to using this tool, and only use if you are sure you know what you are doing!

Command:"php dbReset.php"

Options:

-v Sets verbose mode to print debug information

-i Erases any and all IP information. Tags, regions, and other IPAM preference settings are not changed.

-d Erases any and all DNS information including zones, records, and archive data. Does not affect DNS perference settings, or configured DNS servers.

-r Resets all resource information back to the factory defaults, and removes all associated information (user groups, files, etc).

-u <user name> User name that will have Global Admin permissions restored on the default resources

-h print help

Import

IPPlan Importer:

File: ipplan_import.php

Description: The IPPlan Importer is a command-line tool to import IPs from an IPPlan database into ProVision. Details on how to use this tool are available at IPPlan Importer.

Command: "php ipplan_import.php"

Options:

For options, refer to IPPlan Importer.

PHPIPAM Importer:

File: phpipam_import.php

Description: Generates a CSV or imports data from a phpipam instance

Command: "php phpipam_import.php"

Options: Contact 6connect Support if you have any questions

Internal 6connect Tools

The following tools contained in the /tools folder are for 6connect internal use only:

File: scp_wrapper.php - Internal 6connect tool, not meant for command line use.

File: s64_restart_master - Internal 6connect tool, not meant for command line use.

File: s64_restart_slave - Internal 6connect tool, not meant for command line use.

File: async_import.php - Internal 6connect tool, not meant for command line use.

File: importBigDump.php - Internal 6connect tool supporting IPPlan Importer, not meant for command line use.

File: ipplan_functions.php - Internal 6connect tool supporting IPPlan Importer, not meant for command line use.

File: dns_transfer_server.php - Internal 6connect tool, not meant for command line use.

File: observium_import.php - Internal 6connect tool, not meant for command line use.

File: digAndLookup.php

File: pullLAMPInfo.php

Additional Information:

- IPPlan Importer

IPPlan Importer

IPPlan Importer

- IPPlan Importer
  - Overview
  - Before you begin:
  - Connector Method (Results in .csv files only):
  - Importer Method (Results in full import):
    - Run & Set the RIR:
  - Additional Run Options:
    - Load SQL Without Importing
    - Generate CSV Without Loading SQL
    - Generate CSV No Overwrite
    - Generate CSV With RIR
    - Run Only Import
    - Run Import No Overwrite

Overview

The IPPlan Importer is a command-line tool to import IPs from an IPPlan database into ProVision. This tool can be used via two approaches: generating .csv files via the tool only, then using the ProVision IP Import UI to import the csv files (Connector), or as a full command-line import solution, bypassing the ProVision UI entirely (Importer).

Options:

1) Generate .csv files to use with the IP Import - Upload/Import from CSV tool through the ProVision UI.

2) Import IPs directly into ProVision without accessing the UI.

Before you begin:

You will need:

- A MySQL export of IPPlan created from mysqldump, located on the same server / accessible from the IPPlan Importer tool.
- Administrative access to your ProVision instance folder

Accessing the Tool and Showing Help Instructions:

The tool is located in tools/ipplan_import.php. A help document is provided listing example commands and detailed situation-specific options.

To execute it, you must preface it with "php" program. From from your instance's root folder, the command would be:

"php tools/ipplan_import.php --help"

If you are in the tools folder, the command would be:

"php ipplan_import.php --help"

This will bring up the help / instruction document. It is also provided below.

Click here to see the IPPLan Importer help text...

Help File Text

```
usage: php ipplan_import.php [options] <SQL dump file>

Examples:

For creating CSV's for use in ProVision IP importer:
    php ipplan_import.php filename.SQL --csv

For creating CSV's, but do not load SQL file if one already has been:

    php ipplan_import.php filename.SQL --csv --no-overwrite

For importing data from SQL dump. Note that you must specify RIR:
    php ipplan_import.php filename.SQL --rir=1918
```

```
For importing data from SQL dump, if one hasn't been loaded already:
    php ipplan_import.php filename.SQL --rir=1918 --no-overwrite

Caution: By default, SQL dump will be loaded into database 'provision_import_ipplan'. If database
already exists it will be dropped, unless --no-overwrite option is specified.

options:

  --only-import - Skip loading SQL file. Instead, import existing data in database
'provision_import_ipplan'.

  --only-sql - Load SQL dump into database 'provision_import_ipplan'. --only-import can be run at
a later time to use loaded database.

  --rir=RIR - (required for import) RIR for IP's

  OPTIONS IF LOADING SQL FILE:

  --no-overwrite - Load SQL dump only if database 'provision_import_ipplan' is currently non-
existent. Otherwise, drop existing database.

  OPTIONS IF IMPORTING DATA:

  --csv - generate CSV's instead of importing directly to ProVision.
```

Connector Method (Results in .csv files only):

This method creates .csv files that you can use with the IP Import - Upload/Import from CSV tool through the ProVision UI.

1. From the tools/ folder, run the tool with the --csv option. Be sure to reference your sql export file name and location:

   "php ipplan_import.php ../ipplanv6.sql --csv"

   (where '../ipplanv6.sql' is the path to your SQL file)
   This will generate two files: ipplan_aggregates.csv and ipplan_hosts.csv. Both can be used to import in the IP import section of ProVision.
2. Copy / Download the generated .csv files to a browsable directory on your local machine, to be accessed by the ProVision UI.
3. Through the ProVision UI, import the ipplan_aggregates.csv file as described in IP Import - Upload/Import from CSV.
4. Through the ProVision UI, import the ipplan_hosts.csv file as described in IP Import - Upload/Import from CSV.

Note that you must import ipplan_aggregates first. into IP Import - Upload/Import from CSV to create the aggregates ipplan_hosts.csv will use.

Then, import ipplan_hosts.csv as described in IP Import - Upload/Import from CSV.

Importer Method (Results in full import):

This method has the tool process the import task. Using this case, you must set the RIR in the command line for all of the IP's.

Run & Set the RIR:

1. From the tools/ folder, run the tool with the --rir option, referencing your SQL file location:

   "php ipplan_import.php ../ipplanv6.sql --rir=1918"
   (where '../ipplanv6.sql' is the path to your SQL file)

This will load your IPPlan database file into your mysql server and then import the hosts into ProVision. They will each be given the RIR you specified, as well as this text in the Notes field: 'IPPlan import'

The import may need to run for a number of minutes, depending on the size of your data. For reference, an import of 2100 hosts inside of 150 aggregates took approximately 10 minutes to complete during our testing.

Additional Run Options:

Additional run options for various combination of conditions are detailed below. The command is the text within the quotes only.

Load SQL Without Importing

You can load the SQL file, but NOT run an import, with the --only-sql option (feel free to include --no-overwrite option as well in case you don't want to overwrite a prior loaded SQL file):

"ipplan_import.php ../ipplanv6.sql --only-sql"

"ipplan_import.php ../ipplanv6.sql --only-sql --no-overwrite"

Generate CSV Without Loading SQL

If you have already loaded the sql file previously, you can generate CSV files without having to load the SQL file:

"ipplan_import.php --only-import --csv"

Generate CSV No Overwrite

If you want to just generate CSV files without re-loading the database file, you can as well:

"ipplan_import.php ../ipplanv6.sql --no-overwrite --csv"

Generate CSV With RIR

You can also generate csv files with the rir option:

"ipplan_import.php ../ipplanv6.sql --no-overwrite --csv --rir=1918"

Run Only Import

If you have already loaded your IPPlan database with the tool before, you can do --only-import:

"php ipplan_import.php --rir=1918 --only-import"

Run Import No Overwrite

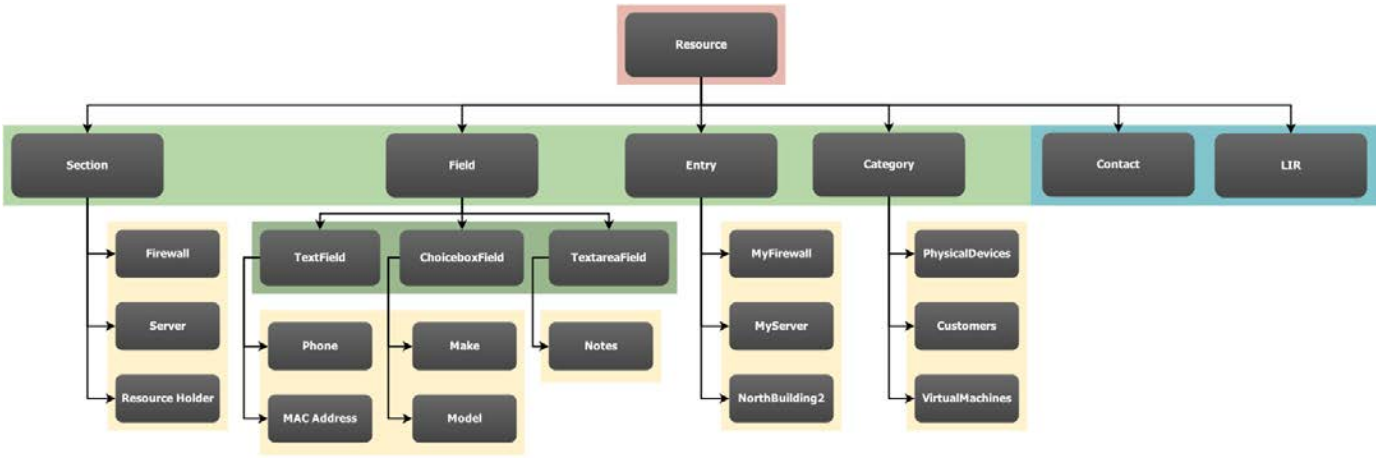
If you are not sure whether you have already loaded your IPPlan database with the tool before, you can specify --no-overwrite to NOT load the sql file if it was loaded already, or do load if it wasn't:

"ipplan_import.php ../ipplanv6.sql --rir=1918 --no-overwrite"

Resource Concepts 1

Overview

In Provision, the Resource System (RS) is an expression of object-oriented programming. In this context, the term “resource” is equivalent to the term “object”, where an object is an instance of a class. Traditionally in OOP, there is an Object class that is the root of the class hierarchy. In the RS, the Resource class is the root class. Every class in the system has Resource as a superclass and all resource objects implement the methods of that class.



The diagram above shows examples of resource sub-types. The items on a green or blue background are types of resources; they each have their own corresponding Class. An item on a yellow background is an example of an object that could have been instantiated from the class (resource type) that it's part of.

Additional Information:

- Resource Classes Doc
- Database Layout 1
- Asset System

Resource Classes Doc

Classes

> "A class--the basic building block of an object-oriented language such as Java--is a template that describes the data and behavior associated with instances of that class. When you instantiate a class you create an object that looks and feels like other instances of the same class."
>
> *Mary Campione and Kathy Walrath, The Java Tutorial: Object-Oriented Programming for the Internet, The Java Series (Reading, Mass.: Addison Wesley, 1996)*

- Classes
  - Class Resource
    - Properties
  - Examples
    - 1 - PHP
    - 2 - API request

Class Resource

```
class Resource {
    public int    $id;
    public string $name;
    public string $slug;
    public string $type;
    public int    $parent_id;
    public int    $category_id;

    protected array $attr   = array();
    protected bool  $loaded = FALSE;

    public object get_attr( string $key );
    public void   set_attr( string $key, object $value );
    public bool   loaded();
}
```

Properties

As you can see from the database layout, the public properties of the Resource class are all part of the main **resource** table. The two protected properties **attr** and **loaded** are created at runtime. There are many situations where only the core information is required. To improve performance, attribute data is ignored when it is not required. Attributes are stored in the database as longtext; non-primitive types (such as arrays) are serialized and stored as a string.

```
$attr
A key-value store of the attributes that exist in the resource_attr table.

$loaded
A boolean value which is used to indicate whether or not the attributes have been loaded.
```

Why do some attributes have names that start with an underscore?

This is the convention for storing metadata. Most attributes are for storing data that is created by the user and is available to be directly edited by the user. When we want to store system data, configuration options, or just data that isn't meant for human consumption - we store it as metadata. An attribute is identified as being metadata by the convention of starting the name/key of the attribute with an underscore character (e.g. _meta). If you are interfacing with the API, you will frequently come across metadata. You're welcome to modify the metadata of a resource (if you know what you're doing) or add metadata attributes for known metadata keys, but you shouldn't create your own attributes with keys that begin with an underscore. Future versions of ProVision will use new metadata keys without warning, and if there is a naming conflict, your data could be lost.

Examples

These examples show the different methods that can be used to find and load a Resource object. They also show different data structures that are used to represent the object.

1 - PHP

Internal code example

To help users better understand how ProVision works, some of the examples in this documentation are of internal processes. They can contain code that only works when used as part of the core system and thus is not applicable to 3rd party development. The API is currently the only way for external tools to integrate with ProVision. Any example that contains internal code should be clearly labeled. Some common characteristics of these examples are code that doesn't use the API and code written in PHP (most example code will be in JavaScript).

This example uses the ResourceQuery class to find a resource object and then prints the result. It is included to show the similarity between finding a resource via the API and what happens under the hood.

```
$params = array(
    'slug' => 'tlr'
);
$resourceQuery = new ResourceQuery();
$resource = $resourceQuery->query($params);

var_dump($resource);
/*
array (size=1)
  0 =>
    object(Resource)[27]
      protected 'id' => string '1' (length=1)
      protected 'name' => string 'TLR' (length=3)
      protected 'slug' => string 'tlr' (length=3)
      protected 'type' => string 'resource' (length=8)
      protected 'parent_id' => null
      protected 'category_id' => null
      protected 'attr' =>
        array (size=0)
          empty
      protected 'loaded' => boolean true
*/
```

2 - API request

This is a standard API request, the request data is urlencoded and the result is JSON

/api/v1/api.php?target=resource&action=get&slug=TLR

```
{
  "success": 1,
  "message": "Search successful",
  "data": [
    {
      "id": "1",
      "name": "TLR",
      "slug": "tlr",
      "type": "resource",
      "parent_id": null,
      "category_id": null,
      "attr": {}
    }
  ]
}
```

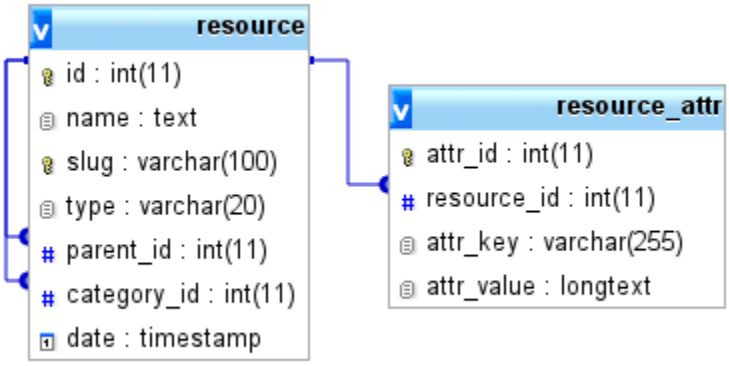
Database Layout 1

Database Layout

Details of the database and tables used by the RS are not necessary and should have no bearing on usage or API based development. However, a visualization of these tables may help some users better understand how the RS works, so they are provided below.

- Database Layout
  - Figure
  - Relations
  - Structure in SQL

Figure



Relations

`resource`.`category_id` -> `resource`.`id`

`resource`.`parent_id` -> `resource`.`id`

`resource_attr`.`resource_id` -> `resource`.`id`

Structure in SQL

resource

```
--
-- Table structure for table `resource`
--
CREATE TABLE IF NOT EXISTS `resource` (
`id` int(11) NOT NULL,
  `name` text NOT NULL,
  `slug` varchar(100) NOT NULL,
  `type` varchar(20) NOT NULL,
  `parent_id` int(11) DEFAULT NULL,
  `category_id` int(11) DEFAULT NULL,
  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=InnoDB  DEFAULT CHARSET=utf8 AUTO_INCREMENT=1115 ;
--
-- RELATIONS FOR TABLE `resource`:
--   `category_id`
--       `resource` -> `id`
--   `parent_id`
--       `resource` -> `id`
--
--
-- Indexes for dumped tables
--
--
```

```
-- Indexes for table `resource`
--
ALTER TABLE `resource`
 ADD PRIMARY KEY (`id`), ADD UNIQUE KEY `slug` (`slug`), ADD KEY `category_id` (`category_id`), ADD
KEY `parent_id` (`parent_id`);
--
-- AUTO_INCREMENT for dumped tables
--
--
-- AUTO_INCREMENT for table `resource`
--
ALTER TABLE `resource`
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT,AUTO_INCREMENT=1115;
--
-- Constraints for dumped tables
--
--
-- Constraints for table `resource`
--
ALTER TABLE `resource`
ADD CONSTRAINT `resource_ibfk_1` FOREIGN KEY (`category_id`) REFERENCES `resource` (`id`) ON DELETE
SET NULL ON UPDATE CASCADE,
ADD CONSTRAINT `resource_ibfk_2` FOREIGN KEY (`parent_id`) REFERENCES `resource` (`id`) ON DELETE SET
NULL ON UPDATE CASCADE;
```

resource_attr

```
--
-- Table structure for table `resource_attr`
--
CREATE TABLE IF NOT EXISTS `resource_attr` (
`attr_id` int(11) NOT NULL,
  `resource_id` int(11) NOT NULL,
  `attr_key` varchar(255) NOT NULL,
  `attr_value` longtext NOT NULL
) ENGINE=InnoDB  DEFAULT CHARSET=utf8 AUTO_INCREMENT=6744 ;
--
-- RELATIONS FOR TABLE `resource_attr`:
--   `resource_id`
--       `resource` -> `id`
--
--
-- Indexes for dumped tables
--
--
-- Indexes for table `resource_attr`
--
ALTER TABLE `resource_attr`
 ADD PRIMARY KEY (`attr_id`), ADD KEY `item_id` (`resource_id`);
--
-- AUTO_INCREMENT for dumped tables
--
--
-- AUTO_INCREMENT for table `resource_attr`
--
ALTER TABLE `resource_attr`
MODIFY `attr_id` int(11) NOT NULL AUTO_INCREMENT,AUTO_INCREMENT=6744;
--
-- Constraints for dumped tables
--
--
-- Constraints for table `resource_attr`
--
ALTER TABLE `resource_attr`
ADD CONSTRAINT `resource_attr_ibfk_1` FOREIGN KEY (`resource_id`) REFERENCES `resource` (`id`) ON
DELETE CASCADE ON UPDATE CASCADE;
```

Asset System

The Asset System

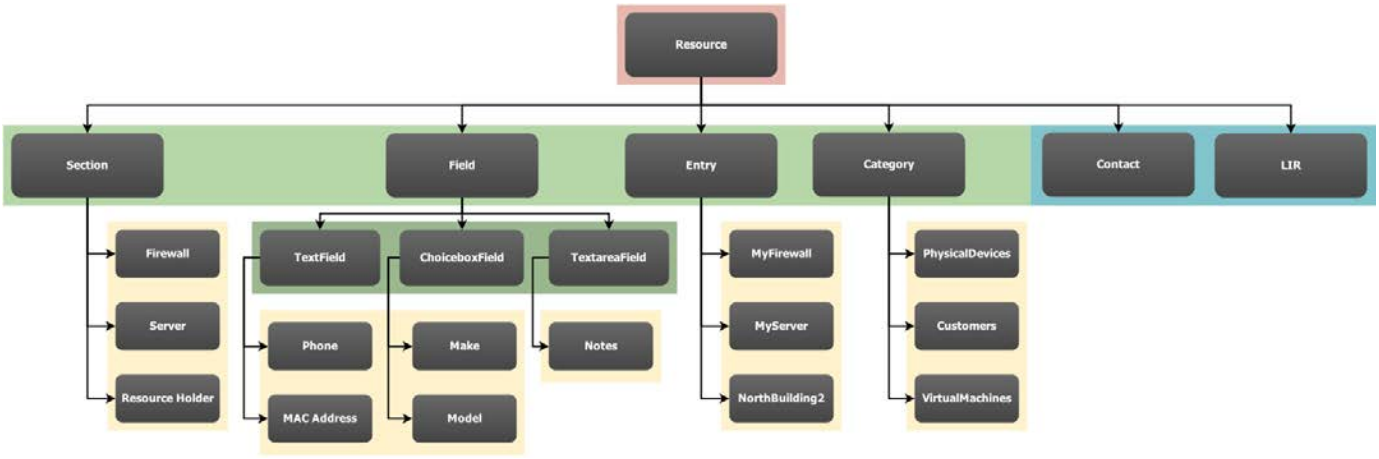
- The Asset System
  - Prerequisites
  - Overview
  - Introduction
  - Components
    - Section
    - Entry
    - Field
    - Category
    - Gadgets

Prerequisites

Some knowledge of object orientated programming (OOP) is recommend to understand the following description of the Asset System. If you are unfamiliar with OOP concepts, I would recommend reading a tutorial such as this one (http://docs.oracle.com/javase/tutorial/java/concepts/index.html) provided by Oracle or this one (http://msdn.microsoft.com/en-us/library/ca22fyhc(v=vs.90).aspx) provided by Microsoft, to help you understand terms like class, object, instantiate, property, method, and others.

Overview

The asset system is a content management system (CMS) that is built as an extension to the resource system. It's the main use of the resource system, and to many, the terms "asset system" and "resource system" can seem synonymous. In the diagram below, the Resource class is at the top in red. The child-classes that make up the asset system are in green. Yellow is used for examples of objects (not classes) that could/would have been instantiated from their Class. And the items in blue are examples of resource child-classes (resource types) that exist outside of the asset system.



Introduction

When writing software, the developer creates classes. A class is like a blueprint for objects. The class defines the properties and methods that the future objects will have, and like blueprints, multiple objects can be created from a single class. The Resource Class is a class, and each resource "type" (e.g. Section, Field, Contact, ect.) has a class, something which has been written in core code and cannot be changed by the user. The purpose of the asset system is to reproduce this fundamental low-level class-object system in such a way that the user can create their own classes, properties, methods, and objects without needing to dive into the code.

Components

Section

Sections are like classes, they are the templates/blueprints of the asset system. To create the structure of the blueprint, the user assigns fields (i.e. properties) and sometimes gadgets (i.e. methods) to the section.

Entry

Entries are the objects of the asset system. An entry cannot be created without a section to use as its blueprint. Creating an entry from a section is like instantiating an object from a class.

Field

Fields are the properties of the class. Field has its own child-classes; this is to accommodate the different types of fields. For example, when creating a class *Car*, the developer might give the *Car* class the property *String color*. In a similar fashion, a user of the Asset System could create a Section called *Truck*, a TextField called *color*, and then assign that textfield to the section. When the user goes to create an entry from the section *Truck*, they'll be given the option to include a text value for the field *color*.

Fields also have a use beyond acting as properties for classes. The field object (in this case *color*) is a resource object in it's own right. This means it can be modified independently of the sections that have assigned it and the entries that are using it. For example, a field which shows a dropdown box of several options could be modified to include more options; any entry which is using that field would automatically receive those new changes. Or consider a simple textfield object called "MAC Address" that is used by several sections and entries. If that field was modified to include a filter that checks the input for a valid MAC string, any entry using that field would get those improved validation checks.

Also, because the same field object can be assigned to multiple sections, it's easier to find entries by their values because they're all using the exact same field object. The alternative would have to be a blind text search to try and find different objects but with contextually similar values, and that method is notoriously unreliable.**This is why it's encouraged to assign the same field object to different sections as opposed to just making new fields each time.**

Fields are like what you might call class properties or class variables, but they've also got a lot more functionality available for when you need it.

Category

Categories are just an organizational tool. There is a clearly defined relationship between Sections, Entries, and Fields, but Categories exist on their own. If you look on the Classes page, you'll see that every Resource has the same 6 fundamental properties and 3 of them are ID values. The first is the ID that belongs to the resource itself, the second is the ID of the resource's parent, and the third is the ID of the Category that the resource belongs to (if any). There isn't a strict hierarchy here, how you use categories is entirely up to you. You can create categories, child categories, and careful plan exactly how you want the resources in your system to be organized. Or you can ignore the whole thing completely and just let every resource have the default category of "uncategorized." Many user find that the ability to create hierarchal parent-child relationships with entries, and then filter down results even further by Section, leaves the use of Categories unnecessary. But if you want to use them, it's there.

Gadgets

Gadgets are not resources, which is why they're not included in the chart at the top of the page. Gadgets are self contained applications and are limited to only using HTML, CSS, and JavaScript. All they know about the page that they're loaded on is the ID of the resource. However, because gadgets can interact with the API via JavaScript/AJAX, they're the perfect way to add new features to the asset system in a maintainable and modular way. At it's core, the asset system just allows users to create entries and then modify their text based attributes through a simple form. The ability for gadgets (such as the IPAM-Gadget) to interact with the API, is what makes the asset system so powerful.

Currently, the only gadgets that can be assigned to sections are gadgets that have been created by 6Connect. However our API is robust enough that almost anything you can do through ProVision could be recreated in the form of an isolated gadget. And because they're just made from html and javascript, it shouldn't be too strenuous for anyone to write a gadget of their own. If you want to create your own gadgets, it would be recommended to email us first with an outline of what you're trying to do. Then the recommended procedure would be to first create it as a standalone HTML/Javascript webpage that connects to our API (you may need to disable cross domain request security in your browser to make the AJAX connections work). Once you have your standalone page working, the process to turn that into an embeddable gadget is trivial.

Note: Gadgets are initialized as AngularJS applications. Both the AngularJS and jQuery libraries will be loaded on the page and available to use, but it is highly recommend to make the entire gadget in the form of an AngularJS app. But as noted above, it's best to contact us first so we can help you in the right direction.

Portable Gadgets

Portable Gadgets

- Portable Gadgets
  - Overview
    - Global Search:
      - Global Search Option Types:
    - IPAM Search:
      - IPAM Option Types:
    - DHCP Search:
      - DHCP Option Types:
    - Log:
      - More Options:
  - Additional Information:

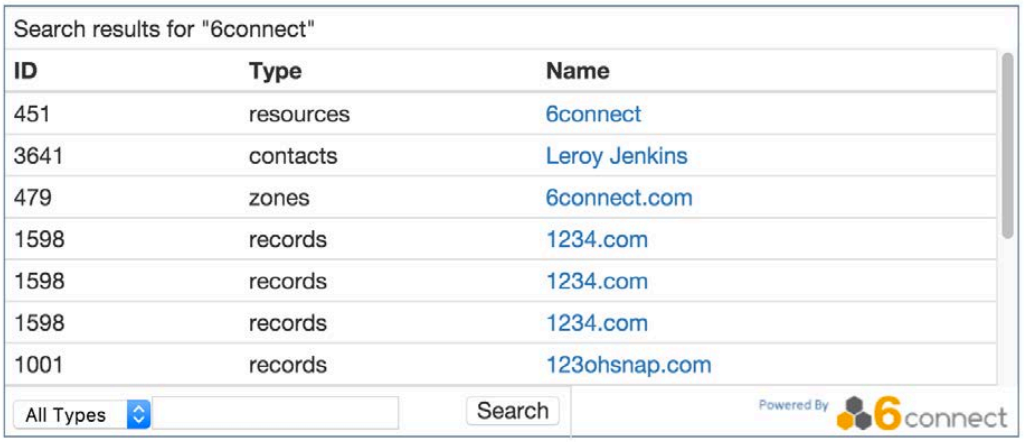
Overview

ProVision's Portable Gadgets are drop-in code snippets that use the ProVision API to bring in data to other systems or web pages. Portable Gadgets allow for quick data access and increased integration.

Currently, four Gadgets are available: Global Search, IPAM Search, DHCP Search, and Log. Each Gadget comes in various default option types for display style, number of records return, or behavior. Additional client-side style customizations may be made to further integrate the gadget with company styles.

Global Search:

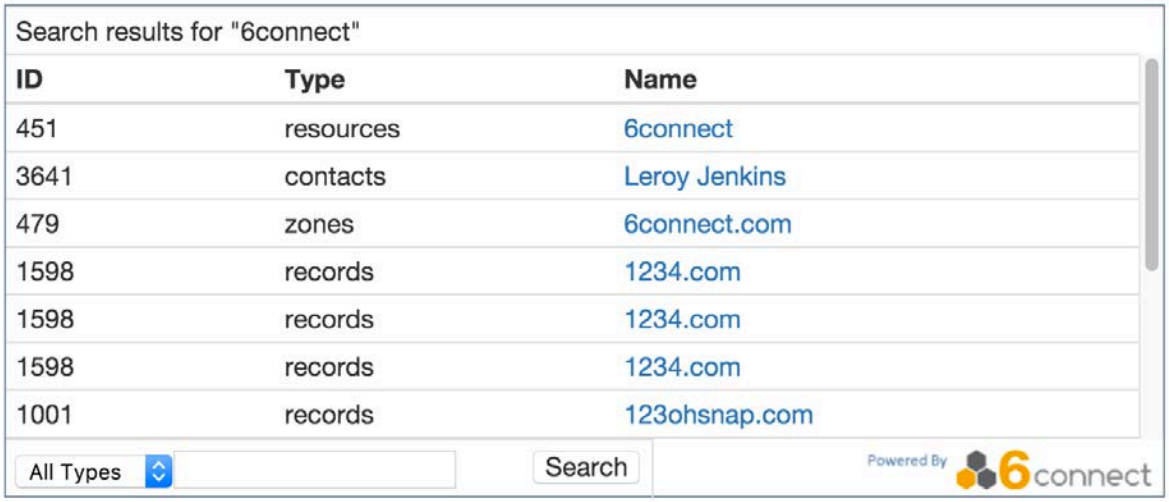
The Global Search Gadget utilizes ProVision's global search API to searching by name string and return ID, Type, and the Name. The search may be limited to specific types (IPAM, Resources, Zones, Records, Contacts, or All Types) if desired.



Global Search Option Types:

Type I.

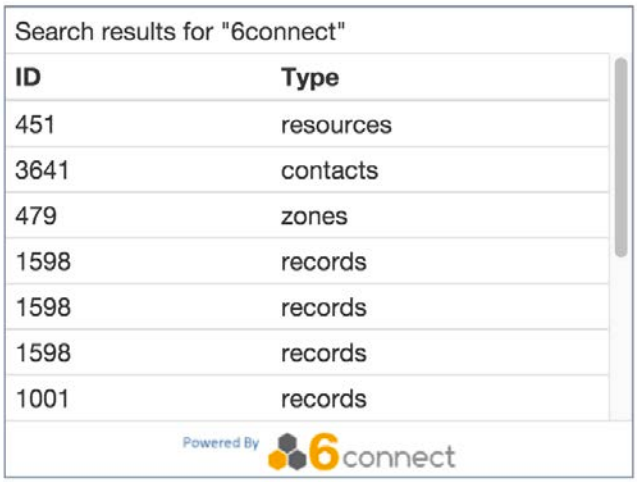
Click here to expand...



- Returns 5 records per Type
- Auto Width, height set to 300px

Type II.

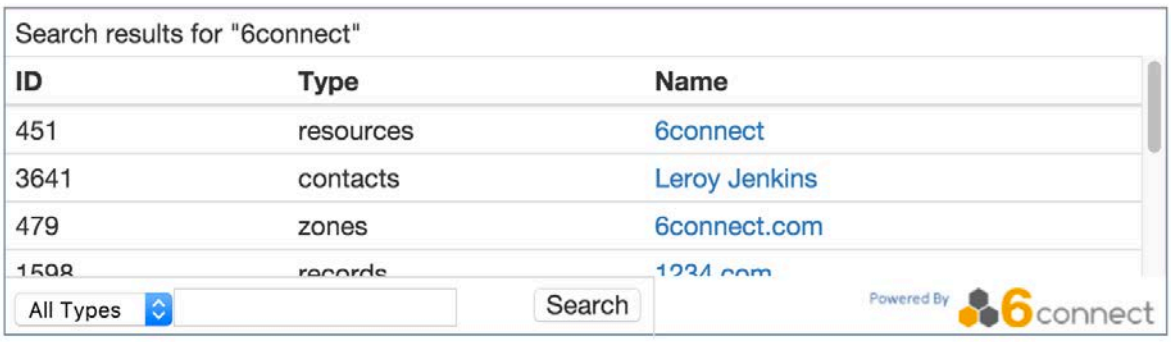
Click here to expand...



- Returns 5 records per Type
- No search input provided
- Fixed size (400px wide x 300px high)

Type III.

Click here to expand...



- Returns 3 records per Type

- Auto Width, height set to 200px

Type IV.

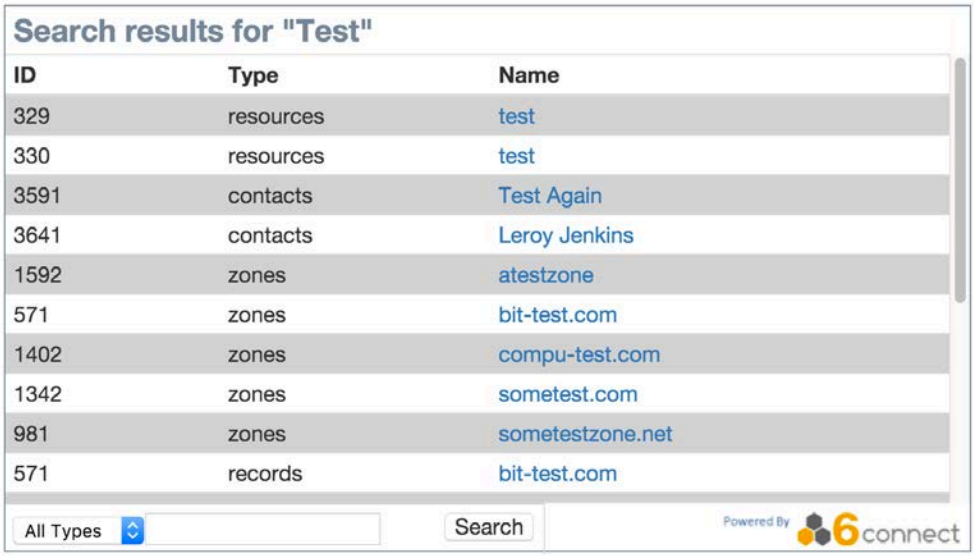
Click here to expand...



- No search on load
- Links stay in current window
- Fixed size (400px wide x 200px high)

Type V.

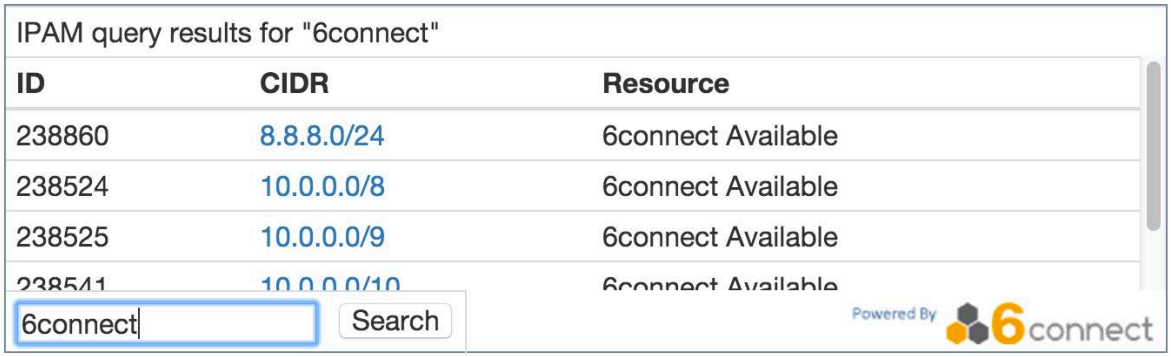
Click here to expand...



- Type I functionality restyled with CSS

IPAM Search:

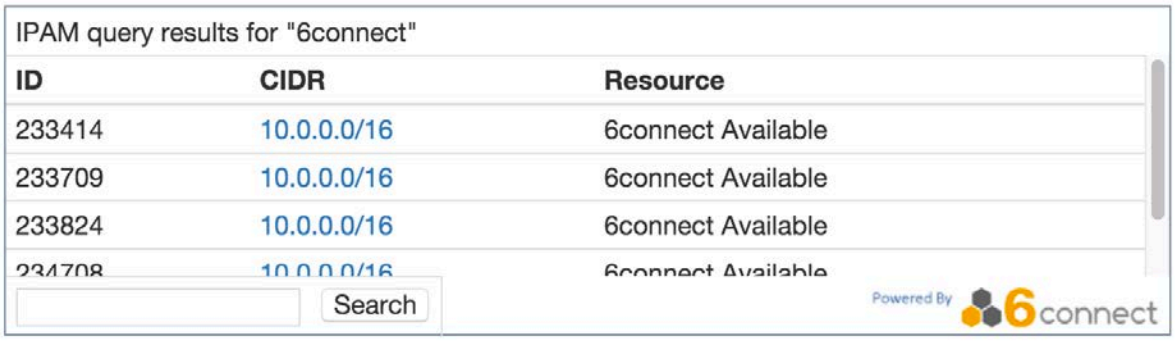
The IPAM Search Gadget utilizes ProVision's IPAM API to return ID, CIDR, and Resource Name results searching by an IPAM Resource name string.



IPAM Option Types:

Type I.

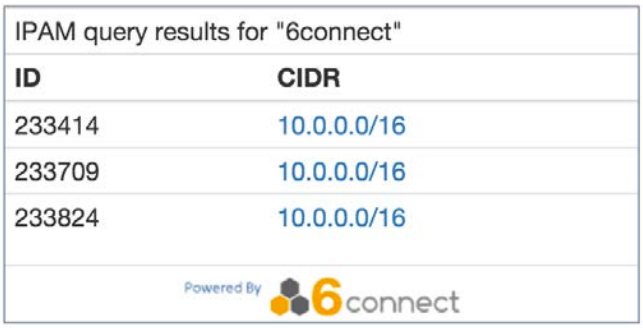
Click here to expand...



- Returns 5 records per Type
- Auto Width, height set to 200px

Type II.

Click here to expand...



- Returns 3 records per Type
- No search input provided
- Fixed size (400px wide x 200px high)

Type III.

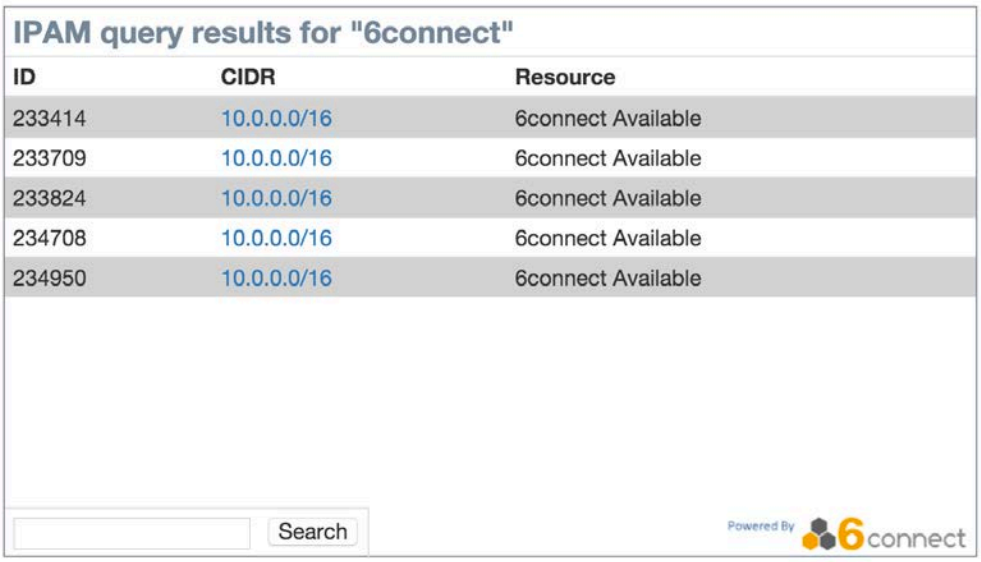
Click here to expand...



- No search on load
- Links stay in current window
- Auto Width, height set to 300px

Type IV.

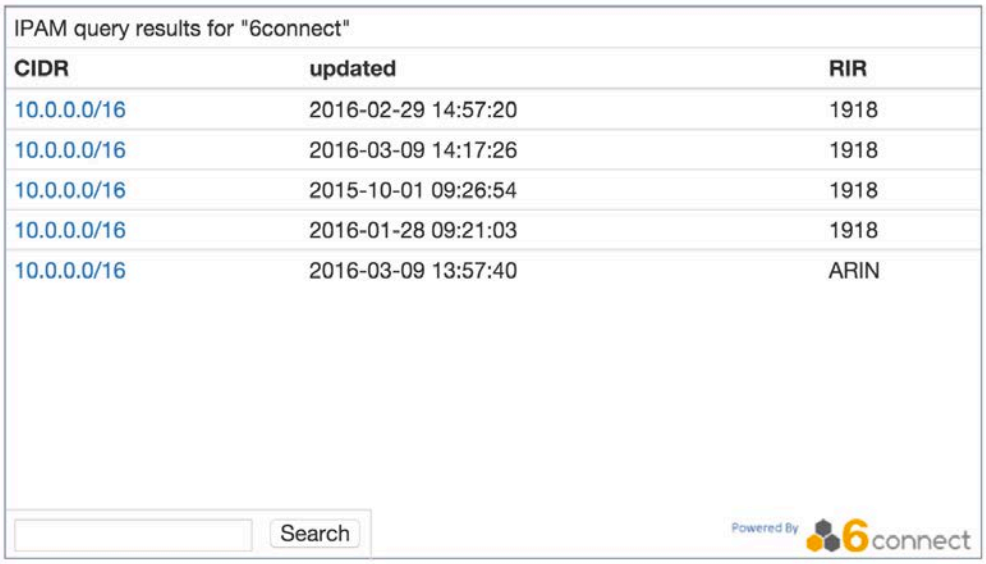
Click here to expand...



- Type I functionality restyled with CSS
- Auto Width, height set to 400px

Type V.

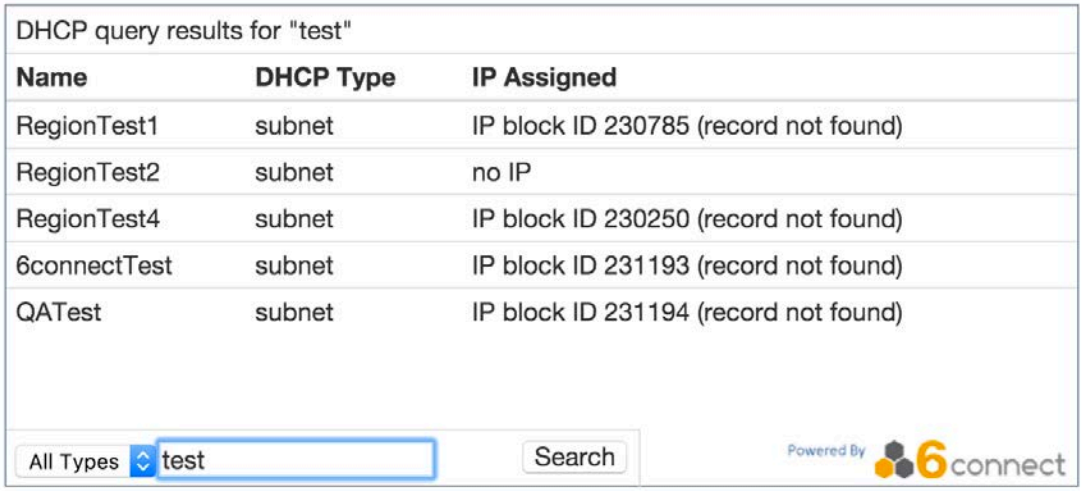
Click here to expand...



- Customizable Fields
- Auto Width, height set to 300px

DHCP Search:

The DHCP Search Gadget utilizes ProVision's DHCP API to return results of varying types when searching for a name string. The search may be limited to specific types if desired.



DHCP Option Types:

Type I.

Click here to expand...



- Returns 5 records per Type
- Auto Width, height set to 300px

Type II.

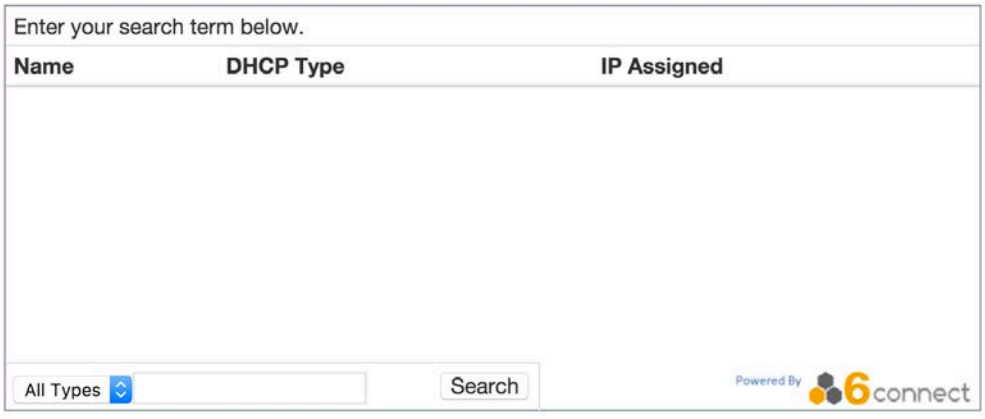
Click here to expand...



- Returns 3 records per Type
- No search input provided
- Fixed size (400px wide x 200px high)

Type III.

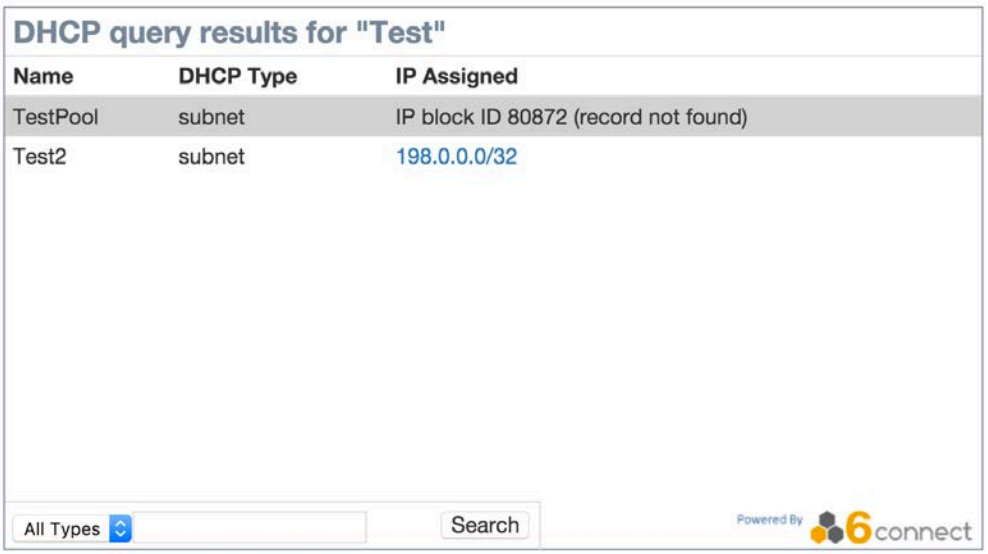
Click here to expand...



- No search on load
- Links stay in current window
- Returns 5 records per Type
- Auto Width, height set to 300px

Type IV.

Click here to expand...

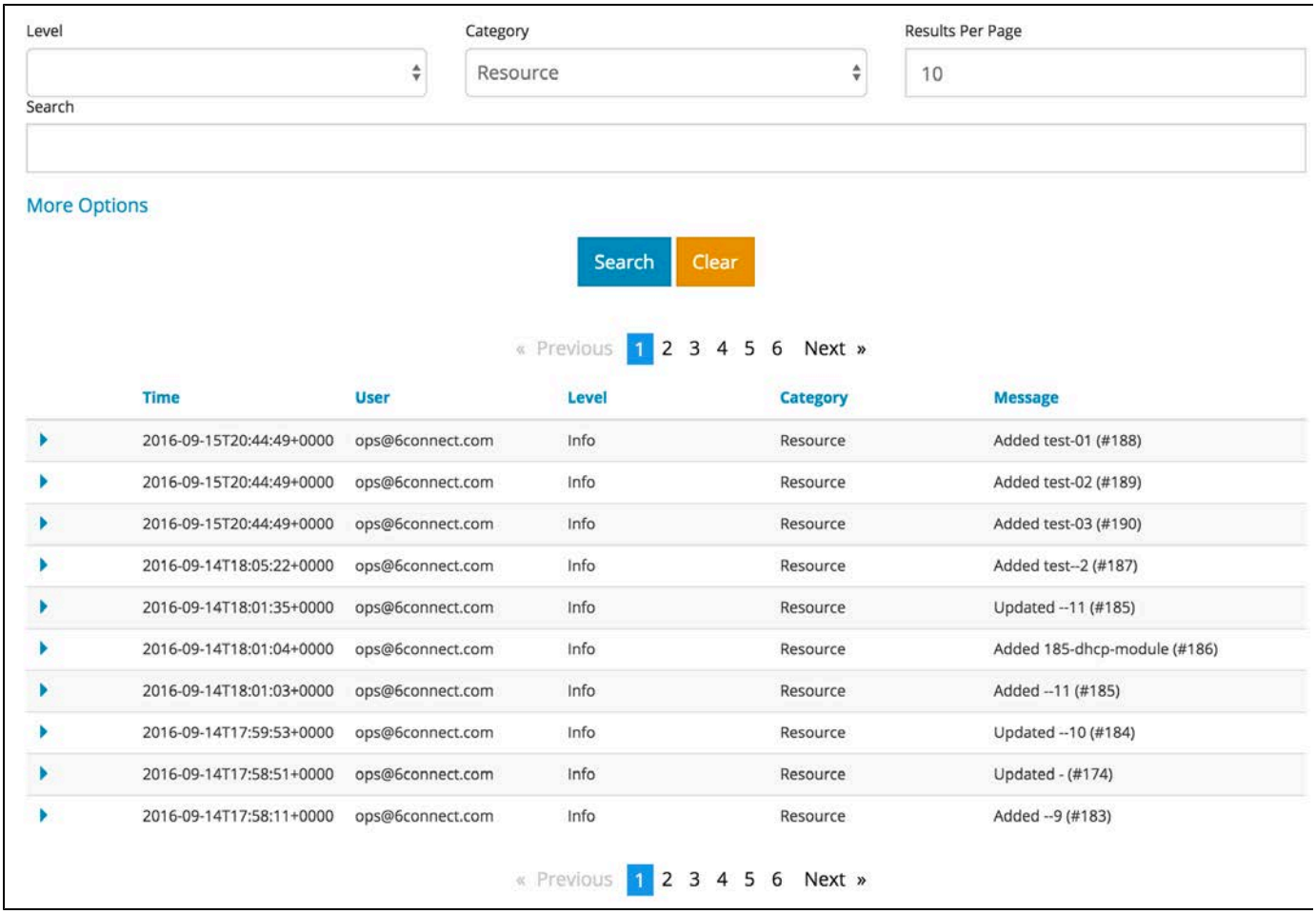


- Type I functionality restyled with CSS
- Returns 5 records per Type
- Auto Width, height set to 300px

Log:

The Log Gadget provides detailed information on actions performed in ProVision. Actions may be filtered by Level, Category, Log ID, User, and Time (specific or a range), as well as results-per-page option set. Additional filtering options allow targeting of specific log entries.

Clicking on the blue arrow at the beginning of each log entry shows additional details about that log item.



The following filters and options are available:

Level: Under the "Level" dropdown box, select "All Levels", "Emergency", "Alert", "Critical", "Error", "Warning", "Notice", "Info", or "Debug".

Category: Under the "Category" dropdown box, select "All Categories", "User", "IPAM", "Resource", "Resource Holder", "DNS", "Peering", "Assistant", "API", "NTP", "Device", or "Reporting".

Results per page: In the "Results per page" text box, type the desired number of log entries to see per page. By default, this value is set to 100.

Search: Type a search string, such as the name of a resource, then click the "Search" button.

More Options:

Additional detailed filter and search options are available under the "More Options" link.

Log ID: Retrieve a specific log entry from its log_id parameter.

IP: Search by IP of the machine that created the event in the log.

Username: Filter logs by ProVision username, or by "Unknown" user.

Time: Filter log entries by exact time created. Must be in datetime format (2016-08-14 16:41:18).

Time Minimum: The earliest day and time of log entries to show in results. Must be in datetime format (2016-08-14 16:41:18).

Time Maximum: The latest day and time of log entries to show in results. Must be in datetime format (2016-08-14 16:41:18).

Additional Information:

- Getting Started With Portable Gadgets
- Portable Gadget Customization

Getting Started With Portable Gadgets

Getting Started

- Getting Started
  - Adding a Portable Gadget to a Web Page
    - Requirements
    - Before You Begin
    - Step 1: Reference dependency files
    - Step 2: Create Gadget
    - Step 3: Initialize the Gadget
      - a) Preparing the settings
      - b) Initialize the gadget HTML element
    - Basic Page Example

Adding a Portable Gadget to a Web Page

Requirements

Setting up Portable Gadgets requires a ProVision instance URL and valid API credentials (API key and secret key), as well as some familiarity with HTML, CSS, and Javascript.

Before You Begin

Before you begin, make sure you have access to the source code for the web page on which you want the portable gadget to appear, or create a simple new web page for testing purposes with minimal tags like this (with the location of steps included as comments):

```
<html>
        <head>
                <!-- STEP 1 WILL GO HERE -->

                <!-- STEP 3a WILL GO HERE -->
                <!-- STEP 3b WILL GO HERE -->
        </head>
        <body>
                <!-- STEP 2 WILL GO HERE -->
        </body>
</html>
```

Step 1: Reference dependency files

Include the following two scripts the <HEAD> block. They should reference your provision installation.

In this example, "https://cloud.6connect.com/6c_123" is the root folder of a 6connect installation, so replace that with your own 6connect instance URL.

```
<script type='text/javascript' src="https://cloud.6connect.com/6c_123/portable/init.js"></script>
<link rel="stylesheet" type="text/css" href="https://cloud.6connect.com/6c_123/portable/css/provision-
portable.css">
```

Step 2: Create Gadget

Place the html tag for your gadget where you want it to appear in the page <body>, and give it a unique id.

Your gadget should be an html element in your page body (any contents inside the element will be removed). This example has a div referred to as "gadget" (as the ID).

```
<!-- provision gadget -->
<div id="gadget"></div>
```

Step 3: Initialize the Gadget

a) Preparing the settings

First, a settings variable must be prepared with the details of the gadget.

It must have the following keys for the provision url and api keys:

- provision_location
- provision_api_key
- provision_secret_key

...as well as the name of one of the gadget to load:

- search -- does a search with the global search API
- dhcp -- does a search with the DHCP API
- ipam -- does a search with the IPAM API
- pv-logs – details actions taken in ProVision.

Here is an example for a search gadget, loading with the search term "6connect". By default it will load up to five results per type.

```
<script type='text/javascript'>
var settings = {
provision_location: "https://cloud.6connect.com/6c_123",
provision_api_key: "00-ABCDEFGHIJKLMN00",
provision_secret_key: "00abcd11ef22ghij3300klmno123",

search: "6connect",
};
</script>
```

Here is an example to load as an IPAM gadget, with an option to limit results to 10 records:

```
<script type='text/javascript'>
var settings = {
provision_location: "https://cloud.6connect.com/6c_123",
provision_api_key: "00-ABCDEFGHIJKLMN00",
provision_secret_key: "00abcd11ef22ghij3300klmno123",

ipam: "6connect",
limit: 10
};
</script>
```

An example with the DHCP gadget:

```
<script type='text/javascript'>
var settings = {
provision_location: "https://cloud.6connect.com/6c_123",
provision_api_key: "00-ABCDEFGHIJKLMN00",
provision_secret_key: "00abcd11ef22ghij3300klmno123",

dhcp: "Test",
};
</script>
```

b) Initialize the gadget HTML element

The gadget will initialize when scripted to do so with the provision() command. For the example html in section II, we can instantiate this gadget by running the following (as long as settings was prepared):

```
<script type='text/javascript'>
provision('#gadget', settings)
</script>
```

The first parameter targets the html element that is the gadget, using jQuery's selector format. In this case, it is an element with the ID "gadget".

Basic Page Example

Here is a simple example html page with a search gadget. In this example, a default search term of "6connect" is included, and the return limited 10 records.

```
<html>
        <head>
                <script type='text/javascript' src="https://cloud.6connect.com/6c_123/portable/init.
js"></script>
                <link rel="stylesheet" type="text/css" href="https://cloud.6connect.com/6c_123
/portable/css/provision-portable.css">
                <script type='text/javascript'>
                        var settings = {
                                provision_location: "https://cloud.6connect.com/6c_123",
                                provision_api_key: "00-ABCDEFGHIJKLMN00",
                                provision_secret_key: "00abcd11ef22ghij3300klmno123",

                                search: "6connect",
                                limit: 10
                        };
                        provision('#search_gadget', settings);
                </script>
        </head>
        <body>
                <div id="search_gadget"></div>
        </body>
</html>
```

Portable Gadget Customization

Customizing Portable Gadgets

- Customizing Portable Gadgets
- Customization Options
    - Required Settings:
    - Optional Settings:
- Sample CSS Customization
    - Alternating Row Formatting:

Customization Options

Portable Gadgets come with both built-in customization options as well as the ability to edit styles locally through CSS.

Required Settings:

The required fields are:

provision_location: the location of the provision installation
provision_api_key: available from the Admin API Tab in ProVision
provision_secret_key: available from the Admin API Tab in ProVision

Other required fields (by gadget)

Search:
    search: "string"
IPAM:
   ipam: "string"
DHCP:
   dhcp: "string"

Where, in each case, "string" is the search term used on the initial loading of the gadget. If it is an empty string (literally, ""), then gadget will load without a pre-defined search.

Optional Settings:

These settings must be included as parameters when initializing the gadget. See Getting Started With Portable Gadgets.

limit: positive integer, or -1

The record return limit of the gadget. In some gadgets (search and DHCP) there are multiple types, so it limits per type. The default limit, if this option is not provided, is 5. If you specify limit a -1 (or basically any negative number), it will not enforce a limit.

links_change_window: true

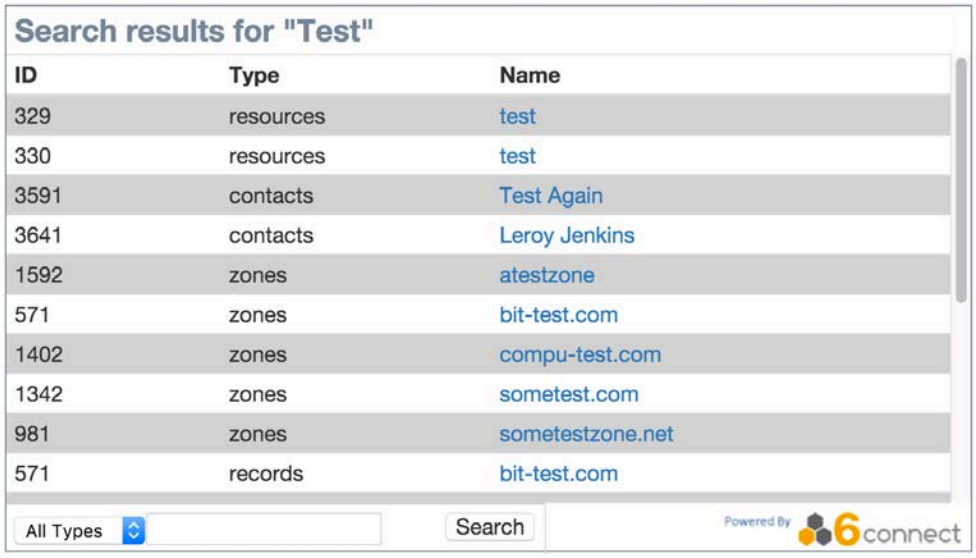
setting this to true (not string "true" but actual true type in javascript) will make links change the current window. The default behavior (not having this option or false), would have links open a new tab.

interact: false

By default, there will be a search box on all these gadgets. But with this option set to false (again, using false not string "false") will remove that box so search cannot be changed.

Sample CSS Customization

Alternating Row Formatting:




```
<style>
.pg-subtitle {
    color: lightslategrey;
    font-size:1.4em;
    font-weight: bold;
}

.pg-field-name a:hover {
    text-decoration: none;
    color: white;
    background-color: gray;
}

.pg-results tr:nth-child(2n+1) {
    background: white;
}
.pg-results tr:nth-child(2n+0) {
    background: lightgray;
}
</style>
```

Help & Support

Help & Support

For setup assistance or additional information, you can contact our support team at support@6connect.com.

For tutorials, frequently asked questions, feedback, or additional resources such as import templates and previous documentation versions, please follow the links listed below.

Table of Contents

- Tutorials
- FAQ
- Additional Resources
- Feedback and Feature Requests

Tutorials

Tutorials

Here we have grouped together video tutorials for various tasks and UI components. We link to these in the Getting Started area in the documentation, but you can also browse them individually depending on your needs. If you have suggestions for content - please send them to support@6connect.com.

Table of Contents

- Common Tasks
- UI Tours

Common Tasks

Here is a list of shortcuts to walkthroughs and information on performing common tasks in ProVision:

IPAM

IP Blocks

Adding/Editing blocks

Aggregating/Splitting blocks

SWIP configuration and use

RPSL configuration and use

IP Import

Import Aggregate Blocks

IP Import from CSV

DNS

Importing DNS Data

Import DNS Zones:

BIND DNS Zone Upload / Import

PowerDNS Zone Import

Peering

Peering Common Tasks

Adding routers

Adding sessions

Importing sessions

Resources

Importing Resource Data

Resource Import from CSV

Resource Importer Walkthrough

UI Tours

- General
  - Getting to know ProVision: First Steps - Part1
    - ProVision v5.1.x
  - Getting to know ProVision: First Steps - Part2
    - ProVision v5.1.x
- Administration
  - Admin Preferences Page Overview
    - ProVision v6.0.x
    - Previous Versions:
    - ProVision v5.1.x
  - Managing Group and User Permissions
    - ProVision v5.1.x
  - DNS BIND Zone Importing
    - ProVision v5.1.x

General

Getting to know ProVision: First Steps - Part1

ProVision v5.1.x

Getting to know ProVision: First Steps - Part2

ProVision v5.1.x

Administration

Admin Preferences Page Overview

ProVision v6.0.x

Previous Versions:

Click here to expand...

ProVision v5.1.x

Managing Group and User Permissions

ProVision v5.1.x

DNS BIND Zone Importing

ProVision v5.1.x

FAQ

FAQ

How can take a manual backup of ProVision?

From the command line in your ProVision instance, you can get a manual SQL backup using the following way:

```
mysqldump -uUSER -pPASSWORD DATABASE > bkup.sql
```

You can get the values for USER, PASSWORD, and DATABASE by reading them from the following file:

```
[ProVision Root]/data/globals.php
```

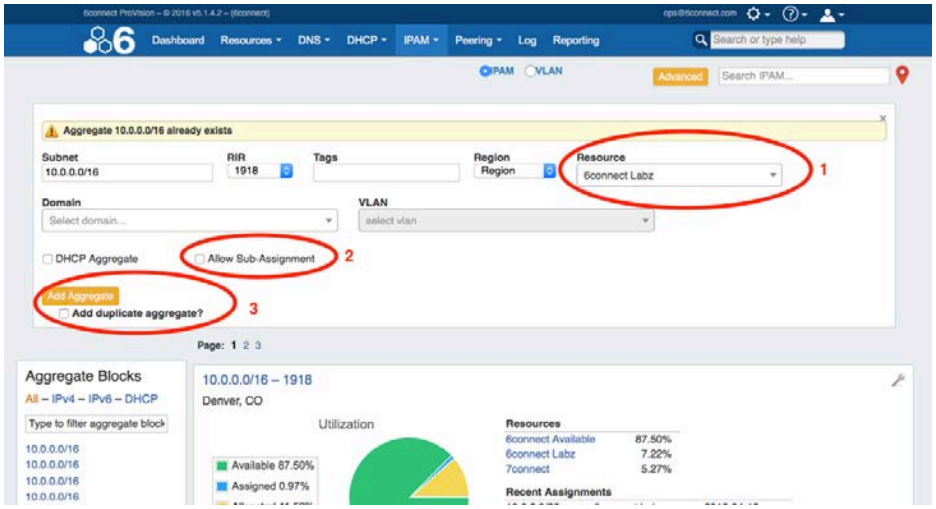
An example of a completed command:

```
mysqldump -uroot -p6connect_DBPSWD qa_5_1_5 > bkup.sql
```

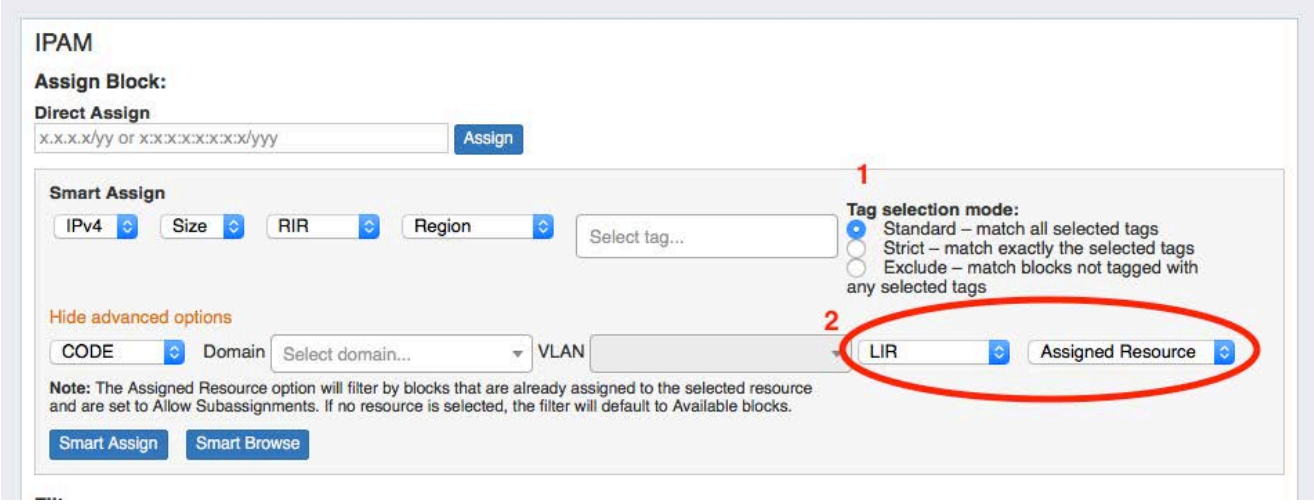
You can then use this .sql file to restore the old version should you need to revert the installation.

How can I manage overlapping/duplicate IP blocks?

When managing duplicate/overlapping IP ranges withing ProVision, there are several methods you can use to ensure that blocks are "unique" and still leverage functions like Smart Assign. The easiest approach is to use metadata for the aggregate (LIR, Region, Resource, etc.). From the ProVision GUI, you can use the "Add Aggregate" function on the IPAM tab (image below) to add a duplicate aggregate. By assigning the new Aggregate to a Resource (#1) you will tie the block and a Resource together. You most likely wish to make assignments out of this new aggregate, so you can enable subassignments (#2) before adding the aggregate (#3).



Now the block will be assigned to a Resource, and you can still provision smaller blocks out of these duplicate parent blocks. When using the IPAM gadget from the Resource screen, just go the Advanced section and you will have more options available for assigning IP data. If you have "strict" enabled, your Smart Assign function should work as expected when you use the LIR or Assigned Resource fields for blocks.



Does ProVision support configuring a DNS server SOA that is different from the actual server name?

Yes - in DNS Admin, simple specify the DNS Servers like normal and then attach the zones as needed with the desired SOA.

On the dashboard, I see "n+1" users - why?

The users list includes a "system user" that is only used by ProVision internally in the application.

I have already SWIPed subnets to ARIN. What happens if I try to SWIP from ProVision, but the block is already SWIPed?

In the case when a user already has SWIPped blocks to ARIN, 6connect checks prior to actually performing a SWIP. In the process, if the IP block is already SWIPped, it will check for existing ARIN customer data and update the 6connect data to reflect what ARIN has on file. Once that is complete, the user can then perform a de-SWIP function using ProVision.

How does 6connect avoid duplicate assignments or resolve conflicts?

When you make an API request to assign a block, if the block is already assigned to another resource, you will receive an error. If your process is to search for and then assign blocks, the Smart Assign API call may be very helpful. That call combines the search and assignment into one action.

My VM works, but I am getting a "URL Not Found" error when using ProVision

Please make sure that URL rewriting is enabled in your instance (apache mod_rewrite)

My DNS zone views aren't working as they should!

In some legacy instances we have seen zone record-view linkages come out of alignment and result in unexpected behavior.

BACKUP YOUR DATABASE

Please note that the following mysql commands modify your database! Please take a backup copy of your database before performance any database modifications.

First, verify the error with the following mysql commands:

```
SELECT count(*) FROM `zone_server_linkage` as t1
INNER JOIN `records` as t2 ON t1.`zoneid` = t2.`zone_id`
INNER JOIN `dns_views` as t3 ON t1.`serverid` = t3.`server_id` AND
`name` = '_6connectDefault'
LEFT JOIN `dns_view_record_linkage` as t4 ON t2.`id` = t4.`record_id`
AND t3.`id` = t4.`view_id`
WHERE t4.`id` IS NULL;
```

If the reply comes back non-zero, then your database is most likely exhibiting unexpected behavior.

The following mysql commands will re-align all the record-view linkages:

```
INSERT INTO `dns_view_record_linkage` SELECT '', t2.`id` as `record_id`,
t3.`id` as `view_id` FROM `zone_server_linkage` as t1
INNER JOIN `records` as t2 ON t1.`zoneid` = t2.`zone_id`
INNER JOIN `dns_views` as t3 ON t1.`serverid` = t3.`server_id` AND
`name` = '_6connectDefault'
LEFT JOIN `dns_view_record_linkage` as t4 ON t2.`id` = t4.`record_id`
AND t3.`id` = t4.`view_id`
WHERE t4.`id` IS NULL;
```

Contact support(support@6connect.com) if you have any additional questions or this does not resolve the issue.

How can I 'reserve' IP space?

To create a reserved pool of IP space, you can create a Section called "Reserved", add the IPAM gadget to it, then create an Entry with that Section to be the address group. From there, use the IPAM gadget and the IPAM Manage page to assign and unassign IP space from that pool.

The workflow for this would be:

1. Assign IP space to the "Reserved" Section.
2. When you are ready to pull space from "Reserved", unassign the desired block. This moves it to the holding tank.
3. Override the holding tank to make the space "available". This can be done in the IPAM manager via the "Override Holding" wrench option, or a manual 'pull out of holding' API call.
4. Assign the block to the desired Resource.

How do I change the URL of my ProVision instance?

Depending on your version of ProVision, you may need both steps. Edit the file <6connect web root>/data/globals.php and:

1) Change the $hostname variable to the new value

2) Change the $base_url to the new value

Please note that you may also need to update the SSL certs, httpd settings, etc.

Additional Resources

- Import Templates
- List of Abbreviations

Import Templates

Import Templates

Downloadable Import Templates

Below you can find CSV templates for uploading Resource, Contact and IP data.

For DNS Import examples and a walkthrough, visit the DNS Import page.

File	Modified
File IP-import-sample_v1.csv	Feb 21, 2017 by Christina Force
File import-zone-assign.csv	Feb 21, 2017 by Christina Force
Text File customer-import-sample.csv	Feb 21, 2017 by Christina Force
File contact-import-sample_v1.csv	Feb 21, 2017 by Christina Force

Drag and drop to upload or browse for files

Download All

List of Abbreviations

List of Abbreviations:

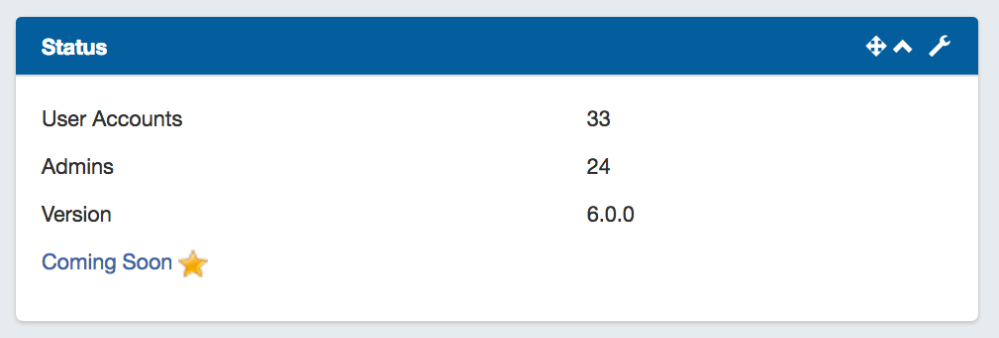
Edit Document

API	Application Program Interface
CLI	Command-line interface
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DNSSec	Domain Name System Security Extensions
IP address	Internet Protocol address
IPAM	Internet Protocol address management
LDAP	Lightweight Directory Access Protocol)
SDK	software development kit
SSH	Secure Shell

Abbreviation List.xlsx

Feedback and Feature Requests

For information on future releases, click on the "Coming Soon" link on the Dashboard Status Widget.



You can also submit product feedback and feature requests to support@6connect.com