

V0.1 08-08-08: deanj

Welcome to the XMDNS User Guide!

This guide should help familiarize one with XMDNS and how to add/modify/delete hosts. Server install information can be found in the document xmdns-cookbook-consolidated.

=====  
Overview  
=====

XMDNS uses Subversion to track changes made to the DNS system which allows problems to be found and versions to be rolled back if needed. It also allows multiple users to submit changes to the system in a trackable and collaborative manner. The X in XMDNS stands for XML. The XML markup language is used to store the DNS records which allows for checking the records for errors and invalid entries and notifying the user to correct the errors.

Each user must check out a copy of the repository in order to make modifications. The files to be modified are the local copy which is then checked for errors and valid records, and then checked back into the repository. Once the repository is updated the DNS servers periodically check for updates and pull down any new changes. Once the DNS servers pull down changes they reload their records if, and only if the data is valid and sane. If there are errors with the new DNS records the last known good version of the records are the ones published by the DNS servers. There is then no interruption of service.

DHCP records can also be generated from XMDNS and integrated with DHCP servers in a similar manner to the DNS integration.

XMDNS uses a menu driven system to add/modify/delete DNS records which helps simplify the complexity of DNS which can exist when using DNS Views and supporting multiple networks and domains.

=====  
Initial Repository Checkout:  
=====

The first order of business is checking out the repository. This task needs to only be done once, so if you already have a version of the repository then please skip this step and move to the 'Common Tasks' or '05-update-source --Updating' step.

To check out the repository one will need the Subversion client. SVN is installed by default on Mac OS X Leopard.

**svn co https://server.with.repository/svn/hosts ~/src/xmdns**

Please substitute the appropriate server name in the above command.

The word 'hosts' in the command is the name of the repository on the server. If the Cookbook has been followed /svn/hosts is valid. If the repository has been named something other than the Cookbook

default please substitute that name.

The '~/src/xmdns' in the example command is the directory store the local copy of the repository. This directory can be called whatever the users wishes and stored anyplace appropriate.

=====  
Common Tasks, short version:  
=====

For more detailed information for each step please move onto the '05-update-source -- Updating' section.

To add/remove/modify host entries you can choose to use the menu driven system or use the command line add host script.

**cd ~/src/xmdns**  
**./05-update-source**  
**./09-find-open-ip** (optional for finding free IP addresses within a range)  
**./10-edit-hosts** or **./10-add-host** (for quickly adding a host)  
**./30-update-zones**  
**./35-check-changes** (this is optional but recommended)  
**./60-commit-changes**

Each stage will guide you to the next.

=====  
05-update-source – Updating the local copy  
=====

Ensuring the local copy of the repository is up to date with the latest version of the working (server side) copy is critical to performing any task.

To update the local repository:

**cd ~/src/xmdns** (or wherever the local copy is..)  
**./05-update-source**

=====  
09-find-open-ip  
=====

This step is optional and used to search for free IP addresses within whatever range is specified. 10-edit-hosts automatically runs this script when the selection to add a new host is chosen.

If running 10-add-host it is suggested to run 09-find-open-ip first to find a free IP address.

Running 09-find-open-ip with the help flag:

## **./09-find-open-ip -h**

Usage:

find\_open\_ip.py [-vh] [-c max\_num] -s IP-portion

Examples:

Find an open IP in the 108 network

% find\_open\_ip.py -s 150.135.108

Find up to 10 open IPs on the 110 network

% find\_open\_ip.py -c 10 -s 150.135.110

Find open IPs on the 110 network and print

all scanned IPs up to the first free IP

% find\_open\_ip.py -v -c 10 -s 150.135.110

Find open IPs on the 100 network and print

the hex equivalent (useful for IPv6)

% find\_open\_ip.py -v --hex -c 10 -s 150.135.110

=====  
./10-add-host – Adding hosts  
=====

Using the 10-add-host script allows for adding a complete host entry with one command. This command also parses the XML and make sure the data is sane before making the requested changes.

This command is limited in scope and does not support every type of DNS record including not supporting IPv6 or text records. For full DNS record support please use 10-edit-hosts.

This command will generate the help information menu:

## **./10-add-host -h**

Usage:

./editor/bin/add.py --help

./editor/bin/add.py --hostname hostname {OPTIONS}

OPTIONS:

--domain domain

--a ip.v.4.address

--mx '## full.name.of.host.'

--description '...'

--macAddress 00:11:22:33:44:55

This command should be familiar to anyone who has used the LPL addhost script.

=====  
10-edit-hosts – Adding/modifying/deleting hosts  
=====

The 10-edit-hosts is the primary method for interaction with any type of record in the XMDNS system.

10-edit-hosts is a menu driven application and is invoked by running the command:

### **./10-edit-hosts**

The usual method for using 10-edit-hosts is to do either add a host, or do a search for a host and then modify that entry. Once all modifications are done saving the session (from the top level menu) and then quitting the program.

Menu breakdown:

The first menu within 10-edit-hosts offers a few options, Add new host, lookup/select host, save changes, and quit.

Selecting the add new host option will open a submenu which lists the different networks available. The network where the new host is to be added must be selected. Once the network selection has occurred the hostname of the new system will be requested. Once the hostname (shortname) is entered another menu will be displayed. This newly added host has no properties (other than its name) so selecting Modify will allow attributes to be added or deleted.

If a host is currently selected and it is being worked with the hosts name and domain will appear at the top of the screen so it is known which host is being modified.

Adding records to a host is very intuitive. The 'add record' menu lists all of the available record types that can be added to a host entry. When adding an IPv4 record a list of the first five available IP addresses within that range will be listed. Please use 09-find-open-ip to find a specific IP range if the low ranges are not desired.

Deleting records is also quite intuitive with a nice menu driven system. Each record for a host has a separate number which can be selected for deletion.

Once all changes to the host are made exit out of the host modification menu, and then exit out of the the host selection. At this point another host can be added, or looked up and modified. If no more changes are necessary Save Changes and quit.

Lookup/select host allows for a search for specific hosts by hostname, ip address, domain, or network. If the search method has more than one result within the current search parameters the option 'list hosts' will appear which will display all of those records. The search can be further refined if necessary.

Save changes will save any changes made and write out the data in xml.

Quit will quit the program. If changes have been made they must be saved before quitting the program. If an error has occurred and a change should not be saved quitting the program without saving will not modify the dataset. When quitting 10-edit-hosts the list of changed records will be displayed. It is suggested that this output be copied to the clipboard for later use as notes in the Subversion repository.

This is just a basic overview of the 10-edit-hosts menu system. 10-edit-hosts is intuitive and easy to work with so figuring out how to do any task is quite easy.

### =====

### 30-update-zones

### =====

This step is to be completed once all changes have been made by either 10-edit-hosts or 10-add-host. This step processes the main XML file and generates the zone files necessary for Bind for all networks.

### **./30-update-zones**

### =====

### 35-check-changes

### =====

35-check-changes outputs the changes that were just made for review by the operator. The changes are broken up by network (or view) for readability.

This step is optional but recommended.

### **./35-check-changes**

### =====

### 60-commit-changes

### =====

This is the final step documents the changes and submits the changes back to the Subversion repository. 60-commit-changes will open an editor which by default is vi. To set a different editor set the EDITOR environment variable to an alternate before running this command.

It is suggested to add the text output list of changes from Quitting 10-edit-hosts at this step. This makes tracking changes easy in case of a problem.

If 10-add-host was run instead of 10-edit-hosts or if the list of changed entries is not available to be added to the change log please describe what changes were made.

### **./60-commit-changes**

This is the last step in the process of adding or modifying DNS entries. The DNS servers will update their records (twice an an hour by the default install suggestion in the cook book document) and the new records will be available.

If immediate changes are desired the commands below can be run on the DNS server. These commands assume the cookbook was followed:

```
/opt/coolstack/bin/svn update /etc/bind  
/usr/sbin/rndc reload  
/usr/sbin/rndc status
```

The records will then the published on that server.

## =====

### Uncommon Tasks:

## =====

Adding new Views, Domains or networks is not automated within 10-edit-hosts. Adding any of these needs to be done manually and should be done carefully.

Entries can be added or modified directly instead of using 10-add-host or 10-edit-hosts but it is not suggested that this be done routinely.

To add hand crafted entries that do not fit in the normal scheme:

```
edit views/{network}/headers/db-{zone name}  
svn commit  
update ns servers (automatically done via cron)
```

To add a new domain:

```
edit res/db/domain_list.xml  
  (pattern match to add new entry from existing domain)  
run res/db/update_other
```

To add a new network:

```
edit res/db/network_list.xml  
  (pattern match to add new entry from existing network)  
run res/db/update_other
```

---

## Important Files:

---

This section is included for reference.

This tree shows how files are included from /etc/bind/named.conf  
most files are automatically generated, exceptions are noted below.

```
| /etc/bind/named.conf  -- hand generated general/global parameters to bind
|                       are placed here
\| /etc/bind/named.acls -- groups of hosts/networks are defined here for
|                       later use
| /etc/bind/named.views -- each view of the DNS tree is defined here with
|                       options
\| /etc/bind/views/{network}/named.conf
|                       -- Simplified named.conf file for each network view
\| /etc/bind/reverse/global-named.conf
||                       -- config for reverse resolution of
||                       all networks in all contexts
\| /etc/bind/reverse/headers/ipv[4|6]-{network}
| |                       -- config for reverse resolution of a
| |                       single network
| \| /etc/bind/reverse/zones/ipv[4|6]-{network}
| |                       -- config for reverse resolution of a
| |                       single network
\| /etc/bind/views/{network}/extra-reverse.conf
|                       -- config for special zones
\| /etc/bind/views/{network}/headers/db-{zone name}
|                       -- header for zone
\| /etc/bind/views/{network}/zones/db-{zone name}
|                       -- zone definition file
```