Mac OS X Server
Command-Line
Administration

For Version 10.3 or Later

Contents

About This Book

Notation Conventions

The following conventions are used throughout this book.

Summary

Notation	Indicates
`monospaced` font	A command or other terminal text
`$`	A shell prompt
`[text_in_brackets]`	An optional parameter
`(one|other)`	Alternative parameters (type one or the other)
<u>underlined</u>	A parameter you must replace with a value
`[...]`	A parameter that may be repeated
`<anglebrackets>`	A displayed value that depends on your server configuration

Commands and Other Terminal Text

Commands or command parameters that you might type, along with other text that normally appears in a Terminal window, are shown in `this` font. For example,

You can use the `doit` command to get things done.

When a command is shown on a line by itself as you might type it in a Terminal window, it follows a dollar sign that represents the shell prompt. For example,

`$ doit`

To use this command, type "doit" without the dollar sign at the command prompt in a Terminal window, then press the Return key.

Command Parameters and Options

Most commands require one or more parameters to specify command options or the item to which the command is applied.

Parameters You Must Type as Shown

If you need to type a parameter as shown, it appears following the command in the same font. For example,

```
$ doit -w later -t 12:30
```

To use the command in the above example, type the entire line as shown.

Parameter Values You Provide

If you need to supply a value, its placeholder is underlined and has a name that indicates what you need to provide. For example,

```
$ doit -w later -t hh:mm
```

In the above example, you need to replace hh with the hour and mm with the minute, as shown in the previous example.

Optional Parameters

If a parameter is available but not required, it appears in square brackets. For example,

```
$ doit [-w later]
```

To use the command in the above example, type either doit or doit -w later. The result might vary but the command will be performed either way.

Alternative Parameters

If you need to type one of a number of parameters, they're separated by a vertical line and grouped within parentheses (|). For example,

```
$ doit -w (now|later)
```

To perform the command, you must type either doit -w now or doit -w later.

Default Settings

Descriptions of server settings usually include the default value for each setting. When this default value depends on other choices you've made (such as the name or IP address of your server, for example), it's enclosed in angle brackets <>.

For example, the default value for the IMAP mail server is the host name of your server. This is indicated by mail:imap:servername = "<hostname>".

Commands Requiring Root Privileges

Throughout this guide, commands that require root privileges begin with sudo.

Typing Commands

<div style="text-align: right">1</div>

How to use Terminal to execute commands, connect to a remote server, and view online information about commands and utilities.

To access a UNIX shell command prompt, you open the Terminal application. In Terminal, you can use the `ssh` command to log in to other servers. You can use the `man` command to view online documentation for most common commands.

Using Terminal

To enter shell commands or run server command-line tools and utilities, you need access to a UNIX shell prompt. Both Mac OS X and Mac OS X Server include Terminal, an application you can use to start a UNIX shell command-line session on the local server or on a remote server.

To open Terminal:
▪ Click the Terminal icon in the dock or double-click the application icon in the Finder (in /Applications/Utilities).

Terminal presents a prompt when it's ready to accept a command. The prompt you see depends on Terminal and shell preferences, but often includes the name of the host you're logged in to, your current working directory, your user name, and a prompt symbol. For example, if you're using the default bash shell and the prompt is

```
server1:~ admin$
```

you're logged in to a computer named "server1" as the user named "admin" and your current directory is the admin's home directory (~).

Throughout this manual, wherever a command is shown as you might type it, the prompt is abbreviated as `$`.

To type a command:

▪ Wait for a prompt to appear in the Terminal window, then type the command and press Return.

If you get the message command not found, check your spelling. If the error recurs, the program you're trying to run might not be in your default search path. Add the path before the program name or change your working directory to the directory that contains the program. For example:

```
[server:/] admin$ serversetup -getAllPort
serversetup: Command not found.
[server:/] admin$ /System/Library/ServerSetup/serversetup -getAllPort
1
Built-in Ethernet
[server:/] admin$ cd /System/Library/ServerSetup
[server:/System/Library/ServerSetup] admin$ ./serversetup -getAllPort
1
Built-in Ethernet
[server:/System/Library/ServerSetup] admin$ cd /
[server:/] admin$ PATH = "$PATH:/System/Library/ServerSetup"
[server:/] admin$ serversetup -getAllPort
1
Built-in Ethernet
```

Correcting Typing Errors

To correct a typing error before you press Return to issue the command, use the Delete key or press Control-H to erase unwanted characters and retype.

To ignore what you have typed and start again, press Control-U.

Repeating Commands

To repeat a command, press Up-Arrow until you see the command, then press Return.

To repeat a command with modifications, press Up-Arrow until you see the command, press Left-Arrow or Right-Arrow to skip over parts of the command you don't want to change, press Delete to remove characters, type regular characters to insert them, then press Return to execute the command.

Including Paths Using Drag-and-Drop

To include a fully-qualified file name or directory path in a command, stop typing where the item is required in the command and drag the folder or file from a Finder window into the Terminal window.

Commands Requiring Root Privileges

Many commands used to manage a server must be executed by the root user. If you get a message such as "permission denied," the command probably requires root privileges.

To issue a single command as the root user, begin the command with `sudo`. For example:

```
$ sudo serveradmin list
```

You're prompted for the root password if you haven't used `sudo` recently. The root user password is set to the administrator user password when you install Mac OS X Server.

To switch to the root user so you don't have to repeatedly type `sudo`, use the `su` command:

```
$ su root
```

You're prompted for the root user password and then are logged in as the root user until you log out or use the `su` command to switch to another user.

Important: As the root user, you have sufficient privileges to do things that can cause your server to stop working properly. Don't execute commands as the root user unless you understand clearly what you're doing. Logging in as an administrative user and using `sudo` selectively might prevent you from making unintended changes.

Throughout this guide, commands that require root privileges begin with `sudo`.

Sending Commands to a Remote Server

Secure Shell (SSH) lets you send secure, encrypted commands to a server over the network. You can use the `ssh` command in Terminal to open a command-line connection to a remote server. While the connection is open, commands you type are performed on the remote server.

Note: You can use any application that supports SSH to connect to Mac OS X Server.

To open a connection to a remote server:

1 Open Terminal.

2 Type the following command to log in to the remote server:

```
ssh -l username server
```
where <u>username</u> is the name of an administrator user on the remote server and <u>server</u> is the name or IP address of the server.

Example: `ssh -l admin 10.0.1.2`

3 If this is the first time you've connected to the server, you're prompted to continue connecting after the remote computer's RSA fingerprint is displayed. Type `yes` and press Return.

4 When prompted, type the user's password (the user's password on the remote server) and press Return.

The command prompt changes to show that you're now connected to the remote server. In the case of the above example, the prompt might look like

```
[10.0.1.2:~] admin$
```

5 To send a command to the remote server, type the command and press Return.

To close a remote connection

▪ Type `logout` and press Return.

Sending a Single Command

You can authenticate and send a command using a single typed line by appending the command you want to execute to the basic `ssh` command.

For example, to delete a file you could type

```
$ ssh -l admin server1.company.com rm /Users/admin/Documents/report
```

or

```
$ ssh -l admin@server1.company.com "rm /Users/admin/Documents/report"
```

You're prompted for the user's password.

Updating SSH Key Fingerprints

The first time you connect to a remote server using SSH, the local computer asks if it can add the remote server's "fingerprint" (a security key) to a list of known remote computers. You might see a message like this:

```
The authenticity of host "server1.company.com" can't be established.
RSA key fingerprint is a8:0d:27:63:74:f1:ad:bd:6a:e4:0d:a3:47:a8:f7.
Are you sure you want to continue connecting (yes/no)?
```

Type `yes` and press Return to finish authenticating.

If you later see a warning message about a "man-in-the-middle" attack when you try to connect, it might be because the key on the remote computer no longer matches the key stored on the local computer. This can happen if you:
• Change your SSH configuration
• Perform a clean install of the server software
• Start up from a Mac OS X Server CD

To connect again, delete the entries corresponding to the remote computer (which can be stored by both name and IP address) in the file `~/.ssh/known_hosts`.

Important: Removing an entry from the known_hosts file bypasses a security mechanism that helps you avoid imposters and "man-in -the-middle" attacks. Be sure you understand why the key on the remote computer has changed before you delete its entry from the known_hosts file.

Notes on Communication Security and `servermgrd`

When you use the Server Admin GUI application or the `serveradmin` command-line tool, you're communicating with a local or remote `servermgrd` process.
• `servermgrd` uses SSL for encryption and client authentication but not for user authentication, which uses HTTP basic authentication along with Directory Services.
• `servermgrd` uses a self-signed (test) SSL certificate installed by default in /etc/servermgrd/ssl.crt/. You can replace this with an actual certificate.
• The default certificate format for SSLeay/OpenSSL is PEM, which actually is Base64 encoded DER with header and footer lines (from www.modssl.org).
• `servermgrd` checks the validity of the SSL certificate only if the "Require valid digital signature" option is checked in Server Admin preferences. If this option is enabled, the certificate must be valid and not expired or Server Admin will refuse to connect.
• The SSLOptions and SSLRequire settings determine what SSL encryption options are used. By default, they're set as shown below but can be changed at any time by editing /etc/servermgrd/servermgrd.conf, port 311.

```
SSLCertificateFile /private/etc/servermgrd/ssl.crt/server.crt
SSLCertificateKeyFile /private/etc/servermgrd/ssl.key/server.key
SSLCipherSuite
    ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
SSLOptions +StdEnvVars
```

Using Telnet

Because it isn't as secure as SSH, Telnet access isn't enabled by default.

To enable Telnet access:

```
$ service telnet start
```

To disable Telnet access:

```
$ service telnet stop
```

Getting Online Help for Commands

Onscreen help is available for most commands and utilities.

Note: Not all techniques work for all commands, and some commands have no onscreen help.

To view onscreen information about a command, try the following:
*   Type the command without any parameters or options. This will often list a summary of options and parameters you can use with the command.
    Example:

    ```
    $ sudo serveradmin
    ```

*   Type man command, where command is the command you're curious about. This usually displays detailed information about the command, its options, parameters, and proper use.
    Example:

    ```
    $ man serveradmin
    ```

    For help using the man command, type:

    ```
    $ man man
    ```

*   Type the command followed by a -help, -h, --help, or help parameter.
    Examples:

    ```
    $ hdiutil help
    $ dig -h
    $ diff --help
    ```

Notes About Specific Commands and Tools

serversetup

The `serversetup` utility is located in /System/Library/ServerSetup. To run this command, you can type the full path, for example:

```
$ /System/Library/ServerSetup/serversetup –getAllPort
```

Or, if you want to use the utility to perform several commands, you can change your working directory and type a shorter command:

```
$ cd /System/Library/ServerSetup
$ ./serversetup –getAllPort
$ ./serversetup –getDefaultInfo
```

or add the directory to your search path for this session and type an even shorter command:

```
$ PATH = "$PATH:/System/Library/ServerSetup"
$ serversetup –getAllPort
```

To permanently add the directory to your search path, add the path to the file /etc/profile.

serveradmin

You can use the `serveradmin` tool to perform many service-related tasks. You'll see it used throughout this guide.

Determining Whether a Service Needs to be Restarted

Some services need to be restarted after you change certain settings. If a change you make using a service's `writeSettings` command requires that you restart the service, the output from the command includes the setting `<svc>:needsRecycleOrRestart` with a value of `yes`.

Important: The `needsRecycleOrRestart` setting is displayed only if you use the `serveradmin` svc:command = writeSettings command to change settings. You won't see it if you use the `serveradmin settings` command.

Installing Server Software and Finishing Basic Setup

2

Commands you can use to install, set up, and update Mac OS X Server software on local or remote computers.

Installing Server Software

You can use the `installer` command to install Mac OS X Server or other software on a computer. For more information, see the man page.

Automating Server Setup

Normally, when you install Mac OS X Server on a computer and restart, the Server Assistant opens and asks you to provide the basic information necessary to get the server up and running (for example, the name and password of the administrator user, the TCP/IP configuration information for the server's network interfaces, and how the server uses directory services). You can automate this initial setup task by providing a configuration file that contains these settings. Servers starting up for the first time look for this file and use it to complete initial server setup without user interaction.

Creating a Configuration File Template

An easy way to prepare configuration files to automate the setup of a group of servers is to start with a file saved using the Server Assistant. You can save the file as the last step when you use the Server Assistant to set up the first server, or you can run the Server Assistant later to create the file. You can then use that first file as a template for creating configuration files for other servers. You can edit the file directly or create scripts to create customized configuration files for any number of servers that use similar hardware.

To save a template configuration file during server setup:

1 In the final pane of the Server Assistant, after you review the settings, click Save As.

2 In the dialog that appears, choose Configuration File next to "Save as" and click OK.

So you can later edit the file, don't select "Save in Encrypted Format."

3 Choose a location to save the file and click Save.

To create a template configuration file at any time after initial setup:

1 Open the Server Assistant (in /Applications/Server).

2 In the Welcome pane, choose "Save setup information in a file or directory record" and click Continue.

3 Enter settings on the remaining panes, then, after you review the settings in the final pane, click Save As.

4 In the dialog that appears, choose Configuration File next to "Save as" and click OK.

   So you can later edit the file, don't select "Save in Encrypted Format."

5 Choose a location to save the file and click Save.

Creating Customized Configuration Files from the Template File

After you create a template configuration file, you can modify it directly using a text editor or write a script to automatically generate custom configuration files for a group of servers.

The file uses XML format to encode the setup information. The name of an XML key reveals the setup parameter it contains.

The following example shows the basic structure and contents of a configuration file for a server with the following configuration:

• An administrative user named "Administrator" (short name "admin") with a user ID of 501 and the password "secret"
• A computer name and host name of "server1.company.com"
• A single Ethernet network interface set to get its address from DHCP
• No server services set to start automatically

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>AdminUser</key>
  <dict>
    <key>exists</key>
    <false/>
    <key>name</key>
    <string>admin</string>
    <key>password</key>
    <string>secret</string>
    <key>realname</key>
    <string>Administrator</string>
    <key>uid</key>
    <string>501</string>
  </dict>
  <key>ComputerName</key>
  <string>server1.company.com</string>
```

```
<key>DS</key>
<dict>
    <key>DSClientInfo</key>
    <string>2 - NetInfo client - broadcast dhcp static -192.168.42.250
    network</string>
    <key>DSClientType</key>
    <string>2</string>
    <key>DSType</key>
    <string>2 - directory client</string>
</dict>
<key>HostName</key>
<string>server1.company.com</string>
<key>InstallLanguage</key>
<string>English</string>
<key>Keyboard</key>
<dict>
    <key>DefaultFormat</key>
    <string>0</string>
    <key>DefaultScript</key>
    <string>0</string>
    <key>ResID</key>
    <integer>0</integer>
    <key>ResName</key>
    <string>U.S.</string>
    <key>ScriptID</key>
    <integer>0</integer>
</dict>
<key>NetworkInterfaces</key>
<array>
    <dict>
        <key>ActiveAT</key>
        <key>ActiveTCPIP</key>
        <key>DNSDomains</key>
        <array>
            <string>company.com</string>
        </array>
        <key>DNSServers</key>
        <array>
            <string>192.168.100.10</string>
        </array>
        <key>DeviceName</key>
        <string>en0</string>
        <key>EthernetAddress</key>
        <string>00:0a:93:bc:6d:1a</string>
        <key>PortName</key>
        <string>Built-in Ethernet</string>
        <key>Settings</key>
        <dict>
            <key>DHCPClientID</key>
```

```
                <string></string>
                <key>Type</key>
                <string>DHCP Configuration</string>
            </dict>
        </dict>
    </array>
    <key>NetworkTimeProtocol</key>
    <dict>
        <key>UsingNTP</key>
        <false/>
    </dict>
    <key>Rendezvous</key>
    <dict>
        <key>RendezvousEnabled</key>
        <true/>
        <key>RendezvousName</key>
        <string>beasbe3</string>
    </dict>
    <key>SerialNumber</key>
    <string>a-123-bcd-456-efg-789-hij-012-klm-345-n</string>
    <key>ServicesAutoStart</key>
    <dict>
        <key>Apache</key>
        <false/>
        <key>File</key>
        <false/>
        <key>MacManager</key>
        <false/>
        <key>Mail</key>
        <false/>
        <key>Print</key>
        <false/>
        <key>QTSS</key>
        <false/>
        <key>WebDAV</key>
        <false/>
    </dict>
    <key>TimeZone</key>
    <string>US/Pacific</string>
    <key>VersionNumber</key>
    <integer>1</integer>
</dict>
</plist>
```

Note: The actual contents of a configuration file depend on the hardware configuration of the computer on which it's created. This is one reason you should start from a template configuration file created on a computer similar to those you plan to set up.

Naming Configuration Files

The Server Assistant recognizes configuration files with these names:
- `MAC-address-of-server`.plist
- `IP-address-of-server`.plist
- `hardware-serial-number-of-server`.plist
- `full-host-name-of-server`.plist
- `generic.plist`

The Server Assistant uses the file to set up the server with the matching address, name, or serial number. If the Server Assistant cannot find a file named for a particular server, it will use the file named `generic.plist`.

Storing a Configuration File in an Accessible Location

The Server Assistant looks for configuration files in the following locations:

`/Volumes/vol/Auto Server Setup/`

where `vol` is any device volume mounted in the /Volumes directory.

Devices you can use to provide configuration files include
- A partition on one of the server's hard disks
- An iPod
- An optical (CD or DVD) drive
- A USB or FireWire drive
- Any other portable storage device that mounts in the /Volumes directory

Changing Server Settings

After initial setup, you can use a variety of commands to view or change Mac OS X Server configuration settings.

For information on changing general system preferences, see Chapter 4, "Setting General System Preferences," on page 31.

For information on changing network settings, see Chapter 5, "Network Preferences," on page 37.

For information on changing service-specific settings, see the chapter that covers the service.

Viewing, Validating, and Setting the Software Serial Number

You can use the `serversetup` command to view or set the server's software serial number or to validate a server software serial number. The `serversetup` utility is located in /System/Library/ServerSetup.

To display the server's software serial number:

```
$ serversetup -getSerialNumber
```

To set the server software serial number:

```
$ sudo serversetup -setSerialNumber serialnumber
```

Parameter	Description
serialnumber	A valid Mac OS X Server software serial number, as found on the software packaging that comes with the software.

To validate a server software serial number:

```
$ serversetup -verifySerialNumber serialnumber
```

Displays 0 if the number is valid, 1 if it isn't.

Updating Server Software

You can use the `softwareupdate` command to check for and install software updates over the web from Apple's website.

To check for available updates:

```
$ softwareupdate --list
```

To install an update:

```
$ softwareupdate --install update-version
```

Parameter	Description
update-version	The hyphenated product version string that appears in the list of updates when you use the `--list` option.

To view command help:

```
$ softwareupdate --help
```

Moving a Server

Try to place a server in its final network location (subnet) before setting it up for the first time. If you're concerned about unauthorized or premature access, you can set up a firewall to protect the server while you're finalizing its configuration.

If you must move a server after initial setup, you need to change settings that are sensitive to network location before the server can be used. For example, the server's IP address and host name—stored in both directories and configuration files that reside on the server—must be updated.

When you move a server, consider these guidelines:
• Minimize the time the server is in its temporary location so the information you need to change is limited.
• Don't configure services that depend on network settings until the server is in its final location. Such services include Open Directory replication, Apache settings (such as virtual hosts), DHCP, and other network infrastructure settings that other computers depend on.
• Wait to import final user accounts. Limit accounts to test accounts so you minimize the user-specific network information (such as home directory location) that will need to change after the move.
• After you move the server, use the `changeip` tool to change IP addresses, host names, and other data stored in Open Directory NetInfo and LDAP directories on the server. See "Changing a Server's IP Address" on page 39. You may need to manually adjust some network configurations, such as the local DNS database, after using the tool.
• Reconfigure the search policy of computers (such as user computers and DHCP servers) that have been configured to use the server in its original location.

Restarting or Shutting Down a Server

3

Commands you can use to shut down or restart a local or remote server.

Restarting a Server

You can use the `reboot` or `shutdown -r` command to restart a server at a specific time. For more information, see the man pages.

Examples

To restart the local server:

```
$ shutdown -r now
```

To restart a remote server immediately:

```
$ ssh -l root server shutdown -r now
```

To restart a remote server at a specific time:

```
$ ssh -l root server shutdown -r hhmm
```

Parameter	Description
server	The IP address or DNS name of the server.
hhmm	The hour and minute when the server restarts.

Automatic Restart

You can also use the `systemsetup` command to set up the server to start automatically after a power failure or system freeze. See "Viewing or Changing Automatic Restart Settings" on page 33.

Changing a Remote Server's Startup Disk

You can change a remote server's startup disk using SSH.

To change the startup disk:

Log in to the remote server using SSH and type

```
$ bless -folder "/Volumes/disk/System/Library/CoreServices" -setOF
```

Parameter	Description
disk	The name of the disk that contains the desired startup volume.

For information on using SSH to log in to a remote server, see "Sending Commands to a Remote Server" on page 16.

Shutting Down a Server

You can use the `shutdown` command to shut down a server at a specific time. For more information, see the man page.

Examples

To shut down a remote server immediately:

```
$ ssh -l root server shutdown -h now
```

To shut down the local server in 30 minutes:

```
$ shutdown -h +30
```

Parameter	Description
server	The IP address or DNS name of the server.

Setting General System Preferences

4

Commands you can use to set system preferences, usually set using the System Preferences GUI application.

Computer Name

You can use the `systemsetup` command to view or change a server's computer name (the name used to browse for AFP share points on the server), which would otherwise be set using the Sharing pane of System Preferences.

Viewing or Changing the Computer Name

To display the server's computer name:

```
$ sudo systemsetup -getcomputername
```

or

```
$ sudo networksetup -getcomputername
```

To change the computer name:

```
$ sudo systemsetup -setcomputername computername
```

or

```
$ sudo networksetup -setcomputername computername
```

Date and Time

You can use the `systemsetup` or `serversetup` command to view or change:

• A server's system date or time
• A server's time zone
• Whether a server uses a network time server

These settings would otherwise be changed using the Date & Time pane of System Preferences.

Viewing or Changing the System Date

To view the current system date:

```
$ sudo systemsetup -getdate
```

or

```
$ serversetup -getDate
```

To set the current system date:

```
$ sudo systemsetup -setdate mm:dd:yy
```

or

```
$ sudo serversetup -setDate mm/dd/yy
```

Viewing or Changing the System Time

To view the current system time:

```
$ sudo systemsetup -gettime
```

or

```
$ serversetup -getTime
```

To change the current system time:

```
$ sudo systemsetup -settime hh:mm:ss
```

or

```
$ sudo serversetup -setTime hh:mm:ss
```

Viewing or Changing the System Time Zone

To view the current time zone:

```
$ sudo systemsetup -gettimezone
```

or

```
$ serversetup -getTimeZone
```

To view the available time zones:

```
$ sudo systemsetup -listtimezones
```

To change the system time zone:

```
$ sudo systemsetup -settimezone timezone
```

or

```
$ sudo serversetup -setTimeZone timezone
```

Viewing or Changing Network Time Server Usage

To see if a network time server is being used:

```
$ sudo systemsetup -getusingnetworktime
```

To enable or disable use of a network time server:

```
$ sudo systemsetup -setusingnetworktime (on|off)
```

To view the current network time server:

```
$ sudo systemsetup -getnetworktimeserver
```

To specify a network time server:

```
$ sudo systemsetup -setnetworktimeserver timeserver
```

Energy Saver Settings

You can use the `systemsetup` command to view or change a server's energy saver settings, which would otherwise be set using the Energy Saver pane of System Preferences.

Viewing or Changing Sleep Settings

To view the idle time before sleep:

```
$ sudo systemsetup -getsleep
```

To set the idle time before sleep:

```
$ sudo systemsetup -setsleep minutes
```

To see if the system is set to wake for modem activity:

```
$ sudo systemsetup -getwakeonmodem
```

To set the system to wake for modem activity:

```
$ sudo systemsetup -setwakeonmodem (on|off)
```

To see if the system is set to wake for network access:

```
$ sudo systemsetup -getwakeonnetworkaccess
```

To set the system to wake for network access:

```
$ sudo systemsetup -setwakeonnetworkaccess (on|off)
```

Viewing or Changing Automatic Restart Settings

To see if the system is set to restart after a power failure:

```
$ sudo systemsetup -getrestartpowerfailure
```

To set the system to restart after a power failure:

```
$ sudo systemsetup -setrestartpowerfailure (on|off)
```

To see how long the system waits to restart after a power failure:

```
$ sudo systemsetup -getWaitForStartupAfterPowerFailure
```

To set how long the system waits to restart after a power failure:

```
$ sudo systemsetup -setWaitForStartupAfterPowerFailure seconds
```

Parameter	Description
seconds	Must be a multiple of 30 seconds.

To see if the system is set to restart after a system freeze:

```
$ sudo systemsetup -getrestartfreeze
```

To set the system to restart after a system freeze:

```
$ sudo systemsetup -setrestartfreeze (on|off)
```

Power Management Settings

You can use the `pmset` command to change a variety of power management settings, including:

- Display dim timer
- Disk spindown timer
- System sleep timer
- Wake on network activity
- Wake on modem activity
- Restart after power failure
- Dynamic processor speed change
- Reduce processor speed
- Sleep computer on power button press

For more information, see the `pmset` man page.

Startup Disk Settings

You can use the `systemsetup` command to view or change a server's computer startup disk, which would otherwise be set using the Startup Disk pane of System Preferences.

Viewing or Changing the Startup Disk

To view the current startup disk:

```
$ sudo systemsetup -getstartupdisk
```

To view the available startup disks:

```
$ sudo systemsetup -liststartupdisks
```

To change the current startup disk:

```
$ sudo systemsetup -setstartupdisk path
```

Sharing Settings

You can use the `systemsetup` command to view or change settings that would otherwise be set using the Sharing pane of System Preferences.

Viewing or Changing Remote Login Settings

You can use SSH to log in to a remote server if remote login is enabled.

To see if the system is set to allow remote login:

```
$ sudo systemsetup -getremotelogin
```

To enable or disable remote login:

```
$ sudo systemsetup -setremotelogin (on|off)
```

or

```
$ serversetup -enableSSH
```

Telnet access is disabled by default because it isn't as secure as SSH. You can, however, enable Telnet access. See "Using Telnet" on page 18.

Viewing or Changing Apple Event Response

To see if the system is set to respond to remote events:

```
$ sudo systemsetup -getremoteappleevents
```

To set the server to respond to remote events:

```
$ sudo systemsetup -setremoteappleevents (on|off)
```

International Settings

You can use the `serversetup` command to view or change language settings that would otherwise be set using the Sharing pane of System Preferences.

Viewing or Changing Language Settings

To view the current primary language:

```
$ serversetup -getPrimaryLanguage
```

To view the installed primary language:

```
$ serversetup -getInstallLanguage
```

To change the install language:

```
$ sudo serversetup -setInstallLanguage language
```

To view the script setting:

```
$ serversetup -getPrimaryScriptCode
```

Login Settings

Disabling the Restart and Shutdown Buttons

To disable or enable the Restart and Shutdown buttons in the login dialog:

```
$ sudo serversetup -setDisableRestartShutdown (0|1)
```

0 disables the buttons.

1 enables the buttons.

To view the current setting:

```
$ serversetup -getDisableRestartShutdown
```

Network Preferences

<div style="text-align: right">

5

</div>

Commands you can use to change a server's network settings.

Network Interface Information

This section describes commands you address to a specific hardware device (for example, `en0`) or port (for example, `Built-in Ethernet`).

If you prefer to work with network port configurations following the approach used in the Network preferences pane of System Preferences, see the commands in "Network Port Configurations" on page 38.

Viewing Port Names and Hardware Addresses

To list all port names:

```
$ serversetup -getAllPort
```

To list all port names with their Ethernet (MAC) addresses:

```
$ sudo networksetup -listallhardwareports
```

To list hardware port information by port configuration:

```
$ sudo networksetup -listallnetworkservices
```

An asterisk in the results (*) marks an inactive configuration.

To view the default (en0) Ethernet (MAC) address of the server:

```
$ serversetup -getMacAddress
```

To view the Ethernet (MAC) address of a particular port:

```
$ sudo networksetup -getmacaddress (devicename|"portname")
```

To scan for new hardware ports:

```
$ sudo networksetup -detectnewhardware
```

This command checks the computer for new network hardware and creates a default configuration for each new port.

Viewing or Changing MTU Values

You can use these commands to change the maximum transmission unit (MTU) size for a port.

To view the MTU value for a hardware port:

```
$ sudo networksetup -getMTU (devicename|"portname")
```

To list valid MTU values for a hardware port:

```
$ sudo networksetup -listvalidMTUrange (devicename|"portname")
```

To change the MTU value for a hardware port:

```
$ sudo networksetup -setMTU (devicename|"portname")
```

Viewing or Changing Media Settings

To view the media settings for a port:

```
$ sudo networksetup -getMedia (devicename|"portname")
```

To list valid media settings for a port:

```
$ sudo networksetup -listValidMedia (devicename|"portname")
```

To change the media settings for a port:

```
$ sudo networksetup -setMedia (devicename|"portname") subtype [option1]
    [option2] [...]
```

Network Port Configurations

Network port configurations are sets of network preferences that can be assigned to a particular network interface and then enabled or disabled. The Network pane of System Preferences stores and displays network settings as port configurations.

Creating or Deleting Port Configurations

To list existing port configuration:

```
$ sudo networksetup -listallnetworkservices
```

To create a port configuration:

```
$ sudo networksetup -createnetworkservice configuration hardwareport
```

To duplicate a port configuration:

```
$ sudo networksetup -duplicatenetworkservice configuration newconfig
```

To rename a port configuration:

```
$ sudo networksetup -renamenetworkservice configuration newname
```

To delete a port configuration:

```
$ sudo networksetup -removenetworkservice configuration
```

Activating Port Configurations

To see if a port configuration is on:

```
$ sudo networksetup -getnetworkserviceenabled configuration
```

To enable or disable a port configuration:

```
$ sudo networksetup -setnetworkserviceenabled configuration (on|off)
```

Changing Configuration Precedence

To list the configuration order:

```
$ sudo networksetup -listnetworkserviceorder
```

The configurations are listed in the order that they're tried when a network connection is established. An asterisk (*) marks an inactive configuration.

To change the order of the port configurations:

```
$ sudo networksetup -ordernetworkservices config1 config2 [config3] [...]
```

TCP/IP Settings

Changing a Server's IP Address

Changing a server's IP address isn't as simple as changing the TCP/IP settings. Address information is set throughout the system when you set up the server. To make sure that all the necessary changes are made, use the `changeip` command.

To change a server's IP address:

1 Run the `changeip` tool:

```
$ changeip [(directory|-)] old-ip new-ip [old-hostname new-hostname]
```

Parameter	Description
directory	If the server is an Open Directory master or replica, or is connected to a directory system, you must include the path to the directory domain (directory node). For a standalone server, type "-" instead.
old-ip	The current IP address.
new-ip	The new IP address.
old-hostname	(optional) The current DNS host name of the server.
new-hostname	(optional) The new DNS host name of the server.

For more information or examples, see the man page.

2 Use the `networksetup` or `serversetup` command (or the Network pane of System Preferences) to change the server's IP address in its network settings.

3 Restart the server.

Viewing or Changing IP Address, Subnet Mask, or Router Address

You can use the `serversetup` and `networksetup` commands to change a computer's TCP/IP settings.

Important: Changing a server's IP address isn't as simple as changing the TCP/IP settings. You must first run the `changeip` utility to make sure necessary changes are made throughout the system. See "Changing a Server's IP Address" on page 39.

To list TCP/IP settings for a configuration:

```
$ sudo networksetup -getinfo "configuration"
```

Example:

```
$ networksetup -getinfo "Built-In Ethernet"
Manual Configuration
IP Address: 192.168.10.12
Subnet mask: 255.255.0.0
Router: 192.18.10.1
Ethernet Address: 1a:2b:3c:4d:5e:6f
```

To view TCP/IP settings for port en0:

```
$ serversetup -getDefaultinfo (devicename|"portname")
```

To view TCP/IP settings for a particular port or device:

```
$ serversetup -getInfo (devicename|"portname")
```

To change TCP/IP settings for a particular port or device:

```
$ sudo serversetup -setInfo (devicename|"portname") ipaddress subnetmask
    router
```

To set manual TCP/IP information for a configuration:

```
$ sudo networksetup -setmanual "configuration" ipaddress subnetmask router
```

To validate an IP address:

```
$ serversetup -isValidIPAddress ipaddress
```

Displays 0 if the address is valid, 1 if it isn't.

To validate a subnet mask:

```
$ serversetup -isValidSubnetMask subnetmask
```

To set a configuration to use DHCP:

```
$ sudo networksetup -setdhcp "configuration" [clientID]
```

To set a configuration to use DHCP with a manual IP address:

```
$ sudo networksetup -setmanualwithdhcprouter "configuration" ipaddress
```

To set a configuration to use BootP:

```
$ sudo networksetup -setbootp "configuration"
```

Viewing or Changing DNS Servers

To view the DNS servers for port en0:

```
$ serversetup -getDefaultDNSServer (devicename|"portname")
```

To change the DNS servers for port en0:

```
$ sudo serversetup -setDefaultDNSServer (devicename|"portname") server1
    [server2] [...]
```

To view the DNS servers for a particular port or device:

```
$ serversetup -getDNSServer (devicename|"portname")
```

To change the DNS servers for a particular port or device:

```
$ sudo serversetup -setDNSServer (devicename|"portname") server1 [server2]
    [...]
```

To list the DNS servers for a configuration:

```
$ sudo networksetup -getdnsservers "configuration"
```

To view the DNS search domains for port en0:

```
$ serversetup -getDefaultDNSDomain (devicename|"portname")
```

To change the DNS search domains for port en0:

```
$ sudo serversetup -setDefaultDNSDomain (devicename|"portname") domain1
    [domain2] [...]
```

To view the DNS search domains for a particular port or device:

```
$ serversetup -getDNSDomain (devicename|"portname")
```

To change the DNS search domains for a particular port or device:

```
$ sudo serversetup -setDNSDomain (devicename|"portname") domain1 [domain2]
    [...]
```

To list the DNS search domains for a configuration:

```
$ sudo networksetup -getsearchdomains "configuration"
```

To set the DNS servers for a configuration:

```
$ sudo networksetup -setdnsservers "configuration" dns1 [dns2] [...]
```

To set the search domains for a configuration:

```
$ sudo networksetup -setsearchdomains "configuration" domain1 [domain2]
    [...]
```

To validate a DNS server:

```
$ serversetup -verifyDNSServer server1 [server2] [...]
```

To validate DNS search domains:

```
$ serversetup -verifyDNSDomain domain1 [domain2] [...]
```

Enabling TCP/IP

To enable TCP/IP on a particular port:

```
$ serversetup -EnableTCPIP [(devicename|"portname")]
```

If you don't provide an interface, en0 is assumed.

To disable TCP/IP on a particular port:

```
$ serversetup -DisableTCPIP [(devicename|"portname")]
```

If you don't provide an interface, en0 is assumed.

AppleTalk Settings

Enabling and Disabling AppleTalk

To enable AppleTalk on a particular port:

```
$ serversetup -EnableAT [(devicename|"portname")]
```

If you don't provide an interface, en0 is assumed.

To disable AppleTalk on a particular port:

```
$ serversetup -DisableAT [(devicename|"portname")]
```

If you don't provide an interface, en0 is assumed.

To enable AppleTalk on en0:

```
$ serversetup -EnableDefaultAT
```

To disable AppleTalk on en0:

```
$ serversetup -DisableDefaultAT
```

To make AppleTalk active or inactive for a configuration:

```
$ sudo networksetup -setappletalk "configuration" (on|off)
```

To check AppleTalk state on en0:

```
$ serversetup -getDefaultATActive
```

To see if AppleTalk is active for a configuration:

```
$ sudo networksetup -getappletalk
```


Proxy Settings

Viewing or Changing FTP Proxy Settings

To view the FTP proxy information for a configuration:

```
$ sudo networksetup -getftpproxy "configuration"
```

To set the FTP proxy information for a configuration:

```
$ sudo networksetup -setftpproxy "configuration" domain portnumber
```

To view the FTP passive setting for a configuration:

```
$ sudo networksetup -getpassiveftp "configuration"
```

To enable or disable FTP passive mode for a configuration:

```
$ sudo networksetup -setpassiveftp "configuration" (on|off)
```

To enable or disable the FTP proxy for a configuration:

```
$ sudo networksetup -setftpproxystate "configuration" (on|off)
```

Viewing or Changing Web Proxy Settings

To view the web proxy information for a configuration:

```
$ sudo networksetup -getwebproxy "configuration"
```

To set the web proxy information for a configuration:

```
$ sudo networksetup -setwebproxy "configuration" domain portnumber
```

To enable or disable the web proxy for a configuration:

```
$ sudo networksetup -setwebproxystate "configuration" (on|off)
```

Viewing or Changing Secure Web Proxy Settings

To view the secure web proxy information for a configuration:

```
$ sudo networksetup -getsecurewebproxy "configuration"
```

To set the secure web proxy information for a configuration:

```
$ sudo networksetup -setsecurewebproxy "configuration" domain portnumber
```

To enable or disable the secure web proxy for a configuration:

```
$ sudo networksetup -setsecurewebproxystate "configuration" (on|off)
```

Viewing or Changing Streaming Proxy Settings

To view the streaming proxy information for a configuration:

```
$ sudo networksetup -getstreamingproxy "configuration"
```

To set the streaming proxy information for a configuration:

```
$ sudo networksetup -setstreamingproxy "configuration" domain portnumber
```

To enable or disable the streaming proxy for a configuration:

```
$ sudo networksetup -setstreamingproxystate "configuration" (on|off)
```

Viewing or Changing Gopher Proxy Settings

To view the gopher proxy information for a configuration:

```
$ sudo networksetup -getgopherproxy "configuration"
```

To set the gopher proxy information for a configuration:

```
$ sudo networksetup -setgopherproxy "configuration" domain portnumber
```

To enable or disable the gopher proxy for a configuration:

```
$ sudo networksetup -setgopherproxystate "configuration" (on|off)
```

Viewing or Changing SOCKS Firewall Proxy Settings
To view the SOCKS firewall proxy information for a configuration:
```
$ sudo networksetup -getsocksfirewallproxy "configuration"
```

To set the SOCKS firewall proxy information for a configuration:
```
$ sudo networksetup -setsocksfirewallproxy "configuration" domain portnumber
```

To enable or disable the SOCKS firewall proxy for a configuration:
```
$ sudo networksetup -setsocksfirewallproxystate "configuration" (on|off)
```

Viewing or Changing Proxy Bypass Domains
To list the proxy bypass domains for a configuration:
```
$ sudo networksetup -getproxybypassdomains "configuration"
```

To set the proxy bypass domains for a configuration:
```
$ sudo networksetup -setproxybypassdomains "configuration" [domain1] domain2
    [...]
```

AirPort Settings

Viewing or Changing Airport Settings
To see if AirPort power is on or off:
```
$ sudo networksetup -getairportpower
```

To turn AirPort power on or off:
```
$ sudo networksetup -setairportpower (on|off)
```

To display the name of the current AirPort network:
```
$ sudo networksetup -getairportnetwork
```

To join an AirPort network:
```
$ sudo networksetup -setairportnetwork network [password]
```

Computer, Host, and Rendezvous Name

Viewing or Changing the Computer Name
To display the server's computer name:
```
$ sudo systemsetup -getcomputername
```

or

```
$ sudo networksetup -getcomputername
```

or

```
$ serversetup -getComputername
```

To change the computer name:

```
$ sudo systemsetup -setcomputername computername
```

or

```
$ sudo networksetup -setcomputername computername
```

or

```
$ sudo serversetup -setComputername computername
```

To validate a computer name:

```
$ serversetup -verifyComputername computername
```

Viewing or Changing the Local Host Name

To display the server's local host name:

```
$ serversetup -getHostname
```

To change the server's local host name:

```
$ sudo serversetup -setHostname hostname
```

Viewing or Changing the Rendezvous Name

To display the server's Rendezvous name:

```
$ serversetup -getRendezvousname
```

To change the server's Rendezvous name:

```
$ sudo serversetup -setRendezvousname rendezvousname
```

The command displays a 0 if the name was changed.

Note: If you use the Server Admin GUI application to connect to a server using its Rendezvous name, then change the server's Rendezvous name, you will need to reconnect to the server the next time you open the Server Admin application.

Working With Disks and Volumes

6

Commands you can use to prepare, use, and test disks and volumes.

Mounting and Unmounting Volumes

You can use the `mount_afp` command to mount an AFP volume. For more information, type `man mount_afp` to see the man page.

Mounting Volumes

You can use the `mount` command with parameters appropriate to the type of file system you want to mount, or use one of these file-system-specific mount commands:

- `mount_afp` for Apple File Protocol (AppleShare) volumes
- `mount_cd9660` for ISO 9660 volumes
- `mount_cddafs` for CD Digital Audio format (CDDA) volumes
- `mount_hfs` for Apple Hierarchical File System (HFS) volumes
- `mount_msdos` for PC MS-DOS volumes
- `mount_nfs` for Network File System (NFS) volumes
- `mount_smbfs` for Server Message Block (SMB) volumes
- `mount_udf` for Universal Disk Format (UDF) volumes
- `mount_webdav` for Web-based Distributed Authoring and Versioning (WebDAV) volumes

For more information, see the related man pages.

Unmounting Volumes

You can use the `umount` command to unmount a volume. For more information, see the man page.

Checking for Disk Problems

You can use the `diskutil` or `fsck` command (`fsck_hfs` for HFS volumes) to check the physical condition and file system integrity of a volume. For more information, see the related man pages.

Monitoring Disk Space

When you need more vigilant monitoring of disk space than the log rolling scripts provide, you can use the `diskspacemonitor` command-line tool. It lets you monitor disk space and take action more frequently than once a day when disk space is critically low, and gives you the opportunity to provide your own action scripts.

`diskspacemonitor` is disabled by default. You can enable it by opening a Terminal window and typing `sudo diskspacemonitor on`. You may be prompted for your password. Type `man diskspacemonitor` for more information about the command-line options.

When enabled, `diskspacemonitor` uses information in a configuration file to determine when to execute alert and recovery scripts for reclaiming disk space:

- The configuration file is /etc/diskspacemonitor/diskspacemonitor.conf. It lets you specify how often you want to monitor disk space and thresholds to use for determining when to take the actions in the scripts. By default, disks are checked every 10 minutes, an alert script executed when disks are 75% full, and a recovery script executed when disks are 85% full. To edit the configuration file, log in to the server as an administrator and use a text editor to open the file. See the comments in the file for additional information.

- By default, two predefined action scripts are executed when the thresholds are reached.
  The default alert script is /etc/diskspacemonitor/action/alert. It runs in accord with instructions in configuration file /etc/diskspacemonitor/alert.conf. It sends email to recipients you specify.

  The default recovery script is /etc/diskspacemonitor/action/recover. It runs in accord with instructions in configuration file /etc/diskspacemonitor/recover.conf.

  See the comments in the script and configuration files for more information about these files.

- If you want to provide your own alert and recovery scripts, you can. Put your alert script in /etc/diskspacemonitor/action/alert.local and your recovery script in /etc/diskspacemonitor/action/recovery.local. Your scripts will be executed before the default scripts when the thresholds are reached.

To configure the scripts on a server from a remote Mac OS X computer, open a Terminal window and log in to the remote server using SSH.

Reclaiming Disk Space Using Log Rolling Scripts

Three predefined scripts are executed automatically to reclaim space used on your server for log files generated by

• Apple file service
• Windows service
• Web service
• Web performance cache
• Mail service
• Print service

The scripts use values in the following configuration files to determine whether and how to reclaim space:

• The script /etc/periodic/daily/600.daily.server runs daily. Its configuration file is /etc/diskspacemonitor/daily.server.conf.
• The script /etc/periodic/weekly/600.weekly.server is intended to run weekly, but is currently empty. Its configuration file is /etc/diskspacemonitor/weekly.server.conf.
• The script /etc/periodic/monthly/600.monthly.server is intended to run monthly, but is currently empty. Its configuration file is /etc/diskspacemonitor/monthly.server.conf.

As configured, the scripts specify actions that complement the log file management performed by the services listed above, so don't modify them. All you need to do is log in as an administrator and use a text editor to define thresholds in the configuration files that determine when the actions are taken:

• the number of megabytes a log file must contain before its space is reclaimed
• the number of days since a log file's last modification that need to pass before its space is reclaimed

Specify one or both thresholds. The actions are taken when either threshold is exceeded.

There are several additional parameters you can specify. Refer to comments in the configuration files for information about all the parameters and how to set them. The scripts ignore all log files except those for which at least one threshold is present in the configuration file.

To configure the scripts on a server from a remote Mac OS X computer, open a Terminal window and log in to the remote server using SSH. Then open a text editor and edit the scripts.

You can also use the `diskspacemonitor` command-line tool to reclaim disk space.

Managing Disk Journaling

Checking to See if Journaling is Enabled

You can use the `mount` command to see if journaling is enable on a volume.

To see if journaling is enabled:

```
$ mount
```

Look for `journaled` in the attributes in parentheses following a volume. For example:

```
/dev/disk0s9 on / (local, journaled)
```

Turning on Journaling for an Existing Volume

You can use the `diskutil` command to enable journaling on a volume without affecting existing files on the volume.

Important: Always check the volume for disk errors using the `fsck_hfs` command before you turn on journaling.

To enable journaling:

```
$ diskutil enableJournal volume
```

Parameter	Description
volume	The volume name or device name of the volume.

Example

```
$ mount
/dev/disk0s9 on / (local, journaled)
/dev/disk0s10 on /Volumes/OS 9.2.2 (local)
$ sudo fsck_hfs /dev/disk0s10/
** /dev/rdisk0s10
** Checking HFS plus volume.
** Checking extents overflow file.
** Checking Catalog file.
** Checking Catalog hierarchy.
** Checking volume bitmap.
** Checking volume information.
** The volume OS 9.2.2 appears to be OK.
$ diskutil enableJournal /dev/disk0s10
Allocated 8192K for journal file.
Journaling has been enabled on /dev/disk0s10
$ mount
/dev/disk0s9 on / (local, journaled)
/dev/disk0s10 on /Volumes/OS 9.2.2 (local, journaled)
```

Enabling Journaling When You Erase a Disk

You can use the `newfs_hfs` command to set up and enable journaling when you erase a disk.

To enable journaling when erasing a disk:

```
$ newfs_hfs -J -v volname device
```

Parameter	Description
volname	The name you want the new disk volume to have.
device	The device name of the disk.

Disabling Journaling

To disable journaling:

```
$ diskutil disableJournal volume
```

Parameter	Description
volume	The volume name or device name of the volume.

Erasing, Partitioning, and Formatting Disks

You can use the `diskutil` command to partition, erase, or format a disk. For more information, see the man page.

Setting Up a Case-Sensitive HFS+ File System

You can use the `diskutil` tool to format a drive for case-sensitive HFS.

Note: Volumes you format as case-sensitive HFS are also journaled.

To format a Mac OS Extended volume as case-sensitive HFS+:

```
$ sudo diskutil eraseVolume "Case-sensitive HFS+" newvolname volume
```

Parameter	Description
newvolname	The name given to the reformatted, case-sensitive volume.
volume	The path to the existing volume to be reformatted. For example, `/Volumes/HFSPlus`

For more information, see the man page for `diskutil`.

Imaging and Cloning Volumes Using ASR

You can use Apple Software Restore (ASR) to copy a disk image onto a volume or prepare existing disk images with checksum information for faster copies. ASR can perform file copies, in which individual files are restored to a volume unless an identical file is already there, and block copies, which restore entire disk images. The `asr` utility doesn't create the disk images. You can use `hdiutil` to create disk images from volumes or folders.

You must run ASR as the root user or with sudo root permissions. You cannot use ASR on read/write disk images.

To image a boot volume:

1 Install and configure Mac OS X on the volume as you want it.

2 Restart from a different volume.

3 Make sure the volume you're imaging has permissions enabled.

4 Use `hditutil` to make a read-write disk image of the volume.

5 Mount the disk image.

6 Remove cache files, host-specific preferences, and virtual memory files. You can find example files to remove on the `asr` man page.

7 Unmount the volume and convert the read-write image to a read-only compressed image.

   ```
   hdiutil convert -format UDZO pathtoimage -o compressedimage
   ```

8 Prepare the image for duplication by adding checksum information:

   ```
   sudo asr -imagescan compressedimage
   ```

To restore a volume from an image:

```
$ sudo asr -source compressedimage -target targetvolume -erase
```

See the `asr` man page for command syntax, limitations, and image preparation instructions.

Working With Users and Groups   7

Commands you can use to set up and manage users and groups in Mac OS X Server.

Creating Server Administrator Users

You can use the `serversetup` command to create administrator users for a server. To create regular users, see "Importing Users and Groups" on page 54.

To create a user:

```
$ serversetup -createUser fullname shortname password
```

The name, short name, and password must be typed in the order shown. If the full name includes spaces, type it in quotes.

The command displays a 1 if the full name or short name is already in use.

To create a user with a specific UID:

```
$ serversetup -createUserWithID fullname shortname password userid
```

The name, short name, password, and UID must be typed in the order shown. If the full name includes spaces, type it in quotes.

The command displays a 1 if the full name, short name, or UID is already in use or if the UID you specified is less than 100.

To create a user with a specific UID and home directory:

```
$ serversetup -createUserWithIDIP fullname shortname password userid
      homedirpath
```

The name, short name, password, and UID must be typed in the order shown. If the full name includes spaces, type it in quotes.

The command displays a 1 if the full name, short name, or UID is already in use or if the UID you specified is less than 100.

Importing Users and Groups

You can use the `dsimportexport` command to import user and group accounts.

Note: Despite its name, `dsimportexport` can't be used to export user records.

The utility is in /Applications/Server/Workgroup Manager.app/Contents/Resources.

For information on the formats of the files you can import, see "Creating a Character-Delimited User Import File" on page 55.

```
$ dsimportexport (-g|-s|-p) file directory user password (O|M|I|A) [options]
```

Parameter	Description
-g|-s|-p	You must specify one of these to indicate the type of file you're importing: -g for a character-delimited file -s for an XML file exported from Users & Groups in Mac OS X Server version 10.1.x -p for an XML file exported from AppleShare IP version 6.x
file	The path of the file to import.
directory	The path to the Open Directory node where the records will be added.
user	The name of the directory administrator.
password	The password of the directory administrator.
O|M|I|A	Specifies how user data is handled if a record for an imported user already exists in the directory: O: Overwrite the matching record. M: Merge the records. Empty attributes in the directory assume values from the imported record. I: Ignore imported record and leave existing record unchanged. A: Append data from import record to existing record.
options	Additional command options. To see available options, execute the `dsimportexport` command with no parameters.

To import users and groups:

1   Create a file containing the accounts to import, and place it in a location accessible from the importing server. You can export this file from an earlier version of Mac OS X Server or AppleShare IP 6.3, or create your own character-delimited file. See "Creating a Character-Delimited User Import File" on page 55.

Open Directory supports up to 100,000 records. For local NetInfo databases, make sure the file contains no more than 10,000 records.

2   Log in as the administrator of the directory domain into which you want to import accounts.

3 Open the Terminal application and type the `dsimportexport` command. The tool is located in /Applications/Utilities/Workgroup Manager.app/Contents/Resources.

To include the space in the path name, precede it with a backslash (\). For example:

```
/Applications/Utilities/Workgroup\ Manager.app/Contents/Resources
    /dsimportexport -h
```

4 If you want, use the `createhomedir` tool to create home directories for imported users. See "Creating a User's Home Directory" on page 63.

Creating a Character-Delimited User Import File
You can create a character-delimited file by hand, using a script, or by using a database or spreadsheet application.

The first record in the file, the record description, describes the format of each account record in the file. There are three options for the record description:
• Write a full record description
• Use the shorthand `StandardUserRecord`
• Use the shorthand `StandardGroupRecord`

The other records in the file describe user or group accounts, encoded in the format described by the record description. Any line of a character-delimited file that begins with "#" is ignored during importing.

Writing a Record Description
The record description specifies the fields in each record in the character-delimited file, specifies the delimiting characters, and specifies the escape character that precedes special characters in a record. Encode the record description using the following elements in the order specified, separating them with a space:
• End-of-record indicator (in hex notation)
• Escape character (in hex notation)
• Field separator (in hex notation)
• Value separator (in hex notation)
• Type of accounts in the file (`DSRecTypeStandard:Users` or `DSRecTypeStandard:Groups`)
• Number of attributes in each account record
• List of attributes

For user accounts, the list of attributes must include the following, although you can omit UID and PrimaryGroupID if you specify a starting UID and a default primary group ID when you import the file:
• RecordName (the user's short name)
• Password
• UniqueID (the UID)
• PrimaryGroupID
• RealName (the user's full name)

In addition, you can include
- UserShell (the default shell)
- NFSHomeDirectory (the path to the user's home directory on the user's computer)
- Other user data types, described under "User Attributes" on page 57

For group accounts, the list of attributes must include
- RecordName (the group name)
- PrimaryGroupID (the group ID)
- GroupMembership

Here is an example of a record description:

```
0x0A 0x5C 0x3A 0x2C DSRecTypeStandard:Users 7
RecordName Password UniqueID PrimaryGroupID
RealName NFSHomeDirectory UserShell
```

Here is an example of a record encoded using the above description:

```
jim:Adl47E$:408:20:J. Smith, Jr.,
     M.D.:/Network/Servers/somemac/Homes/jim:/bin/csh
```

The record consists of values, delimited by colons. Use a double colon (::) to indicate a value is missing.

Here is another example, which shows a record description and user records for users whose passwords are to be validated using the Password Server. The record description should include a field named `dsAttrTypeStandard:AuthMethod`, and the value of this field for each record should be `dsAuthMethodStandard:dsAuthClearText`:

```
0x0A 0x5C 0x3A 0x2C dsRecTypeStandard:Users 8
dsAttrTypeStandard:RecordName dsAttrTypeStandard:AuthMethod
dsAttrTypeStandard:Password dsAttrTypeStandard:UniqueID
dsAttrTypeStandard:PrimaryGroupID dsAttrTypeStandard:Comment
dsAttrTypeStandard:RealName dsAttrTypeStandard:UserShell
skater:dsAuthMethodStandard\:dsAuthClearText:pword1:374:11:comment:
Tony Hawk:/bin/csh
mattm:dsAuthMethodStandard\:dsAuthClearText:pword2:453:161::
Matt Mitchell:/bin/tcsh
```

As these examples illustrate, you can use the prefix `dsAttrTypeStandard:` when referring to an attribute, or you can omit the prefix.

Using the `StandardUserRecord` Shorthand
When the first record in a character-delimited import file contains `StandardUserRecord`, the following record description is assumed:

```
0x0A 0x5C 0x3A 0x2C DSRecTypeStandard:Users 7
RecordName Password UniqueID PrimaryGroupID
RealName NFSHomeDirectory UserShell
```

An example user account looks like this:

```
jim:Adl47E$:408:20:J. Smith, Jr.,
     M.D.:/Network/Servers/somemac/Homes/jim:/bin/csh
```

Using the `StandardGroupRecord` Shorthand

When the first record in a character-delimited import file contains `StandardGroupRecord`, the following record description is assumed:

```
0x0A 0x5C 0x3A 0x2C DSRecTypeStandard:Groups 4
RecordName Password PrimaryGroupID GroupMembership
```

Here is an example of a record encoded using the description:

```
students:Ad147:88:jones,alonso,smith,wong
```

User Attributes

The following table lists standard XML data structures for attributes in user records.

Attribute	Format	Sample values
`RecordName:` A list of names associated with a user; the first is the user's short name, which is also the name of the user's home directory Important: All attributes used for authentication must map to RecordName.	First value: ASCII characters A–Z, a–z, 0–9, _- Second value: UTF-8 Roman text	`Dave` `David Mac` `DMacSmith` Non-zero length, 1 to 16 values. Maximum 255 bytes (85 triple-byte to 255 single-byte characters) per instance. First value must be 1 to 30 bytes for clients using Macintosh Manager, or 1 to 8 bytes for clients using Mac OS X version 10.1 and earlier.
`RealName:` A single name, usually the user's full name; not used for authentication	UTF-8 text	`David L. MacSmith, Jr.` Non-zero length, maximum 255 bytes (85 triple-byte to 255 single-byte characters).
`UniqueID:` A unique user identifier, used for access privilege management	Signed 32-bit ASCII string of digits 0–9	Range is 100 to 2,147,483,648. Values below 100 are typically used for system accounts. Zero is reserved for use by the system. Normally unique among entire population of users, but sometimes can be duplicated. Warning: A non-integer value is interpreted as 0, which is the UniqueID of the root user.
`PrimaryGroupID:` A user's primary group association	Unsigned 32-bit ASCII string of digits 0–9	Range is 1 to 2,147,483,648. Normally unique among entire population of group records. If blank, 20 is assumed.
`NFSHomeDirectory:` Local file system path to the user's home directory	UTF-8 text	`/Network/Servers/example/Users/` `K-M/Tom King` Non-zero length. Maximum 255 bytes.

Attribute	Format	Sample values

`HomeDirectory:`
The location of an AFP-based home directory
	Structured UTF-8 text	```<home_dir>```
		```<url>```
		```afp://server/sharepoint```
		```</url>```
		```<path>```
		```usershomedirectory```
		```</path>```
		```</home_dir>```
		
		In the following example, Tom King's home directory is K-M/Tom King, which resides beneath the share point directory, Users:
		
		```<home_dir>```
		```<url>```
		```afp://example.com/Users```
		```</url>```
		```<path>```
		```K-M/Tom King```
		```</path>```
		```</home_dir>```

`HomeDirectoryQuota:` The disk quota for the user's home directory	Text for the number of bytes allowed	If the quota is 10MB, the value will be the text string `1048576`.

`MailAttribute:`
A user's mail service configuration (refer to "Mail Attributes in User Records" on page 60 for information on individual fields in this structure)
	Structured text	```<dict>```
		```<key>kAttributeVersion</key>```
		```<string>Apple Mail 1.0</string>```
		```<key>kAutoForwardValue</key>```
		```<string>user@example.com</string>```
		```<key>kIMAPLoginState</key>```
		```<string>IMAPAllowed</string>```
		```<key>kMailAccountLocation</key>```
		```<string>domain.example.com</string>```
		```<key>kMailAccountState</key>```
		```<string>Enabled</string>```
		```<key>kNotificationState</key>```
		```<string>NotificationStaticIP</string>```
		```<key>kNotificationStaticIPValue</key>```
		```<string>[1.2.3.4]</string>```
		```<key>kPOP3LoginState</key>```
		```<string>POP3Allowed</string>```
		```<key>kSeparateInboxState</key>```
		```<string>OneInbox</string>```
		```<key>kShowPOP3InboxInIMAP</key>```
		```<string>HidePOP3Inbox</string>```
		```</dict>```

`PrintServiceUserData` A user's print quota statistics	UTF-8 XML plist, single value	

Attribute	Format	Sample values
`MCXFlags:` If present, MCXSettings is loaded; if absent, MCXSettings isn't loaded; required for a managed user.	UTF-8 XML plist, single value	
`MCXSettings:` A user's managed preferences	UTF-8 XML plist, single value	
`AdminLimits` The privileges allowed by Workgroup Manager to a user that can administer the directory domain	UTF-8 XML plist, single value	
`Password:` The user's password	UNIX crypt	
`Picture:` File path to a recognized graphic file to be used as a display picture for the user	UTF-8 text	Maximum 32,676 bytes.
`Comment:` Any documentation you like	UTF-8 text	`John is in charge of product marketing.`

`UserShell:`
The location of the default shell for command-line interactions with the server
	Path name	`/bin/tcsh`
		`/bin/sh`
		None (this value prevents users with accounts in the directory domain from accessing the server remotely via a command line)
		Non-zero length.

`Authentication Authority:` Describes the user's authentication methods, such as Open Directory or crypt password; not required for a user with only a crypt password; absence of this attribute signifies legacy authentication (crypt with Authentication Manager, if it's available).	ASCII text	Values describe the user's authentication methods. Can be multivalued (for example, basic and ShadowHash). Each value has the format *vers*; *tag*; *data* (where *vers* and *data* may be blank). Crypt password: ;basic; Open Directory authentication: ;ApplePasswordServer; *HexID, server's public key IPaddress:port* Shadow password (local directory domain only): ;ShadowHash;
`AuthenticationHint:` Text set by the user to be displayed as a password reminder	UTF-8 text	`Your guess is as good as mine.` Maximum 255 bytes.

Mail Attributes in User Records

The following table lists the standard XML data structures for a user mail attribute, part of a standard user record.

MailAttribute field	Description	Sample values

`AttributeVersion`	A required case-insensitive value that must be set to `AppleMail 1.0`.	`<key>`
		`kAttributeVersion`
		`</key>`
		`<string>`
		`AppleMail 1.0`
		`</string>`

`MailAccountState`	A required case-insensitive keyword describing the state of the user's mail. It must be set to one of these values: `Off`, `Enabled`, or `Forward`.	`<key>`
		`kMailAccountState`
		`</key>`
		`<string>`
		`Enabled`
		`</string>`

`POP3LoginState`	A required case-insensitive keyword indicating whether the user is allowed to access mail via POP. It must be set to one of these values: `POP3Allowed` or `POP3Deny`.	`<key>`
		`kPOP3LoginState`
		`</key>`
		`<string>`
		`POP3Deny`
		`</string>`

`IMAPLoginState`	A required case-insensitive keyword indicating whether the user is allowed to access mail using IMAP. It must be set to one of these values: `IMAPAllowed` or `IMAPDeny`.	`<key>`
		`kIMAPLoginState`
		`</key>`
		`<string>`
		`IMAPAllowed`
		`</string>`

`MailAccountLocation`	A required value indicating the domain name or IP address of the ProductName responsible for storing the user's mail.	`<key>`
		`kMailAccountLocation`
		`</key>`
		`<string>`
		`domain.example.com`
		`</string>`

`AutoForwardValue`	A required field only if MailAccountState has the value `Forward`. The value must be a valid RFC 822 email address.	`<key>`
		`kAutoForwardValue`
		`</key>`
		`<string>`
		`user@example.com`
		`</string>`

MailAttribute field	Description	Sample values
NotificationState	An optional keyword describing whether to notify the user whenever new mail arrives. If provided, it must be set to one of these values: `NotificationOff`, `NotificationLastIP`, or `NotificationStaticIP`. If this field is missing, `NotificationOff` is assumed.	```<key>kNotificationState</key><string>NotificationOff</string>```
NotificationStaticIP Value	An optional IP address, in bracketed, dotted decimal format (`[xxx.xxx.xxx.xxx]`). If this field is missing, `NotificationState` is interpreted as `NotificationLastIP`. The field is used only when `NotificationState` has the value `NotificationStaticIP`.	```<key>kNotificationStaticIPValue</key><string>[1.2.3.4]</string>```
SeparateInboxState	An optional case-insensitive keyword indicating whether the user manages POP and IMAP mail using different inboxes. If provided, it must be set to one of these values: `OneInbox` or `DualInbox`. If this value is missing, the value `OneInbox` is assumed.	```<key>kSeparateInboxState</key><string>OneInbox</string>```
ShowPOP3InboxInIMAP	An optional case-insensitive keyword indicating whether POP messages are displayed in the user's IMAP folder list. If provided, it must be set to one of these values: `ShowPOP3Inbox` or `HidePOP3Inbox`. If this field is missing, the value `ShowPOP3Inbox` is assumed.	```<key>kShowPOP3InboxInIMAP</key><string>HidePOP3Inbox</string>```

Checking a Server User's Name, UID, or Password

You can use the following commands to check the name, UID, or password of a user in the server's local directory.

Note: These tasks only apply to the local directory on the server.

To see if a full name is already in use:

```
$ serversetup -verifyRealName "longname"
```

The command displays a 1 if the name is already in the directory, 0 if it isn't.

To see if a short name is already in use:

```
$ serversetup -verifyName shortname
```

The command displays a 1 if the name is already in the directory, 0 if it isn't.

To see if a UID is already in use:

```
$ serversetup -verifyUID userid
```

The command displays a 1 if the UID is already in the directory, 0 if it isn't.

To test a user's password:

```
$ serversetup -verifyNamePassword shortname password
```

The command displays a 1 if the password is good, 0 if it isn't.

To view the names associated with a UID:

```
$ serversetup -getNamesByID userid
```

No response means UID not valid.

To generate the default UNIX short name for a user long name:

```
$ serversetup -getUNIXName "longname"
```

Creating a User's Home Directory

Normally, you can create a user's home directory by clicking the Create Home Now button on the Homes pane of Workgroup Manager. You can also create home directory folders using the `createhomedir` tool. Otherwise, Mac OS X Server creates the user's home directory when the user logs in for the first time.

You can use `createhomedir` to create
• A home directory for a particular user (`-u` option)
• Home directories for all users in a directory domain (`-n` or `-l` option)
• Home directories for all users in all domains in the directory search path (`-a` option)

For more information, type `man createhomedir` to view the man page.

In all cases, the home directories are created on the server where you run the tool.

To create a home directory for a particular user:
```
$ createhomedir [(-a|-l|-n domain)] -u userid
```

To create a home directory for users in the local domain:
```
$ createhomedir -l
```

To create a home directory for users in the local domain:
```
$ createhomedir [(-a|-l|-n domain)] -u userid
```

You can also create a user's home directory using the serversetup tool.

To create a home directory for a particular user:
```
$ serversetup -createHomedir userid
```

The command displays a 1 if the user ID you specify doesn't exist.

Mounting a User's Home Directory

You can use the `mnthome` command to mount a user's home directory. For more information, see the man page.

Creating a Group Folder

You can use the `CreateGroupFolder` command to set up group folders. For more information see the man page.

Checking a User's Administrator Privileges

To see if a user is a server administrator:
```
$ serversetup -isAdministrator shortname
```

The command displays a 0 if the user has administrator privileges, 0 if the user doesn't.

Working With File Services

8

Commands you can use to create share points and manage AFP, NFS, Windows (SMB), and FTP services in Mac OS X Server.

Share Points

You can use the `sharing` tool to list, create, and modify share points.

Listing Share Points

To list existing share points:

```
$ sharing -l
```

In the resulting list, there's a section of properties similar to the following for each share point defined on the server. (1 = yes, true, or enabled. 0 = false, no, or disabled.)

```
name:           Share1
path:           /Volumes/100GB
        afp:    {
                name:   Share1
                shared: 1
                guest access:   0
                inherit perms:  0
        }
        ftp:    {
                name:   Share1
                shared: 1
                guest access:   1
        }
        smb:    {
                name:   Share1
                shared: 1
                guest access:   1
                inherit perms:  0
                oplocks:        0
                strict locking: 0
                directory mask: 493
                create mask:    420 }
```

Creating a Share Point

To create a share point:

```
$ sharing -a path [-n customname] [-A afpname] [-F ftpname]
      [-S smbname] [-s shareflags] [-g guestflags] [-i inheritflags]
      [-c creationmask] [-d directorymask] [-o oplockflag]
      [-t strictlockingflag]
```

Parameter	Description
path	The full path to the directory you want to share.
customname	The name of the share point. If you don't specify this custom name, it's set to the name of the directory, the last name in path.
afpname	The share point name shown to and used by AFP clients. This name is separate from the share point name.
ftpname	The share point name shown to and used by FTP clients.
smbname	The share point name shown to and used by SMB clients.
shareflags	A three-digit binary number indicating which protocols are used to share the directory. The digits represent, from left to right, AFP, FTP, and SMB. 1=shared, 0=not shared.
guestflags	A group of three flags indicating which protocols allow guest access. The flags are written as a three-digit binary number with the digits representing, from left to right, AFP, FTP, and SMB. 1=guests allowed, 0=guests not allowed.
inheritflags	A group of two flags indicating whether new items in AFP or SMB share points inherit the ownership and access permissions of the parent folder. The flags are written as a two-digit binary number with the digits representing, from left to right, AFP and SMB. 1=inherit, 0=don't inherit.
creationmask	The SMB creation mask. Default=0644.
directorymask	The SMB directory mask. Default=0755.
oplockflag	Specifies whether opportunistic locking is allowed for an SMB share point. 1=enable oplocks, 0=disable oplocks. For more information on oplocks, see the file services administration guide.
strictlockingflag	Specifies whether strict locking is used on an SMB share point. 1=enable strict locking, 0=disable. For more information on strict locking, see the file services administration guide.

Examples

```
$ sharing -a /Volumes/100GB/Art
```

Creates a share point named Art, shared using AFP, FTP, and SMB, and using the name Art for all three types of clients.

```
$ sharing -a /Volumes/100GB/Windows\ Docs -n WinDocs -S Documents -s
      001 -o 1
```

Shares the directory named Windows Docs on the disk 100GB. The share point is named WinDocs for server management purposes, but SMB users see it as Documents. It's shared using only the SMB protocol with oplocks enabled.

Modifying a Share Point

To change share point settings:

```
$ sharing -e sharepointname [-n customname] [-A afpname] [-F ftpname]
    [-S smbname] [-s shareflags] [-g guestflags] [-i inheritflags]
    [-c creationmask] [-d directorymask] [-o oplockflag]
    [-t strictlockingflag]
```

Parameter	Description
sharepointname	The current name of the share point.
Other parameters	See the parameter descriptions under "Creating a Share Point" on page 66.

Disabling a Share Point

To disable a share point:

```
$ sharing -r sharepointname
```

Parameter	Description
sharepointname	The current name of the share point.

AFP Service

Starting and Stopping AFP Service

To start AFP service:

```
$ sudo serveradmin start afp
```

To stop AFP service:

```
$ sudo serveradmin stop afp
```

Checking AFP Service Status

To see if AFP service is running:

```
$ sudo serveradmin status afp
```

To see complete AFP status:

```
$ sudo serveradmin fullstatus afp
```

Viewing AFP Settings

To list all AFP service settings:

```
$ sudo serveradmin settings afp
```

To list a particular setting:

```
$ sudo serveradmin settings afp:setting
```

Parameter	Description
setting	Any of the AFP service settings. For a complete list of settings, type `serveradmin settings afp` or see "List of AFP Settings" on this page.

To list a group of settings:

You can list a group of settings that have part of their names in common by typing only as much of the name as you want, stopping at a colon (:), and typing an asterisk (*) as a wildcard for the remaining parts of the name. For example,

```
$ sudo serveradmin settings afp:loggingAttributes:*
```

Changing AFP Settings

You can change AFP service settings using the `serveradmin` command.

To change a setting:

```
$ sudo serveradmin settings afp:setting = value
```

Parameter	Description
setting	An AFP service setting. To see a list of available settings, type `$ sudo serveradmin settings afp` or see "List of AFP Settings" on this page.
value	An appropriate value for the setting. Enclose text strings in double quotes (for example, `"text string"`).

To change several settings:

```
$ sudo serveradmin settings
afp:setting = value
afp:setting = value
afp:setting = value
[...]
Control-D
```

List of AFP Settings

The following table lists AFP settings as they appear using `serveradmin`.

Parameter (afp:)	Description
activityLog	Turn activity logging on or off. Default = no
activityLogPath	Location of the activity log file. Default = /Library/Logs/AppleFileService/ AppleFileServiceAccess.log

Parameter (`afp:`)	Description
`activityLogSize`	Rollover size (in kilobytes) for the activity log. Only used if `activityLogTime` isn't specified. Default = `1000`
`activityLogTime`	Rollover time (in days) for the activity log. Default = `7`
`admin31GetsSp`	Set to true to force administrative users on Mac OS X to see share points instead of all volumes. Default = `yes`
`adminGetsSp`	Set to true to force administrative users on Mac OS 9 to see share points instead of all volumes. Default = `no`
`afpServerEncoding`	Encoding used with Mac OS 9 clients. Default = `0`
`afpTCPPort`	TCP port used by AFP on server. Default = `548`
`allowRootLogin`	Allow user to log in as root. Default = `no`
`attemptAdminAuth`	Allow an administrator user to masquerade as another user. Default = `yes`

`authenticationMode`	Authentication mode. Can be:
	 `standard`
	 `kerberos`
	 `standard_and_kerberos`
	 Default = `"standard_and_kerberos"`

`autoRestart`	Whether the AFP service should restart automatically when abnormally terminated. Default = `yes`
`clientSleepOnOff`	Allow client computers to sleep. Default = `yes`
`clientSleepTime`	Time (in hours) that clients are allowed to sleep. Default = `24`
`createHomeDir`	Create home directories. Default = `yes`
`errorLogPath`	The location of the error log. Default = `/Library/Logs/AppleFileService/AppleFileServiceError.log`
`errorLogSize`	Rollover size (in kilobytes) for the error log. Only used if `errorLogTime` isn't specified. Default = `1000`
`errorLogTime`	Rollover time (in days) for the error log. Default = `0`

Parameter (`afp:`)	Description
`guestAccess`	Allow guest users access to the server.
	Default = `yes`
`idleDisconnectFlag:` 　`adminUsers`	Enforce idle disconnect for administrative users.
	Default = `yes`
`idleDisconnectFlag:` 　`guestUsers`	Enforce idle disconnect for guest users.
	Default = `yes`
`idleDisconnectFlag:` 　`registeredUsers`	Enforce idle disconnect for registered users.
	Default = `yes`
`idleDisconnectFlag:` 　`usersWithOpenFiles`	Enforce idle disconnect for users with open files.
	Default = `yes`
`idleDisconnectMsg`	The idle disconnect message.
	Default = `""`
`idleDisconnectOnOff`	Enable idle disconnect.
	Default = `no`
`idleDisconnectTime`	Idle time (in minutes) allowed before disconnect.
	Default = `10`
`kerberosPrincipal`	Kerberos server principal name.
	Default = `"afpserver"`
`loggingAttributes:` 　`logCreateDir`	Record directory creations in the activity log.
	Default = `yes`
`loggingAttributes:` 　`logCreateFile`	Record file creations in the activity log.
	Default = `yes`
`loggingAttributes:` 　`logDelete`	Record file deletions in the activity log.
	Default = `yes`
`loggingAttributes:` 　`logLogin`	Record user logins in the activity log.
	Default = `yes`
`loggingAttributes:` 　`logLogout`	Log user logouts in the activity log.
	Default = `yes`
`loggingAttributes:` 　`logOpenFork`	Log file opens in the activity log.
	Default = `yes`
`loginGreeting`	The login greeting message.
	Default = `""`
`loginGreetingTime`	The last time the login greeting was set or updated.
`maxConnections`	Maximum number of simultaneous user sessions allowed by the server.
	Default = `-1` (unlimited)
`maxGuests`	Maximum number of simultaneous guest users allowed.
	Default = `-1` (unlimited)

Parameter (`afp:`)	Description
`maxThreads`	Maximum number of AFP threads. (Must be specified at startup.)
	Default = `40`
`noNetworkUsers`	Indication to client that all users are users on the server.
	Default = `no`
`permissionsModel`	How permissions are enforced. Can be set to:
	`classic_permissions`
	`unix_with_classic_admin_permissions`
	`unix_permissions`
	Default = `"classic_permissions"`
`recon1SrvrKeyTTLHrs`	Time-to-live (in hours) for the server key used to generate reconnect tokens.
	Default = `168`
`recon1TokenTTLMins`	Time-to-live (in minutes) for a reconnect token.
	Default = `10080`
`reconnectFlag`	Allow reconnect options. Can be set to:
	`none`
	`all`
	`no_admin_kills`
	Default = `"all"`
`reconnectTTLInMin`	Time-to-live (in minutes) for a disconnected session waiting reconnection.
	Default = `1440`
`registerAppleTalk`	Advertise the server using AppleTalk NBP.
	Default = `yes`
`registerNSL`	Advertise the server using Rendezvous.
	Default = `yes`
`sendGreetingOnce`	Send the login greeting only once.
	Default = `no`
`shutdownThreshold`	Don't modify. Internal use only.
`specialAdminPrivs`	Grant administrative users super user read/write privileges.
	Default = `no`
`SSHTunnel`	Allow SSH tunneling.
	Default = `yes`
`TCPQuantum`	TCP message quantum.
	Default = `262144`
`tickleTime`	Frequency of tickles sent to client.
	Default = `30`
`updateHomeDirQuota`	Enforce quotas on the users volume.
	Default = `yes`

Parameter (`afp:`)	Description
useAppleTalk	Don't modify. Internal use only.
useHomeDirs	Default = no

List of AFP `serveradmin` Commands

In addition to the standard `start`, `stop`, `status`, and `settings` commands, you can use `serveradmin` to issue the following service-specific AFP commands.

Command (`afp:command=`)	Description
cancelDisconnect	Cancel a pending user disconnect. See "Canceling a User Disconnect" on page 74.
disconnectUsers	Disconnect AFP users. See "Disconnecting AFP Users" on page 73.
getConnectedUsers	List settings for connected users. See "Listing Connected Users" on this page.
getHistory	View a periodic record of file data throughput or number of user connections. See "Listing AFP Service Statistics" on page 75.
getLogPaths	Display the locations of the AFP service activity and error logs.
sendMessage	Send a text message to connected AFP users. See "Sending a Message to AFP Users" on page 73.
syncSharePoints	Update share point information after changing settings.
writeSettings	Equivalent to the standard `serveradmin settings` command, but also returns a setting indicating whether the service needs to be restarted. See "Determining Whether a Service Needs to be Restarted" on page 19.

Listing Connected Users

You can use the `serveradmin getConnectedUsers` command to retrieve information about connected AFP users. In particular, you can use this command to retrieve the session IDs you need to disconnect or send messages to users.

To list connected users:

```
$serveradmin command afp:command = getConnectedUsers
```

Output

The following array of settings is displayed for each connected user:

```
afp:usersArray:_array_index:i:disconnectID = <disconnectID>
afp:usersArray:_array_index:i:flags = <flags>
afp:usersArray:_array_index:i:ipAddress = <ipAddress>
afp:usersArray:_array_index:i:lastUseElapsedTime = <lastUseElapsed>
afp:usersArray:_array_index:i:loginElapsedTime = <loginElapsedTime>
afp:usersArray:_array_index:i:minsToDisconnect = <minsToDisconnect>
afp:usersArray:_array_index:i:name = <name>
afp:usersArray:_array_index:i:serviceType = <serviceType>
afp:usersArray:_array_index:i:sessionID = <sessionID>
afp:usersArray:_array_index:i:sessionType = <sessionType>
afp:usersArray:_array_index:i:state = <state>
```

Sending a Message to AFP Users

You can use the `serveradmin sendMessage` command to send a text message to connected AFP users. Users are specified by session ID.

To send a message:

```
$ sudo serveradmin command
afp:command = sendMessage
afp:message = "message-text"
afp:sessionIDsArray:_array_index:0 = sessionid1
afp:sessionIDsArray:_array_index:1 = sessionid2
afp:sessionIDsArray:_array_index:2 = sessionid3
[...]
Control-D
```

Parameter	Description
message-text	The message that appears on client computers.
sessionid*n*	The session ID of a user you want to receive the message. To list the session IDs of connected users, use the `getConnectedUsers` command. See "Listing Connected Users" on page 72.

Disconnecting AFP Users

You can use the `serveradmin disconnectUsers` command to disconnect AFP users. Users are specified by session ID. You can specify a delay time before disconnect and a warning message.

To disconnect users:

```
$ sudo serveradmin command
afp:command = disconnectUsers
afp:message = "message-text"
afp:minutes = minutes-until
afp:sessionIDsArray:_array_index:0 = sessionid1
afp:sessionIDsArray:_array_index:1 = sessionid2
afp:sessionIDsArray:_array_index:2 = sessionid3
[...]
Control-D
```

Parameter	Description
message-text	The text of a message that appears on client computers in the disconnect announcement dialog.
minutes-until	The number of minutes between the time the command is issued and the users are disconnected.
sessionid*n*	The session ID of a user you want to disconnect. To list the session IDs of connected users, use the `getConnectedUsers` command. See "Listing Connected Users" on page 72.

Output
```
afp:command = "disconnectUsers"
afp:messageSent = "<message>"
afp:timeStamp = "<time>"
afp:timerID = <disconnectID>
<user listing>
afp:status = <status>
```

Value	Description
`<message>`	The message sent to users in the disconnect announcement dialog.
`<time>`	The time when the command was issued.
`<disconnectID>`	An integer that identifies this particular disconnect. You can use this ID with the `cancelDisconnect` command to cancel the disconnect.
`<user listing>`	A standard array of user settings for each user scheduled for disconnect. For a description of these settings, see "Listing Connected Users" on page 72.
`<status>`	A command status code: 0 = command successful

Canceling a User Disconnect

You can use the `serveradmin cancelDisconnect` command to cancel a `disconnectUsers` command. Users receive an announcement that they're no longer scheduled to be disconnected.

To cancel a disconnect:
```
$ sudo serveradmin command
afp:command = cancelDisconnect
afp:timerID = timerID
Control-D
```

Parameter	Description
`timerID`	The integer value of the `afp:timerID` parameter output when you issued the `disconnectUsers` command. You can also find this number by listing any user scheduled to be disconnected and looking at the value of the `disconnectID` setting for the user.

Output
```
afp:command = "cancelDisconnect"
afp:timeStamp = "<time>"
afp:status = <status>
```

Value	Description
`<time>`	The time at which the command was issued.
`<status>`	A command status code: 0 = command successful

Listing AFP Service Statistics

You can use the `serveradmin getHistory` command to display a log of periodic samples of the number of connections and the data throughput. Samples are taken once each minute.

To list samples:

```
$ sudo serveradmin command
afp:command = getHistory
afp:variant = statistic
afp:timeScale = scale
Control-D
```

Parameter	Description
statistic	The value you want to display. Valid values: v1 - number of connected users (average during sampling period) v2 - throughput (bytes/sec)
scale	The length of time in seconds, ending with the current time, for which you want to see samples. For example, to see 30 minutes of data, you would specify afp:timeScale = 1800.

Output

```
afp:nbSamples = <samples>
afp:samplesArray:_array_index:0:vn = <sample>
afp:samplesArray:_array_index:0:t = <time>
afp:samplesArray:_array_index:1:vn = <sample>
afp:samplesArray:_array_index:1:t = <time>
[...]
afp:samplesArray:_array_index:i:vn = <sample>
afp:samplesArray:_array_index:i:t = <time>
afp:vnLegend = "<legend>"
afp:currentServerTime = <servertime>
```

Value displayed by getHistory	Description
<samples>	The total number of samples listed.
<legend>	A textual description of the selected statistic. "CONNECTIONS" for v1 "THROUGHPUT" for v2
<sample>	The numerical value of the sample. For connections (v1), this is integer average number of users. For throughput, (v2), this is integer bytes per second.
<time>	The time at which the sample was measured. A standard UNIX time (number of seconds since Sep 1, 1970.) Samples are taken every 60 seconds.

Viewing AFP Log Files

You can use `tail` or any other file listing tool to view the contents of the AFP service logs.

To view the latest entries in a log:

```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the current AFP error and activity logs are located.

To display the log paths:

```
$ sudo serveradmin command afp:command = getLogPaths
```

Output

```
afp:accesslog = <access-log>
afp:errorlog = <error-log>
```

Value	Description
`<access-log>`	The location of the AFP service access log. Default = `/Library/Logs/AppleFileService/ AppleFileServiceAccess.log`
`<error-log>`	The location of the AFP service error log. Default = `/Library/Logs/AppleFileService/ AppleFileServiceError.log`

NFS Service

Starting and Stopping NFS Service

NFS service is started automatically when a share point is exported using NFS. The NFS daemons that satisfy client requests continue to run until there are no more NFS exports and the server is restarted.

Checking NFS Service Status

To see if NFS service and related processes are running:

```
$ sudo serveradmin status nfs
```

To see complete NFS status:

```
$ sudo serveradmin fullstatus nfs
```

Viewing NFS Settings

To list all NFS service settings:

```
$ sudo serveradmin settings nfs
```

To list a particular setting:

```
$ sudo serveradmin settings nfs:setting
```

Changing NFS Service Settings

Use the following parameters with the `serveradmin` command to change settings for the NFS service.

Parameter (nfs:)	Description
`nbDaemons`	Default = 6
	To reduce the number of daemons, you must restart the server after changing this value.
`useTCP`	Default = `yes`
	You must restart the server after changing this value.
`useUDP`	Default = `yes`
	You must restart the server after changing this value.

FTP Service

Starting FTP Service

To start FTP service:

```
$ sudo serveradmin start ftp
```

Stopping FTP Service

To stop FTP service:

```
$ sudo serveradmin stop ftp
```

Checking FTP Service Status

To see if FTP service is running:

```
$ sudo serveradmin status ftp
```

To see complete FTP status:

```
$ sudo serveradmin fullstatus ftp
```

Viewing FTP Settings

To list all FTP service settings:

```
$ sudo serveradmin settings ftp
```

To list a particular setting:

```
$ sudo serveradmin settings ftp:setting
```

To list a group of settings:

You can list a group of settings that have part of their names in common by typing only as much of the name as you want, stopping at a colon (:), and typing an asterisk (*) as a wildcard for the remaining parts of the name. For example,

```
$ sudo serveradmin settings ftp:logCommands:*
```

Changing FTP Settings

You can change FTP service settings using the `serveradmin` application.

To change a setting:

```
$ sudo serveradmin settings ftp:setting = value
```

Parameter	Description
setting	An FTP service setting. To see a list of available settings, type `$ sudo serveradmin settings ftp` or see "FTP Settings" on this page.
value	An appropriate value for the setting.

To change several settings:

```
$ sudo serveradmin settings
ftp:setting = value
ftp:setting = value
ftp:setting = value
[...]
Control-D
```

FTP Settings

Use the following parameters with the `serveradmin` command to change settings for the FTP service.

Parameter (`ftp:`)	
administratorEmailAddress	Default = `"user@hostname"`
anonymous-root	Default = `"/Library/FTPServer/FTPRoot"`
anonymousAccessPermitted	Default = `no`
authLevel	Default = `"STANDARD"`

bannerMessage	Default = `"This is the "Banner"`
	`message for the Mac OS X Server's`
	`FTP server process.`
	
	`FTP clients will receive this`
	`message immediately before being`
	`prompted for a name and password.`
	
	`PLEASE NOTE: Some FTP clients may`
	`exhibit problems if you make this`
	`file too long.`
	
	`---------------------------------"`

chrootType	Default = `"STANDARD"`
enableMacBinAndDmgAutoConversion	Default = `yes`
ftpRoot	Default = `"/Library/FTPServer/FTPRoot"`

Parameter (`ftp:`)	
`logCommands:anonymous`	Default = `no`
`logCommands:guest`	Default = `no`
`logCommands:real`	Default = `no`
`loginFailuresPermitted`	Default = `3`
`logSecurity:anonymous`	Default = `no`
`logSecurity:guest`	Default = `no`
`logSecurity:real`	Default = `no`
`logToSyslog`	Default = `no`
`logTransfers:anonymous:inbound`	Default = `yes`
`logTransfers:anonymous:outbound`	Default = `yes`
`logTransfers:guest:inbound`	Default = `no`
`logTransfers:guest:outbound`	Default = `no`
`logTransfers:real:inbound`	Default = `yes`
`logTransfers:real:outbound`	Default = `yes`
`maxAnonymousUsers`	Default = `50`
`maxRealUsers`	Default = `50`
`showBannerMessage`	Default = `yes`
`showWelcomeMessage`	Default = `yes`

`welcomeMessage`	Default = `"This is the "Welcome" message for the Mac OS X Server's FTP server process.`
	
	`FTP clients will receive this message right after a successful log in.`
	
	`--------------------------------"`

List of FTP `serveradmin` Commands

You can use the following commands with the `serveradmin` application to manage FTP service.

ftp:command=	Description
`getConnectedUsers`	List connected users. See "Checking for Connected FTP Users" on page 80.

ftp:command=	Description
getLogPaths	Show location of the FTP transfer log file. See "Viewing the FTP Transfer Log" on this page.
writeSettings	Equivalent to the standard `serveradmin settings` command, but also returns a setting indicating whether the service needs to be restarted. See "Determining Whether a Service Needs to be Restarted" on page 19.

Viewing the FTP Transfer Log

You can use `tail` or any other file listing tool to view the contents of the FTP transfer log.

To view the latest entries in the transfer log:

```
$ tail log-file
```

The default location of `log-file` is /Library/Logs/FTP.transger.log. You can use the `serveradmin getLogPaths` command to see where the current transfer log is located.

To display the log path:

```
$ sudo serveradmin command ftp:command = getLogPaths
```

Checking for Connected FTP Users

To see how many FTP users are connected:

```
$ ftpcount
```

or

```
$ sudo serveradmin command ftp:command = getConnectedUsers
```

Windows (SMB) Service

Starting and Stopping SMB Service

To start SMB service:

```
$ sudo serveradmin start smb
```

To stop SMB service:

```
$ sudo serveradmin stop smb
```

Checking SMB Service Status

To see if SMB service is running:

```
$ sudo serveradmin status smb
```

To see complete SMB status:

```
$ sudo serveradmin fullstatus smb
```

Viewing SMB Settings

To list all SMB service settings:

```
$ sudo serveradmin settings smb
```

To list a particular setting:

```
$ sudo serveradmin settings smb:setting
```

Parameter	Description
setting	An SMB service setting. To see a list of available settings, type `$ sudo serveradmin settings smb` or see "List of SMB Service Settings" on page 82.

To list a group of settings:

You can list a group of settings that have part of their names in common by typing only as much of the name as you want, stopping at a colon (:), and typing an asterisk (*) as a wildcard for the remaining parts of the name. For example,

```
$ sudo serveradmin settings smb:adminCommands:*
```

Changing SMB Settings

You can change SMB service settings using the `serveradmin` command.

To change a setting:

```
$ sudo serveradmin settings smb:setting = value
```

Parameter	Description
setting	An SMB service setting. To see a list of available settings, type `$ sudo serveradmin settings smb` or see "List of SMB Service Settings" on page 82.
value	An appropriate value for the setting. For a list of values that correspond to GUI controls in the Server Admin application, see "List of SMB Service Settings" on page 82.

To change several settings:

```
$ sudo serveradmin settings
smb:setting = value
smb:setting = value
smb:setting = value
[...]
Control-D
```

List of SMB Service Settings

Use the following parameters with the `serveradmin` command to change settings for the SMB service.

Parameter (`smb:`)	Description
`adminCommands:homes`	Whether home directories are mounted automatically when Windows users log in so you don't have to set up individual share points for each user. Can be set to: `yes|no` Corresponds to the "Enable virtual share points" checkbox in the Advanced pane of Window service settings in the Server Admin GUI application.
`adminCommands:serverRole`	The authentication role played by the server. Can be set to: `"standalone"` `"domainmember"` `"primarydomaincontroller"` Corresponds to the Role pop-up menu in the General pane of Windows service settings in the Server Admin GUI application.
`domain master`	Whether the server is providing domain master browser service. Can be set to: `yes|no` Corresponds to the Domain Master Browser checkbox in the Advanced pane of Window service settings in the Server Admin GUI application.
`dos charset`	The code page being used. Can be set to: `CP437` (Latin US) `CP737` (Greek) `CP775` (Baltic) `CP850` (Latin1) `CP852` (Latin2) `CP861` (Icelandic) `CP866` (Cyrillic) `CP932` (Japanese SJIS) `CP936` (Simplified Chinese) `CP949` (Korean Hangul) `CP950` (Traditional Chinese) `CP1251` (Windows Cyrillic) Corresponds to the Code Page pop-up menu on the Advanced pane of Windows service settings in the Server Admin GUI application.

Parameter (`smb:`)	Description
`local master`	Whether the server is providing workgroup master browser service. Can be set to: `yes`|`no` Corresponds to the Workgroup Master Browser checkbox in the Advanced pane of Window service settings in the Server Admin GUI application.
`log level`	The amount of detail written to the service logs. Can be set to: `0` (Low: errors and warnings only) `1` (Medium: service start and stop, authentication failures, browser name registrations, and errors and warnings) `2` (High: service start and stop, authentication failures, browser name registration events, log file access, and errors and warnings) Corresponds to the Log Detail pop-up menu in the Logging pane of Window service settings in the Server Admin GUI application
map to guest	Whether guest access is allowed. Can be set to: `"Never"` (No guest access) `"Bad User"` (Allow guest access) Corresponds to the "Allow Guest access" checkbox in the Access pane of Window service settings in the Server Admin GUI application
`max smbd processes`	The maximum allowed number of smb server processes. Each connection uses its own smbd process, so this is the same as specifying the maximum number of SMB connections. `0` means unlimited. This corresponds to the "maximum" client connections field in the Access pane of the Windows service settings in the Server Admin GUI application.
`netbios name`	The server's NetBIOS name. Can be set to a maximum of 15 bytes of UTF-8 characters. Corresponds to the Computer Name field in the General pane of the Windows service settings in the Server Admin GUI application.
`server string`	Text that helps identify the server in the network browsers of client computers. Can be set to a maximum of 15 bytes of UTF-8 characters. Corresponds to the Description field in the General pane of the Windows service settings in the Server Admin GUI application.
`wins support`	Whether the server provides WINS support. Can be set to: `yes`|`no` Corresponds to the WINS Registration "Off" and "Enable WINS server" selections in the Advanced pane of the Windows service settings in the Server Admin GUI application.

Parameter (`smb:`)	Description
`wins server`	The name of the WINS server used by the server.
	Corresponds to the WINS Registration "Register with WINS server" selection and field in the Advanced pane of the Windows service settings in the Server Admin GUI application.
`workgroup`	The server's workgroup. Can be set to a maximum of 15 bytes of UTF-8 characters.
	Corresponds to the Workgroup field in the General pane of the Windows service settings in the Server Admin GUI application.

List of SMB `serveradmin` Commands

You can use these commands with the `serveradmin` tool to manage SMB service.

`smb:command=`	Description
`disconnectUsers`	Disconnect SMB users. See "Disconnecting SMB Users" on page 85.
`getConnectedUsers`	List users currently connected to an SMB service. See "Listing SMB Users" on this page.
`getHistory`	List connection statistics. See "Listing SMB Service Statistics" on page 86.
`getLogPaths`	Show location of service log files. See "Viewing SMB Service Logs" on page 87.
`syncPrefs`	Update the service to recognize changes in share points. See "Updating Share Point Information" on page 86.
writeSettings	Equivalent to the standard `serveradmin settings` command, but also returns a setting indicating whether the service needs to be restarted. See "Determining Whether a Service Needs to be Restarted" on page 19.

Listing SMB Users

You can use the `serveradmin getConnectedUsers` command to retrieve information about connected SMB users. For example, you can use this command to retrieve the session IDs you need to disconnect users.

To list connected users:

```
$serveradmin command smb:command = getConnectedUsers
```

Output

The following array of settings is displayed for each connected user:

```
smb:usersArray:_array_index:i:disconnectID = <disconnectID>
smb:usersArray:_array_index:i:sessionID = <sessionID>
smb:usersArray:_array_index:i:connectAt = <connect-time>
smb:usersArray:_array_index:i:service = <service>
smb:usersArray:_array_index:i:loginElapsedTime = <login-elapsed-time>
smb:usersArray:_array_index:i:name = "<name>"
smb:usersArray:_array_index:i:ipAddress = "<ip-address>"
```

Value returned by `getConnectedUsers` (smb:usersArray:_array_index:<*n*>:)	Description
`<sessionID>`	An integer that identifies the user session.
`<connect-time>`	The date and time when the user connected to the server.
`<service>`	The share point the user is accessing.
`<login-elapsed-time>`	The elapsed time since the user connected.
`<name>`	The user's name.
`<ip-address>`	The user's IP address.

Disconnecting SMB Users

You can use the `serveradmin disconnectUsers` command to disconnect SMB users. Users are specified by session ID.

To disconnect users:

```
$ sudo serveradmin command
smb:command = disconnectUsers
smb:sessionIDsArray:_array_index:0 = sessionid1
smb:sessionIDsArray:_array_index:1 = sessionid2
smb:sessionIDsArray:_array_index:2 = sessionid3
[...]
Control-D
```

Parameter	Description
sessionid*n*	The session ID of a user you want to disconnect. To list the session IDs of connected users, use the `getConnectedUsers` command. See "Listing SMB Users" on page 84.

Output

```
smb:command = "disconnectUsers"
smb:status = <status>
```

Value	Description
`<status>`	A command status code: 0 = command successful

Listing SMB Service Statistics

You can use the `serveradmin getHistory` command to display a log of periodic samples of the number of SMB connections. Samples are taken once each minute.

To list samples:
```
$ sudo serveradmin command
smb:command = getHistory
smb:variant = v1
smb:timeScale = scale
Control-D
```

Parameter	Description
v1	The number of connected users (average during sampling period).
scale	The length of time in seconds, ending with the current time, for which you want to see samples. For example, to see 30 minutes of data, you would specify smb:timeScale = 1800.

Output
```
smb:nbSamples = <samples>
smb:samplesArray:_array_index:0:vn = <sample>
smb:samplesArray:_array_index:0:t = <time>
smb:samplesArray:_array_index:1:vn = <sample>
smb:samplesArray:_array_index:1:t = <time>
[...]
smb:samplesArray:_array_index:i:vn = <sample>
smb:samplesArray:_array_index:i:t = <time>
smb:v1Legend = "CONNECTIONS"
smb:currentServerTime = <servertime>
```

Value displayed by getHistory	Description
<samples>	The total number of samples listed.
<legend>	A textual description of the selected statistic.
	"CONNECTIONS" for v1
	"THROUGHPUT" for v2
<sample>	The numerical value of the sample.
	For connections (v1), this is integer average number of users.
	For throughput, (v2), this is integer bytes per second.
<time>	The time at which the sample was measured. A standard UNIX time (number of seconds since Sep 1, 1970.) Samples are taken every 60 seconds.

Updating Share Point Information

After you make a change to an SMB share point using the `sharing` tool, you need to update the SMB service information.

To update SMB share point information:
```
$ sudo serveradmin command smb:command = syncPrefs
```

Viewing SMB Service Logs

You can use `tail` or any other file listing tool to view the contents of the SMB service logs.

To view the latest entries in a log:

```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the current SMB logs are located.

To display the log paths:

```
$ sudo serveradmin command smb:command = getLogPaths
```

Output

```
smb:fileServiceLog = <smb-log>
smb:nameServiceLog = <name-log>
```

Value	Description
`<smb-log>`	The location of the SMB service log. Default = `/var/log/samba/log.smbd`
`<name-log>`	The location of the name service log. Default = `/var/log/samba/log.nmbd`

Working With Print Service

<div style="text-align: right">9</div>

Commands you can use to manage the Print service in
Mac OS X Server.

Starting and Stopping Print Service

To start Print service:

```
$ sudo serveradmin start print
```

To stop Print service:

```
$ sudo serveradmin stop print
```

Checking the Status of Print Service

To see summary status of Print service:

```
$ sudo serveradmin status print
```

To see detailed status of Print service:

```
$ sudo serveradmin fullstatus print
```

Viewing Print Service Settings

To list Print service configuration settings:

```
$ sudo serveradmin settings print
```

To list a particular setting:

```
$ sudo serveradmin settings print:setting
```

To list a group of settings:
You can list a group of settings that have part of their names in common by typing
only as much of the name as you want, stopping at a colon (:), and typing an asterisk
(*) as a wildcard for the remaining parts of the name. For example, to see all settings for
a particular print queue:

```
$ sudo serveradmin settings print:queuesArray:_array_id:queue-id:*
```

where queue-id is an id such as 66F66AdA-060B-5603-9024-FCB57AAB24B1.

Changing Print Service Settings

To change a setting:

```
$ sudo serveradmin settings print:setting = value
```

Parameter	Description
setting	A Print service setting. To see a list of available settings, type `$ sudo serveradmin settings print` or see "Print Service Settings" on this page.
value	An appropriate value for the setting.

To change several settings:

```
$ sudo serveradmin settings
print:setting = value
print:setting = value
print:setting = value
[...]
Control-D
```

Print Service Settings

Use the following parameters with the `serveradmin` command to change settings for the Print service.

Parameter (print:)	Description
serverLogArchiveIntervalDays	Default = 7
<queue arrays>	See "Queue Data Array" on page 91.
serverLogArchiveEnable	Default = no
jobLogArchiveIntervalDays	Default = 7
jobLogArchiveEnable	Default = no

Queue Data Array

Print service settings include an array of values for each existing print queue. The array is a set of 14 parameters that define values for each queue.

`<id>` is the queue ID, for example, `29D3ECF3-17C8-16E5-A330-84CEC733F249`.

Parameter (`print:`)	Description
`queuesArray:_array_id:<id>:` `quotasEnforced`	Default = `no`

`queuesArray:_array_id:<id>:`
`sharingList:_array_index:0:`
`service`
	Default = `"LPR"`

`queuesArray:_array_id:<id>:`
`sharingList:_array_index:0:`
`sharingEnable`
	Default = `no`

`queuesArray:_array_id:<id>:`
`sharingList:_array_index:1:`
`service`
	Default = `"SMB"`

`queuesArray:_array_id:<id>:`
`sharingList:_array_index:1:`
`sharingEnable`
	Default = `no`

`queuesArray:_array_id:<id>:`
`sharingList:_array_index:2:`
`service`
	Default = `"PAP"`

`queuesArray:_array_id:<id>:`
`sharingList:_array_index:2:`
`sharingEnable`
	Default = `no`

`queuesArray:_array_id:<id>:` `shareable`	Default = `yes`. Cannot be changed.
`queuesArray:_array_id:<id>:` `defaultJobPriority`	Not used. Default = `"NORMAL"`
`queuesArray:_array_id:<id>:` `printerName`	Default = `"<printer-name>"` Cannot be changed using serveradmin.
`queuesArray:_array_id:<id>:` `defaultJobState`	Not used. Default = `"PENDING"`
`queuesArray:_array_id:<id>:` `printerURI`	Default = `<uri>` Format depends on type of printer. Cannot be changed using serveradmin.
`queuesArray:_array_id:<id>:` `registerRendezvous`	Default = `yes`
`queuesArray:_array_id:<id>:` `printerKind`	Default = `"<type>"` Cannot be changed using serveradmin.
`queuesArray:_array_id:<id>:` `sharingName`	Default = `"<name>"`

Here is an example of a queue array parameter block:

```
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-
    84CEC733F249:quotasEnforced = no
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-
    84CEC733F249:sharingList:_array_index:0:service = "LPR"
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-
    84CEC733F249:sharingList:_array_index:0:sharingEnable = no
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-
    84CEC733F249:sharingList:_array_index:1:service = "SMB"
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-
    84CEC733F249:sharingList:_array_index:1:sharingEnable = no
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-
    84CEC733F249:sharingList:_array_index:2:service = "PAP"
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-
    84CEC733F249:sharingList:_array_index:2:sharingEnable = no
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-84CEC733F249:shareable =
    yes
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-
    84CEC733F249:defaultJobPriority = "NORMAL"
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-84CEC733F249:printerName
    = "Room 3 Printer"
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-
    84CEC733F249:defaultJobState = "PENDING"
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-84CEC733F249:printerURI
    = "pap://*/Room%203%20Printer/LaserWriter"
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-
    84CEC733F249:registerRendezvous = yes
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-84CEC733F249:printerKind
    = "HP LaserJet 4100 Series "
print:queuesArray:_array_id:29D3ECF3-17C8-16E5-A330-84CEC733F249:sharingName
    = "Room 3 Printer"
```

Print Service `serveradmin` Commands

You can use the following commands with the `serveradmin` application to manage Print service.

print:command=	Description
`getJobs`	List information about the jobs waiting in a queue. See "Listing Jobs and Job Information" on page 94.
`getLogPaths`	Finding the locations of the Print service and job logs. See "Viewing Print Service Log Files" on page 95.
`getQueues`	List Print service queues. See "Listing Queues" on this page.
`setJobState`	Hold or release a job. See "Holding a Job" on page 94.
`setQueueState`	Pauses or release a queue. See "Pausing a Queue" on this page.
`writeSettings`	Equivalent to the standard `serveradmin settings` command, but also returns a setting indicating whether the service needs to be restarted. See "Determining Whether a Service Needs to be Restarted" on page 19.

Listing Queues

You can use the `serveradmin getQueues` command to list Print service queues.

```
$ sudo serveradmin command print:command = getQueues
```

Pausing a Queue

You can use the `serveradmin setQueueState` command to pause or release a queue.

To pause a queue:

```
$ sudo serveradmin command
print:command = setQueueState
print:status = PAUSED
print:namesArray:_array_index:0 = queue
Control-D
```

Parameter	Description
queue	The name of the queue. To find the name of the queue, use the `getQueues` command and look for the value of the `print` setting. See "Listing Queues" on this page.

To release the queue:

```
$ sudo serveradmin command
print:command = setQueueState
print:status = ""
print:namesArray:_array_index:0 = queue
Control-D
```

Listing Jobs and Job Information

You can use the `serveradmin getJobs` command to list information about print jobs.

```
$ sudo serveradmin command
print:command = getJobs
print:maxDisplayJobs = jobs
print:queueNamesArray:_array_index:0 = queue
Control-D
```

Parameter	Description
jobs	The maximum number of jobs to list.
queue	The name of the queue. To find the name of the queue, use the `getQueues` command and look for the value of the `print` setting. See "Listing Queues" on page 93.

For each job, the command lists:

- Document name
- Number of pages
- Document size
- Number of sheets
- Job ID
- Submitting user
- Submitting host
- Job name
- Job state
- Printing protocol
- Job priority

Holding a Job

You can use the `serveradmin setJobState` command to hold or release a job.

To hold a job:

```
$ sudo serveradmin command
print:command = setJobState
print:status = HOLD
print:namesArray:_array_index:0:printer = queue
print:namesArray:_array_index:0:idsArray:_array_index:0 = jobid
Control-D
```

Parameter	Description
queue	The name of the queue. To find the name of the queue, use the `getQueues` command and look for the value of the `print` setting. See "Listing Queues" on page 93.
jobid	The ID of the job. To find the ID of the job, use the `getJobs` command and look for the value of the `jobId` setting. See "Listing Jobs and Job Information" on this page.

To release the job for printing, change its state to PENDING.

To release the job:
```
$ sudo serveradmin command
print:command = setJobState
print:status = PENDING
print:namesArray:_array_index:0:printer = queue
print:namesArray:_array_index:0:idsArray:_array_index:0 = jobid
Control-D
```

Viewing Print Service Log Files

You can use `tail` or any other file listing tool to view the contents of the Print service logs.

To view the latest entries in a log:
```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the current logs are located.

To display the log paths:
```
$ sudo serveradmin command print:command = getLogPaths
```

Output
```
print:logPathsArray:_array_index:0:path = <service-log>
print:logPathsArray:_array_index:0:name = SYSTEMLOG
print:logPathsArray:_array_index:0:path = <job-log-0>
print:logPathsArray:_array_index:0:path = <queue-name-0>
print:logPathsArray:_array_index:0:path = <job-log-1>
print:logPathsArray:_array_index:0:path = <queue-name-1>
[...]
print:logPathsArray:_array_index:0:path = <job-log-n>
print:logPathsArray:_array_index:0:path = <queue-name-n>
```

Value	Description
`<service-log>`	The location of the primary Print service log. Default = `/Library/Logs/PrintService/PrintService.server.log`
`<job-log-n>`	The location of the job log for the corresponding queue. Default = `/Library/Logs/PrintService/PrintService.<queue-name-n>.job.log`
`<queue-name-n>`	The name of the queue.

Working With NetBoot Service 10

Commands you can use to manage the NetBoot service in Mac OS X Server.

Starting and Stopping NetBoot Service

To start NetBoot service:

```
$ sudo serveradmin start netboot
```

If you get the following response:

```
$ netboot:state = "STOPPED"
$ netboot:status = 5000
```

you have not yet enabled NetBoot on any network port.

To stop NetBoot service:

```
$ sudo serveradmin stop netboot
```

Checking NetBoot Service Status

To see if NetBoot service is running:

```
$ sudo serveradmin status netboot
```

To see complete NetBoot status:

```
$ sudo serveradmin fullstatus netboot
```

Viewing NetBoot Settings

To list all NetBoot service settings:

```
$ sudo serveradmin settings netboot
```

Changing NetBoot Settings

You can change NetBoot service settings using the `serveradmin` command.

To change a setting:

```
$ sudo serveradmin settings netboot:setting = value
```

Parameter	Description
setting	A NetBoot service setting. To see a list of available settings, type `$ sudo serveradmin settings netboot` or see "NetBoot Service Settings" on this page.
value	An appropriate value for the setting.

To change several settings:

```
$ sudo serveradmin settings
netboot:setting = value
netboot:setting = value
netboot:setting = value
[...]
Control-D
```

NetBoot Service Settings

General Settings

Use the following parameters with the `serveradmin` command to change settings for the NetBoot service.

Parameter (`netboot:`)	Description
filterEnabled	Specifies whether client filtering is enabled. Default = `"No"`
netBootStorageRecordsArray...	An array of values for each server volume used to store boot or install images. For a description, see "Storage Record Array" on page 99.
netBootFiltersRecordsArray...	An array of values for each computer explicitly allowed or disallowed access to images. For a description, see "Filters Record Array" on page 99.
netBootImagesRecordsArray...	An array of values for each boot or install image stored on the server. For a description, see "Image Record Array" on page 100.
netBootPortsRecordsArray...	An array of values for each server network port used to deliver boot or install images. For a description, see "Port Record Array" on page 101.

Storage Record Array

A volume parameter array:

Parameter (`netboot:`)	Description
netBootStorageRecordsArray:_array_index:<n>: sharepoint	First parameter in an array describing a volume available to serve images. Default = `"No"`
netBootStorageRecordsArray:_array_index:<n>: clients	Default = `"No"`
netBootStorageRecordsArray:_array_index:<n>: ignorePrivs	Default = `"false"`
netBootStorageRecordsArray:_array_index:<n>: volType	Default = `<voltype>` Example: `"hfs"`
netBootStorageRecordsArray:_array_index:<n>: path	Default = `"/"`
netBootStorageRecordsArray:_array_index:<n>: volName	Default = `<name>`
netBootStorageRecordsArray:_array_index:<n>: volIcon	Default = `<icon>`
netBootStorageRecordsArray:_array_index:<n>: okToDeleteClients	Default = `"Yes"`
netBootStorageRecordsArray:_array_index:<n>: okToDeleteSharepoint	Default = `"Yes"`

Filters Record Array

An array of the following values appears in the NetBoot service settings for each computer explicitly allowed or denied access to images stored on the server:

Parameter (`netboot:`)	Description:
netBootFiltersRecordsArray: _array_index:<n>:hostName	The host name of the filtered computer, if available.

netBootFiltersRecordsArray: _array_index:<n>:filterType	Whether the specified computer is allowed or denied access. Options:
	`"allow"`
	`"deny"`

netBootFiltersRecordsArray: _array_index:<n>:hardwareAddress	The Ethernet hardware (MAC) address of the filtered computer.

Image Record Array

An array of the following values appears in the NetBoot service settings for each image stored on the server:

Parameter (`netboot:`)	Description:
`netBootImagesRecordsArray:` `_array_index:<n>:Name`	Name of the image as it appears in the Startup Disk control panel (Mac OS 9) or Preferences pane (Mac OS X).
`netBootImagesRecordsArray:` `_array_index:<n>:IsDefault`	`Yes` specifies this image file as the default boot image on the subnet.
`netBootImagesRecordsArray:` `_array_index:<n>:RootPath`	The path to the .dmg file.
`netBootImagesRecordsArray:` `_array_index:<n>:isEdited`	
`netBootImagesRecordsArray:` `_array_index:<n>:BootFile`	Name of boot ROM file: `booter`.
`netBootImagesRecordsArray:` `_array_index:<n>:Description`	Arbitrary text describing the image.
`netBootImagesRecordsArray:` `_array_index:<n>:SupportsDiskless`	Yes directs the NetBoot server to allocate space for the shadow files needed by diskless clients.
`netBootImagesRecordsArray:` `_array_index:<n>:Type`	`NFS` or `HTTP`.
`netBootImagesRecordsArray:` `_array_index:<n>:pathToImage`	The path to the parameter list file in the .nbi folder on the server describing the image.
`netBootImagesRecordsArray:` `_array_index:<n>:Index`	`1–4095` indicates a local image unique to the server. `4096–65535` is a duplicate, identical image stored on multiple servers for load balancing.
`netBootImagesRecordsArray:` `_array_index:<n>:IsEnabled`	Sets whether the image is available to NetBoot (or Network Image) clients.
`netBootImagesRecordsArray:` `_array_index:<n>:IsInstall`	`Yes` specifies a Network Install image; `False` specifies a NetBoot image.

Port Record Array

An array of the following items is included in the NetBoot service settings for each network port on the server set to deliver images:

Parameter (`netboot:`)	Description
`netBootPortsRecordsArray:_array_index:<m>:isEnabledAtIndex`	First parameter in an array describing a network interface available for responding to netboot requests. Default = `"No"`
`netBootPortsRecordsArray:_array_index:<m>:nameAtIndex`	Default = `"<devname>"` Example: `"Built-in Ethernet"`
`netBootPortsRecordsArray:_array_index:<m>:deviceAtIndex`	Default = `"<dev>"` Example: `"en0"`

Working With Mail Service

Commands you can use to manage the Mail service in
Mac OS X Server.

Starting and Stopping Mail Service

To start Mail service:

```
$ sudo serveradmin start mail
```

To stop Mail service:

```
$ sudo serveradmin stop mail
```

Checking the Status of Mail Service

To see summary status of Mail service:

```
$ sudo serveradmin status mail
```

To see detailed status of Mail service:

```
$ sudo serveradmin fullstatus mail
```

Viewing Mail Service Settings

To list Mail service configuration settings:

```
$ sudo serveradmin settings mail
```

To list a particular setting:

```
$ sudo serveradmin settings mail:setting
```

To list a group of settings:
You can list a group of settings that have part of their names in common by typing
only as much of the name as you want, stopping at a colon (:), and typing an asterisk
(*) as a wildcard for the remaining parts of the name. For example:

```
$ sudo serveradmin settings mail:imap:*
```

Changing Mail Service Settings

You can use `serveradmin` to modify your server's mail configuration. However, if you want to work with the Mail service from the command-line, you'll probably find it more straightforward to work directly with the underlying Postfix and Cyrus mail services.

For information on Postfix, visit www.postfix.org.

For information on Cyrus IMAP/POP, visit asg.web.cmu.edu/cyrus.

You can also use Sherlock or Google to search the web for information on Postfix or Cyrus.

Mail Service Settings

Use the following parameters with the `serveradmin` command to change settings for the Mail service.

Parameter (mail:)	Description
`postfix:message_size_limit`	Default = `10240000`
`postfix:readme_directory`	Default = `no`
`postfix:double_bounce_sender`	Default = `"double-bounce"`
`postfix:default_recipient_limit`	Default = `10000`
`postfix:local_destination_recipient_limit`	Default = `1`
`postfix:queue_minfree`	Default = `0`
`postfix:show_user_unknown_table_name`	Default = `yes`
`postfix:default_process_limit`	Default = `100`
`postfix:export_environment`	Default = `"TZ MAIL_CONFIG"`
`postfix:smtp_line_length_limit`	Default = `990`
`postfix:smtp_rcpt_timeout`	Default = `"300s"`
`postfix:masquerade_domains`	Default = `""`
`postfix:soft_bounce`	Default = `no`
`postfix:pickup_service_name`	Default = `"pickup"`
`postfix:config_directory`	Default = `"/etc/postfix"`
`postfix:smtpd_soft_error_limit`	Default = `10`
`postfix:undisclosed_recipients_header`	Default = `"To: undisclosed-recipients:;"`
`postfix:lmtp_lhlo_timeout`	Default = `"300s"`
`postfix:smtpd_recipient_restrictions`	Default = `"permit_mynetworks,reject_unauth_destination"`
`postfix:unknown_local_recipient_reject_code`	Default = `450`

Parameter (mail:)	Description
`postfix:error_notice_recipient`	Default = `"postmaster"`
`postfix:smtpd_sasl_local_domain`	Default = `no`
`postfix:strict_mime_encoding_domain`	Default = `no`
`postfix:unknown_relay_recipient_reject_code`	Default = 550
`postfix:disable_vrfy_command`	Default = `no`
`postfix:unknown_virtual_mailbox_reject_code`	Default = 550
`postfix:fast_flush_refresh_time`	Default = `"12h"`
`postfix:prepend_delivered_header`	Default = `"command, file, forward"`
`postfix:defer_service_name`	Default = `"defer"`
`postfix:sendmail_path`	Default = `"/usr/sbin/sendmail"`
`postfix:lmtp_sasl_password_maps`	Default = `no`
`postfix:smtp_sasl_password_maps`	Default = `no`
`postfix:qmgr_clog_warn_time`	Default = `"300s"`
`postfix:smtp_sasl_auth_enable`	Default = `no`
`postfix:smtp_skip_4xx_greeting`	Default = `yes`
`postfix:smtp_skip_5xx_greeting`	Default = `yes`
`postfix:stale_lock_time`	Default = `"500s"`
`postfix:strict_8bitmime_body`	Default = `no`
`postfix:disable_mime_input_processing`	Default = `no`
`postfix:smtpd_hard_error_limit`	Default = 20
`postfix:empty_address_recipient`	Default = `"MAILER-DAEMON"`
`postfix:forward_expansion_filter`	Default = `"1234567890!@%-_=+:,./abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"`
`postfix:smtpd_expansion_filter`	Default = `"\t\40!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\\]^_`abcdefghijklmnopqrstuvwxyz{|}~"`
`postfix:relayhost`	Default = `""`
`postfix:defer_code`	Default = 450
`postfix:lmtp_rset_timeout`	Default = `"300s"`
`postfix:always_bcc`	Default = `""`
`postfix:proxy_interfaces`	Default = `""`
`postfix:maps_rbl_reject_code`	Default = 554

Parameter (mail:)	Description
`postfix:line_length_limit`	Default = `2048`
`postfix:mailbox_transport`	Default = 0
`postfix:deliver_lock_delay`	Default = `"1s"`
`postfix:best_mx_transport`	Default = 0
`postfix:notify_classes`	Default = `"resource,software"`
`postfix:mailbox_command`	Default = `""`
`postfix:mydomain`	Default = <domain>
`postfix:mailbox_size_limit`	Default = `51200000`
`postfix:default_verp_delimiters`	Default = `"+="`
`postfix:resolve_dequoted_address`	Default = yes
`postfix:cleanup_service_name`	Default = `"cleanup"`
`postfix:header_address_token_limit`	Default = `10240`
`postfix:lmtp_connect_timeout`	Default = `"0s"`
`postfix:strict_7bit_headers`	Default = no
`postfix:unknown_hostname_reject_code`	Default = 450
`postfix:virtual_alias_domains`	Default = `"$virtual_alias_maps"`
`postfix:lmtp_sasl_auth_enable`	Default = no
`postfix:queue_directory`	Default = `"/private/var/spool/postfix"`
`postfix:sample_directory`	Default = `"/usr/share/doc/postfix/examples"`
`postfix:fallback_relay`	Default = 0
`postfix:smtpd_use_pw_server`	Default = `"yes"`
`postfix:smtpd_sasl_auth_enable`	Default = no
`postfix:mail_owner`	Default = `"postfix"`
`postfix:command_time_limit`	Default = `"1000s"`
`postfix:verp_delimiter_filter`	Default = `"-=+"`
`postfix:qmqpd_authorized_clients`	Default = 0
`postfix:virtual_mailbox_base`	Default = `""`
`postfix:permit_mx_backup_networks`	Default = `""`
`postfix:queue_run_delay`	Default = `"1000s"`
`postfix:virtual_mailbox_domains`	Default = `"$virtual_mailbox_maps"`
`postfix:local_destination_concurrency_limit`	Default = 2
`postfix:daemon_timeout`	Default = `"18000s"`

Parameter (mail:)	Description
`postfix:local_transport`	Default = `"local:$myhostname"`
`postfix:smtpd_helo_restrictions`	Default = no
`postfix:fork_delay`	Default = `"1s"`
`postfix:disable_mime_output_conversion`	Default = no
`postfix:mynetworks:_array_index:0`	Default = `"127.0.0.1/32"`
`postfix:smtp_never_send_ehlo`	Default = no
`postfix:lmtp_cache_connection`	Default = yes
`postfix:local_recipient_maps`	Default = `"proxy:unix:passwd.byname $alias_maps"`
`postfix:smtpd_timeout`	Default = `"300s"`
`postfix:require_home_directory`	Default = no
`postfix:smtpd_error_sleep_time`	Default = `"1s"`
`postfix:helpful_warnings`	Default = yes
`postfix:mail_spool_directory`	Default = `"/var/mail"`
`postfix:mailbox_delivery_lock`	Default = `"flock"`
`postfix:disable_dns_lookups`	Default = no
`postfix:mailbox_command_maps`	Default = `""`
`postfix:default_destination_concurrency_limit`	Default = 20
`postfix:2bounce_notice_recipient`	Default = `"postmaster"`
`postfix:virtual_alias_maps`	Default = `"$virtual_maps"`
`postfix:mailq_path`	Default = `"/usr/bin/mailq"`
`postfix:recipient_delimiter`	Default = no
`postfix:masquerade_exceptions`	Default = `""`
`postfix:delay_notice_recipient`	Default = `"postmaster"`
`postfix:smtp_helo_name`	Default = `"$myhostname"`
`postfix:flush_service_name`	Default = `"flush"`
`postfix:service_throttle_time`	Default = `"60s"`
`postfix:import_environment`	Default = `"MAIL_CONFIG MAIL_DEBUG MAIL_LOGTAG TZ XAUTHORITY DISPLAY"`
`postfix:sun_mailtool_compatibility`	Default = no
`postfix:authorized_verp_clients`	Default = `"$mynetworks"`
`postfix:debug_peer_list`	Default = `""`
`postfix:mime_boundary_length_limit`	Default = 2048
`postfix:initial_destination_concurrency`	Default = 5

Parameter (mail:)	Description
postfix:parent_domain_matches_subdomains	Default = "debug_peer_list,fast_flush_domains,mynetworks,permit_mx_backup_networks,qmqpd_authorized_clients,relay_domains,smtpd_access_maps"
postfix:setgid_group	Default = "postdrop"
postfix:mime_header_checks	Default = "$header_checks"
postfix:smtpd_etrn_restrictions	Default = ""
postfix:relay_transport	Default = "relay"
postfix:inet_interfaces	Default = "localhost"
postfix:smtpd_sender_restrictions	Default = ""
postfix:delay_warning_time	Default = "0h"
postfix:alias_maps	Default = "hash:/etc/aliases"
postfix:sender_canonical_maps	Default = ""
postfix:trigger_timeout	Default = "10s"
postfix:newaliases_path	Default = "/usr/bin/newaliases"
postfix:default_rbl_reply	Default = "$rbl_code Service unavailable; $rbl_class [$rbl_what] blocked using rbl_domain{rbl_reason?; $rbl_reason}"
postfix:alias_database	Default = "hash:/etc/aliases"
postfix:qmgr_message_recipient_limit	Default = 20000
postfix:extract_recipient_limit	Default = 10240
postfix:header_checks	Default = 0
postfix:syslog_facility	Default = "mail"
postfix:luser_relay	Default = ""
postfix:maps_rbl_domains:_array_index:0	Default = ""
postfix:deliver_lock_attempts	Default = 20
postfix:smtpd_data_restrictions	Default = ""
postfix:smtpd_pw_server_security_options:_array_index:0	Default = "none"
postfix:ipc_idle	Default = "100s"
postfix:mail_version	Default = "2.0.7"
postfix:transport_retry_time	Default = "60s"

Parameter (mail:)	Description
`postfix:virtual_mailbox_limit`	Default = `51200000`
`postfix:smtpd_noop_commands`	Default = `0`
`postfix:mail_release_date`	Default = `"20030319"`
`postfix:append_at_myorigin`	Default = `yes`
`postfix:body_checks_size_limit`	Default = `51200`
`postfix:qmgr_message_active_limit`	Default = `20000`
`postfix:mail_name`	Default = `"Postfix"`
`postfix:masquerade_classes`	Default = `"envelope_sender, header_sender, header_recipient"`
`postfix:allow_min_user`	Default = `no`
`postfix:smtp_randomize_addresses`	Default = `yes`
`postfix:alternate_config_directories`	Default = `no`
`postfix:allow_percent_hack`	Default = `yes`
`postfix:process_id_directory`	Default = `"pid"`
`postfix:strict_rfc821_envelopes`	Default = `no`
`postfix:fallback_transport`	Default = `0`
`postfix:owner_request_special`	Default = `yes`
`postfix:default_transport`	Default = `"smtp"`
`postfix:biff`	Default = `yes`
`postfix:relay_domains_reject_code`	Default = `554`
`postfix:smtpd_delay_reject`	Default = `yes`
`postfix:lmtp_quit_timeout`	Default = `"300s"`
`postfix:lmtp_mail_timeout`	Default = `"300s"`
`postfix:fast_flush_purge_time`	Default = `"7d"`
`postfix:disable_verp_bounces`	Default = `no`
`postfix:lmtp_skip_quit_response`	Default = `no`
`postfix:daemon_directory`	Default = `"/usr/libexec/postfix"`
`postfix:default_destination_recipient_limit`	Default = `50`
`postfix:smtp_skip_quit_response`	Default = `yes`
`postfix:smtpd_recipient_limit`	Default = `1000`
`postfix:virtual_gid_maps`	Default = `""`
`postfix:duplicate_filter_limit`	Default = `1000`
`postfix:rbl_reply_maps`	Default = `""`
`postfix:relay_recipient_maps`	Default = `0`
`postfix:syslog_name`	Default = `"postfix"`

Parameter (mail:)	Description
`postfix:queue_service_name`	Default = `"qmgr"`
`postfix:transport_maps`	Default = `""`
`postfix:smtp_destination_concurrency_limit`	Default = `"$default_destination_concurrency_limit"`
`postfix:virtual_mailbox_lock`	Default = `"fcntl"`
`postfix:qmgr_fudge_factor`	Default = `100`
`postfix:ipc_timeout`	Default = `"3600s"`
`postfix:default_delivery_slot_discount`	Default = `50`
`postfix:relocated_maps`	Default = `""`
`postfix:max_use`	Default = `100`
`postfix:default_delivery_slot_cost`	Default = `5`
`postfix:default_privs`	Default = `"nobody"`
`postfix:smtp_bind_address`	Default = `no`
`postfix:nested_header_checks`	Default = `"$header_checks"`
`postfix:canonical_maps`	Default = `no`
`postfix:debug_peer_level`	Default = `2`
`postfix:in_flow_delay`	Default = `"1s"`
`postfix:smtpd_junk_command_limit`	Default = `100`
`postfix:program_directory`	Default = `"/usr/libexec/postfix"`
`postfix:smtp_quit_timeout`	Default = `"300s"`
`postfix:smtp_mail_timeout`	Default = `"300s"`
`postfix:minimal_backoff_time`	Default = `"1000s"`
`postfix:queue_file_attribute_count_limit`	Default = `100`
`postfix:body_checks`	Default = `no`
`postfix:smtpd_client_restrictions:_array_index:0`	Default = `""`
`postfix:mydestination:_array_index:0`	Default = `"$myhostname"`
`postfix:mydestination:_array_index:1`	Default = `"localhost.$mydomain"`
`postfix:error_service_name`	Default = `"error"`
`postfix:smtpd_sasl_security_options:_array_index:0`	Default = `"noanonymous"`
`postfix:smtpd_null_access_lookup_key`	Default = `"<>"`
`postfix:virtual_uid_maps`	Default = `""`
`postfix:smtpd_history_flush_threshold`	Default = `100`
`postfix:smtp_pix_workaround_threshold_time`	Default = `"500s"`

Parameter (mail:)	Description
`postfix:showq_service_name`	Default = `"showq"`
`postfix:smtp_pix_workaround_delay_time`	Default = `"10s"`
`postfix:lmtp_sasl_security_options`	Default = `"noplaintext, noanonymous"`
`postfix:bounce_size_limit`	Default = `50000`
`postfix:qmqpd_timeout`	Default = `"300s"`
`postfix:allow_mail_to_files`	Default = `"alias,forward"`
`postfix:relay_domains`	Default = `"$mydestination"`
`postfix:smtpd_banner`	Default = `"$myhostname ESMTP $mail_name"`
`postfix:smtpd_helo_required`	Default = `no`
`postfix:berkeley_db_read_buffer_size`	Default = `131072`
`postfix:swap_bangpath`	Default = `yes`
`postfix:maximal_queue_lifetime`	Default = `"5d"`
`postfix:ignore_mx_lookup_error`	Default = `no`
`postfix:mynetworks_style`	Default = `"host"`
`postfix:myhostname`	Default = `"<hostname>"`
`postfix:default_minimum_delivery_slots`	Default = `3`
`postfix:recipient_canonical_maps`	Default = `no`
`postfix:hash_queue_depth`	Default = `1`
`postfix:hash_queue_names:_array_index:0`	Default = `"incoming"`
`postfix:hash_queue_names:_array_index:1`	Default = `"active"`
`postfix:hash_queue_names:_array_index:2`	Default = `"deferred"`
`postfix:hash_queue_names:_array_index:3`	Default = `"bounce"`
`postfix:hash_queue_names:_array_index:4`	Default = `"defer"`
`postfix:hash_queue_names:_array_index:5`	Default = `"flush"`
`postfix:hash_queue_names:_array_index:6`	Default = `"hold"`
`postfix:lmtp_tcp_port`	Default = `24`
`postfix:local_command_shell`	Default = `0`
`postfix:allow_mail_to_commands`	Default = `"alias,forward"`
`postfix:non_fqdn_reject_code`	Default = `504`
`postfix:maximal_backoff_time`	Default = `"4000s"`
`postfix:smtp_always_send_ehlo`	Default = `yes`

Parameter (mail:)	Description
postfix:proxy_read_maps	Default = "$local_recipient_maps $mydestination $virtual_alias_maps $virtual_alias_domains $virtual_mailbox_maps $virtual_mailbox_domains $relay_recipient_maps $relay_domains $canonical_maps $sender_canonical_maps $recipient_canonical_maps $relocated_maps $transport_maps $mynetworks"
postfix:propagate_unmatched_extensions	Default = "canonical, virtual"
postfix:smtp_destination_recipient_limit	Default = "$default_destination_ recipient_limit"
postfix:smtpd_restriction_classes	Default = ""
postfix:mime_nesting_limit	Default = 100
postfix:virtual_mailbox_maps	Default = ""
postfix:bounce_service_name	Default = "bounce"
postfix:header_size_limit	Default = 102400
postfix:strict_8bitmime	Default = no
postfix:virtual_transport	Default = "virtual"
postfix:berkeley_db_create_buffer_size	Default = 16777216
postfix:broken_sasl_auth_clients	Default = no
postfix:home_mailbox	Default = no
postfix:content_filter	Default = ""
postfix:forward_path	Default = "$home/.forward${recipien t_delimiter}${extension}, $home/.forward"
postfix:qmqpd_error_delay	Default = "1s"
postfix:manpage_directory	Default = "/usr/share/man"
postfix:hopcount_limit	Default = 50
postfix:unknown_virtual_alias_reject_code	Default = 550
postfix:smtpd_sender_login_maps	Default = ""
postfix:rewrite_service_name	Default = "rewrite"
postfix:unknown_address_reject_code	Default = 450

Chapter 11   Working With Mail Service

Parameter (mail:)	Description
postfix:append_dot_mydomain	Default = yes
postfix:command_expansion_filter	Default = "1234567890!@%-_=+:,./abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ"
postfix:default_extra_recipient_limit	Default = 1000
postfix:lmtp_data_done_timeout	Default = "600s"
postfix:myorigin	Default = "$myhostname"
postfix:lmtp_data_init_timeout	Default = "120s"
postfix:lmtp_data_xfer_timeout	Default = "180s"
postfix:smtp_data_done_timeout	Default = "600s"
postfix:smtp_data_init_timeout	Default = "120s"
postfix:smtp_data_xfer_timeout	Default = "180s"
postfix:default_delivery_slot_loan	Default = 3
postfix:reject_code	Default = 554
postfix:command_directory	Default = "/usr/sbin"
postfix:lmtp_rcpt_timeout	Default = "300s"
postfix:smtp_sasl_security_options	Default = "noplaintext, noanonymous"
postfix:access_map_reject_code	Default = 554
postfix:smtp_helo_timeout	Default = "300s"
postfix:bounce_notice_recipient	Default = "postmaster"
postfix:smtp_connect_timeout	Default = "30s"
postfix:fault_injection_code	Default = 0
postfix:unknown_client_reject_code	Default = 450
postfix:virtual_minimum_uid	Default = 100
postfix:fast_flush_domains	Default = "$relay_domains"
postfix:default_database_type	Default = "hash"
postfix:dont_remove	Default = 0
postfix:expand_owner_alias	Default = no
postfix:max_idle	Default = "100s"
postfix:defer_transports	Default = ""
postfix:qmgr_message_recipient_minimum	Default = 10
postfix:invalid_hostname_reject_code	Default = 501
postfix:fork_attempts	Default = 5
postfix:allow_untrusted_routing	Default = no
imap:tls_cipher_list:_array_index:0	Default = "DEFAULT"

Parameter (mail:)	Description
`imap:umask`	Default = `"077"`
`imap:tls_ca_path`	Default = `""`
`imap:pop_auth_gssapi`	Default = `yes`
`imap:sasl_minimum_layer`	Default = `0`
`imap:tls_cert_file`	Default = `""`
`imap:poptimeout`	Default = `10`
`imap:tls_sieve_require_cert`	Default = `no`
`imap:mupdate_server`	Default = `""`
`imap:timeout`	Default = `30`
`imap:quotawarn`	Default = `90`
`imap:enable_pop`	Default = `no`
`imap:mupdate_retry_delay`	Default = `20`
`imap:tls_session_timeout`	Default = `1440`
`imap:postmaster`	Default = `"postmaster"`
`imap:defaultacl`	Default = `"anyone lrs"`
`imap:tls_lmtp_key_file`	Default = `""`
`imap:newsprefix`	Default = `""`
`imap:userprefix`	Default = `"Other Users"`
`imap:deleteright`	Default = `"c"`
`imap:allowplaintext`	Default = `yes`
`imap:pop_auth_clear`	Default = `no`
`imap:imapidresponse`	Default = `yes`
`imap:sasl_auto_transition`	Default = `no`
`imap:mupdate_port`	Default = `""`
`imap:admins:_array_index:0`	Default = `"cyrus"`
`imap:plaintextloginpause`	Default = `0`
`imap:popexpiretime`	Default = `0`
`imap:pop_auth_any`	Default = `no`
`imap:sieve_maxscriptsize`	Default = `32`
`imap:hashimapspool`	Default = `no`
`imap:tls_lmtp_cert_file`	Default = `""`
`imap:tls_sieve_key_file`	Default = `""`
`imap:sievedir`	Default = `"/usr/sieve"`
`imap:debug_command`	Default = `""`
`imap:popminpoll`	Default = `0`
`imap:tls_lmtp_require_cert`	Default = `no`

Parameter (mail:)	Description
`imap:tls_ca_file`	Default = `""`
`imap:sasl_pwcheck_method`	Default = `"auxprop"`
`imap:postuser`	Default = `""`
`imap:sieve_maxscripts`	Default = `5`
`imap:defaultpartition`	Default = `"default"`
`imap:altnamespace`	Default = `yes`
`imap:max_imap_connections`	Default = `100`
`imap:tls_imap_cert_file`	Default = `""`
`imap:sieveusehomedir`	Default = `no`
`imap:reject8bit`	Default = `no`
`imap:tls_sieve_cert_file`	Default = `""`
`imap:imapidlepoll`	Default = `60`
`imap:srvtab`	Default = `"/etc/srvtab"`
`imap:imap_auth_login`	Default = `no`
`imap:tls_pop3_cert_file`	Default = `""`
`imap:tls_pop3_require_cert`	Default = `no`
`imap:lmtp_overquota_perm_failure`	Default = `no`
`imap:tls_imap_key_file`	Default = `""`
`imap:enable_imap`	Default = `no`
`imap:tls_require_cert`	Default = `no`
`imap:autocreatequota`	Default = `0`
`imap:allowanonymouslogin`	Default = `no`
`imap:pop_auth_apop`	Default = `yes`
`imap:partition-default`	Default = `"/var/spool/imap"`
`imap:imap_auth_cram_md5`	Default = `no`
`imap:mupdate_password`	Default = `""`
`imap:idlesocket`	Default = `"/var/imap/socket/idle"`
`imap:allowallsubscribe`	Default = `no`
`imap:singleinstancestore`	Default = `yes`
`imap:unixhierarchysep`	Default = `"yes"`
`imap:mupdate_realm`	Default = `""`
`imap:sharedprefix`	Default = `"Shared Folders"`
`imap:tls_key_file`	Default = `""`
`imap:lmtpsocket`	Default = `"/var/imap/socket/lmtp"`

Parameter (mail:)	Description
`imap:configdirectory`	Default = `"/var/imap"`
`imap:sasl_maximum_layer`	Default = `256`
`imap:sendmail`	Default = `"/usr/sbin/sendmail"`
`imap:loginuseacl`	Default = `no`
`imap:mupdate_username`	Default = `""`
`imap:imap_auth_plain`	Default = `no`
`imap:imap_auth_any`	Default = `no`
`imap:duplicatesuppression`	Default = `yes`
`imap:notifysocket`	Default = `"/var/imap/socket/notify"`
`imap:tls_imap_require_cert`	Default = `no`
`imap:imap_auth_clear`	Default = `yes`
`imap:tls_pop3_key_file`	Default = `""`
`imap:proxyd_allow_status_referral`	Default = `no`
`imap:servername`	Default = `"<hostname>"`
`imap:logtimestamps`	Default = `no`
`imap:imap_auth_gssapi`	Default = `no`
`imap:mupdate_authname`	Default = `""`
`mailman:enable_mailman`	Default = `no`

Mail `serveradmin` Commands

You can use the following commands with the `serveradmin` application to manage Mail service.

Command (`mail:command=`)	Description
`getHistory`	View a periodic record of file data throughput or number of user connections. See "Listing Mail Service Statistics" on page 117.
`getLogPaths`	Display the locations of the Mail service logs. See "Viewing the Mail Service Logs" on page 118.
`writeSettings`	Equivalent to the standard `serveradmin settings` command, but also returns a setting indicating whether the service needs to be restarted. See "Determining Whether a Service Needs to be Restarted" on page 19.

Listing Mail Service Statistics

You can use the `serveradmin getHistory` command to display a log of periodic samples of the number of user connections and the data throughput. Samples are taken once each minute.

To list samples:

```
$ sudo serveradmin command
mail:command = getHistory
mail:variant = statistic
mail:timeScale = scale
Control-D
```

Parameter	Description
`statistic`	The value you want to display.
	Valid values:
	`v1` - number of connected users (average during sampling period)
	`v2` - data throughput (bytes/sec)
`scale`	The length of time in seconds, ending with the current time, for which you want to see samples. For example, to see 24 hours of data, you would specify `mail:timeScale = 86400`.

Output

```
mail:nbSamples = <samples>
mail:v2Legend = "throughput"
mail:samplesArray:_array_index:0:vn = <sample>
mail:samplesArray:_array_index:0:t = <time>
mail:samplesArray:_array_index:1:vn = <sample>
mail:samplesArray:_array_index:1:t = <time>
[...]
mail:samplesArray:_array_index:i:vn = <sample>
mail:samplesArray:_array_index:i:t = <time>
mail:v1Legend = "connections"
afp:currentServerTime = <servertime>
```

Value displayed by getHistory	Description
`<samples>`	The total number of samples listed.
`<sample>`	The numerical value of the sample.
	For connections (v1), this is integer average number of users.
	For throughput, (v2), this is integer bytes per second.
`<time>`	The time at which the sample was measured. A standard UNIX time (number of seconds since Sep 1, 1970.) Samples are taken every 60 seconds.

Viewing the Mail Service Logs

You can use `tail` or any other file listing tool to view the contents of the Mail service logs.

To view the latest entries in a log:

```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the Mail service logs are located.

To display the log locations:

```
$ sudo serveradmin command mail:command = getLogPaths
```

Output

```
mail:Server Log = <server-log>
mail:Lists qrunner = <lists-log>
mail:Lists post = <postings-log>
mail:Lists smtp = <delivery-log>
mail:Lists subscribe = <subscriptions-log>
mail:SMTP Log = <smtp-log>
mail:POP Log = <pop-log>
mail:Lists error = <listerrors-log>
mail:IMAP Log = <imap-log>
mail:Lists smtp-failure = <failures-log>
```

Value	Description
`<server-log>`	The location of the server log. Default = `srvr.log`
`<lists-log>`	The location of the Mailing Lists log. Default = `/private/var/mailman/logs/qrunner`
`<postings-log>`	The location of the Mailing Lists Postings log. Default = `/private/var/mailman/logs/post`
`<delivery-log>`	The location of the Mailing Lists Delivery log. Default = `/private/var/mailman/logs/smtp`
`<subscriptions-log>`	The location of the Mailing Lists Subscriptions log. Default = `/private/var/mailman/logs/subscribe`
`<smtp-log>`	The location of the server log. Default = `smtp.log`
`<pop-log>`	The location of the server log. Default = `pop3.log`
`<listerrors-log>`	The location of the Mailing Lists Error log. Default = `/private/var/mailman/logs/error`
`<imap-log>`	The location of the server log. Default = `imap.log`
`<failures-log>`	The location of the Mailing Lists Delivery Failures log. Default = `/private/var/mailman/logs/smtp-failure`

Setting Up SSL for Mail Service

Mail service requires some configuration to provide Secure Sockets Layer (SSL) connections automatically. The basic steps are as follows:

- Generate a Certificate Signing Request (CSR) and create a keychain.
- Obtain an SSL certificate from an issuing authority.
- Import the SSL certificate into the keychain.
- Create a passphrase file.

Generating a CSR and Creating a Keychain

To begin configuring Mail service for SSL connections, you generate a CSR and create a keychain by using the command-line tool `certtool`. A CSR is a file that provides information needed to issue an SSL certificate.

1   Log in to the server as root.

2   In the Terminal application, type the following two commands:

```
$ cd /private/var/root/Library/Keychains/
$ /usr/bin/certtool r csr.txt k=certkc c
```

This use of the `certtool` command begins an interactive process that generates a Certificate Signing Request (CSR) in the file csr.txt and creates a keychain named certkc.

3   In the New Keychain Passphrase dialog that appears, enter a passphrase or password for the keychain you're creating, enter the password or passphrase a second time to verify it, and click OK.

Remember this passphrase, because later you must supply it again.

4   When "Enter key and certificate label:" appears in the Terminal window, type a one-word key, a blank space, and a one-word certificate label, then press Return.

For example, you could type your organization's name as the key and `mailservice` as the certificate label.

5   Type `r` when prompted to select a key algorithm, then press Return.

```
Please specify parameters for the key pair you will generate.
   r   RSA
   d   DSA
   f   FEE
Select key algorithm by letter:
```

6   Type a key size at the next prompt, then press Return.

```
Valid key sizes for RSA are 512..2048; default is 512
Enter key size in bits or CR for default:
```

Larger key sizes are more secure, but require more processing time on your server. Key sizes smaller than 1024 aren't accepted by some certificate-issuing authorities.

7 Type `y` when prompted to confirm the algorithm and key size, then press Return.

```
You have selected algorithm RSA, key size (size entered above) bits.
OK (y/anything)?
```

8 Type `b` when prompted to specify how this certificate will be used, then press Return.

```
Enter cert/key usage (s=signing, b=signing AND encrypting):
```

9 Type `s` when prompted to select a signature algorithm, then press Return.

```
...Generating key pair...
Please specify the algorithm with which your certificate will be signed.

   5   RSA with MD5
   s   RSA with SHA1

Select signature algorithm by letter:
```

10 Type `y` when asked to confirm the selected algorithm, then press Return.

```
You have selected algorithm RSA with SHA1.
OK (y/anything)?
```

11 Enter a phrase or some random text when prompted to enter a challenge string, then press Return.

```
...creating CSR...
Enter challenge string:
```

12 Enter the correct information at the next five prompts, which request the various components of the certificate's Relative Distinguished Name (RDN), pressing return after each entry.

```
For Common Name, enter the server's DNS name, such as server.example.com.
For Country, enter the country in which your organization is located.
For Organization, enter the organization to which your domain name is
     registered.
For Organizational Unit, enter something similar to a department name.
For State/Province, enter the full name of your state or province.
```

13 Type `y` when asked to confirm the information you entered, then press Return.

```
Is this OK (y/anything)?
```

When you see a message about writing to csr.txt, you have successfully generated a CSR and created the keychain that Mail service needs for SSL connections.

```
Wrote (n) bytes of CSR to csr.txt
```

Obtaining an SSL Certificate

After generating a CSR and a keychain, you continue configuring Mail service for automatic SSL connections by purchasing an SSL certificate from a certificate authority such as Verisign or Thawte. You can do this by completing a form on the certificate authority's website. When prompted for your CSR, open the csr.txt file using a text editor such as TextEdit. Then copy and paste the contents of the file into the appropriate field on the certificate authority's website. The websites for these certificate authorities are at

- www.verisign.com
- www.thawte.com

When you receive your certificate, save it in a text file named sslcert.txt. You can save this file with the TextEdit application. Make sure the file is plain text, not rich text, and contains only the certificate text.

Importing an SSL Certificate Into the Keychain

To import an SSL certificate into a keychain, use the command-line tool `certtool`. This continues the configuration of Mail service for automatic SSL connections.

1 Log in to the server as root.

2 Open the Terminal application.

3 Go to the directory where the saved certificate file is located.

For example, type `cd /private/var/root/Desktop` and press Return if the certificate file is saved on the desktop of the root user.

4 Type the following command and press Return:

```
certtool i sslcert.txt k=certkc
```

Using `certtool` this way imports a certificate from the file named sslcert.txt into the keychain named certkc.

A message on screen confirms that the certificate was successfully imported.

```
...certificate successfully imported.
```

Creating a Passphrase File

To create a passphrase file, you will use TextEdit, then change the privileges of the file using the Terminal application. This file contains the passphrase you specified when you created the keychain. Mail service will automatically use the passphrase file to unlock the keychain that contains the SSL certificate. This concludes configuring Mail service for automatic SSL connections.

1   Log in to the server as root (if you're not already logged in as root).

2   In TextEdit, create a new file and type the passphrase exactly as you entered it when you created the keychain.

    Don't press Return after typing the passphrase.

3   Make the file plain text by choosing Make Plain Text from the Format menu.

4   Save the file, naming it cerkc.pass.

5   Move the file to the root keychain folder.

    The path is /private/var/root/Library/Keychains/.

    To see the root keychain folder in the Finder, choose Go to Folder from the Go menu, then type `/private/var/root/Library/Keychains/` and click Go.

6   In the Terminal application, change the access privileges to the passphrase file so only root can read and write to this file.

    Do this by typing the following two commands, pressing Return after each one:

    ```
    cd /private/var/root/Library/Keychains/
    chmod 600 certkc.pass
    ```

    Mail service of Mac OS X Server can now use SSL for secure IMAP connections.

7   Log out as root.

    Note: If Mail service is running, you need to stop it and start it again to make it recognize the new certificate keychain.

Setting Up SSL for Mail Service on a Headless Server

If you want to set up SSL for Mail service on a server that doesn't have a display, first follow the instructions in the sections:
• "Generating a CSR and Creating a Keychain" on page 119
• "Obtaining an SSL Certificate" on page 121
• "Importing an SSL Certificate Into the Keychain" on page 121
• "Creating a Passphrase File" on this page

Then copy the keychain file "certkc" and the keychain passphrase file "certkc.pass" to the root keychain folder on the headless server. The path on the headless server is /private/var/root/Library/Keychains/.

Working With Web Technologies

12

Commands you can use to manage Web service in Mac OS X Server.

Starting and Stopping Web Service

To start Web service:

```
$ sudo serveradmin start web
```

To stop Web service:

```
$ sudo serveradmin stop web
```

Checking Web Service Status

To see if Web service is running:

```
$ sudo serveradmin status web
```

To see complete Web service status:

```
$ sudo serveradmin fullstatus web
```

Viewing Web Settings

You can use `serveradmin` to view your server's Web service configuration. However, if you want to work with the Web service from the command-line, you'll probably find it more straightforward to work directly with the underlying Apache web server.

For information on Apache settings, visit www.apache.org.

To list all Web service settings:

```
$ sudo serveradmin settings web
```

To list a particular setting:

```
$ sudo serveradmin settings web:setting
```

To list a group of settings:

You can list a group of settings that have part of their names in common by typing only as much of the name as you want, stopping at a colon (:), and typing an asterisk (*) as a wildcard for the remaining parts of the name. For example,

```
$ sudo serveradmin settings web:IFModule:_array_id:mod_alias.c:*
```

Changing Web Settings

You can use `serveradmin` to modify your server's Web service configuration. However, if you want to work with the Web service from the command-line, you'll probably find it more straightforward to work directly with the underlying Apache web server.

For information on Apache, visit www.apache.org.

`serveradmin` and Apache Settings

The parameters are written differently in the Apache configuration file than they are in `serveradmin`. For example, this block of Apache configuration parameters

```
<IfModule mod_macbinary_apple.c>
  MacBinary On
  MacBinaryBlock html shtml perl pl cgi jsp php phps asp scpt
  MacBinaryBlock htaccess
</IfModule>
```

appear as follows in `serveradmin`

```
web:IfModule:_array_id:mod_macbinary_apple.c:MacBinary = yes
web:IfModule:_array_id:mod_macbinary_apple.c:MacBinaryBlock:_array_index:0 =
     "html shtml perl pl cgi jsp php phps asp scpt"
web:IfModule:_array_id:mod_macbinary_apple.c:MacBinaryBlock:_array_index:1 =
     "htaccess".
```

For information on Apache settings, visit www.apache.org.

Changing Settings Using `serveradmin`

You can change Web service settings using the `serveradmin` command.

To change a setting:

```
$ sudo serveradmin settings web:setting = value
```

Parameter	Description
setting	A Web service setting. To see a list of available settings, type
	`$ sudo serveradmin settings web`
value	An appropriate value for the setting.

To change several settings:

```
$ sudo serveradmin settings
web:setting = value
web:setting = value
web:setting = value
[...]
Control-D
```

Web `serveradmin` Commands

You can use the following commands with the `serveradmin` application to manage Web service.

Command (`web:command=`)	Description
`getHistory`	View Web service statistics. See "Viewing Service Statistics" on page 126.
`getLogPaths`	Finding the access and error logs for each hosted site. See "Viewing Service Logs" on this page.
`getSites`	Listing existing sites. See "Listing Hosted Sites" on this page.

Listing Hosted Sites

You can use the `serveradmin getSites` command to display a list of the sites hosted by the server along with basic settings and status.

To list sites:

```
$ sudo serveradmin command web:command = getSites
```

Viewing Service Logs

You can use `tail` or any other file listing tool to view the contents of Web service access and error logs for each site hosted by the server.

To view the latest entries in a log:

```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the current error and activity logs for each site are located.

To display the log paths:

```
$ sudo serveradmin command web:command = getLogPaths
```

Viewing Service Statistics

You can use the `serveradmin getHistory` command to display a log of periodic samples of the number of requests, cache performance, and data throughput. Samples are taken once each minute.

To list samples:

```
$ sudo serveradmin command
qtss:command = getHistory
qtss:variant = statistic
qtss:timeScale = scale
Control-D
```

Parameter	Description
statistic	The value you want to display. Valid values:
	v1 - number of requests per second
	v2 - throughput (bytes/sec)
	v3 - cache requests per second
	v4 - cache throughput (bytes/sec)
scale	The length of time in seconds, ending with the current time, for which you want to see samples. For example, to see 30 minutes of data, you would specify `qtss:timeScale = 1800`.

Output

```
web:nbSamples = <samples>
web:samplesArray:_array_index:0:vn = <sample>
web:samplesArray:_array_index:0:t = <time>
web:samplesArray:_array_index:1:vn = <sample>
web:samplesArray:_array_index:1:t = <time>
[...]
web:samplesArray:_array_index:i:vn = <sample>
web:samplesArray:_array_index:i:t = <time>
web:vnLegend = "<legend>"
web:currentServerTime = <servertime>
```

Value displayed by getHistory	Description
<samples>	The total number of samples listed.
<legend>	A textual description of the selected statistic.
	"REQUESTS_PER_SECOND" for v1
	"THROUGHPUT" for v2
	"CACHE_REQUESTS_PER_SECOND" for v3
	"CACHE_THROUGHPUT" for v4
<sample>	The numerical value of the sample.
<time>	The time at which the sample was measured. A standard UNIX time (number of seconds since Sep 1, 1970.) Samples are taken every 60 seconds.

Example Script for Adding a Website

The following script shows how you can use `serveradmin` to add a website to the server's Web service configuration. The script uses two files:

- **addsite** The actual script you run. It accepts values for the site's IP address, port number, server name, and root directory and uses `sed` to substitute these values in the settings it reads from the second file (addsite.in) feeds to serveradmin.
- **addsite.in** Contains the actual settings (with placeholders for values you provide when you run addsite) used to create the website.

The addsite File

```
sed -es#_ipaddr#$1#g -es#_port#$2#g -es#_servername#$3#g
    -es#_docroot#$4#g ./addsite.in | /usr/sbin/serveradmin --set -i
```

The addsite.in File

```
web:Sites:_array_id:_ipaddr\:_port__servername = create
web:Sites:_array_id:_ipaddr\:_port__servername:Listen:_array_index:0 =
    "_ipaddr:_port"
web:Sites:_array_id:_ipaddr\:_port__servername:ServerName = _servername
web:Sites:_array_id:_ipaddr\:_port__servername:ServerAdmin =
    admin@_servername
web:Sites:_array_id:_ipaddr\:_port__servername:DirectoryIndex:_array_index:0
    = "index.html"
web:Sites:_array_id:_ipaddr\:_port__servername:DirectoryIndex:_array_index:1
    = "index.php"
web:Sites:_array_id:_ipaddr\:_port__servername:WebMail = yes
web:Sites:_array_id:_ipaddr\:_port__servername:CustomLog:_array_index:0:
    Format = "%{User-agent}i"
web:Sites:_array_id:_ipaddr\:_port__servername:CustomLog:_array_index:0:
    enabled = yes
web:Sites:_array_id:_ipaddr\:_port__servername:CustomLog:_array_index:0:
    ArchiveInterval = 0
web:Sites:_array_id:_ipaddr\:_port__servername:CustomLog:_array_index:0:
    Path = "/private/var/log/httpd/access_log"
web:Sites:_array_id:_ipaddr\:_port__servername:CustomLog:_array_index:0:
    Archive = yes
web:Sites:_array_id:_ipaddr\:_port__servername:Directory:_array_id:
    /Library/WebServer/Documents:Options:Indexes = yes
web:Sites:_array_id:_ipaddr\:_port__servername:Directory:_array_id:
    /Library/WebServer/Documents:Options:ExecCGI = no
web:Sites:_array_id:_ipaddr\:_port__servername:Directory:_array_id:
    /Library/WebServer/Documents:AuthName = "Test Site"
web:Sites:_array_id:_ipaddr\:_port__servername:ErrorLog:ArchiveInterval = 0
web:Sites:_array_id:_ipaddr\:_port__servername:ErrorLog:Path =
    "/private/var/log/httpd/error_log"
web:Sites:_array_id:_ipaddr\:_port__servername:ErrorLog:Archive = no
web:Sites:_array_id:_ipaddr\:_port__servername:Include:_array_index:0 =
    "/etc/httpd/httpd_squirrelmail.conf"
web:Sites:_array_id:_ipaddr\:_port__servername:enabled = yes
```

```
web:Sites:_array_id:_ipaddr\:_port__servername:ErrorDocument:_array_index:0:
    StatusCode = 404
web:Sites:_array_id:_ipaddr\:_port__servername:ErrorDocument:_array_index:0:
    Document = "/nwesite_notfound.html"
web:Sites:_array_id:_ipaddr\:_port__servername:LogLevel = "warn"
web:Sites:_array_id:_ipaddr\:_port__servername:IfModule:_array_id:mod_ssl.c:
    SSLEngine = no
web:Sites:_array_id:_ipaddr\:_port__servername:IfModule:_array_id:mod_ssl.c:
    SSLPassPhrase = ""
web:Sites:_array_id:_ipaddr\:_port__servername:IfModule:_array_id:mod_ssl.c:
    SSLLog = "/private/var/log/httpd/ssl_engine_log"
web:Sites:_array_id:_ipaddr\:_port__servername:DocumentRoot = "_docroot"
web:Sites:_array_id:_ipaddr\:_port__servername
```

To run the script:

```
$ addsite ipaddress port name root
```

Parameter	Description
ipaddress	The IP address for the site.
port	The port number to be used to for HTTP access to the site.
name	The name of the site.
root	The root directory for the site's files and subdirectories.

If you get the message "command not found" when you try to run the script, precede the command with the full path to the script file. For example,

```
/users/admin/documents/addsite 10.0.0.2 80 corpsite
    /users/webmaster/sites/corpsite
```

Or, use `cd` to change to the directory that contains the file and precede the command with `./`. For example:

```
$ cd /users/admin/documents
$ ./addsite 10.0.0.2 80 corpsite /users/webmaster/sites/corpsite
```

Working With Network Services

13

Commands you can use to manage DHCP, DNS, Firewall, NAT, and VPN service in Mac OS X Server.

DHCP Service

Starting and Stopping DHCP Service

To start DHCP service:

```
$ sudo serveradmin start dhcp
```

To stop DHCP service:

```
$ sudo serveradmin stop dhcp
```

Checking the Status of DHCP Service

To see summary status of DHCP service:

```
$ sudo serveradmin status dhcp
```

To see detailed status of DHCP service:

```
$ sudo serveradmin fullstatus dhcp
```

Viewing DHCP Service Settings

To list DHCP service configuration settings:

```
$ sudo serveradmin settings dhcp
```

To list a particular setting:

```
$ sudo serveradmin settings dhcp:setting
```

To list a group of settings:
You can list a group of settings that have part of their names in common by typing only as much of the name as you want, stopping at a colon (:), and typing an asterisk (*) as a wildcard for the remaining parts of the name. For example,

```
$ sudo serveradmin settings dhcp:subnets:*
```

Changing DHCP Service Settings

To change a setting:

```
$ sudo serveradmin settings dhcp:setting = value
```

Parameter	Description
`setting`	A DHCP service setting. To see a list of available settings, type `$ sudo serveradmin settings dhcp` or see "DHCP Service Settings" on this page and "DHCP Subnet Settings Array" on page 131.
`value`	An appropriate value for the setting.

To change several settings:

```
$ sudo serveradmin settings
dhcp:setting = value
dhcp:setting = value
dhcp:setting = value
[...]
Control-D
```

DHCP Service Settings

Use the following parameters with the `serveradmin` command to change settings for the dhcp service.

Parameter (dhcp:)	Description
`logging_level`	`"LOW"`|`"MEDIUM"`|`"HIGH"` Default = `"MEDIUM"` Corresponds to the Log Detail Level pop-up menu in the Logging pane of DHCP service settings in the Server Admin GUI application.
`subnet_status`	Default = `0`
`subnet_defaults:logVerbosity`	`"LOW"`|`"MEDIUM"`|`"HIGH"` Default = `"MEDIUM"`
`subnet_defaults:logVerbosityList:` `_array_index:n`	Available values for the logVerbosity setting. Default = `"LOW,"` `"MEDIUM,"` and `"HIGH"`
`subnet_defaults:WINS_node_type`	Default = `"NOT_SET"`
`subnet_defaults:routers`	Default = `empty_dictionary`
`subnet_defaults:selected_port_key`	Default = `en0`
`subnet_defaults:selected_port_key` `_list:_array_index:n`	An array of available ports.
`subnet_defaults:dhcp_domain_name`	Default = The last portion of the server's host name, for example, `company.com`.

Parameter (dhcp:)	Description
`subnet_defaults:dhcp_domain_name_server:_array_index:n`	Default = The DNS server addresses provided during server setup, as listed in the Network pane of the server's System Preferences.
`subnets:_array_id:<subnetID>...`	An array of settings for a particular subnet. `<subnetID>` is a unique identifier for each subnet. See "DHCP Subnet Settings Array" on this page.

DHCP Subnet Settings Array

An array of the settings listed in the following table is included in the DHCP service settings for each subnet you define. You can add a subnet to the DHCP configuration by using `serveradmin` to add an array of these settings.

About Subnet IDs

In an actual list of settings, `<subnetID>` is replaced with a unique ID code for the subnet. The IDs generated by the server are just random numbers. The only requirement for this ID is that it be unique among the subnets defined on the server.

Subnet Parameter `subnets:_array_id:<subnetID>:`	Description
`descriptive_name`	A textual description of the subnet.
	Corresponds to the Subnet Name field in the General pane of the subnet settings in the Server Admin GUI application.
`dhcp_domain_name`	The default domain for DNS searches, for example, `company.com`.
	Corresponds to the Default Domain field in the DNS pane of the subnet settings in the Server Admin GUI application.
`dhcp_domain_name_server:_array_index:n`	The primary WINS server to be used by clients.
	Corresponds to the Name Servers field in the DNS pane of the subnet settings in the Server Admin GUI application.
`dhcp_enabled`	Whether DHCP is enabled for this subnet.
	Corresponds to the Enable checkbox in the list of subnets in the Subnets pane of the DHCP settings in the Server Admin GUI application.
`dhcp_ldap_url:_array_index:n`	The URL of the LDAP directory to be used by clients.
	Corresponds to the Lease URL field in the LDAP pane of the subnet settings in the Server Admin GUI application.
`dhcp_router`	The IPv4 address of the subnet's router.
	Corresponds to the Router field in the General pane of the subnet settings in the Server Admin GUI application.

Subnet Parameter `subnets:_array_id:<subnetID>:`	Description
`lease_time_secs`	Lease time in seconds. Default = `"3600"` Corresponds to the Lease Time pop-up menu and field in the General pane of the subnet settings in the Server Admin GUI application.
`net_address`	The IPv4 network address for the subnet.
`net_mask`	The subnet mask for the subnet. Corresponds to the Subnet Mask field in the General pane of the subnet settings in the Server Admin GUI application.
`net_range_end`	The highest available IPv4 address for the subnet. Corresponds to the Ending IP Address field in the General pane of the subnet settings in the Server Admin GUI application.
`net_range_start`	The lowest available IPv4 address for the subnet. Corresponds to the Starting IP Address field in the General pane of the subnet settings in the Server Admin GUI application.
`selected_port_name`	The network port for the subnet. Corresponds to the Network Interface pop-up menu in the General pane of the subnet settings in the Server Admin GUI application.
`WINS_NBDD_server`	The NetBIOS Datagram Distribution Server IPv4 address. Corresponds to the NBDD Server field in the WINS pane of the subnet settings in the Server Admin GUI application.

`WINS_node_type`	The WINS node type. Can be set to:
	`" "` (not set, default)
	`BROADCAST_B_NODE`
	`PEER_P_NODE`
	`MIXED_M_NODE`
	`HYBRID-H-NODE`
	Corresponds to the NBT Node Type field in the WINS pane of the subnet settings in the Server Admin GUI application.

`WINS_primary_server`	The primary WINS server to be used by clients. Corresponds to the WINS/NBNS Primary Server field in the WINS pane of the subnet settings in the Server Admin GUI application.

Subnet Parameter `subnets:_array_id:<subnetID>:`	Description
`WINS_scope_id`	A domain name such as `apple.com`. Default = `" "` Corresponds to the NetBIOS Scope ID field in the WINS pane of the subnet settings in the Server Admin GUI application.
`WINS_secondary_server`	The secondary WINS server to be used by clients. Corresponds to the WINS/NBNS Secondary Server field in the WINS pane of the subnet settings in the Server Admin GUI application.

Adding a DHCP Subnet

You may already have a subnet for each port you enabled when you installed and set up the server. You can use the `serveradmin settings` command to check for subnets that the server set up for you; see "Viewing DHCP Service Settings" on page 129.

You can use the `serveradmin settings` command to add other subnets to your DHCP configuration.

Note: Be sure to include the special first setting (ending with `= create`). This is how you tell `serveradmin` to create the necessary settings array with the specified subnet ID.

To add a subnet:

```
$ sudo serveradmin settings
dhcp:subnets:_array_id:subnetID = create
dhcp:subnets:_array_id:subnetID:WINS_NBDD_server = nbdd-server
dhcp:subnets:_array_id:subnetID:WINS_node_type = node-type
dhcp:subnets:_array_id:subnetID:net_range_start = start-address
dhcp:subnets:_array_id:subnetID:WINS_scope_id = scope-ID
dhcp:subnets:_array_id:subnetID:dhcp_router = router
dhcp:subnets:_array_id:subnetID:net_address = net-address
dhcp:subnets:_array_id:subnetID:net_range_end = end-address
dhcp:subnets:_array_id:subnetID:lease_time_secs = lease-time
dhcp:subnets:_array_id:subnetID:dhcp_ldap_url:_array_index:0 = ldap-server
dhcp:subnets:_array_id:subnetID:WINS_secondary_server = wins-server-2
dhcp:subnets:_array_id:subnetID:descriptive_name = description
dhcp:subnets:_array_id:subnetID:WINS_primary_server = wins-server-1
dhcp:subnets:_array_id:subnetID:dhcp_domain_name = domain
dhcp:subnets:_array_id:subnetID:dhcp_enabled = (yes|no)
dhcp:subnets:_array_id:subnetID:dhcp_domain_name_server:_array_index:0 =
     dns-server-1
dhcp:subnets:_array_id:subnetID:dhcp_domain_name_server:_array_index:1 =
     dns-server-2
dhcp:subnets:_array_id:subnetID:net_mask = mask
dhcp:subnets:_array_id:subnetID:selected_port_name = port
Control-D
```

Parameter	Description
subnetID	A unique number that identifies the subnet. Can be any number not already assigned to another subnet defined on the server. Can include embedded hyphens (–).
dns-server-*n*	To specify additional DNS servers, add additional dhcp_name_server settings, incrementing _array_index:*n* for each additional value.
Other parameters	The standard subnet settings described under "DHCP Subnet Settings Array" on page 131.

List of DHCP `serveradmin` Commands

You can use the following command with the `serveradmin` application to manage DHCP service.

Command (dhcp:command=)	Description
getLogPaths	Determine the location of the DHCP service logs.

Viewing the DHCP Service Log

You can use `tail` or any other file listing tool to view the contents of the DHCP service log.

To view the latest entries in a log:

```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the current DHCP log is located.

To display the log path:

```
$ sudo serveradmin command dhcp:command = getLogPaths
```

Output

```
dhcp:systemLog = <system-log>
```

Value	Description
<system-log>	The location of the DNS service log.
	Default = /var/logs/system.log

DNS Service

Starting and Stopping the DNS Service

To start DNS service:

```
$ sudo serveradmin start dns
```

To stop DNS service:

```
$ sudo serveradmin stop dns
```

Checking the Status of DNS Service

To see summary status of DNS service:

```
$ sudo serveradmin status dns
```

To see detailed status of DNS service:

```
$ sudo serveradmin fullstatus dns
```

Viewing DNS Service Settings

To list DNS service configuration settings:

```
$ sudo serveradmin settings dns
```

To list a particular setting:

```
$ sudo serveradmin settings dns:setting
```

To list a group of settings:

Type only as much of the name as you want, stopping at a colon (:), then type an asterisk (*) as a wildcard for the remaining parts of the name. For example,

```
$ sudo serveradmin settings dns:zone:_array_id:localhost:*
```

Changing DNS Service Settings

You can use `serveradmin` to modify your server's DNS configuration. However, you'll probably find it more straightforward to work directly with DNS and BIND using the standard tools and techniques described in the many books on the subject. (See, for example, "DNS and BIND" by Paul Albitz and Cricket Liu.)

DNS Service Settings

To list the settings, see "Viewing DNS Service Settings" on this page.

List of DNS `serveradmin` Commands

Command (`dns:command=`)	Description
`getLogPaths`	Find the location of the DNS service log. See "Viewing the DNS Service Log" on this page.
`getStatistics`	Retrieve DNS service statistics. See "Listing DNS Service Statistics" on page 136.

Viewing the DNS Service Log

You can use `tail` or any other file listing tool to view the contents of the DNS service log.

To view the latest entries in a log:

```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the current DNS log is located. The default is `/Library/Logs/named.log`.

To display the log path:

```
$ sudo serveradmin command dns:command = getLogPaths
```

Listing DNS Service Statistics

You can use the `serveradmin getStatistics` command to display a summary of current DNS service workload.

To list statistics:

```
$ sudo serveradmin command dns:command = getStatistics
```

Sample Output

```
dns:queriesArray:_array_index:0:name = "NS_QUERIES"
dns:queriesArray:_array_index:0:value = -1
dns:queriesArray:_array_index:1:name = "A_QUERIES"
dns:queriesArray:_array_index:1:value = -1
dns:queriesArray:_array_index:2:name = "CNAME_QUERIES"
dns:queriesArray:_array_index:2:value = -1
dns:queriesArray:_array_index:3:name = "PTR_QUERIES"
dns:queriesArray:_array_index:3:value = -1
dns:queriesArray:_array_index:4:name = "MX_QUERIES"
dns:queriesArray:_array_index:4:value = -1
dns:queriesArray:_array_index:5:name = "SOA_QUERIES"
dns:queriesArray:_array_index:5:value = -1
dns:queriesArray:_array_index:6:name = "TXT_QUERIES"
dns:queriesArray:_array_index:6:value = -1
dns:nxdomain = 0
dns:nxrrset = 0
dns:reloadedTime = ""
dns:success = 0
dns:failure = 0
dns:recursion = 0
dns:startedTime = "2003-09-10 11:24:03 -0700"
dns:referral = 0
```

Firewall Service

Starting and Stopping Firewall Service

To start Firewall service:

```
$ sudo serveradmin start ipfilter
```

To stop Firewall service:

```
$ sudo serveradmin stop ipfilter
```

Checking the Status of Firewall Service

To see summary status of Firewall service:

```
$ sudo serveradmin status ipfilter
```

To see detailed status of Firewall service, including rules:

```
$ sudo serveradmin fullstatus ipfilter
```

Viewing Firewall Service Settings

To list Firewall service configuration settings:

```
$ sudo serveradmin settings ipfilter
```

To list a particular setting:

```
$ sudo serveradmin settings ipfilter:setting
```

To list a group of settings:

Type only as much of the name as you want, stopping at a colon (:), then type an asterisk (*) as a wildcard for the remaining parts of the name. For example,

```
$ sudo serveradmin settings ipfilter:ipAddressGroups:*
```

Changing Firewall Service Settings

To change a setting:

```
$ sudo serveradmin settings ipfilter:setting = value
```

Parameter	Description
setting	A IPFilter service setting. See "Firewall Service Settings" on this page.
value	An appropriate value for the setting.

To change several settings:

```
$ sudo serveradmin settings
ipfilter:setting = value
ipfilter:setting = value
ipfilter:setting = value
[...]
Control-D
```

Firewall Service Settings

Use the following parameters with the `serveradmin` command to change settings for the IPFilter service.

Parameter (`ipfilter:`)	Description
ipAddressGroupsWithRules: _array_id:<group>...	An array of settings describing the services allowed for specific IP address groups. See "IPFilter Groups With Rules Array" on page 138.
rules:_array_id:<rule>:...	Arrays of rule settings, one array per defined rule. See "IPFilter Rules Array" on page 141.

Parameter (`ipfilter:`)	Description
`logAllDenied`	Specifies whether to log all denials. Default = `no`
`ipAddressGroups:_array_id: n:address`	The address of a defined IP address group, the first element of an array that defines an IP address group.
`ipAddressGroups:_array_id: n:name`	The name of a defined IP address group, the second element of an array that defines an IP address group.
`logAllAllowed`	Whether to log access allowed by rules. Default = `no`

IPFilter Groups With Rules Array

An array of the following settings is included in the IPFilter settings for each defined IP address group. These arrays aren't part of a standard ipfw configuration, but are created by the Server Admin GUI application to implement the IP Address groups on the General pane of the Firewall service settings. In an actual list of settings, `<group>` is replaced with an IP address group.

Parameter (`ipfilter:`)	Description
`ipAddressGroupsWithRules: _array_id:<group>:rules`	An array of rules for the group.
`ipAddressGroupsWithRules: _array_id:<group>:addresses`	The group's address.
`ipAddressGroupsWithRules: _array_id:<group>:name`	The group's name.
`ipAddressGroupsWithRules: _array_id:<group>:readOnly`	Whether the group is set for read-only.

Defining Firewall Rules

You can use `serveradmin` to set up firewall rules for your server. However, a simpler method is to add your rules to a configuration file used by the service. By modifying the file, you'll be able to define your rules using standard rule syntax instead of creating a specialized array to store the rule's components.

Adding Rules by Modifying `ipfw.conf`

The file in which you can define your rules is /etc/ipfilter/ipfw.conf. The Firewall service reads this file, but doesn't modify it. Its contents are annotated and include commented-out rules you can use as models. Its default contents are listed below.

For more information, read the `ipfw` man page.

The unmodified ipfw.conf file:

```
# ipfw.conf.default - Installed by Apple, never modified by Server Admin app
#
# ipfw.conf - The servermgrd process (the back end of Server Admin app)
# creates this from ipfw.conf.default if it's absent, but does not modify
# it.
#
# Administrators can place custom ipfw rules in ipfw.conf.
#
# Whenever a change is made to the ipfw rules by the Server Admin
# application and saved:
#    1. All ipfw rules are flushed
#    2. The rules defined by the Server Admin app (stored as plists)
#          are exported to /etc/ipfilter/ipfw.conf.apple and loaded into the
#          firewall via ipfw.
#    3. The rules in /etc/ipfilter/ipfw.conf are loaded into the firewall
#          via ipfw.
# Note that the rules loaded into the firewall are not applied unless the
# firewall is enabled.
#
# The rules resulting from the Server Admin app's IPFirewall and NAT panels
# are numbered:
#    10 - from the NAT Service - this is the NAT divert rule, present only
#          when he NAT service is started via the Server Admin app.
#    1000 - from the "Advanced" panel - the modifiable rules, ordered by
#          their relative position in the drag-sortable rule list
#    12300 - from the "General" panel - "allow"" rules that punch specific
#          holes in the firewall for specific services
#    63200 - from the "Advanced" panel - the non-modifiable rules at the
#          bottom of the panel's rule list
#
# Refer to the man page for ipfw(8) for more information.
#
# The following default rules are already added by default:
#
#add 01000 allow all from any to any via lo0
#add 01010 deny all from any to 127.0.0.0/8
#add 01020 deny ip from 224.0.0.0/4 to any in
#add 01030 deny tcp from any to 224.0.0.0/4 in
#add 12300 ("allow" rules from the "General" panel)
#...
#add 63200 deny icmp from any to any in icmptypes 0 in
#add 63300 deny igmp from any to any in
#add 65000 deny tcp from any to any in setup
```

For more information, read the `ipfw` man page.

Adding Rules Using `serveradmin`

If you prefer not to work with the `ipfw.conf` file, you can use the `serveradmin settings` command to add firewall rules to your configuration.

Note: Be sure to include the special first setting (ending with `= create`). This is how you tell `serveradmin` to create the necessary rule array with the specified rule number.

To add a subnet:

```
$ sudo serveradmin settings
ipfilter:rules:_array_id:rule = create
ipfilter:rules:_array_id:rule:source = source
ipfilter:rules:_array_id:rule:protocol = protocol
ipfilter:rules:_array_id:rule:destination = destination
ipfilter:rules:_array_id:rule:action = action
ipfilter:rules:_array_id:rule:enableLocked = (yes|no)
ipfilter:rules:_array_id:rule:enabled = (yes|no)
ipfilter:rules:_array_id:rule:log = (yes|no)
ipfilter:rules:_array_id:rule:readOnly = (yes|no)
ipfilter:rules:_array_id:rule:source-port = port
Control-D
```

Parameter	Description
`rule`	A unique rule number.
Other parameters	The standard rule settings described under "IPFilter Rules Array" on page 141.

Example:

```
$ sudo serveradmin settings
ipfilter:rules:_array_id:1111 = create
ipfilter:rules:_array_id:1111:source = "10.10.41.60"
ipfilter:rules:_array_id:1111:protocol = "udp"
ipfilter:rules:_array_id:1111:destination = "any via en0"
ipfilter:rules:_array_id:1111:action = "allow"
ipfilter:rules:_array_id:1111:enableLocked = yes
ipfilter:rules:_array_id:1111:enabled = yes
ipfilter:rules:_array_id:1111:log = no
ipfilter:rules:_array_id:1111:readOnly = yes
ipfilter:rules:_array_id:1111:source-port = ""
Control-D
```

IPFilter Rules Array

An array of the following settings is included in the IPFilter settings for each defined firewall rule. In an actual list of settings, `<rule>` is replaced with a rule number. You can add a rule by using `serveradmin` to create such an array in the firewall settings (see "Adding Rules Using `serveradmin`" on page 140).

Parameter (`ipfilter:`)	Description
`rules:_array_id:<rule>: source`	The source of traffic governed by the rule.
`rules:_array_id:<rule>: protocol`	The protocol for traffic governed by the rule.
`rules:_array_id:<rule>: destination`	The destination of traffic governed by the rule.
`rules:_array_id:<rule>: action`	The action to be taken.
`rules:_array_id:<rule>: enabled`	Whether the rule is enabled.
`rules:_array_id:<rule>: log`	Whether activation of the rule is logged.
`rules:_array_id:<rule>: readOnly`	Whether read-only is set.
`rules:_array_id:<rule>: source-port`	The source port of traffic governed by the rule.

Firewall `serveradmin` Commands

You can use the following commands with the `serveradmin` application to manage Firewall (ipfilter) service.

Command (`ipfilter:command=`)	Description
getLogPaths	Find the current location of the log used by the service. Default = `/var/log/system.log`
getStandardServices	Retrieve a list of the standard services as they appear on the General pane of the Firewall service settings in the Server Admin GUI application.
writeSettings	Equivalent to the standard `serveradmin settings` command, but also returns a setting indicating whether the service needs to be restarted. See "Determining Whether a Service Needs to be Restarted" on page 19.

Viewing Firewall Service Log

You can use `tail` or any other file listing tool to view the contents of the ipfilter service log.

To view the latest entries in the log:

```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the current ipfilter service log is located.

To display the log path:

```
$ sudo serveradmin command ipfilter:command = getLogPaths
```

Output

```
ipfilter:systemLog = <system-log>
```

Value	Description
`<system-log>`	The location of the ipfilter service log.
	Default = `/var/log/system.log`

Using Firewall Service to Simulate Network Activity

You can use the Firewall service in Mac OS X service in conjunction with Dummynet, a general-purpose network load simulator. For more information on Dummynet, visit ai3.asti.dost.gov.ph/sat/dummynet.html or use Google or Sherlock to search the web.

NAT Service

Starting and Stopping NAT Service

To start NAT service:

```
$ sudo serveradmin start nat
```

To stop NAT service:

```
$ sudo serveradmin stop nat
```

Checking the Status of NAT Service

To see summary status of NAT service:

```
$ sudo serveradmin status nat
```

To see detailed status of NAT service:

```
$ sudo serveradmin fullstatus nat
```

Viewing NAT Service Settings

To list NAT service configuration settings:

```
$ sudo serveradmin settings nat
```

To list a particular setting:

```
$ sudo serveradmin settings nat:setting
```

Changing NAT Service Settings

To change a setting:

```
$ sudo serveradmin settings nat:setting = value
```

Parameter	Description
setting	A NAT service setting. To see a list of available settings, type `$ sudo serveradmin settings nat` or see "NAT Service Settings" on this page.
value	An appropriate value for the setting.

To change several settings:

```
$ sudo serveradmin settings
nat:setting = value
nat:setting = value
nat:setting = value
[...]
Control-D
```

NAT Service Settings

Use the following parameters with the `serveradmin` command to change settings for NAT service.

Parameter (`nat:`)	Description
deny_incoming	yes|no Default = no.
log_denied	yes|no Default = no.
clamp_mss	yes|no Default = yes
reverse	yes|no Default = no
log	yes|no Default = yes
proxy_only	yes|no Default = no
dynamic	yes|no Default = yes
use_sockets	yes|no Default = yes
interface	The network port. Default = "en0"

Parameter (`nat:`)	Description
`unregistered_only`	yes | no
	Default = `no`
`same_ports`	yes | no
	Default = `yes`

NAT `serveradmin` Commands

You can use the following commands with the `serveradmin` application to manage NAT service.

Command (`nat:command=`)	Description
`getLogPaths`	Find the current location of the log used by the NAT service. See "Viewing the NAT Service Log" on this page.
`updateNATRuleInIpfw`	Update the firewall rules defined in the ipfilter service to reflect changes in the NAT settings.
`writeSettings`	Equivalent to the standard `serveradmin settings` command, but also returns a setting indicating whether the service needs to be restarted. See "Determining Whether a Service Needs to be Restarted" on page 19.

Viewing the NAT Service Log

You can use `tail` or any other file listing tool to view the contents of the NAT service log.

To view the latest entries in the log:

```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the current NAT service log is located.

To display the log path:

```
$ sudo serveradmin command nat:command = getLogPaths
```

Output

```
nat:natLog = <nat-log>
```

Value	Description
`<nat-log>`	The location of the NAT service log.
	Default = `/var/log/alias.log`

VPN Service

Starting and Stopping VPN Service

To start VPN service:

```
$ sudo serveradmin start vpn
```

To stop VPN service:

```
$ sudo serveradmin stop vpn
```

Checking the Status of VPN Service

To see summary status of VPN service:

```
$ sudo serveradmin status vpn
```

To see detailed status of VPN service:

```
$ sudo serveradmin fullstatus vpn
```

Viewing VPN Service Settings

To list VPN service configuration settings:

```
$ sudo serveradmin settings vpn
```

To list a particular setting:

```
$ sudo serveradmin settings vpn:setting
```

Changing VPN Service Settings

To change a setting:

```
$ sudo serveradmin settings vpn:setting = value
```

Parameter	Description
setting	A VPN service setting. To see a list of available settings, type
	`$ sudo serveradmin settings vpn`
	or see "List of VPN Service Settings" on page 146.
value	An appropriate value for the setting.

To change several settings:

```
$ sudo serveradmin settings
vpn:setting = value
vpn:setting = value
vpn:setting = value
[...]
Control-D
```

List of VPN Service Settings

Use the following parameters with the `serveradmin` command to change settings for VPN service.

Parameter (`vpn:Servers:`)	Description
`com.<name>.ppp.l2tp:` `Server:VerboseLogging`	Default = `1`
`com.<name>.ppp.l2tp:` `Server:MaximumSessions`	Default = `128`
`com.<name>.ppp.l2tp:` `Server:LogFile`	Default = `"/var/log/ppp/vpnd.log"`
`com.<name>.ppp.l2tp:` `L2TP:IPSecSharedSecretEncryption`	Default = `"Key"`
`com.<name>.ppp.l2tp:` `L2TP:IPSecSharedSecretValue`	Default = `""`
`com.<name>.ppp.l2tp:` `L2TP:IPSecSharedSecret`	Default = `""`
`com.<name>.ppp.l2tp:` `L2TP:Transport`	Default = `"IPSec"`
`com.<name>.ppp.l2tp:` `enabled`	Default = `no`
`com.<name>.ppp.l2tp:` `IPv4:DestAddressRanges`	Default = `_empty_array`
`com.<name>.ppp.l2tp:` `IPv4:OfferedRouteMasks`	Default = `_empty_array`
`com.<name>.ppp.l2tp:` `IPv4:OfferedRouteAddresses`	Default = `_empty_array`
`com.<name>.ppp.l2tp:` `IPv4:OfferedRouteTypes`	Default = `_empty_array`
`com.<name>.ppp.l2tp:` `IPv4:ConfigMethod`	Default = `"Manual"`
`com.<name>.ppp.l2tp:` `DNS:OfferedSearchDomains`	Default = `_empty_array`
`com.<name>.ppp.l2tp:` `DNS:OfferedServerAddresses`	Default = `_empty_array`
`com.<name>.ppp.l2tp:` `DSACL:Group`	Default = `""`
`com.<name>.ppp.l2tp:` `Interface:SubType`	Default = `"L2TP"`
`com.<name>.ppp.l2tp:` `Interface:Type`	Default = `"PPP"`
`com.<name>.ppp.l2tp:` `PPP:LCPEchoFailure`	Default = `5`

Parameter (`vpn:Servers:`)	Description
`com.<name>.ppp.l2tp:` `PPP:DSACLEnabled`	Default = `no`
`com.<name>.ppp.l2tp:` `PPP:VerboseLogging`	Default = `1`

`com.<name>.ppp.l2tp:`
`PPP:AuthenticatorPlugins:`
`_array_index:n`
	Default = `"DSAuth"`

`com.<name>.ppp.l2tp:` `PPP:LCPEchoInterval`	Default = `60`
`com.<name>.ppp.l2tp:` `PPP:LCPEchoEnabled`	Default = `1`
`com.<name>.ppp.l2tp:` `PPP:IPCPCompressionVJ`	Default = `0`

`com.<name>.ppp.l2tp:`
`PPP:AuthenticatorProtocol:`
`_array_index:n`
	Default = `"MSCHAP2"`

`com.<name>.ppp.l2tp:` `PPP:LogFile`	Default = `"/var/log/ppp/vpnd.log"`
`com.<name>.ppp.pptp:` `Server:VerboseLogging`	Default = `1`
`com.<name>.ppp.pptp:` `Server:MaximumSessions`	Default = `128`
`com.<name>.ppp.pptp:` `Server:LogFile`	Default = `"/var/log/ppp/vpnd.log"`
`com.<name>.ppp.pptp:` `enabled`	Default = `no`
`com.<name>.ppp.pptp:` `IPv4:DestAddressRanges`	Default = `_empty_array`
`com.<name>.ppp.pptp:` `IPv4:OfferedRouteMasks`	Default = `_empty_array`
`com.<name>.ppp.pptp:` `IPv4:OfferedRouteAddresses`	Default = `_empty_array`
`com.<name>.ppp.pptp:` `IPv4:OfferedRouteTypes`	Default = `_empty_array`
`com.<name>.ppp.pptp:` `IPv4:ConfigMethod`	Default = `"Manual"`
`com.<name>.ppp.pptp:` `DNS:OfferedSearchDomains`	Default = `_empty_array`
`com.<name>.ppp.pptp:` `DNS:OfferedServerAddresses`	Default = `_empty_array`
`com.<name>.ppp.pptp:` `DSACL:Group`	Default = `""`

Parameter (`vpn:Servers:`)	Description
`com.<name>.ppp.pptp:` `Interface:SubType`	Default = `"PPTP"`
`com.<name>.ppp.pptp:` `Interface:Type`	Default = `"PPP"`
`com.<name>.ppp.pptp:` `PPP:CCPProtocols:_array_index:`*n*	Default = `"MPPE"`
`com.<name>.ppp.pptp:` `PPP:LCPEchoFailure`	Default = 5
`com.<name>.ppp.pptp:` `PPP:MPPEKeySize128`	Default = 1
`com.<name>.ppp.pptp:` `PPP:DSACLEnabled`	Default = `no`
`com.<name>.ppp.pptp:` `PPP:VerboseLogging`	Default = 1

`com.<name>.ppp.pptp:`
`PPP:AuthenticatorPlugins:`
`_array_index:`*n*
	Default = `"DSAuth"`

`com.<name>.ppp.pptp:` `PPP:MPPEKeySize40`	Default = 0
`com.<name>.ppp.pptp:` `PPP:LCPEchoInterval`	Default = 60
`com.<name>.ppp.pptp:` `PPP:LCPEchoEnabled`	Default = 1
`com.<name>.ppp.pptp:` `PPP:CCPEnabled`	Default = 1
`com.<name>.ppp.pptp:` `PPP:IPCPCompressionVJ`	Default = 0

`com.<name>.ppp.pptp:`
`PPP:AuthenticatorProtocol:`
`_array_index:`*n*
	Default = `"MSCHAP2"`

`com.<name>.ppp.pptp:` `PPP:LogFile`	Default = `"/var/log/ppp/vpnd.log"`

List of VPN `serveradmin` Commands

You can use the following commands with the `serveradmin` application to manage VPN service.

Command (`vpn:command=`)	Description
`getLogPaths`	Find the current location of the VPN service log. See "Viewing the VPN Service Log" on this page.
`writeSettings`	Equivalent to the standard `serveradmin settings` command, but also returns a setting indicating whether the service needs to be restarted. See "Determining Whether a Service Needs to be Restarted" on page 19.

Viewing the VPN Service Log

You can use `tail` or any other file listing tool to view the contents of the VPN service log.

To view the latest entries in the log:

```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the current VPN service log is located.

To display the log path:

```
$ sudo serveradmin command vpn:command = getLogPaths
```

Output

```
vpn:vpnLog = <vpn-log>
```

Value	Description
`<vpn-log>`	The location of the VPN service log. Default = `/var/log/vpnd.log`

IP Failover

IP failover allows a secondary server to acquire the IP address of a primary server if the primary server ceases to function. Once the primary server returns to normal operation, the secondary server relinquishes the IP address. This allows your website to remain available on the network even if the primary server is temporarily offline.

Note: IP failover only allows a secondary server to acquire a primary server's IP address. You need additional software tools such as `rsync` to provide capabilities such as mirroring the primary server's data on the secondary server. See the `rsync` man pages for more information.

Requirements

IP failover isn't a complete solution; it is one tool you can use to increase your server's availability to your clients. To use IP failover, you will need to set up the following hardware and software.

Hardware

IP failover requires the following hardware setup:
- Primary server
- Secondary server
- Public network (servers must be on same subnet)
- Private network between the servers (additional network interface card)

Note: Because IP failover uses broadcast messages, both servers must have IP addresses on the same subnet of the public network. In addition, both servers must have IP addresses on the same subnet of the private network.

Software

IP failover requires the following software setup:
- Unique IP addresses for each network interface (public and private)
- Software to mirror primary server data to secondary server
- Scripts to control failover behavior on secondary server (optional)

Failover Operation

When IP failover is active, the primary server periodically broadcasts a brief message confirming normal operation on both the public and private networks. This message is monitored by the secondary server.
- If the broadcast is interrupted on both public and private networks, the secondary server initiates the failover process.
- If status messages are interrupted on only one network, the secondary server sends email notification of a network anomaly, but doesn't acquire the primary server's IP address.

Email notification is sent when the secondary server detects a failover condition, a network anomaly, and when the IP address is relinquished back to the primary server.

Enabling IP Failover

You enable IP failover by adding command lines to the file /etc/hostconfig on the primary and the secondary server. Be sure to enter these lines exactly as shown with regard to spaces and punctuation marks.

To enable IP failover:

1 At the primary server, add the following line to /etc/hostconfig:

```
FAILOVER_BCAST_IPS="10.0.0.255 100.0.255.255"
```

Substitute the broadcast addresses used on your server for the public and private networks. This tells the server to send broadcast messages over relevant network interfaces that the server at those IP addresses is functioning.

2 Restart the primary server so that your changes can take effect.

3 Disconnect the primary server from both the public and private networks.

4 At the secondary server, add the following lines to /etc/hostconfig:

```
FAILOVER_PEER_IP="10.0.0.1"
FAILOVER_PEER_IP_PAIRS="en0:100.0.0.10"
FAILOVER_EMAIL_RECIPIENT="admin@example.com"
```

In the first line substitute the IP address of the primary server on the private network.

In the second line enter the local network interface that should adopt the primary server's public IP address, a colon, then the primary server's public IP address.

(Optional) In the third line, enter the email address for notification messages regarding the primary server status. If this line is omitted, email notifications are sent to the root account on the local machine.

5 Restart the secondary server so your changes can take effect and allow the secondary server to acquire the primary's public IP address.

Important: Before you enable IP Failover, verify on both servers that the port used for the public network is at the top of the Network Port Configurations list in the Network pane of System Preferences. Also verify that the port used for the private network contains no DNS configuration information.

6 Reconnect the primary server to the private network, wait fifteen seconds, then reconnect the primary server to the public network.

7 Verify that the secondary server relinquishes the primary server's public IP address.

Configuring IP Failover

You configure failover behavior using scripts. The scripts must be executable (for example, shell scripts, Perl, compiled C code, or executable AppleScripts). You place these scripts in /Library/IPFailover/<IP address> on the secondary server.

You need to create a directory named with the public IP address of the primary server to contain the failover scripts for that server. For example:

```
/Library/IPFailover/100.0.0.10
```

Notification Only

You can use a script named "Test" located in the failover scripts directory to control whether, in the event of a failover condition, the secondary server acquires the primary's IP address, or simply sends an email notification. If no script exists, or if the script returns a zero result, then the secondary server acquires the primary's IP address. If the script returns a non-zero result, then the secondary server skips IP address acquisition and only sends email notification of the failover condition. The test script is run to determine whether the IP address should be acquired and to determine if the IP address should be relinquished when the primary server returns to service.

A simple way to set up this notification-only mode is to copy the script located at /usr/bin/false to the directory named with your primary server IP address and then change the name of the script to "Test". This script always returns a non-zero result.

Using the Test script, you can configure the primary server to monitor the secondary server, and send email notification if the secondary server becomes unavailable.

Pre and Post Scripts

You can configure the failover process with scripts that can run before acquiring the primary IP address (preacquisition), after acquiring the IP address (postacquisition), before relinquishing the primary IP address (prerelinquish), and after relinquishing the IP address back to the primary server (postrelinquish). These scripts reside in the /Library/IPFailover/<IP address> directory on the secondary server, as previously discussed. The scripts use these four prefixes:
- PreAcq – run before acquiring IP address from primary server
- PostAcq – run after acquiring IP address from primary server
- PreRel – run before relinquishing IP address back to primary server
- PostRel – run after relinquishing IP address back to primary server

Important: Always be sure that the primary server is up and functioning normally before you activate IP failover on the secondary server. If the primary server isn't sending broadcast messages, the secondary server will initiate the failover process and acquire the primary's public IP address.

You may have more than one script at each stage. The scripts in each prefix group are run in the order their file names appear in a directory listing using the `ls` command.

For example, your secondary server may perform other services on the network such as running a statistical analysis application and distributed image processing software. A preacquisition script quits the running applications to free up the CPU for the Web server. A postacquisition script starts the Web server. Once the primary is up and running again, a prerelinquish script quits the Web server, and a postrelinquish script starts the image processing and statistical analysis applications. The sequence of scripted events might look like this:

```
<Failover condition detected>
Test (if present)
PreAcq10.StopDIP
PreAcq20.StopSA
PreAcq30.CleanupTmp
<Acquire IP address>
PostAcq10.StartTimer
PostAcq20.StartApache
<Primary server returns to service>
PreRel10.StopApache
PreRel20.StopTimer
<Relinquish IP address>
PostRel10.StartSA
PostRel20.StartDIP
PostRel30.MailTimerResultsToAdmin
```

Enabling PPP Dial-In

You can use the `pppd` command to set up Point-to-Point Protocol (PPP) dial-in service. For more information, see the man page. The "Examples" section of the man page shows an example of setting up dial-in service.

Working With Open Directory

14

Commands you can use to manage the Open Directory service in Mac OS X Server.

This chapter includes descriptions of general directory tools and tools for working with LDAP, NetInfo, and the Password Server.

General Directory Tools

Testing Your Open Directory Configuration

You can use the `dscl` utility to test your directory services configuration. For more information, type `man dscl` to see the man page.

Modifying an Open Directory Node

You can also use the `dscl` utility to create, modify, or delete directory information in an Open Directory node.

Testing Open Directory Plugins

You can use the `dsperfmonitor` tool to check the performance of the protocol-specific plugins used by Open Directory. It can list the API calls being made to plugins, how long the plugins take to reply, and recent API call errors.

For more information, type `man dsperfmonitor` to see the man page.

Directory services API support is provided by the `DirectoryService` daemon. For more information, type `man DirectoryService` to see the man page.

For information on the data types used by directory services, type `man DirectoryServiceAttributes` to see the man page.

Finally, for information on the internals of Open Directory and its plugins, including source code you can examine or adopt, follow the Open Directory link at www.apple.com/darwin.

Registering URLs With Service Location Protocol (SLP)

You can use the `slp_reg` command to register service URLs using the Service Location Protocol (SLP).

For more information, type `man slp_reg` to see the man page.

SLP registration is handled by the SLP daemon `slpd`. For more information, type `man slpd` to see the man page.

Changing Open Directory Service Settings

Use the following parameters with the `serveradmin` command to change settings for the Open Directory service.

Be sure to add `dirserv:` to the beginning of any parameter you use. For example, to see the role that the server is playing in the directory hierarchy, you would type `serveradmin settings dirserv:LDAPServerType`.

Parameter (`dirserv:`)	Description
`replicationUnits`	Default = `"days"`
`replicaLastUpdate`	Default = `""`
`LDAPDataBasePath`	Default = `""`
`replicationPeriod`	Default = 4
`LDAPSearchBase`	Default = `""`
`passwordOptionsString`	Default = `"usingHistory=0 usingExpirationDate=0 usingHardExpirationDate=0 requiresAlpha=0 requiresNumeric=0 expirationDateGMT=12/31/69 hardExpireDateGMT=12/31/69 maxMinutesUntilChangePassword=0 maxMinutesUntilDisabled=0 maxMinutesOfNonUse=0 maxFailedLoginAttempts=0 minChars=0 maxChars=0 passwordCannotBeName=0"`
`NetInfoRunStatus`	Default = `""`
`LDAPSSLCertificatePath`	Default = `""`
`masterServer`	Default = `""`
`LDAPServerType`	Default = `"standalone"`
`NetInfoDomain`	Default = `""`
`replicationWhen`	Default = `"periodic"`
`useSSL`	Default = `"YES"`
`LDAPDefaultPrefix`	Default = `"dc=<domain>,dc=com"`
`LDAPTimeoutUnits`	Default = `"minutes"`
`LDAPServerBackend`	Default = `"BerkeleyDB"`

LDAP

Configuring LDAP

The following tools are available for configuring LDAP. For more information, see the man page for each tool.

slapconfig

You can use the `slapconfig` utility to configure the `slapd` and `slurpd` LDAP daemons and related search policies. For more information, type `man slapconfig` to see the man page.

Standard Distribution Tools

These tools are included in the standard LDAP distribution.

Program	Used to
`/usr/bin/ldapadd`	Add entries to the LDAP directory.
`/usr/bin/ldapcompare`	Compare a directory entry's actual attributes with known attributes.
`/usr/bin/ldapdelete`	Delete entries from the LDAP directory.
`/usr/bin/ldapmodify`	Change an entry's attributes.
`/usr/bin/ldapmodrdn`	Change an entry's relative distinguished name (RDN).
`/usr/bin/ldappasswd`	Set the password for an LDAP user. Apple recommends using `passwd` instead of `ldappasswd`. For more information, type `man passwd`.
`/usr/bin/ldapsearch`	Search the LDAP directory. See the usage note under "A Note on Using `ldapsearch`" on this page.
`/usr/bin/ldapwhoami`	Obtain the primary authorization identity associated with a user.
`/usr/sbin/slapadd`	Add entries to the LDAP directory.
`/usr/sbin/slapcat`	Export LDAP Directory Interchange Format files.
`/usr/sbin/slapindex`	Regenerate directory indexes.
`/usr/sbin/slappasswd`	Generate user password. hashes.

A Note on Using `ldapsearch`

The `ldapsearch` tool connects to an LDAP server, binds to it, finds entries, and returns attributes of the entries found.

By default, `ldapsearch` tries to connect to the LDAP server using the Simple Authentication and Security Layer (SASL) method. If the server doesn't support this method, you see this error message:

```
ldap_sasl_interactive_bind_s: No such attribute (16)
```

To avoid this, include the `-x` option when you type the command. For example:

```
ldapsearch -h 192.168.100.1 -b "dc=ecxample,dc=com" -x
```

The `-x` option forces `ldapsearch` to use simple authentication instead of SASL.

Idle Rebinding Options

The following two LDAPv3 plugin parameters aren't documented in the open directory administration guide. The parameters are in, or can be added to, the file /library/preferences/directoryservice/DSLDAPv3PlugInConfig.plist.

Delay Rebind

This parameter specifies how long the LDAP plugin waits before attempting to reconnect to a server that fails to respond. You can increase this value to prevent continuous reconnect attempts.

```
<key>Delay Rebind Try in seconds<\key>
<integer>n<\integer>
```

You should find this parameter in the plist file near `<key>OpenClose Timeout in seconds<\key>`. If not, you can add it there.

Idle Timeout

This parameter specifies how long the LDAP plugin will sit idle before disconnecting from the server. You can adjust this value to reduce overloading of the server's connections from remote clients.

```
<key>Idle Timeout in minutes<\key>
<integer>n<\integer>
```

If it doesn't already exist in the plist file, you can add it near `<key>OpenClose Timeout in seconds<\key>`.

Additional Information About LDAP

The LDAP server in Mac OS X Server is based on OpenLDAP. Additional information about OpenLDAP, including an administrator's guide, is available at www.openldap.org.

NetInfo

Configuring NetInfo

You can use the following command-line utilities to manage the NetInfo directory. For more information about a utility, see the related man page.

Utility	Used to
NeST	Configure the directory system of a server.
nicl	Create, view, and modify entries in the NetInfo directory.
nifind	Search the NetInfo directory for a particular entry.
nigrep	Search the NetInfo directory for an expression.
nidump	Export NetInfo data to text or flat files.
niload	Import flat files into the NetInfo directory.
nireport	Print tables of NetInfo directory entries.

For example, you can use the `NeST -setprotocols` command to specify which authentication methods the server's Open Directory Password Server uses.

Password Server

Working With the Password Server

You can use the `mkpassdb` utility to create, modify, or back up the password database used by the Mac OS X Server Password Server. For more information, type `man mkpassdb` to read the man page.

Viewing or Changing Password Policies

You can use the `pwpolicy` command to view or change the authentication policies used by the Mac OS X Server Password Server. For more information, type `man pwpolicy` to see the man page.

Enabling or Disabling Authentication Methods

All password authentication methods supported by Open Directory Password Server are initially enabled. You can disable and enable Open Directory Password Server authentication methods by using the `NeST` tool.

To see a list of available methods:
```
$ NeST -getprotocols
```

To disable or enable a method:
```
$ NeST -setprotocols protocol (on|off)
```

Parameter	Description
protocol	Any of the protocol names listed by `NeST -getprotocols` (for example, SMB-LAN-MANAGER).

For information on the available methods, see the Open Directory administration guide.

Kerberos and Single Sign On

The following tools are available for setting up your Kerberos and Single Sign-On environment. For more information on a tool, see the related man page.

Tool (in usr/sbin/)	Description
kdcsetup	Creates necessary setup files and adds krb5kdc and kadmind servers for the Apple Open Directory KDC.
sso_util	Sets up, interrogates, and tears down the Kerberos configuration within the Apple Single Sign On environment.
kerberosautoconfig	Creates the edu.mit.Kerberos file based on the Open Directory KerberosClient record.

Working With QuickTime Streaming Server

15

Commands you can use to manage QTSS service in Mac OS X Server.

Starting QTSS Service

You can use the `serveradmin` command to start QTSS service, or you can use the `quicktimestreamingserver` command to specify additional service parameters when you start the service.

To start QTSS service:

```
$ sudo serveradmin start qtss
```

or

```
$ sudo quicktimestreamingserver
```

To see a list of `quicktimestreamingserver` command options, type

```
$ sudo quicktimestreamingserver -h
```

Stopping QTSS Service

To stop QTSS service:

```
$ sudo serveradmin stop qtss
```

Checking QTSS Service Status

To see if QTSS service is running:

```
$ sudo serveradmin status qtss
```

To see complete QTSS status:

```
$ sudo serveradmin fullstatus qtss
```

Viewing QTSS Settings

To list all QTSS service settings:

```
$ sudo serveradmin settings qtss
```

To list a particular setting:

```
$ sudo serveradmin settings qtss:setting
```

To list a group of settings:

You can list a group of settings that have part of their names in common by typing only as much of the name as you want, stopping at a colon (:), and typing an asterisk (*) as a wildcard for the remaining parts of the name. For example,

```
$ sudo serveradmin settings qtss:modules:_array_id:QTSSAdminModule:*
```

Changing QTSS Settings

You can change QTSS service settings using the `serveradmin` command or by editing the QTSS parameter list file directly.

To change a setting:

```
$ sudo serveradmin settings qtss:setting = value
```

Parameter	Description
setting	A QTSS service setting. To see a list of available settings, type `$ sudo serveradmin settings qtss` or see "QTSS Settings" on page 163.
value	An appropriate value for the setting.

To change several settings:

```
$ sudo serveradmin settings
qtss:setting = value
qtss:setting = value
qtss:setting = value
[...]
Control-D
```

QTSS Settings

Use the following parameters with the `serveradmin` command to change settings for the QTSS service.

Descriptions of Settings

To see descriptions of most QTSS settings, you can look in the sample settings file /Library/QuickTimeStreaming/Config/streamingserver.xml-sample.

Look for XML module and pref names that match the last two segments of the parameter name.

For example, to see a description of

`modules:_array_id:QTSSFileModule:record_movie_file_sdp`

Look in the sample file for

`<MODULE NAME="QTSSFileModule">...`
        `<PREF NAME="record_movie_file_sdp".`

QTSS parameters you might change:

Parameter (`qtss:`)	Description
`broadcaster:password`	Default = `""`
`broadcaster:username`	Default = `""`
`modules:_array_id:QTSSAccessLogModule:` `request_logfile_dir`	Default = `"/Library/QuickTime` `Streaming/Logs/"`
`modules:_array_id:QTSSAccessLogModule:` `request_logfile_interval`	Default = 7
`modules:_array_id:QTSSAccessLogModule:` `request_logfile_name`	Default = `"StreamingServer"`
`modules:_array_id:QTSSAccessLogModule:` `request_logfile_size`	Default = `10240000`
`modules:_array_id:QTSSAccessLogModule:` `request_logging`	Default = yes
`modules:_array_id:QTSSAccessLogModule:` `request_logtime_in_gmt`	Default = yes

`modules:_array_id:QTSSAccessModule:`
`modAccess_groupsfilepath`
	Default = `"/Library/Quick`
	`TimeStreaming/Config/`
	`qtgroups"`

`modules:_array_id:QTSSAccessModule:` `modAccess_qtaccessfilename`	Default = `"qtaccess"`

`modules:_array_id:QTSSAccessModule:`
`modAccess_usersfilepath`
	Default = `"/Library/Quick`
	`TimeStreaming/Config/`
	`qtusers"`

Parameter (`qtss:`)	Description
`modules:_array_id:QTSSAdminModule:` `AdministratorGroup`	Default = `"admin"`
`modules:_array_id:QTSSAdminModule:` `Authenticate`	Default = `yes`
`modules:_array_id:QTSSAdminModule:` `enable_remote_admin`	Default = `yes`
`modules:_array_id:QTSSAdminModule:` `IPAccessList`	Default = `"127.0.0.*"`
`modules:_array_id:QTSSAdminModule:` `LocalAccessOnly`	Default = `yes`
`modules:_array_id:QTSSFileModule:` `add_seconds_to_client_buffer_delay`	Default = `0`
`modules:_array_id:QTSSFileModule:` `admin_email`	Default = `""`
`modules:_array_id:QTSSFileModule:` `record_movie_file_sdp`	Default = `no`
`modules:_array_id:QTSSHomeDirectoryModule:` `enabled`	Default = `no`
`modules:_array_id:QTSSHomeDirectoryModule:` `movies_directory`	Default = `"/Sites/Streaming"`
`modules:_array_id:QTSSMP3StreamingModule:` `mp3_broadcast_buffer_size`	Default = `8192`
`modules:_array_id:QTSSMP3StreamingModule:` `mp3_broadcast_password`	Default = `""`
`modules:_array_id:QTSSMP3StreamingModule:` `mp3_max_flow_control_time`	Default = `10000`
`modules:_array_id:QTSSMP3StreamingModule:` `mp3_request_logfile_dir`	Default = `"/Library/QuickTime Streaming/Logs/"`
`modules:_array_id:QTSSMP3StreamingModule:` `mp3_request_logfile_interval`	Default = `7`
`modules:_array_id:QTSSMP3StreamingModule:` `mp3_request_logfile_name`	Default = `"mp3_access"`
`modules:_array_id:QTSSMP3StreamingModule:` `mp3_request_logfile_size`	Default = `10240000`
`modules:_array_id:QTSSMP3StreamingModule:` `mp3_request_logging`	Default = `yes`
`modules:_array_id:QTSSMP3StreamingModule:` `mp3_request_logtime_in_gmt`	Default = `yes`
`modules:_array_id:QTSSMP3StreamingModule:` `mp3_streaming_enabled`	Default = `yes`

Parameter (`qtss:`)	Description
`modules:_array_id:QTSSReflectorModule:` `allow_broadcasts`	Default = `yes`
`modules:_array_id:QTSSReflectorModule:` `allow_non_sdp_urls`	Default = `yes`
`modules:_array_id:QTSSReflectorModule:` `BroadcasterGroup`	Default = `"broadcaster"`
`modules:_array_id:QTSSReflectorModule:` `broadcast_dir_list`	Default = `""`
`modules:_array_id:QTSSReflectorModule:` `disable_overbuffering`	Default = `no`
`modules:_array_id:QTSSReflectorModule:` `enable_broadcast_announce`	Default = `yes`
`modules:_array_id:QTSSReflectorModule:` `enable_broadcast_push`	Default = `yes`
`modules:_array_id:QTSSReflectorModule:` `ip_allow_list`	Default = `"127.0.0.*"`
`modules:_array_id:QTSSReflectorModule:` `kill_clients_when_broadcast_stops`	Default = `no`
`modules:_array_id:QTSSReflectorModule:` `minimum_static_sdp_port`	Default = `20000`
`modules:_array_id:QTSSReflectorModule:` `timeout_broadcaster_session_secs`	Default = `20`

`modules:_array_id:QTSSRelayModule:`
`relay_prefs_file`
	Default = `"/Library/Quick`
	`TimeStreaming/Config/`
	`relayconfig.xml"`

`server:authentication_scheme`	Default = `"digest"`
`server:auto_restart`	Default = `yes`
`server:default_authorization_realm`	Default = `"Streaming` `Server"`
`server:do_report_http_connection_ip_address`	Default = `no`
`server:error_logfile_dir`	Default = `"/Library/Quick` `TimeStreaming/Logs/"`
`server:error_logfile_name`	Default = `"Error"`
`server:error_logfile_size`	Default = `256000`
`server:error_logfile_verbosity`	Default = `2`
`server:error_logging`	Default = `yes`
`server:force_logs_close_on_write`	Default = `no`
`server:maximum_bandwidth`	Default = `102400`
`server:maximum_connections`	Default = `1000`
`server:module_folder`	Default = `"/Library/Quick` `TimeStreaming/Modules/"`

Parameter (`qtss:`)	Description
`server:movie_folder`	Default = `"/Library/Quick TimeStreaming/Movies/"`
`server:pid_file`	Default = `"/var/run/Quick TimeStreamingServer.pid"`
`server:reliable_udp`	Default = `yes`
`server:reliable_udp_dirs`	Default = `"/"`
`server:run_group_name`	Default = `"qtss"`
`server:run_num_threads`	Default = `0`
`server:run_user_name`	Default = `"qtss"`
`web_admin:enabled`	Default = `no`
`web_admin:password`	Default = `""`
`web_admin:username`	Default = `""`

QTSS `serveradmin` Commands

You can use the following commands with the `serveradmin` application to manage QTSS service.

Command (`qtss:command=`)	Description
`getConnections`	List current QTSS connections. See "Listing Current Connections" on this page.
`getHistory`	View service statistics. See "Viewing QTSS Service Statistics" on page 167.
`getLogPaths`	Find the current location of the service logs. See "Viewing Service Logs" on page 168.

Listing Current Connections

You can use the `serveradmin getConnectedUsers` command to retrieve information about QTSS connections.

To list connected users:

```
$serveradmin command qtss:command = getConnectedUsers
```

Viewing QTSS Service Statistics

You can use the `serveradmin getHistory` command to display a log of periodic samples of the number of connections and the data throughput. Samples are taken once each minute.

To list samples:

```
$ sudo serveradmin command
qtss:command = getHistory
qtss:variant = statistic
qtss:timeScale = scale
Control-D
```

Parameter	Description
statistic	The value you want to display.
	Valid values:
	v1 - number of connected users (average during sampling period)
	v2 - throughput (bytes/sec)
scale	The length of time in seconds, ending with the current time, for which you want to see samples. For example, to see 30 minutes of data, you would specify `qtss:timeScale = 1800`.

Output

```
qtss:nbSamples = <samples>
qtss:samplesArray:_array_index:0:vn = <sample>
qtss:samplesArray:_array_index:0:t = <time>
qtss:samplesArray:_array_index:1:vn = <sample>
qtss:samplesArray:_array_index:1:t = <time>
[...]
qtss:samplesArray:_array_index:i:vn = <sample>
qtss:samplesArray:_array_index:i:t = <time>
qtss:vnLegend = "<legend>"
qtss:currentServerTime = <servertime>
```

Value displayed by getHistory	Description
<samples>	The total number of samples listed.
<legend>	A textual description of the selected statistic.
	`"CONNECTIONS"` for v1
	`"THROUGHPUT"` for v2
<sample>	The numerical value of the sample.
	For connections (v1), this is integer average number of connections.
	For throughput, (v2), this is integer bytes per second.
<time>	The time at which the sample was measured. A standard UNIX time (number of seconds since Sep 1, 1970). Samples are taken every 60 seconds.

Viewing Service Logs

You can use `tail` or any other file listing tool to view the contents of the QTSS service logs.

To view the latest entries in a log:

```
$ tail log-file
```

You can use the `serveradmin getLogPaths` command to see where the current QTSS error and activity logs are located.

To display the log paths:

```
$ sudo serveradmin command qtss:command = getLogPaths
```

Output

```
qtss:accessLog = <access-log>
qtss:errorLog = <error-log>
```

Value	Description
`<access-log>`	The location of the QTSS service access log. Default = `/Library/QuickTimeStreaming/Logs/StreamingServer.log`
`<error-log>`	The location of the QTSS service error log. Default = `/Library/QuickTimeStreaming/Logs/Error.log`

Forcing QTSS to Re-Read its Preferences

You can force QTSS to re-read its preferences without restarting the server. You must log in as root to perform this task.

To force QTSS to re-read its preferences:

1 List the QTSS processes:

```
$ ps -ax | grep QuickTimeStreamingServer
```

You should see a list similar to the following:

```
949  ??  Ss     0:00.00 /usr/sbin/QuickTimeStreamingServer
950  ??  S      0:00.13 /usr/sbin/QuickTimeStreamingServer
965 std  S+     0:00.00 grep QuickTimeStreamingServer
```

2 Find the larger of the two process IDs (PIDs) for the `QuickTimeStreamingServer` processes (in this case `950`).

3 Send a HUP signal to this process:

```
$ kill -HUP 950
```

Preparing Older Home Directories for User Streaming

If you want to enable QTSS home directory streaming for home directories created using an earlier version of Mac OS X Server (before version 10.3), you need to set up the necessary streaming media folder in each user's home directory. You can use the `createuserstreamingdir` tool to set up the needed /Sites/Streaming folder.

To set up /Sites/Streaming in older home directories:

```
$ createuserstreamingdir user
```

Parameter	Description
user	The user in whose home directory the /Sites/Streaming folder is created.

Index

starting service 145
stopping service 145
viewing service logs 149
viewing service settings 145

W

web proxy settings 43
Web service
changing settings 124
checking status 123
listing sites 125
script to add site 127
starting 123
stopping 123
viewing logs 125
viewing settings 123
viewing statistics 126
websites
script for adding 127
Windows service. *See* SMB service