
Mac OS X Server
Web Technologies
Administration
For Version 10.5 Leopard

Apple
1 Infinite Loop
Cupertino, CA 95014-2084
408-996-1010
www.apple.com



019-0945/2007-09-01

# Contents

# About This Guide

## This guide tells you how to set up and manage a web server, websites, and use open source web technologies.

Mac OS X Server version 10.5 includes Web service that is comprised of multiple web technologies. Web service comes preinstalled on Apple server hardware and offers an integrated, flexible environment for establishing and managing web technologies.

## What's New in Version 10.5

Mac OS X Server v10.5 offers the following enhancements to Web service:

- New and improved Apache 2.2
- Group wikis and blogs
- Easy certificate management in Server Admin
- Control of conventional (forward) and back-end (reverse) proxies
- Back-end proxy balancer, which allows simple deployment of Ruby on Rails or WebObject applications

## What's in This Guide

This guide includes the following chapters:

- Chapter 1, "Web Technologies Overview," highlights key concepts and provides basic information about configuring a server, setting up websites, and understanding specialized web components.
- Chapter 2, "Working with Web Service," describes how to set up your web server for the first time and how to manage web settings and components.
- Chapter 3, "Creating and Managing Websites," provides instructions for setting up and managing websites.
- Chapter 4, "Creating and Managing Wikis and Blogs," describes how to use Server Admin to set up and manage wikis and blogs.
- Chapter 5, "Configuring and Managing Webmail," tells you how to enable and use Webmail on your web server.

- Chapter 6, "Working with WebObjects and Open Source Applications," provides information and instructions related to WebObjects and open source components Apache, Tomcat, and MySQL.
- Chapter 7, "Managing Web Modules," describes the modules included in Mac OS X Server and explains how to install, enable, and view modules.
- Chapter 8, "Solving Web Service Problems," helps you address issues with web technologies and websites.

In addition, the Glossary defines terms you'll encounter as you read this guide.

***Note:*** Because Apple frequently releases new versions and updates to its software, images shown in this book may be different from what you see on your screen.

## Using This Guide

The following list contains suggestions for using this guide:

- Read the guide in its entirety. Subsequent sections might build on information and recommendations discussed in prior sections.
- The instructions in this guide should always be tested in a nonoperational environment before deployment. This nonoperational environment should simulate, as much as possible, the environment where the computer will be deployed.

## Using Onscreen Help

You can get task instructions onscreen in Help Viewer while you're managing Mac OS X Server. You can view help on a server or an administrator computer. (An administrator computer is a Mac OS X computer with Mac OS X Server administration software installed on it.)

**To get help for an advanced configuration of Leopard Server:**

- Open Server Admin or Workgroup Manager and then:
  - Use the Help menu to search for a task you want to perform.
  - Choose Help > Server Admin Help or Help > Workgroup Manager to browse and search the help topics.

The onscreen help contains instructions taken from *Server Administration* and other advanced administration guides described in "Mac OS X Server Administration Guides," next.

**To see the most recent server help topics:**

- Make sure the server or administrator computer is connected to the Internet while you're getting help.

Help Viewer automatically retrieves and caches the most recent server help topics from the Internet. When not connected to the Internet, Help Viewer displays cached help topics.

## Mac OS X Server Administration Guides

*Getting Started* covers installation and setup for standard and workgroup configurations of Mac OS X Server. For advanced configurations, *Server Administration* covers planning, installation, setup, and general server administration. A suite of additional guides, listed below, covers advanced planning, setup, and management of individual services. You can get these guides in PDF format from the Mac OS X Server documentation website:

www.apple.com/server/documentation

| This guide... | tells you how to: |
|---|---|
| *Getting Started* and *Mac OS X Server Worksheet* | Install Mac OS X Server and set it up for the first time. |
| *Command-Line Administration* | Install, set up, and manage Mac OS X Server using UNIX command-line tools and configuration files. |
| *File Services Administration* | Share selected server volumes or folders among server clients using the AFP, NFS, FTP, and SMB protocols. |
| *iCal Service Administration* | Set up and manage iCal shared calendar service. |
| *iChat Service Administration* | Set up and manage iChat instant messaging service. |
| *Mac OS X Security Configuration* | Make Mac OS X computers (clients) more secure, as required by enterprise and government customers. |
| *Mac OS X Server Security Configuration* | Make Mac OS X Server and the computer it's installed on more secure, as required by enterprise and government customers. |
| *Mail Service Administration* | Set up and manage IMAP, POP, and SMTP mail services on the server. |
| *Network Services Administration* | Set up, configure, and administer DHCP, DNS, VPN, NTP, IP firewall, NAT, and RADIUS services on the server. |
| *Open Directory Administration* | Set up and manage directory and authentication services, and configure clients to access directory services. |
| *Podcast Producer Administration* | Set up and manage Podcast Producer service to record, process, and distribute podcasts. |
| *Print Service Administration* | Host shared printers and manage their associated queues and print jobs. |
| *QuickTime Streaming and Broadcasting Administration* | Capture and encode QuickTime content. Set up and manage QuickTime streaming service to deliver media streams live or on demand. |
| *Server Administration* | Perform advanced installation and setup of server software, and manage options that apply to multiple services or to the server as a whole. |

| This guide... | tells you how to: |
|---|---|
| *System Imaging and Software Update Administration* | Use NetBoot, NetInstall, and Software Update to automate the management of operating system and other software used by client computers. |
| *Upgrading and Migrating* | Use data and service settings from an earlier version of Mac OS X Server or Windows NT. |
| *User Management* | Create and manage user accounts, groups, and computers. Set up managed preferences for Mac OS X clients. |
| *Web Technologies Administration* | Set up and manage web technologies, including web, blog, webmail, wiki, MySQL, PHP, Ruby on Rails, and WebDAV. |
| *Xgrid Administration and High Performance Computing Guide* | Set up and manage computational clusters of Xserve systems and Mac computers. |
| *Mac OS X Server Glossary* | Learn about terms used for server and storage products. |

## Viewing PDF Guides Onscreen

While reading the PDF version of a guide onscreen:

- Show bookmarks to see the guide's outline, and click a bookmark to jump to the corresponding section.
- Search for a word or phrase to see a list of places where it appears in the document. Click a listed place to see the page where it occurs.
- Click a cross-reference to jump to the referenced section. Click a web link to visit the website in your browser.

## Printing PDF Guides

If you want to print a guide, you can take these steps to save paper and ink:

- Save ink or toner by not printing the cover page.
- Save color ink on a color printer by looking in the panes of the Print dialog for an option to print in grays or black and white.
- Reduce the bulk of the printed document and save paper by printing more than one page per sheet of paper. In the Print dialog, change Scale to 115% (155% for *Getting Started*). Then choose Layout from the untitled pop-up menu. If your printer supports two-sided (duplex) printing, select one of the Two-Sided options. Otherwise, choose 2 from the Pages per Sheet pop-up menu, and optionally choose Single Hairline from the Border menu. (If you're using Mac OS X v10.4 or earlier, the Scale setting is in the Page Setup dialog and the Layout settings are in the Print dialog.)

You may want to enlarge the printed pages even if you don't print double sided, because the PDF page size is smaller than standard printer paper. In the Print dialog or Page Setup dialog, try changing Scale to 115% (155% for *Getting Started*, which has CD-size pages).

## Getting Documentation Updates

Periodically, Apple posts revised help pages and new editions of guides. Some revised help pages update the latest editions of the guides.

- To view new onscreen help topics for a server application, make sure your server or administrator computer is connected to the Internet and click "Latest help topics" or "Staying current" in the main help page for the application.
- To download the latest guides in PDF format, go to the Mac OS X Server documentation website:

  www.apple.com/server/documentation

## Getting Additional Information

For more information, consult these resources:

- *Read Me documents*—important updates and special information. Look for them on the server discs.
- *Mac OS X Server website* (www.apple.com/server/macosx)—gateway to extensive product and technology information.
- *Mac OS X Server Support website* (www.apple.com/support/macosxserver)—access to hundreds of articles from Apple's support organization.
- *Apple Discussions website* (discussions.apple.com)—a way to share questions, knowledge, and advice with other administrators.
- *Apple Mailing Lists website* (www.lists.apple.com)—subscribe to mailing lists so you can communicate with other administrators using email.

# Web Technologies Overview

# 1

This chapter helps you to become familiar with web technologies and to understand the major components before setting up your services and sites.

The Web service is a complex suite of tools for the configuration and management of the Apache web server, development of websites, and the integration of an application server with a number of open-source components. It is best to familiarize yourself with the complexities of your system before proceeding.

## Web Technologies Overview

Web technologies offer an integrated Internet server solution. Web technologies—also known as Web service in this guide—are easy to set up and manage, so you don't need to be an experienced web administrator to set up multiple websites and configure and monitor your web server.

Web service is based on Apache, an open source HTTP web server. A web server responds to requests for HTML webpages stored on your site. Open source software gives you the capability to view and change the source code to make changes and improvements. This has led to Apache's widespread use, making it one of the most popular web servers on the Internet today.

Web administrators can use Server Admin to administer Web service without knowing about advanced settings or configuration files. Web administrators proficient with Apache can also administer web technologies using Apache's advanced features.

Because Web service in Mac OS X Server is based on Apache, you add advanced features with plug-in modules. Apache modules let you add support for Simple Object Access Protocol (SOAP), Java, and CGI languages such as Python.

## Key Web Features

Web service consists of the following key components (web technologies), which provide a flexible and scalable server environment.

- Apache Web Server
- WebDAV
- CGI Support
- SSL Support
- Dynamic Content with Server Side Includes (SSI)
- Blogs and RSS Support

## Apache Web Server

Apache is an open source HTTP web server that administrators configure using Server Admin.

Apache has a modular design, and the set of modules enabled by default is adequate for most uses. Server Admin controls a few optional modules. Experienced Apache users can add or remove modules and change the server code. For information about modules, see "Apache Web Module Overview" on page 99.

Apache v1.3 is installed in earlier versions of Mac OS X Server. If you are doing a clean installation, use Apache 2. Automatic migration from Apache1 to Apache 2 is a supported feature of the Mac OS X Server v10.5.

> *WARNING:* There are possible side-effects of the Apache 1 to Apache 2 conversion script, particularly for security-related settings, which can impact the security of your upgrade.

## WebDAV

Web-based Distributed Authoring and Versioning (WebDAV) is particularly useful for updating content on a website. Users who have WebDAV access to the server can open files, make changes or additions, and save those revisions. On Mac OS X, users can mount WebDAV volumes and access them seamlessly from the Finder.

For more about using WebDAV for file sharing, see "Using WebDAV to Share Files" on page 55.

## CGI Support

Common Gateway Interface (CGI) scripting provides a means of interaction between the server and clients. For example, CGI scripts let you place an order for a product offered on a website or submit responses to information requests.

It is possible to write CGI scripts in several scripting languages, including Perl and Python. The folder /Library/WebServer/CGI-Executable is the default location for CGI scripts.

### SSL Support

Web service includes support for Secure Sockets Layer (SSL), a protocol that encrypts information being transferred between client and server. SSL works with a digital certificate that provides a certified identity for the server by establishing a secure, encrypted exchange of information.

### Dynamic Content with Server Side Includes (SSI)

Server Side Includes (SSI) provide a method for using the same content on multiple pages in a site. They also can tell the server to run a script or insert specific data into a page. This feature makes updating content much easier, because you revise information in only one place and the SSI command displays that revised information about many pages.

For more information about SSI, see "Enabling Server Side Includes (SSI)" on page 52.

### Blogs and RSS Support

The web server provides blogs as an option for each website. The blogs comply with RSS and Atom XML standards and permit Open Directory authentication. Blog users can choose from several techniques for working with templates and style sheets.

*Important:* To make service access control list (SACL) changes to blog service, it is necessary to use the server interface and not the web interface.

For information about setting access control for blogs using SACLs, see "Setting Blog SACL Permissions for Users" on page 76.

## Before You Begin

This section provides information you need before you set up your web server for the first time. Read this section even if you are an experienced web administrator. Some features and behaviors might be different from what you expect.

### Selecting a Version of Apache

With a clean installation, Apache v2.2.4 will be installed. With an upgrade installation, you start with v1.3 but can move to v2.2.4 when you are ready to do so.

### Configuring Your Web Server

You use Server Admin to set up and configure most features of your web server. If you are an experienced Apache administrator and need to work with features of the Apache web server that aren't included in Server Admin, change the relevant configuration files.

However, Apple does not provide technical support for modifying Apache configuration files. If you alter a file, be sure to make a backup copy first. Then revert to the copy if you have problems.

## Providing Secure Transactions

If you want to provide secure transactions on your server, you must set up SSL protection. SSL lets you send encrypted, authenticated information across the Internet. For example, if you want to authorize credit card transactions through your website, you can use SSL to protect the information that's passed to and from your site.

*Important:* You can't use the performance cache for a website if SSL is enabled for that site.

For instructions on how to set up secure transactions, see "Enabling Secure Sockets Layer (SSL)" on page 43.

## Setting Up Websites

Before hosting a website, you must:

- Register your domain name with a domain name authority
- Create a folder for your website on the server
- Create a default page in the folder for users to see when they connect
- Verify that DNS is properly configured if you want clients to access your website by name

When you are ready to publish, or enable, your site, use Server Admin. The Sites pane, located within Web service, lets you add a new site and select a variety of settings for each site you host.

For more information about using WebDAV for file sharing, see "Website Management" on page 49.

## Hosting More Than One Website

You can host more than one website simultaneously on your web server. Depending on how you configure your sites, they might share the same domain name, IP address, or port. The unique combination of domain name, IP address, and port identifies each separate site.

Your domain names must be registered with a domain name authority such as InterNIC. Otherwise, the website associated with the domain won't be visible on the Internet. (There is a fee for each extra name you register.)

For more information about multiple sites, see "Managing Multiple Sites on One Server" on page 56.

For more information about WebDAV, see "Understanding WebDAV" on page 19.

For more information about MIME formats, see "Understanding Multipurpose Internet Mail Extension (MIME)" on page 20.

## Understanding WebDAV

If you use WebDAV to provide live authoring on your website, you must create realms and set access privileges for users. Each site you host can be divided into a number of realms, each with its own set of users and groups that have browsing or authoring privileges.

### Setting WebDAV Privileges

The Apache process running on the server must have access to the website's files and folders. To provide this access, Mac OS X Server installs a user named www and a group named www in the server's Users & Groups List. The Apache processes that serve webpages run as the www user and as members of the www group.

You must give the www group Read access to files in websites so the server can transfer the files to browsers when users connect to the sites. The Apache process runs with an effective user id and group id of www and needs access to the files and directories in the WebDAV realm and in the /var/run/davlocks/ folder.

### Understanding WebDAV Security

In Mac OS X Server v10.5, WebDAV lets you use a web server as a file server. Clients use their browsers from multiple locations, on many types of computers, to access and share files on the server. For more information about using WebDAV for file sharing, see "Using WebDAV to Share Files" on page 55.

WebDAV also lets users update files on a website while the site is running. When WebDAV is enabled, the web server must have write access to the files and folders in the site users are updating.

Both features of WebDAV—providing a file server with browser access, and website updating—have significant security implications when other sites are running on the server, because individuals responsible for one site might be able to change other sites.

You can avoid this problem by carefully setting access privileges for the site files using the File Sharing pane of Server Admin. Mac OS X Server uses a predefined group www, which contains the Apache processes. You must give the www group Read & Write access to files on the website. You also need to assign these files Read & Write access by the website administrator (Owner) and No Access to Everyone. For more information, see *File Services Administration*.

## Defining Realms

When you define a realm, which is typically a folder (or file system), the access privileges you set for the realm apply to all contents of that folder. If a new realm is defined for a folder in the existing realm, only the new realm privileges apply to that folder and its contents. For information about creating realms and setting access privileges, see "Using Realms to Control Access" on page 40.

***Note:*** When an assigned user or group possesses fewer permissions than the permissions that have been assigned to user Everyone, that user or group is deleted upon a refresh. This happens because the access assigned to Everyone preempts the access assigned to specific users or groups with fewer permissions than those possessed by Everyone.

The greater permissions always take precedence. Consequently, the list of assigned users and groups with fewer permissions are not saved in the Realms pane upon refresh if their permissions are determined to be preempted by the permissions assigned to Everyone. After the refresh the names are no longer listed in the list on the right in the Realms pane. Also, for a brief period of time, user Everyone will switch its displayed name to "no-user."

## Understanding Multipurpose Internet Mail Extension (MIME)

Multipurpose Internet Mail Extension (MIME) is an Internet standard for specifying what happens when a web browser requests a file with certain characteristics. You can choose the response you want the web server to make based on the file's suffix. Your choices depend partly on what modules you have installed on your web server. Each combination of a file suffix and its associated response is known as a *MIME type mapping*.

### MIME Suffixes

A *suffix* describes the type of data in a file. Here are some examples:

- txt for text files
- cgi for Common Gateway Interface files
- gif for GIF (graphics) files
- php for PHP: Hypertext Preprocessor (embedded HTML scripts) used for Webmail, and so on
- tiff for TIFF (graphics) files

Mac OS X Server includes a default set of MIME type suffixes. This set includes all the suffixes in the mime.types file distributed with Apache, with a few additions. If a suffix you need is not listed or does not have the behavior you want, use Server Admin to add the suffix to the set or to change its behavior.

***Note:*** Do not add or change MIME suffixes by editing configuration files.

## Web Server Responses (Content Handlers)

When a file is requested, the web server handles the file using the response specified for the file's suffix. Responses, also known as content handlers, can be either an action or a MIME type. Likely responses include:

- Return file as MIME type (you enter the mapping you want to return)
- Send-as-is (send the file exactly as it exists)
- Cgi-script (run a CGI script you designate)
- Imap-file (generate an IMAP mail message)
- Mac-binary (download a compressed file in MacBinary format)

MIME type mappings are divided into two subfields separated by a forward slash, such as text/plain.

Mac OS X Server includes a list of default MIME type mappings. You can edit these and add others using Server Admin.

When you specify a MIME type as a response, the server identifies the type of data requested and sends the response you specify. For example, if the browser requests a file with the suffix "jpg," and its associated MIME type mapping is image/jpeg, the server knows it needs to send an image file and that its format is JPEG. The server doesn't need to do anything except serve the data requested.

Actions are handled differently. If you've mapped an action to a suffix, your server runs a program or script, and the result is served to the requesting browser. For example, if a browser requests a file with the suffix "cgi," and its associated response is the action cgi-script, your server runs the script and returns the resulting data to the requesting browser.

# Working with Web Service

# 2

This chapter shows you how to use Server Admin to set up Web service and to manage web settings and components.

Mac OS X Server combines the latest open source and standards-based Internet services in a complete, easy-to-use web hosting solution. Use Server Admin to configure Web service and set up web components depending on your organization's needs.

## Setup Overview

Here is an overview of the basic steps for setting up Web service.

**Step 1: Read "Before You Begin"**
For issues you should consider before setting up Web service on your network, read "Before You Begin" on page 17.

**Step 2: Turn Web service on**
Before configuring, Web service must be turned on. See "Turning Web Service On" on page 24.

**Step 3: Configure web general settings**
Configure General settings to set connection settings and enable Tomcat. See "Configuring General Settings" on page 24.

**Step 4: Configure web MIME types**
Using MIME types you can set up how your web server responds when your browser requests certain file types. See "Configuring MIME Types Settings" on page 25.

**Step 5: Configure web proxy settings**
Use proxy settings to enable a proxy that sends requests to and from the web server. See "Configuring Proxy Settings" on page 27.

**Step 6: Configure web modules**
Use modules settings to select or deselect which web modules are available for the web server. See "Configuring Modules Settings" on page 28.

**Step 7: Configure web services**
Use web service settings to set up common settings shared between wikis, blogs, web calendars, and web based mailing list archives for groups. See "Configuring Web Services Settings" on page 29.

**Step 8: Start Web service**
After you configure Web service, start the service to make it available. See "Starting Web Service" on page 29.

## Turning Web Service On

Before you can configure Web settings, you must turn on web service in Server Admin.

**To turn Web service on:**

1. Open Server Admin and connect to the server.
2. Click Settings, then click Services.
3. Select the Web checkbox.
4. Click Save.

## Setting Up Web Service

Use Server Admin to change Web service settings. The following sections describe the tasks for configuring and starting Web service.

There are five groups of settings on the Settings pane for Web service in Server Admin:

- **General.** Set Web service connection and spare server settings.
- **MIME Types.** Set up multipurpose internet mail extension (MIME) types and content handlers.
- **Proxy.** Configure proxy settings for the web server.
- **Modules.** Select which web modules are available for Web service.
- **Web Services.** Configure settings common Web services that are hosted on any site.

The following sections describe how to configure these settings, and a final section tells you how to start Web service when you finish.

### Configuring General Settings

You use the General settings pane in web service to configure Web server connection settings, spare server settings, and to enable or disable Tomcat.

For more information on web server connection settings, see "Performance Tuning" on page 32.

**To configure Web service General settings:**

1. Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Settings, then click General.

5 Enter the maximum simultaneous connections.

The default setting is 1024 connections.

This is the number of concurrent connections that are allowed to access your web server.

6 Enter the time in seconds for the connection timeout.

The default setting is 300 seconds.

This is the length of time before a connection to your web server times out. This happens when a user is viewing web pages but not interacting with the site.

7 Enter the number of minimum and maximum spare servers.

Spare server settings regulate the creation of idle spare server processes.

For maximum spare servers, if more than the maximum number of spare servers are idle, the server stops adding spare servers beyond the maximum limit.

For minimum spare servers, if there are fewer than the minimum spare servers required, the server adds spare servers at a rate of one per second.

8 Enter the number of servers to start.

This is the number of spare servers that get created at startup.

9 For your site to permit persistent connections, select the Allow Persistent Connections checkbox and configure the persistent connection settings:

Set the Maximum persistent connections. The default is 500 connections.

Set the Persistent connection timeout length in seconds. The default is 15 seconds.

10 Select the Enable Tomcat checkbox to turn Tomcat on.

11 Click Save.

## Configuring MIME Types Settings

MIME is an Internet standard for specifying what happens when a web browser requests a file with specific characteristics. The MIME Types pane in Server Admin lets you set up how your web server responds when a browser requests certain file types.

Content handlers are similar and also use suffixes to determine how a file is handled. The file suffix describes the type of data in the file. Each suffix and its associated response (such as text/plain and text/richtext) are known as a MIME type mapping or a content handler mapping.

The server includes the MIME type in its response to a browser to describe the information being sent. The browser can then use its list of MIME preferences to determine how to handle the information.

The server's default MIME type is text/html, which specifies that a file contains HTML text.

The web server is set up to handle the most common MIME types and content handlers. You can add, edit, or delete MIME type and content handler mappings. In Server Admin, these files are displayed in two lists: MIME Types and Content Handlers. You can edit items in each list and add or delete items in either list.

**To configure MIME Types settings:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Settings, then click MIME Types.
5. Add, delete, or edit MIME Type mappings.

   Click the Add (+) button to add a mapping to the MIME Types list. Enter each part of the name (separated by a slash), then double-click "new" in the Suffixes list and enter a suffix name. Use the Add (+) or Delete (–) button (next to the Suffixes list) to add or delete suffixes in the Suffixes list. Then click OK.

   To delete a MIME Type mapping, select it from the MIME Types list and click the Delete (–) button.

   To edit a MIME Type mapping, select the mapping from the MIME Types list and click the Edit (/) button. Make your changes to the mapping, then click OK.
6. Add, delete, or edit Content Handlers mappings.

   Click the Add (+) button to add a mapping to the Content Handlers list. Enter the name, then double-click "new" in the Suffixes list and enter a suffix name. Use the Add (+) or Delete (–) button (next to the Suffixes list) to add or delete suffixes in the Suffixes list. Then click OK.

   To delete a Content Handlers mapping, select it from the Content Handlers list and click the Delete (–) button.

   To edit a Content Handlers mapping, select the mapping from the Content Handlers list and click the Edit (/) button. Make your changes to the mapping, then click OK.

***Note:*** If you add or edit a handler that has a Common Gateway Interface (CGI) script, make sure you have enabled CGI execution for your site in the Options pane of the Sites pane.

7 Click Save.

## Configuring Proxy Settings

You use the Proxy settings pane in Web service to configure a forward proxy. A forward proxy is located between the web server and client browsers and passes requests for information between clients and server. The client must be configured to use the forward proxy to access other sites.

A forward proxy is commonly used to provide Internet access to internal client computers that are restricted by a firewall. A forward proxy lets users verify a local server for frequently used files. A forward proxy can be used to block access to specific sites for internal clients and can improve performance.

You can also use a forward proxy to speed response times and reduce network traffic. The proxy stores recently accessed files in a cache on your web server. Browsers on your network verify the cache before retrieving files from more distant servers.

For additional security you should restrict access to your server by setting up this forward proxy. This is particularly true if your server hosts internal and external websites. If your web server is set up to act as a proxy, you can prevent the server from caching objectionable websites.

***Important:*** To take advantage of this feature, client computers must specify your web server as their proxy server in their browser preferences.

When setting up a forward proxy, make sure you create and enable a website for the proxy. You might want to disable logging on the proxy site or configure the site to record its access log in a separate file from your other sites' access logs. The site does not need to be on port 80 but setting up web clients is easier if it is because browsers use port 80 by default.

Mac OS X Server v10.5 provides forward and reverse proxy. The reverse proxy is configured in the Web service Sites pane. For information about setting up a reverse proxy, see "Setting Up a Reverse Proxy" on page 47.

**To configure Web service forward proxy settings:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Settings, then click Proxy.

5 Select the Enable Forward Proxy checkbox.

If a forward proxy server is enabled, each site on the server can be used as the proxy.

6 Select the Control Access To Proxy checkbox to limit access and then enter the domain name that is permitted access in the "Allowed Domain" field.

Generally, when limiting who can use your web server as a proxy, limit access to a specific domain. Users in that domain obtain access.

7 In the Cache Folder field, enter the pathname for the cache folder.

You can also click the Browse button and browse for the folder you want to use.

If you are administering a remote server, File service must be running on the remote server to use the Browse button.

If you change the folder location from the default, you must select the new folder in Finder. Choose File > Get Info, and change the owner and group to www.

8 Set the disk cache target size and set an interval for emptying the cache.

When the cache reaches this size, the oldest files are deleted from the cache folder.

9 To add a host to block, click the Add (+) button and enter its URL.

Add the names of all hosts you want to block.

You can import a list of websites by dragging the list to the list of blocked hosts. The list must be a text file with the host names separated by commas or tabs (also known as csv and tsv strings). Make sure the last entry in the file is terminated with a carriage return/line feed; otherwise, it is overlooked.

10 Click Save.

## Configuring Modules Settings

You use the Modules settings pane in Web service to configure the web modules your server will use.

The Web service in Mac OS X Server is modular. This means that administrators have more flexibility in the web technologies that are added to the service. For more information on web modules, see "Working with Web Modules" on page 99.

**To configure Web service modules settings:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Settings, then click Modules.

5 Select the Enable checkbox next to each module that you want the server to use.

For information on how to add, change, or delete modules, see "Working with Web Modules" on page 99.

6 Click Save.

## Configuring Web Services Settings

You use the Web Services settings pane in Web service to configure common web server settings that are hosted on any site.

Web services include wikis, blogs, web calendars, and web-based mailing list archives for groups. These services are independently enabled for each website you host.

**To configure Web service settings for your server:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Settings, then click Web Services.
5. In the Data Store field, enter the folder where Web service will store information.

   The default folder is /Library/Collaboration/. Click Choose to browse for a different folder.
6. In the Maximum attachment size field, enter the maximum attachment size for files that can be attached to the web services.

   The default file size is 50 MB.
7. From the Default Wiki and Blog Theme pop-up menu, choose the theme for your wiki.

   A theme controls the appearance of a wiki and blog. Themes determine the color, size, location, and other attributes of wiki and blog elements. Each theme is implemented using a style sheet. The default theme is used when a wiki or blog is initially created, but blog owners can change the theme. For more information, see "Customizing Wiki Themes and Layouts" on page 71.
8. Click Save.

## Starting Web Service

You start Web service from Server Admin. When you make configuration changes to Web service and you save your changes, the web server is restarted, causing those changes to be recognized by the httpd process.

**To start Web service:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Start Web (below the Servers list).

The service runs until you stop it and restarts if your server is restarted.

### From the Command Line

You can also start Web service using the `serveradmin` command in Terminal. For more information, see the Web service chapter of *Command-Line Administration*.

## Managing Web Service

This section describes typical day-to-day tasks you might perform after you set up Web service on your server. Initial setup information appears in "Setting Up Web Service" on page 24.

For more information about Website management, see "Website Management" on page 49.

### Checking Web Service Status

Use Server Admin to check the status of Web service.

**To view Web service status:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. To see information such as whether the service is running, when it started, Apache Server version, the number of requests per second, and server throughput, click Overview.
5. To review access and error logs, click Logs.

   To choose which log to view, select the logs in the list. The corresponding log appears below.
6. To see graphs of connected users or throughput, click Graphs.

   Use the pop-up menus to choose which graph to view and the duration of time to graph data for.
7. To see a list of websites, click Sites.

   The list includes the domain name, address, port, and whether the site is enabled.

### From the Command Line

You can also view the status of Web service by using the `ps` or `top` command in Terminal, or by looking at the log files in the /Library/Logs/wikid/ or /var/log/apache2/ folder using the `cat` or `tail` command. For more information, see the File services chapter of *Command-Line Administration*.

## Viewing Web Service Logs

Use Server Admin to view the error and access logs for Web service, if you have enabled them. Web service in Mac OS X Server uses the standard Apache log format, so you can also use a third-party log analysis tool to interpret the log data.

**To view logs:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Logs, then choose between an access or error log by selecting the log from the list of logs.

   To search for specific entries, use the Filter field in the lower right.

### From the Command Line

You can also view Web service logs in the /Library/Logs/wikid/ or /var/log/apache2/ folder by using the `cat` or `tail` command in Terminal. For more information, see the Web service chapter of *Command-Line Administration*.

## Viewing Web Graphs

Use Server Admin to view Web service graphs.

**To view web graphs:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. To see graphs of connected users or throughput, click Graphs.

   To choose which graph to view and the duration of time to graph data for, use the pop-up menus.
5. To update the data in the graphs, click the Refresh button (below the Servers list).

## Stopping Web Service

Use Server Admin to stop Web service. This disconnects all users, so connected users may lose unsaved changes in open files.

**To stop Web service:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Stop Web (below the Servers list).

#### From the Command Line

You can also stop Web service immediately using the `serveradmin` command in Terminal. For more information, see the Web services chapter of *Command-Line Administration*.

## Performance Tuning

You can limit the period of time that users are connected to the server. You can also specify the number of connections to websites on the server at one time.

### Setting Simultaneous Connections for the Web Server

You can specify the number of simultaneous connections to your web server. When the maximum number of connections is reached, new requests receive a message that the server is busy.

Simultaneous connections are concurrent HTTP client connections. Browsers often request several parts of a webpage at the same time, and each request creates a connection. As a result, a high number of simultaneous connections can be reached if the site has pages with multiple elements and many users are trying to reach the server at one time.

**To set the maximum number of connections to your web server:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Settings, then click General.

5 Enter a number in the "Maximum simultaneous connections" field.

The range for maximum simultaneous connections is 1 to 1024. The default is 500, but you can set the number higher or lower, taking into consideration the desired performance of your server.

6 Enter the time in seconds for the Connection timeout.

The default is 300 seconds.

This is the length of time before a connection to your web server times out. This happens when a user is viewing web pages but not interacting with the site.

7 Enter the number of minimum and maximum spare servers.

The spare server settings regulate the creation of idle spare server processes.

For maximum spare servers, if more than the maximum number of spare servers are idle, the server stops adding spare servers beyond the maximum limit.

For minimum spare servers, if there are fewer than the minimum spare servers required, the server adds spare servers at a rate of one per second.

8 Enter the number of servers to start.

This is the number of spare servers that get created at startup.

9 Click Save.

## Setting Persistent Connections for the Web Server

You can set up your web server to respond to multiple requests from a client computer without closing the connection each time. Repeatedly opening and closing connections isn't efficient and decreases performance.

Most browsers request a persistent connection from the server, and the server keeps the connection open until the browser closes the connection. This means the browser is using a connection even when no information is being transferred. The Apache documentation refers to persistent connects as Keep-Alive connections.

You can authorize more persistent connections—and avoid sending a Server Busy message to other users—by increasing the number of authorized persistent connections.

*Important:* Persistent connections are not compatible with the performance cache.

**To set the number of persistent connections:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Settings, then click General.

5 Select Allow Persistent Connections if it is not selected.

6 Enter a number in the "Maximum persistent connections" field.

The range for maximum persistent connections is 1 to 2048.

7 Click Save.

Web service restarts when you save the changes.

## Setting a Connection Timeout Interval

You can specify a time period after which the server can drop a connection that is inactive.

**To set the connection timeout interval:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Settings, then click General.
5. In the "Persistent connection timeout" field, enter a number to specify the amount of time that can pass between requests before the session is disconnected by the web server.

   The range for connection timeout is 0 to 9999 seconds.
6. Click Save.

# Creating and Managing Websites 3

## This chapter helps you create and manage websites that are hosted on your web server.

With Web service configured and your web server running, you can create websites. You create and modify websites on your server with Server Admin. Creating a website establishes the framework that you use to provide web hosted content in various formats.

## Website Setup Overview

Here is an overview of the basic steps for setting up a website.

**Step 1: Configure your web server**

The default configuration works for most web servers that host a single website, but you can configure all basic features of Web service and websites using Server Admin. For more information, see Chapter 2, "Working with Web Service."

For more advanced configuration options, see Chapter 6, "Working with WebObjects and Open Source Applications."

To host user websites, you must configure at least one website.

**Step 2: Set up the web folder**

When your server software is installed, a folder located at /Library/WebServer/Documents/ is set up in the file system. Put items you want to make available through a website in the web folder. You can create subfolders in the web folder to organize the information, and it is generally recommended that you do so if you create additional virtual hosts.

In addition, each registered user has a Sites folder in the user's home folder. Graphics or HTML pages stored in the user's Sites folder are served from http://server.example.com/~username/.

For more information, see "Setting Up the Web Folder" on page 37.

**Step 3: Assign privileges for your website**

The Apache processes that serve webpages must have Read access to the files and Read/Execute access to the folders. (In the case of folders, Execute access means the ability to read the names of files and folders contained in that folder.)

Those Apache processes run as user www—a special user created for Apache when Mac OS X Server is installed. User www is a member of group www, so for the Apache process to access the content of the website, the files and folders must be readable by user www.

You must give group www at least Read-Only access to files in your website so it can transfer those files to browsers when users connect to the site. This applies to all parent folders as well. In other words, the folder containing your web content and the folder containing that folder, and so on, must be readable and searchable by user or group www.

You can do this by:

- Making the files and folders readable and searchable by everyone regardless of their user or group ownership.
- Making group www the owner of files and folders and making sure that the files and folders are readable and searchable by the owner.
- Making group www the owner of files and folders and making sure the files and folders are readable and searchable by the group.
- Making sure the files and folders are readable and searchable by everyone (world), regardless of their ownership and group settings. This is the default case.

For information about assigning privileges, see *File Services Administration*.

**Step 4: Create the website**

Use Server Admin to create a website. After the site is created, configure the settings for your network environment and web requirements. For details, see "Creating a Website" on page 38.

**Step 5: Set the default page**

When users connect to your website, they see the default page. When you first install the software, the file index.html in the Documents folder is the default page. Replace this file with the first page of your website and name it index.html.

To name the file something else, add that name to the list of default index files and move its name to the top of the list in the General pane of the site settings window of Server Admin. For instructions about specifying default index file names, see "Setting the Default Webpage" on page 39.

**Step 6: (Optional) Configure website Apache options**

Use the Sites Options pane to configure Apache web options. For details, see "Configuring Website Apache Options" on page 39.

**Step 7: (Optional) Creating realms to control website access**
You can create a realm to control access to locations or folders in a website. Use the Sites Realms pane to configure your website realms. For details, see "Using Realms to Control Access" on page 40.

**Step 8: Enable website access and error logs**
Use the Logging pane in the Sites pane to enable access and error logs for your website. For details, see "Enabling Access and Error Logs for a Website" on page 42.

**Step 9: (Optional) Enable SSL**
Use the Security pane in the Sites pane to enable SSL for your website. For details, see "Enabling Secure Sockets Layer (SSL)" on page 43.

**Step 10: (Optional) Creating website aliases and redirects**
Use the Aliases pane in the Sites pane to configure website aliases and redirects. For details, see "Managing Access to Sites Using Aliases" on page 45.

**Step 11: (Optional) Set up a reverse proxy**
Use the Proxy pane in the Sites pane to configure a reverse proxy for your website. For details, see "Setting Up a Reverse Proxy" on page 47.

**Step 12: (Optional) Enable optional website features**
Use the Web Services pane in the Sites pane to enable optional web services. For details, see "Enabling Optional Web Services" on page 48.

**Step 13: Connect to your website**
To make sure the website is working properly, open your browser and try to connect to your website over the Internet. If your site isn't working correctly, see Chapter 8, "Solving Web Service Problems," on page 107.

## Setting Up Your Website

The following sections provide instructions for setting up your website.

### Setting Up the Web Folder

To make files available through a website, put the files in the web folder for the site. To organize the information, you can create subfolders inside the web folder. The folder is located at /Library/WebServer/Documents/.

In addition, each registered user has a Sites folder in the user's home folder. Graphics or HTML pages stored here are served from http://server.example.com/~username/.

**To set up the web folder for your website:**

1 Open the web folder on your web server.

By default, the documents folder is located at /Library/WebServer/Documents/.

2 Replace the index.html file with the main page for your website.

Make sure the name of your main page matches the default document name you set in the Sites General pane. For details, see "Setting the Default Webpage" on page 39.

3 Copy files you want available on your website to the web folder.

## Creating a Website

Use Server Admin to create a website framework. This allows content from the web folder to be hosted by your web server. Before you can create a website, you must produce the content for the site and set up your site folders.

**To create a website:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Sites, then click the Add (+) button to add a new site.

5 In the Sites General pane, enter the fully qualified DNS name of your website in the Domain Name field.

***Note:*** You can leave the domain name blank and the IP address set to "any" and the site remains operational.

6 Enter the IP address and port number for the site.

The default port number is 80. If you are using SSL, the port is 443. Make sure the number you choose is not in use by another service on the server.

To enable your website on the server, the website must have a unique name, IP address, and port number combination. For more information see "Hosting More Than One Website" on page 18.

> ***WARNING:*** Do not try to access the server through the direct ports. Instead, allow your access to be proxied through Apache as it is set up. For instance, Server Admin provides no obvious way to configure wikis, and will return the xmlrpc error. In addition, do not access the wiki server on port 8086 or 8087.

7 Enter the path to the folder you set up for this website.

You can also click the Choose button and browse for the folder you want to use.

8 In the Error Document field, enter the page you want to appear when a web page error occurs.

9 (Optional) In the Administrator Email field, enter the administrator mail address.

The server sends website error messages to this mail address.

10 Click Save.

## Setting the Default Webpage

The default page appears when a user connects to your website by specifying a folder or host name instead of a file name.

You can have more than one default page (known as a default index file in Server Admin) for a website. If multiple index files are listed for a website, the web server uses the first one listed in the web folder for that website.

**To set the default webpage:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Sites, then select the website in the list.

5 Click General below the websites list.

6 At the right of the Default Index Files list, click the Add (+) button and enter a name (but do not use spaces in the name.)

A file with this name must be in the web folder.

7 To set the file as the default page the server displays, drag that file to the top of the list.

8 Click Save.

***Note:*** If you plan to use only one index page for a site, you can leave index.html as the default index file and change the content of the existing file with that name in /Library/WebServer/Documents/.

## Configuring Website Apache Options

The default page appears when a user connects to your website by specifying a folder or host name instead of a file name.

**To configure website Apache options:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Sites, then select the website in the list.

5 Click Options below the websites list.

6 Select any of the following Apache options your website requires:

**Folder Listing:** Displays a list of folders when users specify the URL and no default webpage (such as index.html) is present. Instead of viewing a default webpage, the server shows a list of the web folder's contents. Folder listings appear only if no default document is found.

**WebDAV:** Turns Web-based Distributed Authoring and Versioning (WebDAV) on, which allows users to make changes to websites while the sites are running. If you enable WebDAV you must also assign access privileges for the sites and for the web folders.

**CGI Execution:** Permits Common Gateway Interface (CGI) programs or scripts to run on your web server. CGI programs or scripts define how a web server interacts with external content-generating programs. For more information, see "Enabling a Common Gateway Interface (CGI) Script" on page 51.

**Server Side Includes (SSI):** Permits SSI directives placed in web pages to be evaluated on the server while the website is active. You can add dynamically generated content to your web pages while the files are being viewed by users. For more information, see "Enabling Server Side Includes (SSI)" on page 52.

**Allow All Overrides:** Instructs Web service to look for additional configuration files inside the web folder for each request.

**Spotlight Searching:** Allows web browsers to search the content of your website. For details on configuring website indexing, see "Creating Indexes for Searching Website Content" on page 52.

7 Click Save.

## Using Realms to Control Access

You can use realms to control access and provide security to locations or folders within a website. Realms are locations at the URL or they are files in the folder that users can view.

If WebDAV is enabled, users with authoring privileges can also change content in the realm. You set up the realms and specify the users and groups that have access to them.

When an assigned user or group possesses fewer permissions than the permissions that have been assigned to user Everyone, that user or group is deleted upon a refresh. This happens because the access assigned to Everyone preempts the access assigned to specific users or groups with fewer permissions than those possessed by Everyone. The greater permissions always take precedence.

Consequently, the list of assigned users and groups with fewer permissions are not saved in the Realms pane upon refresh if their permissions are determined to be preempted by the permissions assigned to Everyone. After the refresh the names are no longer listed in the list on the right in the Realms pane. Also, for a brief period of time, user Everyone will switch its displayed name to "no-user."

**To use a realm to control website access:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Sites, then select the website in the list.
5. Click Realms below the websites list.
6. Click the Add (+) button to create a realm.

   The realm is the part of the website users can access.
7. In the Realm Name field, enter the realm name.

   This is the name users see when they log in to the website.
8. From the Authentication pop-up menu, choose a method of authentication.

   Basic authentication is on by default. Don't use basic authentication for sensitive data because it sends your password to the server unencrypted.

   Digest authentication is more secure than basic authentication because it uses an encrypted hash of your password.

   Kerberos authentication is the most secure because it implements server certificates to authenticate. If you want Kerberos authentication for the realm, you must join the server to a Kerberos domain.
9. Enter the realm location or folder you are restricting access to:

   Choose Location from the pop-up menu and enter a URL to the location in the website that you want to restrict access to.

   Choose Folder from the pop-up menu and enter the path to the folder that you want to restrict access to.

   You can also click the Browse button to locate the folder you want to use.
10. Click OK.
11. Select the new realm and click Add (+) to open the Users & Groups panel.

    To switch between the Users list and the Groups list, click Users or Groups in the panel.
12. To add users or groups to a realm, drag users to the list on the right in the Realms pane.

    When users or members of a group you've added to the realm connect to the site, they must supply their user name and password.
13. Limit realm access to specified users and groups by setting the following permissions using the up and down arrows in the Persmissions column.

    **Browse Only:** Permits users or groups to browse the website.

**Browse and Read WebDAV:** Permits users or groups to browse the website and also read the website files using WebDAV.

**Browse and Read/Write WebDAV:** Permits users or groups to browse the website and also read and write to website files using WebDAV.

**None:** Prevents users or groups from using any permissions.

14 Click Save.

Use the Realms pane to delete a user or group by selecting the name and clicking the Delete (–) button.

## Enabling Access and Error Logs for a Website

When enabled, Web service keeps access and error logs for your website. You can set up error and access logs for individual websites that you host on your server. However, enabling logs can slow server performance.

The access log contains an entry for each access to the website, indicating what page was accessed, by whom, and whether the access was successful, along with other details.

The error log contains information about failed accesses, or various conditions of interest to the administrator. This log prioritizes messages using severity levels ranging from debug to critical. Server Admin can limit the messages logged by the level of severity. By default, messages are logged at a "warning" level threshold.

In addition to per-site logs, there is an access log and an error log for the wikid process, which provides logging for wikis.

Finally, if you upgraded to Mac OS X Server v10.5 from Mac OS X Server v10.3 or Mac OS X Server v10.4, and the Apache Mode was changed from Apache 1 to Apache 2, there will be a Web Service migration log, which details the actions taken by the Apache 1.3 -> 2.2 translation script.

**To enable access and error logs for a website:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Sites, then select the website in the list.

5 Click Logging below the websites list.

6 Select Enable Access Log to enable this log.

7 Set how often you want the Access log to be archived by selecting the "Archive every __ days" checkbox and entering the number of days.

8. In the Location field, enter the path to the folder where you want to store access logs.

   If you are working with multiple websites, you can name separate logs for each website. You might want to include the site domain name in the log name for easy recognition when reviewing logs. If you have only two websites, you might want to use a single log (with the default name the server uses).

   You can also click the Browse button to locate the folder you want to use.

   If you are administering a remote server, File service must be running on the remote server to use the Browse button.

9. From the Format pop-up menu, choose a log format.
10. If necessary, edit the format string.

    ***Note:*** The Help button next to the format string opens the Apache documentation web page (http://httpd.apache.org/docs/mod/mod_log_config.html), which explains parameters for format strings.

11. Set how often you want the Error log to be archived by selecting the "Archive every ___ days" checkbox for the Error log and entering the number of days.
12. In the Error log Location field, enter the path to the folder where you want to store error logs.

    You can also click the Browse button to locate the folder you want to use.

13. Choose the level of error in the Level pop-up menu to set which error message priority gets logged.
14. Click Save.

## Enabling Secure Sockets Layer (SSL)

Secure Sockets Layer (SSL) provides security for a site and its users by authenticating the server, encrypting information, and maintaining message integrity.

SSL is a per-site setting that lets you send encrypted, authenticated information across the Internet. For example, if you want to permit credit card transactions through a website, you can protect the information that's passed to and from that site.

The SSL layer is below application protocols (for example, HTTP) and above TCP/IP. This means that when SSL is operating on the server and on the client computer, all information is encrypted before being sent.

The Apache web server in Mac OS X Server uses a public key-private key combination to protect information. A browser encrypts information using a public key provided by the server. Only the server has a private key that can decrypt that information.

The web server supports SSLv2, SSLv3, and TLSv1. More information about these protocol versions is available at www.modssl.org.

When SSL is implemented on a server, a browser connects to it using the https prefix in the URL, rather than http. The "s" indicates that the server is secure.

When a browser initiates a connection to an SSL-protected server, it connects to a specific port (443) and sends a message that describes the encryption ciphers it recognizes. The server responds with its strongest cipher, and the browser and server then continue exchanging messages until the server determines the strongest cipher that it and the browser can recognize.

The server then sends its certificate (an ISO X.509 certificate) to the browser. This certificate identifies the server and uses it to create an encryption key for the browser to use. At this point a secure connection has been established and the browser and server can exchange encrypted information.

Before you can enable SSL protection for a website, you must obtain the proper certificates. For detailed information about certificates and their management, see *Server Administration*.

**To set up SSL for a website:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Sites, then select the website in the list.
5. Click Security below the websites list.
6. In the Security pane, select Enable Secure Sockets Layer (SSL).

   When you turn on SSL, a message appears, noting that the port is changed to 443.
7. In the Certificate pop-up menu, choose the certificate you want.

   If the certificate is protected by a passphrase, the name of the certificate must match the virtual host name. If the names don't match, Web service won't restart.
8. If you choose Custom Configuration or want to edit a certificate, you might have to do the following:
   a. Click the Edit (/) button and supply the correct information in each field for the certificate.
   b. If you received a ca.crt file from the certificate authority, click the Edit (/) button and paste the text from the ca.crt file in the Certificate Authority File field.

      ***Note:*** The ca.crt file might be required but might not be sent directly to you. This file must be available on the website of the certificate authority.
   c. In the Private Key Passphrase field, enter a passphrase and click OK.
9. Click Save.

10 Confirm that you want to restart Web service.

Server Admin lets you enable SSL with or without saving the SSL passphrase. If you did not save the passphrase with the SSL certificate data, the server prompts you for the passphrase upon restart but won't accept manually entered passphrases. Use the Security pane for the site in Server Admin to save the passphrase with the SSL certificate data. For more information, see "Using a Passphrase with SSL Certificates" on page 53.

## Managing Access to Sites Using Aliases

You can manage access to websites by using aliases and redirect commands. An alias is an alternative name for a website, which can be useful in simplifying the name users must enter to connect to the site. You can have multiple aliases for a single site.

For example, with a host named example.com you might want to provide a server alias named www.example.com.

The Server Admin Sites Aliases panel mixes two types of aliases.

- The top half of the panel is for web server aliases that give an alternate name to the website or virtual host.
- The bottom half of the panel is for URL aliases and redirects, which are more fine-grained.

By default, the Sites Aliases panel lists a Web Server Alias * (wildcard) directive. To perform name-based virtual hosting, remove the wildcard. If you do not remove the wildcard, browsers trying to access your virtual hosts will access the default host instead.

***Note:*** Server aliases and virtual hosts must be DNS names and they must resolve to the IP address of the website.

A redirect command specifies that when users ask for a specific folder or file on a site, their browser is sent to a different location that you designate.

For example, you could set up a redirect so that if the user enters a URL such as www.example.com/images/boats.jpg and the site has an images folder containing the boats.jpg file, the browser gets redirected to www.apple.com.

By default, the Sites Aliases panel lists the following redirects:

- /collaboration - used to provide the CSS required by Apple's wiki and blog pages and default index.html and Spotlight displays
- /icons/ - used to direct browsers to the standard collection of icons shipped with Apache
- /error/ - used to direct browsers to the standard collection of error pages shipped with Apache

The examples below show aliases and redirects.

| Type | Pattern | Path | Description |
|---|---|---|---|
| Alias | /images | /Volumes/Data/imgs | If you make a file system change but don't want to update all image URLs in your HTML files, this instructs www.example.com/images/boat.jpg to take the file from /Volumes/Data/imgs/boat.jpg. |
| Alias Match | ^/(.*)\.gif | /Library/WebServer/Documents/gifs$1.jpg | If you store all gifs in a specific folder but they must be referenced from the web server root, this instructs the alias www.example.com/logo.gif to serve the file located at /Library/WebServer/Documents/gifs/logo.gif. |
| Redirect | /webstore | https://secure.example.com/webstore | This redirects all queries for webstore to the secure server. |
| Redirect Match | (.*)\.jpg | http://imageserver.example.com$1.jpg | If you host static content such as images on a new server, this redirects all requests for files ending in .jpg to a different server. |

Further information and other examples of aliases and redirects are available at http://httpd.apache.org/docs/mod/mod_alias.html.

**To create or edit aliases the site responds to:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Sites, then select the website in the list.
5. Click Aliases below the websites list.
6. To create aliases, click the Add (+) button under the Web Server Aliases list or select an alias and click the Edit button.
7. In the Server Alias field, enter an alias and click OK.
8. To create a redirect, click the Add (+) button under URL Aliases and Redirects list or select a redirect and click the Edit (/) button.
9. Choose one of the following options from the Type pop-up menu.

   **Alias:** Maps from the URL term to a location in the file system.

   **Alias Match:** Maps a regular expression pattern for a path to a location in the file system.

**Redirect:** Maps a URL term to redirect to another server.

**Redirect Match:** Maps a regular expression pattern for a path to redirect to another server.

10 In the Pattern field, enter the pattern for the alias or redirect.

This is the pattern input from the incoming URL.

11 In the Path field, enter the path for the alias or redirect and click OK.

This is the path in the file system or the redirect that gets sent back to the requester.

12 Click Save.

## Setting Up a Reverse Proxy

You set up a reverse proxy using the Proxy pane in the Sites pane of Server Admin. A reverse proxy differs from a forward proxy by appearing to client computers as a normal web server. The client computers make requests to the web server. The reverse proxy then determines the location to send the requests to and returns web content as if it were the web server. Client computers do not need configuration changes to use a reverse proxy.

You can use a reverse proxy to provide Internet users access to a server located behind a firewall. A reverse proxy can also balance network traffic among several back-end servers or provide caching for a slower back-end server. Administrators also use a reverse proxy to bring several servers into the same URL space.

Mac OS X Server v10.5 provides both forward and reverse proxy. The forward proxy is configured in the Web service Settings pane. For information about setting up a forward proxy, see "Configuring Proxy Settings" on page 27.

**To enable reverse proxy:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Sites, then select the website in the list.

5 Click Proxy below the websites list.

6 Select the Enable Reverse Proxy checkbox.

7 In the Proxy Path field, enter the proxy pathname.

8 In the Sticky Session Identifier field, enter a sticky session identifier or choose one from the pop-up menu.

A sticky session identifier is used to bind a user that is browsing your site to the server that the session started on. This keeps users that are browsing a website that is supported by multiple web servers connected to the server that they started with.

9 To add balancer members, click the Add (+) button below the Balancer Members list; enter a Server URL (worker URL) and define its route and load factor, then click OK.

A balancer member is a server (designated by its worker URL) that shares the network traffic generated by website sessions. Multiple balancers share the website traffic by binding and routing a predetermined load to each server. This prevents a single server from being inundated by web traffic and it improves performance.

The route of the worker URL is a value appended to the sticky session ID.

The load factor is a number between 1 and 100 that defines how much load the worker will handle.

10 Add additional balancer members as necessary, depending on your network requirements.

11 Click Save.

## Enabling Optional Web Services

You can enable additional web services such as wikis, blogs, or webmail.

**To enable optional web services:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Sites, then select the website in the list.

5 Click Web Services below the websites list.

6 Select the Webmail checkbox to enable webmail for your website.

Webmail adds mail functionality for each user of your website. For more information about setting up Webmail, see "Configuring Webmail" on page 79.

7 Select the Blog checkbox to enable blogs for your website.

A blog is a chronological journal on your website that is updated with content added by users. For more information, see "Setting Up User and Group Blogs" on page 75.

8 Select the Wiki and blog checkbox to enable group website functionality.

This website functionality makes it easy for groups to create and distribute information in their own shared websites. For details, see "Setting Up a Wiki" on page 64.

9 Select the Web calendar checkbox if you want calendar functionality for your website.

Users can access a group calendar to track meetings and deadlines.

For details, see "Setting Up a Web Calendar" on page 72.

10 Select the Mailing list web archive checkbox if you want mailing list functionality on your website.

A mailing list is a discussion group that uses mass mail to facilitate communication. For details, see "Setting Up Mailing List Web Archives" on page 80.

11 Click the Add (+) button below the Users/Group list to add users and groups that will create wikis on your site.

Select the Moderator checkbox for each user or group that you want to designate as a moderator.

If the list is empty, all users can create wikis.

12 Click Save.

### Connecting to Your Website

After you configure your website, view the site with a web browser to verify that everything appears as intended.

**To connect to your website:**

1 Open a web browser and enter the web address of your server.

You can use the IP address or the DNS name of the server. If SSL is enabled, use "https" in the URL instead of "http."

2 If you are not using the default port, enter the port number.

3 If you've restricted access to specific users, enter a valid user name and password.

> *WARNING:* Do not try to access the server through the direct ports. Instead, allow your access to be proxied through Apache as it is set up. For instance, Server Admin provides no obvious way to configure wikis and will return the xmlrpc error. Do not access the wiki server on port 8086 or 8087.

4 Verify that the website default index page appears.

## Website Management

This section describes typical tasks you might perform after you create a website on your server. Initial website setup information appears in "Setting Up Your Website" on page 37.

### Viewing Website Settings

You can use the Sites pane of Server Admin to see a list of your websites. The Sites pane lists configuration information for each site, including:

- Whether a site is enabled

- The DNS name and IP address for a site
- The port being used for the site

**To view website settings:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Sites, then select the website in the list.

You can view or change the settings for a site by selecting the site in the Sites pane list and clicking a setting pane.

## Changing the Web Folder for a Site

The web folder is used as the root for the site (known as DocumentRoot in Apache). In other words, the default folder is the top level of the file system structure for the site.

**To change the web folder for a site hosted on your server:**

1. Log in to the server you want to administer.

   You need access to the file system on the server.
2. Drag the contents of your previous web folder to your new web folder.
3. Open Server Admin and connect to the server.
4. Click the triangle to the left of the server.

   The list of services appears.
5. From the expanded Servers list, select Web.
6. Click Sites, then select the website in the list.
7. In the website General pane, enter the path to the web folder in the Web Folder field, or click the Browse button and navigate to the new web folder location.
8. Click Save.

## Changing the Access Port for a Website

By default, the server uses port 80 for connections to websites on your server. You might need to change the port used for an individual website (for example, if you want to set up a streaming server on port 80).

Make sure the number you choose does not conflict with ports being used on the server (for FTP, Apple File Service, SMTP, and others). If you change the port number for a website you must change all URLs that point to the web server to include the new port number you choose.

***Note:*** If you turn SSL on for a site, the port for that site is changed to 443. If you turn SSL off, the port changes to 80, regardless of what it was previously. A message on the screen alerts you to the port change when you turn off SSL.

**To set the port for a website:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Sites, then select the website in the list.
5. In the General pane, enter the port number in the Port field.
6. Click Save.

> ***WARNING:*** Do not try to access the server through the direct ports. Instead, allow your access to be proxied through Apache as it is set up. For instance, Server Admin provides no obvious way to configure wikis and will return an xmlrpc error. Do not access the wiki server on port 8086 or 8087.

## Enabling a Common Gateway Interface (CGI) Script

Common Gateway Interface (CGI) scripts (or programs) send information between your website and applications that provide different services for the site.

If a CGI script is to be used by only one site, install the CGI in the Documents folder for the site. The CGI name must end with the suffix ".cgi."

If a CGI script is to be used by all sites on the server, install it in the /Library/WebServer/CGI-Executable folder. In this case, clients must include /cgi-bin/ in the URL for the site (for example, http://www.example.com/cgi-bin/test.cgi).

Make sure the file permissions for the CGI script permit it to be executed by the user www. Because the script typically isn't owned by www, Everyone should be able to execute it.

**To enable a CGI for a website:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Sites, then select the website in the list.
5. In the Options pane, select CGI Execution.
6. Click Save.

***Note:*** Disabling CGIs for a site does not disable CGIs in the CGI-Executables folder.

## Enabling Server Side Includes (SSI)

Enabling Server Side Includes (SSI) permits a block of HTML code or other information to be shared by different webpages on your site. SSIs can also function like CGIs and carry out commands or scripts on the server.

**To enable SSI in Server Admin:**

1. In Server Admin, click Web in the list for the server you want.
2. Click Sites in the button bar.
3. In the Sites pane, click the site in the list.
4. In the Options pane, select Server Side Includes (SSI).
5. Click Save.

## Creating Indexes for Searching Website Content

Use the Sites Options pane in Server Admin to enable Spotlight searching on your website. This turns the mod_spotlight_apple Apache module on, which allows browsers to search through an index of your website content.

The Spotlight indexing mechanism is turned off by default in Mac OS X Server v10.5. Spotlight searching is still enabled, but you must turn indexing on to provide Spotlight-based search capability for Web service.

**To create an index for your website:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Sites, then select the website in the list.
5. In the Options pane, select the Spotlight Searching checkbox.

   This enables mod_spotlight and causes the Spotlight processes to create an index for the Document Root of the website. Several minutes after indexing is enabled for the first time, the index is available and searching is possible.
6. Click Save.
7. Copy the template.spotlight file from /Library/WebServer/Document into DocumentRoot of each virtual host for which you want the Spotlight search to be available.

   You can customize the title, maximum permitted hits, and other aspects of the presentation by modifying a copy of this file.
8. Advise clients to append ".spotlight" to the URL for websites.

An example URL is httpd://vhost1.example.com/.spotlight. This presents a simple search page that searches the contents of DocumentRoot for the website. Results are sorted with the most relevant hits first, although no relevance score is presented.

## Monitoring Website Activity

Use website logs to monitor your website activity and server events. You can configure logs to record events as messages for specific website activity. Website logs are used to track who accesses a website and what errors occur on a website. This information is useful when troubleshooting problems or monitoring malicious activity.

For more information on setting up logs, see "Enabling Access and Error Logs for a Website" on page 42.

**To view website logs:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Logs, then select the log for your website in the list.

   The log messages display below the log list.

   Switch between logs by selecting them in the list.
5. Search the contents of a log by entering a search term in the Filter field located in the lower right corner below the log.

## Using a Passphrase with SSL Certificates

If you manage SSL certificates using Server Admin and you use a passphrase for your certificates, Server Admin ensures that the passphrase is stored in the system keychain.

When a website is configured to use the certificate and that web server is started, the getsslpassphrase(8) utility extracts the passphrase from the system keychain and passes it to the web server, as long as the certificate name matches the virtual host name.

(If you do not want to rely on this mechanism, you can have the Apache web server prompt you for the passphrase when you start or restart it. Use the `serveradmin` command-line tool to configure this.)

**To configure Apache to prompt you for a passphrase when it starts:**

1. Open Terminal and enter the following command.

```
$ sudo serveradmin settings web:IfModule:_array_id:mod_ssl.c:SSL
     PassPhraseDialog=builtin
```

2. Start Apache with the command:

```
$ sudo serveradmin start web
```

3 When prompted, enter the certificate passphrase.

## Using WebDAV to Manage Website Content

WebDAV lets you or your users make changes to websites while the sites are running. With WebDAV, users or groups can collaboratively manage website files and folders. For more information on how WebDAV works, see "Understanding WebDAV" on page 19.

Work with WebDAV as explained in the following sections:

- "Enabling WebDAV on Websites" on page 54
- "Using WebDAV to Share Files" on page 55
- "Configuring Web Content File and Folder Permissions" on page 55

### Enabling WebDAV on Websites

If you enable WebDAV, you must also assign access privileges for the sites and web folders.

**To enable WebDAV for a site:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select Web.

4 Click Sites, then select the website in the list.

5 Click Options below the websites list.

6 Select the WebDAV checkbox.

This option turns WebDAV on, allowing users to make changes to websites while the sites are running. If you enable WebDAV, you must also assign access privileges for the sites and web folders.

***Note:*** If you turned off the WebDAV module in the Modules pane of Server Admin, you must turn it on again before WebDAV takes effect for a site. This is true even if the WebDAV option is selected in the Options pane for the site. For more about enabling modules, see "Apache Web Module Overview" on page 99.

7 Click Save.

After WebDAV is turned on, you can use realms to control access to the website. For more information about configuring realms, see "Using Realms to Control Access" on page 40.

## Using WebDAV to Share Files

You can use WebDAV to permit authorized users to connect to a website and to share files on that site. The steps below provide a brief example of setting up and sharing files using WebDAV.

- Turn on WebDAV for the site in Server Admin. See "Enabling WebDAV on Websites" on page 54.
- Set up realms for the site in Server Admin to control access to the site. See "Using Realms to Control Access" on page 40.

  For example, you could create a folder for shared documents inside the website folder and give specific people Browse and Read/Write access to that folder.
- Tell authorized users how to connect to the site using the WebDAV client built into Mac OS X (or Mac OS X Server).

Users can connect to the website using a WebDAV-enabled application, such as the Finder in Mac OS X, Adobe GoLive, Adobe Dreamweaver, or Microsoft Internet Explorer.

Browsers are not generally WebDAV-enabled, but a browser can access a WebDAV-enabled site and perform read operations (limited by realm permissions configured on the web server), because WebDAV is a superset of HTTP.

Write operations cannot be performed by a web browser. They require a WebDAV client, such as Goliath, or the client built into the Mac OS X file system and typically used through the Finder. For more information about Goliath, see www.webdav.org/goliath.

**To use Finder to connect to a website using WebDAV:**

1. Open Finder.
2. Choose Go > Connect to Server.
3. In the Server Address field, enter the HTTP URL.

   The URL for connecting is http://<serverURL>:<server port>/<folder, or folder where collaborative files are stored>.
4. Click Connect.

***Note:*** To connect from another platform, see the platform-specific documentation for the relevant WebDAV client. Microsoft platforms use an authentication mechanism that can make it difficult or impossible to mount WebDAV volumes from Mac OS X.

## Configuring Web Content File and Folder Permissions

You can use file and folder permissions to control WebDAV access to website content that is located by default in the /Library/WebServer/Documents/ folder.

Mac OS X Server imposes the following constraints on web content files and folders:

- For security reasons, web content files and folders must not be writable by Everyone.

- Web content files and folders are owned by user Root and Group Admin by default, so they are modifiable by an administrator but not by user or group www.
- To use WebDAV, web content files must be readable and writable by user or group www, and folders must be readable, writable, and executable by user or group www.
- If you need to change web content files and folders while you are logged in as an administrator, those files or folders must be modifiable by the administrator.

To use WebDAV you must enable it in Server Admin. When enabled, Server Admin changes the group ownership of the WebDAV folder to www.

If you are using WebDAV and you want to make changes to web content files or folders while logged in as an administrator, you must change the web content file and folder permissions to admin, make your edits, and then restore the file and folder permissions to www.

**To add sites to your web server while using WebDAV:**

1. Change the group privileges of the folder containing your websites to admin.

   The default folder location is /Library/Webserver/Documents/.
2. Add your new site folder.
3. Change the group privileges of the folder containing your websites back to www.

## Managing Multiple Sites on One Server

You can create multiple sites on the same web server, at the same IP address (also referred to as virtual hosts), or at separate, secondary IP addresses (referred to as multihoming).

Virtual hosts are multiple sites on the same server. These sites can be name-based (such as www.example.com) or they can use IP addresses (such as 10.201.42.73). You can use Server Admin to manage name-based and IP-based websites.

If you configure multiple sites on your server using the Sites pane in Server Admin, each site is considered a virtual host. For more information on setting up a site, see "Creating a Website" on page 38.

A multihomed site is a site that has more than one connection to the Internet. Multihoming is typically done to improve reliability and performance. Those multiple connections might be through the same Internet service provider (ISP) or through multiple ISPs, and they might involve multiple IP addresses or one address.

### Using Aliases to Have a Site Respond to Multiple Names

If you want a website to respond to multiple names, choose one name as the primary and add the other names as aliases.

To set up a website this way, use the primary name as the site name in Server Admin (by clicking the site and entering the primary name in the General pane for the site, then adding the other names in the Aliases pane for that site). For the procedure, see "Managing Access to Sites Using Aliases" on page 45.

For example, if you want your website to respond to example.com, www.example.com, and widget.example.com, you could set it up as follows (the names and IP addresses are examples only):

**Primary name:** www.example.com (entered in the General pane for the site)

**Secondary names:** example.com and widget.example.com (entered in the Web Server Aliases list for the site)

Make sure your DNS server aliases your web server address to all three domain names.

### Websites and Multiple Network Interfaces

By default, the web server is configured with a single wildcard website or virtual host. Such a website is useful for these reasons:

- It responds on all network interfaces and on all IP addresses on all those interfaces.
- It responds to the DNS name that maps to one of those addresses.

Other websites can be added using the Sites pane in Server Admin. When websites are added, the administrator can associate a specific IP address or a wildcard address with each website.

If the web server has multiple interfaces and multiple addresses, configuring Apache to use them is a matter of configuring websites to use the desired addresses. An even simpler scenario is to let the wildcard website respond to all addresses, which it does by default.

## User Content on Websites

Mac OS X client has a Web Sharing feature, which allows a user to place content in the Sites folder of his or her home folder and have it visible on the web. Mac OS X Server also has a much broader web service capability, which can include a form of personal web sharing, but there are important differences between Mac OS X client and Mac OS X Server.

### Web Service Configuration

By default, on Mac OS X Server Web service ignores files in the /etc/httpd/users/ folder, and folder listings are not enabled for users. All folder listings in Web service use Apache's FancyIndexing directive, which makes folder listings more readable.

In Server Admin, the Options pane in the Sites pane for each site has a Folder Listing checkbox. This setting enables folder listings for a specific virtual host by adding a "+Indexes" flag to Apache's Options directive for that virtual host. If folder listings are not explicitly enabled for each site (virtual host), file indexes are not shown.

The site-specific settings do not apply outside the site; therefore, site-specific settings do not apply to home directories. If you want users to have folder-indexing capability on their home directories, you must add suitable directives to Apache's configuration files.

For a specific user, you add the following directives inside the <IfModule mod_userdir.c> block in the httpd.conf file:

```
<Directory "/Users/refuser/Sites">
    Options Indexes MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

## Default Content

The default content for the user's Sites folder is an index.html file along with a few images. This index.html file has text that describes the Personal Web Sharing feature of Mac OS X client. The user must replace the index.html file with one suited to the content of his or her Sites folder.

## Accessing Web Content

After the home folder is created, the content of the Sites folder in the user's home folder is visible when Web service is running. If your server is named example.com and the user's short name is refuser, the content of the Sites folder can be accessed at http://example.com/~refuser.

If the user has multiple short names, one name can also be used after the tilde (~) to access that same content.

If the user places a content file named foo.html in his or her Sites folder, that file must be available at http://example.com/~refuser/foo.html.

If the user places multiple content files in his or her Sites folder and cannot change index.html to include links to those files, the user might benefit from the automatic folder indexing described previously. If the "Enable folder listing" setting is enabled, an index listing of file names is visible to browsers at http://example.com/~refuser.

Indexing settings also apply to subfolders placed in the user's Sites folder. If the user adds a content subfolder named Example to the Sites folder and an index.html file is present inside the Example folder, or if folder indexing is enabled for that user's site, the folder is made available to browsers at http://example.com/~refuser/Example.

## Securing Web Content on Case Insensitive File Systems

The recommended practice for serving web content whose access is controlled via the Realm mechanism is to serve it from case-sensitive volumes, such as UFS or HFSX, where a folder named "Protected' and another folder named "PrOtECted" are two different folders.

If you use the default case-insensitive HFS file system to serve access-controlled web content, consider using location-based realms rather than folder-based realms. However, if you need to use folder-based realms on a case-insensitive file system, Apple provides a layer of protection for that scenario, for both Apache 1.3 and Apache 2.2, by using mod_hfs_apple.

The HFS Extended volume format commonly used for Mac OS X Server preserves the case of file names but does not distinguish between a file or folder named "Example" and one named "eXaMpLe." Without mod_hfs_apple, this insensitivity could be an issue when your web content resides on such a volume and you are attempting to restrict access to all or part of your web content using security realms.

If you set up a security realm requiring browsers to use a name and a password for Read-Only access to content in a folder named "Protected," browsers would need to authenticate to access the following URLs:

- http://example.com/Protected
- http://example.com/Protected/secret
- http://example.com/Protected/sECreT

However, they could bypass it by using something like the following:

- http://example.com/PrOtECted
- http://example.com/PrOtECted/secret
- http://example.com/PrOtECted/sECreT

Fortunately, mod_hfs_apple prevents those types of efforts to bypass the security realm, and this module is enabled by default.

***Note:*** mod_hfs_apple operates on folders; it is *not* intended to prevent access to individual files. A file named "secret" can be accessed as "seCREt". This is correct behavior, and does not permit bypassing security realms.

# Creating and Managing Wikis and Blogs

# 4

## This chapter shows you how to use Server Admin to create and manage a wiki and blog on your website.

Mac OS X v10.5 makes it easy for groups to collaborate and communicate through their own wiki-powered intranet website, complete with group calendar, blog, and mailing list archive functions.

Users can create and edit wiki pages, tag and cross-reference material, upload files and images, add comments, and search content with drag-and-drop ease.

## Wiki Overview

Wikis allow you to create project-specific websites for a group. Groups can then collaborate and communicate through wiki-powered intranet websites, complete with a group calendar, blog, and mailing list.

You can select from more than 20 built-in themes with different colors, fonts, and layout styles. You can customize these templates with your own banner image and a custom sidebar title that displays pages with a user-defined tag at initial login.

After you set up your wiki it is easy to add, delete, and edit your content. No syntax or markup knowledge is required—the wiki comes with full drag-and-drop support.

You can insert hyperlinks, link between pages, add images, attach files, and change webpage formatting. Because wikis feature RSS support, group members can be automatically notified when content is added or edited. The wiki maintains a complete change history, so you can always revert to a previous version of your document.

After you create a wiki on your website and give access to group members, everyone can contribute to the site. The group's owner and administrator can grant access controls for viewing and editing.

Users can access a group calendar to track meetings and deadlines or send messages to a mailing list to keep others informed. The blog feature is ideal for brainstorming or commenting on work.

## About Wiki Pages

The following is a list of the wiki pages and their description:

- **Groups page:** Page that links to all groups hosted by the wiki website.
- **Wiki Home page:** The home page of a group's wiki. Contains links to pages, a group calendar, group blog, and search, as well as to the default sidebars (for example, "What's hot" and "Recent Changes").
- **Document pages:** Pages that group members create. They are dynamically linked to and from other webpages and are found by using search and tags.
- **Calendar pages:** Pages that show a group calendar that uses iCal service to provide a shared calendar to group members. For more information on iCal service, see *iCal Service Administration*.
- **Mailing list pages:** Pages that provide a web archive of a group's mailing list traffic. For more information about Webmail, see Chapter 5, "Configuring and Managing Webmail."
- **Blog Pages:** Pages that show a user or group blog. Blogs are created and updated when users or members of the group (that have Read and Write access to a group blog) add comments to the blogs. For more information about blogs, see "Setting Up User and Group Blogs" on page 75.

## About Wiki Security

The level of website security determines the level of wiki security. Wiki security is established when the website that the wiki is configured on is secure.

Methods you can use to help secure data moving to and from your wiki include the following:

- Set up SSL for the website your wiki is running on. SSL provides security for a site and its users by authenticating the server, encrypting information, and maintaining message integrity. For more information, see "Enabling Secure Sockets Layer (SSL)" on page 43.
- Restrict user and groups that can create wiki pages on your website by adding users and groups to the Web services list. For more information, see "Enabling Web Calendar Service for a Website" on page 72.

## About Wiki File and Folder Hierarchy

By default wiki content is stored in the /Library/Collaboration/ folder. This folder can be changed in the Web service Settings Web Services pane in Server Admin.

The following list shows the default wiki file and folder hierarchy. This includes where all wiki files are stored and the folder structure for the wiki content. In the list, *groupname* is the name of the group, *pagename* is the name of the wiki page, and *page* is the name of the webpage.

- /Library/Collaboration/ contains all files for the wiki.

- /Library/Collaboration/Groups/*groupname*/ contains all files for one group's services.
- /Library/Collaboration/Groups/*groupname*/wiki/*pagename*.page/ contains the component files of a wiki page.
- /Library/Collaboration/Groups/*groupname*/wiki/*pagename*.page/*page*.html contains the main text of the wiki (html content).
- /Library/Collaboration/Groups/*groupname*/wiki/*pagename*.page/*page*.plist contains the metadata for the wiki page.
- /Library/Collaboration/Groups/*groupname*/wiki/*pagename*.page/revisions.db contains the version history database for that wiki page.
- /Library/Collaboration/Groups/*groupname*/*pagename*.page/images/ contains the images for that wiki page.
- /Library/Collaboration/Groups/*groupname*/*pagename*.page/attachments/ contains all attachments for that wiki page.

## Wiki Setup Overview

Here is an overview of the basic steps for setting up a wiki.

**Step 1: Configure your web server**

The default configuration works for most web servers that host a single website but you can configure all basic features of Web service and websites using Server Admin. For more information, see Chapter 2, "Working with Web Service."

Before you create and configure wikis or blogs, set up default web services settings. For details, see "Configuring Web Services Settings" on page 29.

**Step 2: Set up your website**

With your web service configured and running, you can create websites. Creating a website establishes the framework that you use to provide web-hosted content in various formats, including wikis and blogs. For details, see Chapter 3, "Creating and Managing Websites."

**Step 3: Enable wiki web services for your website**

To create a wiki, you must enable the wiki web service on your website. For details, see "Enabling Web Calendar Service for a Website" on page 72.

**Step 4: Create groups for the wiki**

After the wiki web service is enabled on your website, you must create groups in Workgroup Manager or Directory and give them access to the wiki web service for the wiki site. For more information about using Directory, see Directory help.

Wikis are provisioned for groups as they are created and enabled for web services in Workgroup Manager. You can modify the Write and View permissions for users within the group for wiki pages. For details, see *User Management*.

**Step 5: Connect to your wiki**

To make sure the wiki is working properly, open your browser and try to connect to it over the Internet. For details, see "Connecting to a Wiki" on page 65.

## Setting Up a Wiki

The following sections provide instructions for setting up a wiki on your website.

### Enabling Wiki Web Services for a Website

You can enable wiki web services on your website. In addition, blogs, calendaring, and mailing list web services are available for your site.

Wiki will not work without a local Open Directory master. Your server can be connected to another directory server simultaneously, but for wiki to work, your server must be an Open Directory master. For more information, see *Open Directory Administration*.

**To enable wiki services on your website:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Sites, then select the website in the list.
5. Click Web Services below the websites list.
6. Select the Blog checkbox to enable user blogs for your website.

   This gives users the ability to create and maintain personal blog pages. A blog is a chronological journal on your website that is updated with content added by users. For more information, see "Setting Up User and Group Blogs" on page 75.
7. Select the "Wiki and blog" checkbox to enable group website functionality.

   This website functionality makes it easy for groups of people to create and distribute information in their own shared websites. This also enables group blogs on your wiki pages.
8. If you want calendar functionality for your website, select the "Web calendar" checkbox.

   Users can access a group calendar to track meetings and deadlines. For details, see "Setting Up a Web Calendar" on page 72.
9. If you want mailing list functionality on your website, select the "Mailing list web archive" checkbox.

   A mailing list is a discussion group that uses mass email to facilitate communication. For details, see "Setting Up Mailing List Web Archives" on page 80.

10 Click the Add (+) button below the Users/Group list to add users and groups who will create wikis on your site, then select the Moderator checkbox for each user or group in the list that you want to designate as a moderator.

If you leave the list empty, all users can create wikis.

11 Click Save.

### Connecting to a Wiki

If wiki web service is enabled on your website you can connect to the wiki.

**To connect to your wiki:**

1 Open a web browser and navigate to website.

2 To access the wiki, click Groups on the top of the webpage.

The available group wikis are listed on the page. As groups are created and enabled, their wikis will appear on this page.

3 Select the wiki you want to connect to.

This opens the wiki page for the group.

## Changing Wiki Settings

If you are the group owner or server administrator, you can change wiki settings such as how wiki pages look and whether readers can add comments to pages. You can also add a sidebar to the homepage for showing specific content.

**To change wiki settings:**

1 Open your wiki homepage.

Log in as a group owner or administrator.

2 Click Settings in the Admin Functions sidebar.

3 Click the Edit (/) button in the toolbar to change the following options.

**Title:** The name of your site.

**Theme:** The theme of your site. Click Choose to change the appearance of your site. The theme is applied dynamically as pages are served to users. If the theme uses a banner image, you can upload a JPEG or PNG image.

**Custom Sidebar:** The title of a custom sidebar on the homepage and the tag that causes items to appear in the sidebar list.

**Comments:** The users, if any, that can post comments to wiki pages. You can also turn on moderation of comments to prevent a comment from appearing until a moderator approves it.

**Podcast:** Whether podcast entries are allowed in the blog, and the category they should appear when a user subscribes to the podcast using iTunes Music Store.

4 Click "save."

## Managing Wiki Pages

This section describes typical day-to-day tasks you might perform after you set up a wiki on your website. Initial wiki setup information appears in "Setting Up a Wiki" on page 64.

### Adding Document Pages

You can add document pages to your wiki from your Internet browser.

**To create a wiki page:**

1 Click "wiki" if you're not already viewing a wiki page, or navigate to the wiki page that you want to add a new page to.

2 Click the New Page (+) button in the toolbar.

3 In the New Page dialog that appears, enter the page title and click Create.

The editing toolbar appears and a new page is created.

4 Delete the sample text and enter your own content.

5 Click "save" in the editing toolbar when you're finished.

In the comment field, you can enter a note about the changes you made. This note appears in the history for the page.

### Editing Document Pages

You can edit wiki pages from your Internet browser.

**To edit a wiki page:**

1 Navigate to the page you want to edit.

2 Click the Edit (/) button in the toolbar.

3 When the content appears, edit it using the tools in the editing toolbar.

4 Click Save in the editing toolbar to save your changes.

When you save changes to a wiki page, you can also enter a comment that will appear in the page history.

Your new content replaces the previous page content.

### Deleting Document Pages

You can delete wiki pages from your Internet browser.

**To delete a wiki page:**

1 Navigate to the page you want to delete.

2 Click the Delete Page (–) button in the toolbar.

3 Click Delete to confirm.

The page disappears but the content is retained so the page can be restored if needed. Administrators and group owners can permanently delete the page.

## Adding a Link to a Wiki Page

You can add hyperlinks that link to other wiki pages or to other websites.

**To add a link to a wiki page:**

1 Navigate to the page you want to add a link to.

2 Click the Edit (/) button in the toolbar.

3 Select the text you want to use as the link text.

4 Click the Link button in the editing toolbar and then choose an option from the pop-up menu that appears:

**New Page:** Links to a wiki page that doesn't exist (you'll be asked to create the page).

**Search:** Searches for a wiki page that contains the link text you selected, or allows you to enter different text to search for.

**Enter URL:** Links to a page on another website. Enter a complete URL.

**Unlink:** Removes the link, if the text you've selected is already linked.

5 To make a link that creates a mail message, select Enter URL from the Link pop-up menu, and then enter a link in this form: mailto:annejohnson@example.com.

6 When you finish, click "save" in the editing toolbar.

## Inserting a Table on a Wiki Page

Use the table editor to insert or delete a table on a wiki page. You can also add or delete table columns, rows, header columns, and header rows, and enter data in table cells.

**To insert a table on a wiki page:**

1 Navigate to the page you want to insert a table on.

2 Click the Edit (/) button in the toolbar to enter edit mode.

3 Position the insertion point where you want to insert the table, then click Insert Table in the toolbar to reveal the table editor.

4 Click OK to add the table to the page.

Press Tab to move from cell to cell.

5 When you finish, click "save" in the editing toolbar.

## Adding Tags to Wiki Pages

Tags lets you identify, categorize, and quickly find related wiki and blog pages. Use tags to group and identify related items and provide an easy-to-use organizational system so everyone contributing to the wiki can keep up with the latest changes and news.

For example, you might add a tag to each page that indicates its department or project. Any user who can edit the site content can add or remove tags. You can search for tagged items to quickly find what you're looking for.

**To add a tag to a page:**

1. Navigate to the page you want to tag.
2. Click the Add Tag (+) button.

   A text field appears.
3. Enter the tag you want to add and then press Return.

   If the tag already exists, select it when it appears, then press Return.
4. Continue adding tags as desired.

   You can also add tags while editing a page, and delete tags that no longer apply.

## Removing Tags from Wiki Pages

If a page has been accidentally or incorrectly tagged, you can remove individual tags.

**To remove a tag from a page:**

1. View the page.

   You cannot remove a tag from the search results, tag view, or history list.
2. Drag the tag you want to delete from the tag bar and then release the mouse button.

   It will disappear in a puff of smoke.

To remove a tag while editing a page, click the small x that appears when you move the pointer near the tag.

## Attaching a File to Wiki Pages

The best way to attach a file for downloading is to create an archive (zip) of the file before uploading it to the server. This is essential if you're attaching a folder of files or complex types of files such as Keynote presentations.

**To attach a file to a page so others can download it:**

1. Navigate to the page you want to attach a file to.
2. Click the Edit (/) button in the toolbar.
3. Position the insertion point where you want the file attachment to appear on the page.
4. Click the Attach File (paperclip) button in the editing toolbar.
5. Click Choose and select the file to attach.

6 Click Attach to upload the file.

When the upload finishes a file download button appears on the page with the name of the file on it.

7 When you're finished editing the page, click "save" in the editing toolbar.

A user can now click the name to download the file.

If you're attaching a media file, such as an image, and you want others to see it without downloading it first, use the Insert Media button instead of the Attach FIle button. The Insert Media button lets you upload QuickTime image or audio files.

## Finding Tagged Wiki Pages

There are two ways to quickly find tagged pages:

- If the page you're viewing has the tag you want to find, click the tag to search for other pages.
- If the page you're viewing is not tagged with what you want to find, click Search (magnifying glass), then choose a tag from the pop-up menu, or choose All Tags to see more selections.

## Searching Wiki Pages

Use the search feature to find items on your wiki.

**To find items in your wiki:**

1 Click the Search (magnifying glass) button.

2 Enter a word or phrase in the text field that appears.

A quick search (find-as-you-type) feature displays a list of pages that have the word or phrase you entered in the title as you type in the search field.

3 If you don't find what you are looking for in the quick search list, press Return to search titles and content.

You will see a list of pages and calendar events that contain the phrase you entered.

After the results appear, you can further refine your search by choosing to show only wiki or blog pages that contain the tags you select. Use the tag checkboxes on the right side of the page.

To view the search results in a different order, use the pop-up menus to choose a new sort order. You can also choose to search all pages or limit the search to just the blog, wiki, calendar events (only the name and location are searched), or mail messages.

## Viewing or Replacing Older or Deleted Wiki Pages

Every time a wiki page is updated, the previous version is retained so you can undo changes or reactivate an older version of a page.

**To view previous versions of a wiki page:**

1. Navigate to the page you want to view older versions of.
2. Click the View Document History (>) button.

   This button is visible only when you're viewing, not editing, a page.
3. Select any version in the list to view its contents.
4. Click the Compare button to compare the selected version with its previous version.
5. Click the View Alone button to stop comparing versions.
6. To replace the current version of a page with an older version, select the version and click Restore.

   This creates a new revision of the page. Attachments and media associated with the older version are also restored.
7. To exit without changing the current version of a page, click the Hide Document History (V) button.

### Restoring Deleted Wiki Pages

Wiki files are stored in a file system, so backup is just like backing up ordinary files. The default files are kept in the /Library/Collaboration/Groups/<groupname>/ folder.

Each wiki maintains its own history. You can restore any page individually by using the wiki history function.

**To restore a deleted wiki page:**

1. Perform any search, then choose Deleted Entries in the Search Summary area of the results page.
2. Click the title of a deleted page.
3. Click Undelete Page to restore the deleted page.

## Customizing Wiki

You can customize your wiki pages to have a personal look. For example, you may want to include organization logos or employee images in your wikis.

### Choosing Font Styles and Formatting

Use the editing toolbar to style and format text when you're editing a page.

The following tools are available:

- **Paragraph Style:** To change the style of the current paragraph or selected paragraphs, click the Paragraph Style button and choose a new style from the pop-up menu.
- **Text Style:** To change the style of a word or just a few characters, select the text and then choose an inline style from the text Text Style menu.

- **List Style:** To move the left margin of the current paragraph or selected paragraphs in or out, or to apply a numbered or bulleted list style, choose an option from the pop-up menu. To number several lines or paragraphs sequentially, select them all before choosing Ordered List.

**To modify the text style:**

1. Navigate to the page you want to modify.
2. Click the Edit (/) button in the toolbar to enter edit mode.
3. Select the text you would like to change.
4. Use the text style tools in the edit toolbar to change the text.
5. Click "save" in the editing toolbar.

### Customizing Wiki Themes and Layouts

Wiki administrators can create themes using Cascading Style Sheets (CSS) and add custom XSL templates to change the content of pages. Don't modify the default templates because they may be changed or replaced during a software update. You can use the default templates as examples.

You should use the wireframe theme as a starting point. The wireframe theme is located in the /Library/Application/Support/Apple/WikiServer/Themes/wireframe.wikitheme/ folder.

A plist file comes with each theme folder. You can modify theme characteristics such as the sidebar and banner images in the plist file. You can also include javascripts to run on your wiki pages.

In the CSS file, you can specify display variables such as font definitions and link colors.

## Getting Help Using the Wiki

There is a pointer to online help content in the wiki.

**To get online help from your wiki:**

1. Open any wiki page.
2. Click Help at the bottom of the page.
3. Select the Web service topic for a list of quick search topics.
4. For more specific help, click the Search (magnifying glass) button, enter a topic, and press Return.

   A list of topics appears.

## Setting Up a Web Calendar

The following sections provide instructions for setting up a web calendar on your website.

Web calendar is a web service for groups that lets users access a group calendar to track meetings and deadlines from the web.

The web calendar uses the iCal service in Mac OS X Server v10.5. iCal service must be configured and running for a wiki to use the group calendar. For more information, see *iCal Service Administration*.

### Enabling Web Calendar Service for a Website

You can enable Web Calendar service on your website. You must enable Wiki and blog group web services before you can enable Web Calendar service.

**To turn web calendar on for your website:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Sites.
5. In the Sites list, click the site where you want web calendaring service enabled.
6. Click Web Services.
7. In the Services for Groups section, select the Web calendar checkbox.
8. Click Save.

### Navigating the Web Calendar

The calendar includes several navigation buttons that make it easy to skip through months or weeks, or return to today's date.

In the calendar, there are two types of views:  week and month.

- In week view, you see all events scheduled for a given week, including all-day events and timed events. You can also quickly create all-day or timed events.
- In month view, you see only all-day events and create all-day events by default.

To switch between week and month views, click "week" or "month" in Calendar.

**To navigate through the calendar:**

1. Open to the web calendar.
2. Click "month" to switch to month view or "week" to switch to week view.
3. In month view, click Previous (<) to view the previous month or Next (>) to view the next month.

To view another month, in month view, click the name of the month (between the Previous and Next buttons), and choose the month you want to view.

4 In week view, click Previous (<) to view the previous week or Next (>) to view the next week.

5 To open a minicalendar, in week view, click the name of the week and navigate to a specific week or month:

To go to the previous month in the minicalendar, click Previous (up arrow).

To go to the next month in the minicalendar, click Next (down arrow).

6 To view the week that includes today's date, click Today (diamond).

7 To view a specific week, click one of the days in that week.

8 To view a specific month, click the name of the month.

## Creating Timed Calendar Events

Timed events are events such as meetings or appointments. You can create timed events in week view or month view, but they're easier to create in week view.

**To create a timed event in the web calendar:**

1 Navigate to the web calendar.

2 In week view, drag from the start time to the end time of the event.

3 Enter the name of the event in the Summary field and the location of the event in the Location field.

To change the start date of the event, click the date, and in the calendar that appears, click a date. To change the start date to today's date, click Today (diamond); to view the previous month, click Previous (up arrow); to view the next month, click Next (down arrow).

To change the start time, use the Start Time pop-up menus.

4 Deselect "All-Day (Banner) event."

5 To change the duration of the event, use the Duration pop-up menus.

If the duration is more than 24 hours, enter a number in the "days" field.

6 Click OK.

## Editing Calendar Events

When you edit an event in Calendar, you can change all information associated with the event, such as name, location, date, and duration. You can also change an event from an all-day event to a timed event or vice-versa.

**To edit a web calendar event:**

1 Navigate to the web calendar.

2 In week or month view, click the event.

3 To change the name of the event, enter the name in the Summary field.

4 To change the location of the event, enter a new location in the Location field.

5 To change the start date of the event, click the date and in the calendar that appears, click the date of the event.

To change the start date to today's date, click Today (diamond); to view the previous month, click Previous (up arrow); to view the next month, click Next (down arrow).

6 To change the start time of the event, use the pop-up menus next to the date.

The pop-up menus are in HH:MM format.

7 To change the event from all-day to timed or vice versa, select or deselect "All-Day (Banner) event."

8 To change the duration of the event, use the Duration pop-up menus.

If it is an all-day event, or if the duration is more than 24 hours, enter a number in the "days" field.

9 Click OK to save your settings.

## Deleting Web Calendar Events

When you delete an event in Calendar, it is permanently deleted and you can't undo the deletion. However, you can create the event again.

**To delete an event from the web calendar:**

1 Navigate to the web calendar.

2 In week or month view, click the event.

3 Click Delete.

4 Click OK.

## Using the Web Calendar with iCal

You can subscribe to a web calendar in iCal and configure your iCal calendar to retrieve updates from the web calendar. In iCal, the web calendar is read-only, so you can't edit it.

The web server firewall must allow traffic through port 8008.

**To subscribe to a web calendar in iCal:**

5 In iCal, choose Calendar > Subscribe.

6 Enter http://*serverurl*:8008/calendars/groups/*groupname*/calendar/.

Replace *serverurl* with the URL of your web server, such as www.example.com. Replace *groupname* with the name of your group.

7 Click Subscribe.

8 Authenticate using your web name and password and then click OK.

9 In the Title field, enter a name for your calendar.

10 To enable autoupdating of your iCal calendar, select Refresh and choose the updating frequency.

11 Click OK.

## Setting Up User and Group Blogs

A blog is like a diary or journal, with entries that are arranged in the order they were created in. On the other hand, a wiki contains shared content that doesn't appear in chronological order. The type of information you want to put on your site helps determine whether it appears in a wiki or in a blog.

The following sections provide instructions for setting up user and group blogs on your website.

### Enabling Blog Service for a Website

You can enable user and group blog service on your website. Mac OS X Server includes a group wiki and a group blog. These are enabled together. Group blogs let users in a group access and post entries to the same blog.

Users can also publish their own personal blog using Web services associated with their server account. This gives users the ability to maintain personal blogs on their own user pages.

For more information, see "Enabling Web Calendar Service for a Website" on page 72.

### Adding a Blog Page

Entries appear in the order they were created in, with the most recent entry appearing first. Only the most current entries appear on the main page of the blog. Older entries are still available and can be found and viewed by searching or navigating using the date controls.

**To add a blog page:**

1 Click "blog" if you're not already viewing a blog page.

2 Click the New Page (+) button in the toolbar.

3 In the New Entry dialog that appears, enter a title.

4 If the blog is configured for podcasting and you have an audio or video file for this entry, select Podcast and choose the file.

5 Click Create.

The editing toolbar appears and a new page is created.

6 Delete the sample text and enter your content.

7 When you finish, click Save in the editing toolbar.

## Setting Blog SACL Permissions for Users

Web services administrators can use service access control lists (SACLs) to specify which users have access to blogs. Use Server Admin to set SACL permissions.

*Important:* To change SACL settings for blogs, you must use the server interface, not the Web interface.

**To set user SACL permissions for a blog:**

1. Open Server Admin and connect to the server.
2. Select the server.

   The list of services appears.
3. Click Settings.
4. Click Access.
5. Click Services, if it is not already displayed.
6. Select the level of restriction that you want for the services.

   To restrict access to all services, select "For all services."

   To set access permissions for blogs, select "For selected services below" and then select Blog from the Service list.
7. Select the level of restriction you want for users and groups.

   To provide unrestricted access, click "Allow all users and groups."

   To restrict access to specific users and groups, select "Allow only users and groups below," click the Add (+) button to open the Users and Groups pane, and then drag users and groups to the list.
8. Click Save.

# Configuring and Managing Webmail

5

This chapter shows you how to enable Webmail for the websites on your server in order to provide access to basic mail operations via a web connection.

Webmail adds basic mail functions to your website. If your web service hosts more than one website, Webmail can provide access to mail service on all sites. The mail service looks the same on all sites.

## Webmail Overview

The Webmail software is included in Mac OS X Server and is disabled by default.

The Webmail software is based on SquirrelMail (v1.4.9a), which is a collection of open source scripts run by the Apache server. For more information about SquirrelMail, see www.squirrelmail.org.

### Webmail User Services

If you enable webmail, you users can:

- Compose and send messages
- Receive messages
- Forward or reply to received messages
- Maintain a signature that is appended to each sent message
- Create, delete, and rename folders and move messages between folders
- Attach files to outgoing messages
- Retrieve attached files from incoming messages
- Manage a private address book
- Set webmail preferences, including the color scheme displayed in the web browser

Users access the Webmail page of your website by appending /webmail to the URL of your site (for example, http://mysite.example.com/webmail/).

To use your Webmail service, a user must have an account on your mail server. Therefore, you must have the mail service set up if you want to offer Webmail on your websites.

Users log in to Webmail with the name and password they use for logging in to their regular mail service. Webmail does not provide its own authentication. For more information about mail service users, see *Mail Service Administration*.

When users log in to Webmail, their passwords are sent over the Internet in clear text (not encrypted) unless the website is configured to use SSL. For instructions on configuring SSL for website, see "Enabling Secure Sockets Layer (SSL)" on page 43.

More information about Webmail is available in the SquirrelMail user manual, located at http://squirrelmail.org/wiki/DocumentationHome.

### Webmail and Your Mail Server

Webmail relies on your mail server to provide the mail service. Webmail merely provides access to the mail service through a web browser. Webmail cannot provide mail service independent of a mail server.

Webmail uses the mail service of your Mac OS X Server by default. You can designate a different mail server using Terminal and UNIX command-line tools. For instructions, see "Configuring Webmail" on page 79.

### Webmail Protocols

Webmail uses the following standard mail protocols that your mail server must support:

- Internet Message Access Protocol (IMAP), for retrieving incoming mail
- Simple Mail Transfer Protocol (SMTP), for exchanging mail with other mail servers (sending outgoing mail and receiving incoming mail)

The SquirrelMail configuration script authorizes setting the IMAP server type:

- The setting `macosx = Mac OS X MailServer` refers to the older Apple MailServer in Mac OS X Server v10.2.
- In Mac OS X v10.3, v10.4, and v0.5, the correct setting (set by default) is `cyrus = Cyrus IMAP Server`.

Webmail does not support retrieving incoming mail using Post Office Protocol (POP). Even if your mail server supports POP, Webmail does not.

## Enabling Webmail

Use Server Admin to enable Webmail for your websites hosted on your web server. Any changes you make take effect when you restart Web service.

*Important:* Webmail will not work on a site if the mail protocols and Mail service are not configured and started.

**To enable Webmail for a site:**

1. Make sure your mail service is started and configured to provide IMAP and SMTP service.
2. Make sure IMAP mail service is enabled for the user accounts of the users you want to have Webmail access.

   For details on mail settings in user accounts, see *User Management*.
3. Open Server Admin and connect to the server.
4. Click the triangle to the left of the server.

   The list of services appears.
5. From the expanded Servers list, select Web.
6. Click Sites.
7. In the Sites list, click the site you want to have Webmail enabled.
8. Click Web Services.
9. In the Services for Users section, select the Webmail checkbox.
10. Click Save.

When you turn Webmail on, the PHP module is enabled (if it was not already). If you turn webmail off, PHP remains on until you turn it off. For more information, see "PHP" on page 104.

## Configuring Webmail

After enabling Webmail to provide basic mail functions on your website, you can change settings to integrate Webmail with your site. You can do this by editing the SquirrelMail configuration file, /etc/squirrelmail/config/config.php, or by using Terminal with root privileges to run the interactive configuration script. This Perl script operates by reading original values from config.php and writing new values back to config.php.

You can configure the following SquirrelMail options to integrate Webmail with your site:

- **Organization Name:** The name that appears on the main Webmail page when a user logs in. The default is Mac OS X Server Webmail.
- **Organization Logo**: The relative or absolute path to an image file.
- **Organization Title**: The title of the web browser window while viewing a Webmail page. The default is Mac OS X Server Webmail.
- **Trash Folder**: The name of the IMAP folder where Mail service puts messages when the user deletes them. The default is Deleted Messages.

- **Sent Folder**: The name of the IMAP folder where Mail service puts messages after sending them. The default is Sent Messages.
- **Draft Folder**: The name of the IMAP folder where Mail service puts the user's draft messages. The default is Drafts.

***Important:*** If you use the interactive configuration script to change SquirrelMail settings, you must also use the script to enter the domain name of your server. If this is not done, Webmail can't send messages.

Webmail configuration settings apply to all websites hosted by your web service.

**To configure Webmail options using a Perl configuration script:**

1. Open Terminal and enter the following command:

```
$ sudo /etc/squirrelmail/config/conf.pl
```

2. Access and change the SquirrelMail settings as needed using the interactive menu options.
3. Change the domain name to your server's real domain name, such as example.com.

   The domain name is the first item on the SquirrelMail script's Server Settings menu.

   If you don't enter the server's domain name correctly, the interactive script replaces the original value, getenv(SERVER_NAME), with the same value but enclosed in single quotes. The quoted value no longer works as a function call to retrieve the domain name, and as a result Webmail can't send messages.
4. Save your data after you complete the configuration changes.
5. Quit the interactive script.

Webmail configuration changes do not require restarting Web service unless users are logged in to Webmail.

To further customize the appearance (for example, to provide a specific appearance for each website, you must know how to write PHP scripts. In addition, you must be familiar with the SquirrelMail plug-in architecture and you must write your own SquirrelMail plug-ins.

## Setting Up Mailing List Web Archives

Mailing lists are discussion groups that use mail distribution to facilitate communication between users. Mailing lists distribute a single mail message to multiple recipients and can be administered by someone other than the workgroup or server administrator. More importantly, mailing list subscribers do not need an account (mail or file access) on the list's server. Any mail address can be added to the list.

You can create and maintain mailing lists with a web-based interface for users. You can also configure mailing list archiving, content filtering, and digest delivery options for mailing lists. For more information about mailing lists, see *Mail Service Administration*.

Messages sent to a mailing list can be archived and browsed from your website at a later time. The messages are grouped into archival volumes by time and date. You must turn mailing list web archive service on for your website to access the mailing list archive through your web server.

**To turn mailing list archiving on for your website:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Sites.
5. In the Sites list, click the site where you want mailing list web archive service enabled.
6. Click Web Services.
7. In the Services for Groups section, select the "Mailing list web archive" checkbox.
8. Click Save.

# Working with WebObjects and Open Source Applications

# 6

This chapter helps you become familiar with WebObjects and the open source applications Mac OS X Server uses to administer and deliver web services.

WebObjects service is the application server component of Mac OS X Server. WebObjects offers versatile web development tools that let you extend your web server in a variety of ways.

In addition, several open source applications provide essential features for Web service. These applications include:

- Apache web server
- Tomcat servlet container
- MySQL database
- Ruby on Rails

## Working with WebObjects Service

WebObjects is the Apple solution for rapid development and deployment of ecommerce and other Internet applications. WebObjects applications can connect to multiple databases and dynamically generate HTML content.

The following topics cover WebObjects administration:

## WebObjects Overview

Mac OS X Server includes the WebObjects run-time libraries and an unlimited deployment license to facilitate developing standards-based web services and Java server applications. You can optionally purchase WebObjects development tools from the Apple Store (store.apple.com), Apple's retail stores, and authorized Apple resellers.

You can set WebObjects to start when the server starts. This ensures that WebObjects modules start after a power failure or after the server shuts down.

For more information and documentation on WebObjects, go to www.apple.com/webobjects or developer.apple.com/documentation/WebObjects.

## Turning WebObjects Service On

Before you can configure your application server, you must turn WebObjects service on in Server Admin.

**To turn WebObjects service on:**

1. Open Server Admin and connect to the server.
2. Click Settings, then click Services.
3. Select the WebObjects checkbox.
4. Click Save.

## Setting Up WebObjects Service

Use WebObjects service settings in Server Admin to specify a WebObjects Task Daemon (wotaskd) port, enable Java monitoring, and set the monitor port. WebObjects deployment uses wotaskd to manage the application instances running on your application server.

> *WARNING:* To avoid security problems, any computer that runs JavaMonitor and wotaskd should always be behind a firewall. In addition, only one server per subnet should run JavaMonitor at any time.

The main task of wotaskd is to start application instances when the server is restarted. To accomplish this, wotaskd must be restarted when the server starts up, which is done by configuring wotaskd as a service started when the computer starts up. By default, a wotaskd process running on port 1085 is configured as a service on all supported platforms.

**To configure WebObjects service settings:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. Click WebObjects.

4 Click Settings.

5 Specify the wotaskd port or the Monitor port as desired.

Monitor and wotaskd are part of the WebObjects deployment strategy. Each machine that is running a WebObjects application should have wotaskd running on it. To configure these applications for deployment, run the Monitor application. After WebObjects is configured, only wotaskd must remain running.

Each wotaskd instance is only responsible for WebObjects applications running on the same host. The Web server adaptor (or WebObjects HTTP adaptor) communicates with the wotaskds instance on each host to discover the WebObjects applications that are available.

Instances of WebObjects communicate their state to wotaskd through the use of TCP lifebeats, and wotaskd controls instances through special DirectActions calls.

6 If required, turn Monitor on by selecting the Enable Monitor checkbox.

You must run monitor from a machine that is running wotaskd and that is a managed server. Never run more than one instance of Monitor for a set of servers. Always run Monitor and wotaskd behind a firewall.

7 Click Save.

## Starting WebObjects Service

You start WebObjects service from Server Admin.

**To start WebObjects service:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select WebObjects.

4 Click Start WebObjects (below the Servers list).

The service runs until you stop it and restarts if your server is restarted.

### From the Command Line

You can also start or stop WebObjects using the `serveradmin` command in Terminal by entering the following commands:

```
$ serveradmin start webobjects
$ serveradmin stop webobjects
```

## Checking the Status of WebObjects Service

You can use Server Admin to monitor WebObjects service.

**To check the status of WebObjects service:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select WebObjects.

4 Click Overview to see if WebObjects service is running, the time it started if it is running, and to see if Monitor is running.

### Stopping WebObjects Service

You can use Server Admin to stop WebObjects service.

**To stop WebObjects service:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select WebObjects.

4 Click Stop WebObjects (below the Servers list).

### Opening the Monitor

Monitor is a web-based tool that helps you manage and monitor applications running on WebObjects service. You use this tool to set up applications for deployment through your web server and to control load balancing across multiple web servers.

For more information, open Monitor and select Help.

**To open Monitor:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select WebObjects.

4 Click Settings and select the Enable Monitor checkbox.

5 Click Save.

6 Open a web browser and enter the address for Monitor:

http://localhost:<Monitor port>

## Working with Apache

Apache is the open source HTTP web server provided with Mac OS X Server. You can use Server Admin to manage most web server operations, but in some instances you may want to add or change parts of the Apache server. In such situations, you must modify Apache configuration files and change or add Apache modules.

Mac OS X Server v10.5 supports two versions of the Apache web server—Apache 1.3 and Apache 2.2. Both Apache 1.3 and Apache 2.2 are supported by Server Admin. Apache 2.2 runs as a 64-bit process on appropriate hardware, but Apache 1.3 is 32-bit only.

The two versions can be manually configured to run side by side as long as they do not both attempt to listen on the same IP address / service port combination. Only one version is managed by Server Admin, and running both concurrently is not supported.

Server Admin manages only one version of Apache at a time. In a clean installation it's always Apache 2.2. In an upgrade, it is Apache 1.3, until you convert to Apache 2.2.

The locations of key Apache files are listed in the following table.

| File Description | Apache 1.3 Location | Apache 2.2 Location |
|---|---|---|
| Configuration file for Web service | /etc/httpd/ folder | /etc/apache2/ folder |
| Site configuration files | /etc/httpd/sites/ folder | /etc/apache2/sites/ folder |
| Executable file | /usr/sbin/httpd-1.3 | /usr/sbin/httpd |
| Web modules | /usr/libexec/httpd/ folder | /usr/libexec/apache2/ folder |
| Error log | /var/log/httpd/ folder | /var/log/apache2/ folder (with a symlink that lets the folder be viewed as /Library/Logs/WebServer/) |
| Temporarily disabled virtual hosts | /etc/httpd/sites_disabled/ folder | /etc/apache2/sites_disabled/ folder |
| Static content for both Apache versions default to /Library/WebServer/Documents/ | | |
| CGIs for both Apache versions default to /Library/WebServer/CGI-Executables/ | | |

All files in /etc/httpd/sites/ for Apache 1.3 or in /etc/apache2/sites/ for Apache 2.2 are read and processed by Apache when it performs a hard or soft (graceful) restart. Each time you save changes, the server does a graceful restart.

If you edit a file using a text editor that creates a temporary or backup copy, the server restart may fail because two files with almost identical names are present. To avoid this problem, delete temporary or backup files created when editing files in this folder.

## Editing Apache Configuration Files

You can edit Apache configuration files if you need to work with features of the Apache web server that are not part of Server Admin. To edit configuration files, you must be an experienced Apache administrator and you must be familiar with text-editing tools. Be sure to make a copy of the original configuration file before editing it.

The httpd.conf configuration file handles all directives controlled by Server Admin. You can edit this file as long as you follow the text conventions and comments in the file.

This file also has a directive to include the .../sites/ folder. That folder contains all virtual hosts for that server. The files are named with the unique identifier of the virtual host (for example, 0000_17.221.43.127_80_www.example.com.conf). You disable specific sites by moving them to the sites_disabled folder and then restarting Web service. You can also edit site files as long as the conventions in the file are followed.

One hidden file in the sites_disabled folder is named "default_default.conf." This file is used as the template for all new virtual hosts created in Server Admin. An administrator can edit the template file to customize it, taking care to follow the conventions established in the file.

For more information about Apache and its modules, see "Apache Web Module Overview" on page 99.

## Restoring the Default Configuration

It is possible to restore a factory setting or default configuration of Apache without reinstalling Mac OS X Server. The various .default files in the Apache configuration directories are put there for this purpose and are installed as a Read-Only files to discourage administrators from modifying them.

**To restore the default configuration:**

1. Open Terminal.
2. Enter the following command:

```
$ sudo serveradmin web:command=writeSettings \web:variant=withDefaults
```

A ReadMe.txt file that describes the Apache configuration is available:

- The Apache 2.2 readme.txt file is installed in the /etc/apache2/ folder.
- The Apache 1.3 readme.txt file is installed in the /etc/httpd/ folder.

This file contains instructions for manually going from Apache 2.2 to 1.3, if that becomes necessary.

## Using the apachectl Script

The default way to start and stop Apache on Mac OS X Server is to use Server Admin.

There are two versions of `apachectl` commands:

- `apachectl` controls Apache 2.2. Apache 2.2 runs as a 64-bit process on appropriate hardware.
- `apachectl-1.3` controls Apache 1.3. Apache 1.3 is 32-bit only.

If you want to use the `apachectl` script to start and stop Web service instead of using Server Admin, be aware of the following:

- The web performance cache is enabled by default in Mac OS X Server v10.5. For upgrade installations, with Apache 1.3, the web performance cache is enabled in v10.5 only if it was enabled prior to the upgrade. When Web service starts, the main web service process (httpd) and a webperfcache process start. (The webperfcache process serves static content from a memory cache and relays requests to httpd when necessary.) The `apachectl` script that comes with Mac OS X Server is unaware of webperfcache, so if you have not disabled the performance cache, you must also use the `webperfcachectl` script to start and stop webperfcache.
- When Apache is started using the `apachectl` script, the soft process limit is 100, the default limit. When you use CGI scripts, this limit may not be high enough. In this case, you can start Web service using Server Admin, which sets the soft process limit to 2048. Alternatively, you can enter `ulimit -u 2048` before using `apachectl`.
- The `apachectl` script does not start Apache when the server restarts.

Because of the issues noted above, if you must control Apache from a script, the recommended approach is to use the `serveradmin` command-line tool.

**To start Apache from a script:**

1. Open your script.
2. Enter the following command:

```
serveradmin start web
```

   This starts Apache and the performance cache (if necessary) and it marks /etc/hostconfig to start Web service on restart.

3. Save and run your script.

**To stop Apache from the command line:**

1. Open your script.
2. Enter the following command:

```
serveradmin stop web
```

   This stops Apache and the performance cache (if necessary), and it marks /etc/hostconfig not to start Web service on restart.

3. Save and run your script.

## About Apache Multicast DNS Registration

Do not use Apache multicast DNS registration with the server.

***Important:*** Do not try to turn on Apache multicast DNS (mdns) registration for the server. It does not support virtual hosts, and the server uses virtual hosts.

## Using Apache Axis

Apache Extensible Interaction System (Axis) is an implementation of Simple Object Access Protocol (SOAP). More about SOAP can be found at www.w3.org/TR/SOAP. More about Axis can be found at:ws.apache.org/axis.

You can use Apache Axis by writing web applications that use the Axis libraries and then deploy the applications in Tomcat. Unlike Tomcat, Axis is not usually used as an application server.

Mac OS X Server v10.5 includes a preinstalled version of Apache Axis (v1.1), which operates with the preinstalled Tomcat (v4.1.x).

The Axis libraries are in the /System/Library/Axis/ folder. By default, Apple installs an example Axis web application into Tomcat. The web application, known as axis, is found in /Library/Tomcat/webapps/axis/.

After you enable Tomcat in the Web Service Settings pane in Server Admin, you can validate the preinstalled Apache Axis by accessing:

http://example.com:9006/axis/

Replace "example.com" with your host name. Note the nonstandard Tomcat port.

The first time you exercise the preinstalled Axis by accessing http://example.com:9006/axis/ and selecting the link entitled "Validate the local installation's configuration," you will see the following error messages:

- Warning: could not find class javax.mail.internet.MimeMessage from file mail.jar Attachments will not work.

  See ava.sun.com/products/javamail.

- Warning: could not find class org.apache.xml.security.Init from file xmlsec.jar XML Security is not supported

  See xml.apache.org/security.

Follow the instructions that accompany the warning messages if you require those optional components.

Consult the *Axis User's Guide* to learn more about using Axis in your own web applications. This guide is located at ws.apache.org/axis/java/user-guide.html.

# Working with Tomcat

Tomcat adds Java servlet and JavaServer Pages (JSP) capabilities to Mac OS X Server. Java servlets are Java-based applications that run on your server, in contrast to Java applets, which run on the user's computer. JavaServer Pages let you embed Java servlets in your HTML web pages.

The Java Servlet and JavaServer Pages specifications are developed by Sun Microsystems under the Java Community Process. The current production series is the Tomcat 4.1.x series, which implements Java Servlet 2.3 and JavaServer Pages 1.2 specifications.

For more information about Tomcat and documentation for this software, see http://tomcat.apache.org/.

For information about Java Servlets that you can use on your web server, see:

- java.sun.com/products/servlet
- java.sun.com/products/jsp

By default, the Tomcat management console and status service are turned off. Consult the Apache Tomcat documentation (http://tomcat.apache.org/tomcat-6.0-doc/index.html) to properly enable and secure these service for your deployment environment. It is recommended that Web service be secured behind a firewall.

For more resources, consult the O'Reilly book *Tomcat the Definitive Guide* (www.oreilly.com).

## Setting Tomcat as the Application Container

You use Server Admin to work with Tomcat. You can set Tomcat to start when the server starts. This ensures that the Tomcat module starts after a power failure or after the server shuts down.

You can use Server Admin or Terminal to enable Tomcat.

**To start Tomcat using Server Admin:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Settings, then click General.
5. Select the Enable Tomcat checkbox.
6. Click Save.

### From the Command Line

You can also start Tomcat in Terminal by entering the following commands:

```
$ cd /Library/Tomcat/bin
$ ./startup.sh start
```

To verify that Tomcat is running, use a browser to access port 9006 on your website server by entering the URL for your site followed by :9006. If Tomcat is running, this URL shows the Tomcat home page.

## Working with MySQL

MySQL provides a relational database management solution for your web server. With this open source software, you can link data in different tables or databases and provide the information on your website.

The MySQL Manager application is replaced by the MySQL service in Server Admin.

### Turning MySQL Service On

Before you can configure your database manager, you must turn MySQL service on in Server Admin.

**To turn MySQL service on:**

1. Open Server Admin and connect to the server.
2. Click Settings, then click Services.
3. Select the MySQL checkbox.
4. Click Save.

### Setting Up MySQL Service

Use MySQL service Settings in Server Admin to specify the database location, to enable network connections, and to set the MySQL root password.

**To configure MySQL service settings:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. Click MySQL.
4. Click Settings.
5. Select the "Allow network connections" checkbox to permit users to access MySQL service.

   This grants users access to database information through the web server.
6. Enter the path to the location of your database in the Database location field.

   You can also click the Choose button and browse for the folder you want to use.
7. Click Save.

### Starting MySQL Service

You start MySQL service from Server Admin.

**To start MySQL service:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select MySQL.

4 Click Start MySQL (below the Servers list).

The service runs until you stop it and restarts if your server is restarted.

## Checking the Status of MySQL Service

You can use Server Admin to monitor MySQL service.

**To check the status of MySQL service:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select MySQL.

4 Click Overview to see if MySQL service is running, the time it started if it is running, and if network connections are allowed.

## Viewing MySQL Service and Admin Logs

MySQL service keeps two types of logs, a MySQL service log and MySQL admin logs:

- The MySQL service log records the time of events such as when MySQL service is started and stopped.
- The MySQL admin log records information such as when clients connect or disconnect and each SQL statement received from clients. This log is located at /Library/Logs/MySQL.log.

You can view MySQL service logs using Server Admin.

**To view MySQL service logs:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 Click MySQL.

4 Click Logs.

Use the Filter field to search for specific entries.

## Stopping MySQL Service

You can use Server Admin to stop MySQL service.

**To stop MySQL service:**

1 Open Server Admin and connect to the server.

2 Click the triangle to the left of the server.

The list of services appears.

3 From the expanded Servers list, select MySQL.

4 Click Stop MySQL (below the Servers list).

### Upgrading MySQL

Mac OS X Server v10.5 includes the latest version of MySQL, v5.0. Because it's preinstalled, you won't find it in /usr/local/mysql. Instead, its elements are distributed in the file system according to standard UNIX file layout as follows:

- MySQL executables are located in the /usr/sbin/ and /usr/bin/ folders.
- MySQL man pages are located in the /usr/share/man/ folder.
- Other MySQL parts are located in the /usr/share/mysql/ folder.

When installed, the MySQL database resides in the /var/mysql/ folder.

At some point a newer version of MySQL will be posted to www.mysql.com. At that time you can download the source and build it (if you have the developer packages installed) or you can download the relevant binary distribution and install it, following the instructions posted on that website.

By default, such installations reside in the /usr/local/mysql/ folder. If you install your own version of MySQL, you'll have two versions of MySQL present on your system. This causes no harm as long as you don't try to run the two versions at the same time.

Be sure to use commands intended for the new version by specifying the full path (starting with /usr/local/mysql/), or make sure your shell's path variable is set to search in your local folder first.

## Working with Ruby on Rails

Ruby on Rails is a web application framework, becoming very popular because of its ease of development, scalability, and support for the Model-View-Controller architecture, and because it uses Ajax via the Prototype and Script.aculo.us libraries. Details can be found at www.rubyonrails.org.

In Mac OS X Server v10.5, Ruby on Rails is preinstalled with several useful gems (component packages), including the Mongrel web server.

The Mongrel web server comes with the `mongrel_rails` tool to manage it. Mac OS X Server v10.5 supports the deployment of Ruby on Rails applications in the following ways:

- It includes an enhanced version of the `mongrel_rails` tool called `mongrel_rails_persist`, which creates a `launchd` plist file to run Mongrel persistently (across reboots) and causes it to register with Bonjour.

This is helpful because it allows the Server Admin Web Site Proxy panel to find instances of Mongrel running on the same machine, and presents their URLs in the Balancer Members popup. More details about `mongrel_rails_persist` are available on its main page.

- It allows administration of Apache 2.2 mod_proxy_balancer in the Server Admin web service Sites Proxy panel. This allows several instances of Mongrel (or another back-end http server) to be accessed via a single URL and allows Apache to distribute its load to those services in a configured proportion.
- It includes mod_fastcgi for customers who have used it to solve configuration issues and prefer to use it over mod_proxy_balancer. This module is disabled by default.

## Managing the Deployment of Ruby on Rails Applications

You can use Server Admin to manage the deployment of Ruby on Rails applications with the Apache 2.2 mod_proxy_balancer module.

You can dedicate your website (virtual host) to Ruby on Rails or you can share your website with Ruby on Rails. The following scenarios describe how to do this:

- In the first scenario, the website is dedicated to the Ruby on Rails web application.
- In the second scenario, the website is shared with the Ruby on Rails application.

In these scenarios, we use the default wild-card website which is the website that has the asterisk in the address column of the websites list, as an example. There are other variations depending on how you organize your websites and how you organize your Ruby on Rails applications, but these two scenarios should illustrate the general mechanism. You can check the knowledge base for additional techniques.

**Scenario 1 - Dedicating a Website (Virtual Host) to the Proxied Web Application**

1. Open Terminal and enter the following commands to create your Ruby on Rails application outside the document root of any existing web virtual host (for example in /Library/WebServer/*MyWebApp*, where *MyWebApp* is the name of your rails application).

```
$ cd /Library/WebServer
$ rails MyWebApp
$ ...
```

2. Start the Mongrel web server using the `mongrel_rails_persist` command:

```
$ sudo mongrel_rails_persist start -p 3001 -c /Library/WebServer/MyWebApp
```

This wrapper for the `mongrel_rails` command registers the instance of Mongrel with Bonjour and provides a launchd plist file so the instance of Mongrel restarts on server startup.

3. Use Safari to browse the local Rails URL to confirm that the web application is responding:

http://127.0.0.1:3001

If you specified a model or scaffold in your Rails application, the URL might be something like:

http://127.0.0.1:3001/*ModelName*

You should see the "Welcome Aboard / You're riding the rails" page.

4. Open Server Admin and connect to the server.
5. Click the triangle to the left of the server.

   The list of services appears.
6. From the expanded Servers list, select Web.
7. Click Sites, then select the website in the list.
8. Click Proxy below the websites list.
9. Select the Enable Reverse Proxy checkbox.
10. Verify that the Proxy path field is set to "/".

    This requires all URLs within the website to be proxied to the balancer group.
11. Leave the Stick Session Identifier field blank unless you have reason to specify a value.
12. To add a balancer member, click the Add (+) button below the Balancer Members list.
13. From the Server URL pop-up menu, designate the URL for the load balancer member.

    Each instance of Mongrel running locally has its URL shown in the pop-up menu, so you should be able to select one.

    Create additional balancer members if you have multiple instances of Mongrel serving your web application on this host or other reachable hosts. Each balancer member corresponds to an instance of Mongrel, running on either the local host or other hosts.
14. If there is only one balancer member, set the Load Factor to 100.

    Use the Load Factor field to distribute the load among balancer members.
15. Leave the Route field blank unless you have a specific reason to enter a value.
16. Click OK.
17. Click Save.
18. Start Web Service, if it is not already running.
19. Use Safari to access the proxy URL to confirm that the web application is responding:

    http://127.0.0.1

    If you specified a model or scaffold in your Rails applicatioin, the URL might be something like:

    http://127.0.0.1/*ModelName*

    It is not necessary to enter a trailing slash.

### Scenario 2 - Sharing a Website (Virtual Host) with the Proxied Web Application

1. Open Terminal and enter the following commands to create your Ruby on Rails application outside the document root of any existing web virtual host (for example in /Library/WebServer/*MyWebApp*, where *MyWebApp* is the name of your rails application).

```
$ cd /Library/WebServer
$ rails MyWebApp
$ ...
```

2. Start the Mongrel web server using the `mongrel_rails_persist` command and using the `--prefix` argument:

```
$ sudo mongrel_rails_persist start -p 3001 --prefix /rails -c /Library/
    WebServer/MyWebApp
```

3. Use Safari to access the local Rails URL to confirm that the web application is responding:

   http://127.0.0.1:3001/rails/

   If you specified a model or scaffold in your Rails application, the URL might be something like:

   http://127.0.0.1/rails/*ModelName*

   You should see the "Welcome Aboard / You're riding the rails" page.

4. Open Server Admin and connect to the server.
5. Click the triangle to the left of the server.

   The list of services appears.

6. From the expanded Servers list, select Web.
7. Click Sites, then select the website in the list.
8. Click Proxy below the websites list.
9. Select the Enable Reverse Proxy checkbox.
10. In the Proxy path field enter the prefix you specified to mongrel_rails_persist, but with both a leading and trailing back slash. In our example, this would be /rails/.
11. Leave the Sticky Session Identifier field blank unless you have a reason to specify a value.
12. To add a balancer member, click the Add (+) button below the Balancer Members list.
13. From the Server URL pop-up menu, designate the URL for the load balancer member.

    Each instance of Mongrel running locally has its URL shown in the pop-up menu, so you should be able to select one (for example, http://127.0.0.1:3001/rails).

14. If there is only one balancer member, set the Load Factor to 100.

    Use the Load Factor field to distribute the load among balancer members.

15 Leave the Route field blank unless you have a specific reason to enter a value.

16 Click OK.

17 Click Save.

18 Start Web Service, if it is not already running.

19 Use Safari to access the proxy URL to confirm that the web application is responding:

http://127.0.0.1/rails/

If you specified a model or scaffold in your Rails application, the URL might be something like:

http://127.0.0.1/rail/*ModelName*

If you find that a trailing slash is required, you can use the Server Admin Web Alias panel for the site, and add a RedirectMatch entry that maps /rails to /rails/.

20 Use Safari to access to the local URL to confirm that other content is available at other URLs within the website:

http://127.0.0.1

# Managing Web Modules

# 7

This chapter familiarizes you with Apache web modules that provide key features and controls for Web service.

The Apache web server includes a series of modules that control the server's operation. In addition, Mac OS X Server provides modules with specialized functions for the Macintosh.

## Apache Web Module Overview

Modules plug in to the Apache web server software and add functionality to your website. Apache comes with several standard modules, but you can purchase additional modules from software vendors or download them from the Internet. You can find information about available Apache modules at www.apache.org/docs/mod.

***Note:*** Rails is not based on a separate web module. The discussion of Rails, wherever it appears, refers to mod_proxy_balancer, which is a standard Apache 2.2 module.

## Working with Web Modules

The Apache web server has a modular design that enables you to expand the core functionality of your web server by enabling additional modules. Modules can be enabled or disabled using Server Admin.

Although enabling or disabling Apache web modules is easy in Server Admin, generally you should have a specific functionality goal and fully understand the implications of enabling or disabling modules.

Some web modules are mutually exclusive or are interdependent. Here are some examples:

- auth_digest_module and digest_module must never be enabled simultaneously.
- proxy_module must be enabled if proxy_connect_module, proxy_ftp_module, proxy_http_module, proxy_ajp_module, or proxy_balancer_module are enabled.
- dav_module and dav_fs_module should be in the same state.

- encoding_module requires that headers_module, dav_module, and dav_fs_module are enabled.
- cache_module is required for mem_cache_module and disk_cache_module.

*Important:* Web modules used with Apache 1.3 are different from web modules used with Apache 2.2.

## Viewing Web Modules

You can view a list of modules in use or available for use on the server.

**To view web modules:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Settings, then click Modules.
5. Scroll through the modules list to see modules in use or available for use on the web server.

## Adding Web Modules

You can use Server Admin to add web modules to your web server.

Before you can add a web module to the server, the module must be installed. To install a module, follow the instructions that came with the module software. The web server loads modules from the /usr/libexec/httpd/ folder.

**To add web modules to the server:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Settings, then click Modules.
5. Click the Add (+) button to add a module to the list of available modules.
6. In the Module Name field, enter the module name.
7. Select the Enabled checkbox if you want the module enabled.
8. In the Module Path field, enter the path to the installed module or click the browse button to select the folder.
9. Click OK.
10. Click Save.

## Enabling Web Modules

You can use Server Admin to enable modules for your web server.

**To enable Web service modules:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Settings, then click Modules.
5. Scroll through the modules list to see the set of modules in use or available for use on the web server.
6. Click the Enable checkbox next to the module you want to enable.
7. Click Save.

## Changing Web Modules

You can use Server Admin to change web modules on your server.

**To modify web module settings:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Settings, then click Modules.
5. Scroll through the modules list to see the set of modules in use or available for use on the web server.
6. Select the module you want to change and click the Edit (/) button.

   You can also duplicate an existing module and modify its settings by selecting the module and clicking the Duplicate button, then changing the duplicate module settings.
7. In the Module Name field, enter the module name.
8. If you want the module enabled or disabled for your web server, select or unselect the Enabled checkbox.
9. In the Module Path field, enter the path to the installed module or click the browse button to select the folder.
10. Click OK.
11. Click Save.

### Deleting Web Modules

You can use Server Admin to remove web modules from your server.

**To delete web modules:**

1. Open Server Admin and connect to the server.
2. Click the triangle to the left of the server.

   The list of services appears.
3. From the expanded Servers list, select Web.
4. Click Settings, then click Modules.
5. Scroll through the modules list to see the set of modules in use or available for use on the web server.
6. Select the module you want to remove and click the Delete (–) button.
7. Click Save.

## Macintosh-Specific Modules

Web service in Mac OS X Server installs modules specific to the Macintosh. These modules are described in this section.

### mod_macbinary_apple

This module packages files in MacBinary format, which allows Macintosh files to be downloaded directly from your website. A user can download a MacBinary file using a regular web browser by adding ".bin" to the URL used to access the file.

This is present only in Apache 1.3.

### mod_spotlight_apple

This module allow Apache to perform relevance-ranked searches of the website using Spotlight. After you index your site, you can provide a search field for users to search your website.

Clients must add .spotlight to your website's URL to access a page that allows them to search your site (For example, http://www.example.com/.spotlight).

### mod_auth_apple

This module allows a website to authenticate users by looking for them in file system service domains in the server's search policy. When authentication is enabled, website visitors are prompted for a user name and password before they can access information about the site.

### mod_hfs_apple

This module requires users to enter URLs for HFS volumes using the correct case (lowercase or uppercase). This module adds security for case-insensitive volumes.

### mod_digest_apple

This module enables digest authentication for a WebDAV realm. This is the older, default digest authentication module, based on Apache's mod_digest but modified to use Open Directory rather than htdigest files. It is the default digest module because it works with Open Directory masters running Mac OS X Server v10.4.

This form of recommendation is recommended over basic authentication because it does not cause passwords to be sent in clear text format.

### mod_auth_digest_apple

This module enables digest authentication for a WebDAV realm. This is the newer digest authentication module, based on Apache's mod_auth_digest but modified to use Open Directory rather than the htdigest files. It is disabled by default because it requires that the Open Directory master to use Mac OS X v10.5.

### mod_spnego

This module provides Kerberos authentication for Open Directory users via the SPNEGO/Negotiate protocol.

### mod_encoding

This open source module, customized by Apple, along with a modification to WebDAV module mod_dav, allows WebDAV files to include Japanese characters in their names.

### mod_bonjour

This module allows administrators to control how websites are registered with multicast DNS.

## Open Source Modules

Mac OS X Server includes several popular open source web modules. These include Tomcat and PHP.

### Tomcat

This module, which uses Java-like scripting, is the official reference implementation for Java Servlet and JavaServer Pages developed under the Java Community Process.

Tomcat must be enabled before it can be used.

For more information about Tomcat, as well as how to enable Tomcat, see "Working with Tomcat" on page 90.

## PHP

PHP Hypertext Preprocessor (PHP) lets you handle dynamic web content by using a server-side, HTML-embedded scripting language resembling C. Web developers embed PHP code in HTML code, allowing programmers to integrate dynamic logic directly in an HTML script rather than writing a program that generates HTML.

PHP provides functions similar to those of CGI scripts but it supports a variety of database formats and can communicate across networks by using many protocols.

The PHP libraries are included in Mac OS X Server but are disabled by default.

Unlike client-side JavaScript, PHP code is executed on the server. PHP is also used to implement Webmail on Mac OS X Server. For more information about this module, see www.php.net.

***Important:*** If you perform an upgrade on your server and keep Apache 1.3 with PHP 4.4.x, you should switch to Apache 2.2 with PHP 5.x before August 8, 2008. This is when PHP stops supporting PHP 4.4.

## mod_perl

This module integrates the verify Perl interpreter into the web server, letting existing Perl CGI scripts run without modification. This integration means that the scripts run faster and consume fewer system resources.

For more information about this module, see perl.apache.org.

## mod_encoding

To improve WebDAV's interoperability with non-ASCII file names, Web service includes the open-source Apache module named mod_encoding.

By default, mod_encoding is disabled. The module is installed and configuration directives are present in the Apache config file, but they are not activated because the LoadModule and AddModule directives that inform Apache about mod_encoding are disabled.

To support non-ASCII file names, you must enable mod_encoding. Make sure dav_module is also enabled.

The mod_encoding module extends Apache's functionality and is controlled by a set of configuration directives.

The Apache configuration file supplied with Web service contains a specific set of directives that should be sufficient for most needs. To modify those directives you must to use a text editor and edit the /etc/httpd/httpd.conf file.

The following describes the directives supported by mod_encoding.

**EncodingEngine directive**: This directive enables and disables mod_encoding. Correct operation of mod_encoding also requires that the special version of mod_dav, mod_dav_encoding, be enabled as well.

| Syntax | Default | Context | Compatibility |
|---|---|---|---|
| `EncodingEngine [ on \| off ]` | Off | Server Config | Apache 1.3.x; Mac OS X Server only |

**AddClientEncoding directive:** Although WebDAV clients are expected to send data in UTF-8 or any other properly detectable style, some clients send data in non-autodetectable, platform-local encoding, thus requiring this directive, which maps encoding names to client types.

This directive specifies encodings expected from each client type. The clients are identified by agent name. The agent name can be specified as a pattern using extended regexp. Never use ".*" for agent name. Instead, use DefaultClientEncoding.

This module uses CoreFoundation's CFString and supports all encoding supported by it. In general, IANA-registered encoding names are supported.

| Syntax | Default | Context | Compatibility |
|---|---|---|---|
| `AddClientEncoding agent-name encoding [ encoding...]` | None | Server Config | Apache 1.3.x; Mac OS X Server only |

**DefaultClientEncoding directive:** This directive tells the default set of encodings what to expect from various clients in general. You don't need to specify UTF-8 because that is the default.

| Syntax | Default | Context | Compatibility |
|---|---|---|---|
| `DefaultClientEncoding encoding [ encoding...]` | UTF-8 | Server Config | Apache 1.3.x; Mac OS X and Mac OS X Server only |

**NormalizeUsername directive:** This directive is introduced to support the behavior of Microsoft Windows XP when accessing a password-protected resource. Windows XP clients prepend "hostname\" to the real username. Enabling this option strips off the "hostname\" part, so only "real" username is passed to the authentication module.

| Syntax | Default | Context | Compatibility |
|---|---|---|---|
| `NormalizeUsername [ on \| off ]` | Off | Server Config | Apache 1.3.x; Mac OS X and Mac OS X Server only |

For additional information about mod_encoding, download your own version and read additional documentation provided in the source distribution from:

www.denpa.org/~go/denpa/200302/mod_encoding+mod_dav-macosx.tar.gz

# Solving Web Service Problems

# 8

## If you experience a problem with Web service or one of its components, use the tips and strategies in this chapter.

From time to time you might encounter a problem when setting up or managing Web services. Situations that might cause a problem for administering Web service or for client connections are outlined here.

## If Users Can't Connect to a Website on Your Server

Try these strategies to uncover the problem:

- Make sure Web service is turned on and the site is enabled.
- View the Overview pane of Web service to verify that the server is running.
- Verify the Apache access and error logs. (If you are not sure what the messages mean, you'll find explanations on the Apache website at www.apache.org.)
- Make sure users are entering the correct URL to connect to the web server.
- Make sure the correct folder is selected as the default web folder. Make sure the correct HTML file is selected as the default document page.
- If your website is restricted to specific users, make sure those users have access privileges to your website.
- Verify that users' computers are configured correctly for TCP/IP. If the TCP/IP settings appear correct, use a pinging utility to verify network connections.
- Verify that the problem is not a DNS problem. Try to connect with the IP address of the server instead of using its DNS name.
- Make sure your DNS server's entry for the website's IP address and domain name are correct.

## If a Web Module Is Not Working as Expected

Try the following strategies to uncover the problem:

- Read the error log in Server Admin for information about why the module might not be working.

- If the module came with your web server, read the Apache documentation for that module and make sure the module is intended to work the way you expected.
- If you installed the module, read the documentation that came with the web module to make sure it is installed correctly and is compatible with your server software.

For more information about supported Apache modules for Mac OS X Server, see "Working with Web Modules" on page 99 and the Apache website at www.apache.org/docs/mod.

## If a CGI Script Does Not Run

Try this strategy to uncover the problem.

View the CGI script's file permissions to make sure the script is executable by www. If not, the script won't run on your server even if you enable CGI execution in Server Admin.

# Glossary



**Apache** An open source HTTP server integrated into Mac OS X Server. You can find detailed information about Apache at www.apache.org.

**application server** Software that runs and manages other applications, usually web applications, that are accessed using a web browser. The managed applications reside on the same computer where the application server runs.

**blog** A webpage that presents chronologically ordered entries. Often used as an electronic journal or newsletter.

**Blojsom** The open source project on which Mac OS X Server v10.4 Weblog service is based.

**certificate** Sometimes called an "identity certificate" or "public key certificate." A file in a specific format (Mac OS X Server uses the X.509 format) that contains the public key half of a public-private keypair, the user's identity information such as name and contact information, and the digital signature of either a **Certificate Authority** (CA) or the key user.

**CGI** Common Gateway Interface. A script or program that adds dynamic functions to a website. A CGI sends information back and forth between a website and an application that provides a service for the site.

**Common Gateway Interface** See **CGI**.

**everyone** Any user who can log in to a file server: a registered user or guest, an anonymous FTP user, or a website visitor.

**HTML** Hypertext Markup Language. The set of symbols or codes inserted in a file to be displayed on a web browser page. The markup tells the web browser how to display a webpage's words and images for the user.

**HTTP** Hypertext Transfer Protocol. The client/server protocol for the World Wide Web. HTTP provides a way for a web browser to access a web server and request hypermedia documents created using HTML.

**Internet Protocol** See **IP**.

**IP** Internet Protocol. Also known as IPv4. A method used with Transmission Control Protocol (TCP) to send data between computers over a local network or the Internet. IP delivers data packets and TCP keeps track of data packets.

**IP address** A unique numeric address that identifies a computer on the Internet.

**JavaScript** A scripting language used to add interactivity to webpages.

**JBoss** A full-featured Java application server that provides support for Java 2 Platform, Enterprise Edition (J2EE) applications.

**Kerberos realm** The authentication domain comprising the users and services that are registered with the same Kerberos server. The registered users and services trust the Kerberos server to verify each other's identities.

**local hostname** A name that designates a computer on a local subnet. It can be used without a global DNS system to resolve names to IP addresses. It consists of lowercase letters, numbers, or hyphens (except as the last characters), and ends with ".local" (For example, bills-computer.local). Although the default name is derived from the computer name, a user can specify this name in the Sharing pane of System Preferences. It can be changed easily, and can be used anywhere a DNS name or fully qualified domain name is used. It can only resolve on the same subnet as the computer using it.

**multicast DNS** A protocol developed by Apple for automatic discovery of computers, devices, and services on IP networks. Called Bonjour (previously Rendezvous) by Apple, this proposed Internet standard protocol is sometimes referred to as ZeroConf or multicast DNS. For more information, visit www.apple.com or www.zeroconf.org. To see how this protocol is used in Mac OS X Server, see **local hostname**.

**MySQL** An open source relational database management tool frequently used by web servers.

**open source** A term for the cooperative development of software by the Internet community. The basic principle is to involve as many people as possible in writing and debugging code by publishing the source code and encouraging the formation of a large community of developers who will submit modifications and enhancements.

**PHP** PHP Hypertext Preprocessor (originally Personal Home Page). A scripting language embedded in HTML that's used to create dynamic webpages.

**port** A sort of virtual mail slot. A server uses port numbers to determine which application should receive data packets. Firewalls use port numbers to determine whether data packets are allowed to traverse a local network. "Port" usually refers to either a TCP or UDP port.

**protocol** A set of rules that determines how data is sent back and forth between two applications.

**proxy server** A server that sits between a client application, such as a web browser, and a real server. The proxy server intercepts all requests to the real server to see if it can fulfill the requests itself. If not, it forwards the request to the real server.

**realm** General term with multiple applications. See **WebDAV realm**, **Kerberos realm**.

**SSL** Secure Sockets Layer. An Internet protocol that allows you to send encrypted, authenticated information across the Internet. More recent versions of SSL are known as TLS (Transport Level Security).

**TCP** Transmission Control Protocol. A method used with the Internet Protocol (IP) to send data in the form of message units between computers over the Internet. IP handles the actual delivery of the data, and TCP keeps track of the units of data (called packets) into which a message is divided for efficient routing through the Internet.

**Tomcat** The official reference implementation for Java Servlet 2.2 and JavaServer Pages 1.1, two complementary technologies developed under the Java Community Process.

**URL** Uniform Resource Locator. The address of a computer, file, or resource that can be accessed on a local network or the Internet. The URL is made up of the name of the protocol needed to access the resource, a domain name that identifies a specific computer on the Internet, and a hierarchical description of a file location on the computer.

**user name** The long name for a user, sometimes referred to as the user's real name.

**WebDAV** Web-based Distributed Authoring and Versioning. A live authoring environment that allows client users to check out webpages, make changes, and then check the pages back in to the site while the site is running.

**WebDAV realm** A region of a website, usually a folder or directory, that's defined to provide access for WebDAV users and groups.

**weblog** See **blog**.

# Index

## N



## O



## P



## R



## S

## T



## U



## V



## W