Chapter No. 4
"Providing Webmail Access"
In this package, you will find:
A Biography of the authors of the book
A preview chapter from the book, Chapter NO.4 "Providing Webmail Access"
A synopsis of the book’s content
Information on where to buy this book

About the Authors

**Ian Haycox** is a freelance IT consultant based in France and actively contributes to open source projects. He has twenty-five years of software development experience in the enterprise integration, telecommunications, banking, and television sectors.

Ian has a degree in Computer Science from the University of Hertfordshire, UK, and now runs his own web design company (**http://www.ianhaycox.com/**) and Linux programming consultancy.

---

My thanks to Debbie for supplying me with copious amount of coffee and cheese sandwiches.

---

**Alistair McDonald** is a software developer and IT consultant. He has worked as a freelancer in the UK for 15 years, developing cross-platform software systems in C, C++, Perl, Java, and SQL. He has been using open source software for over 20 years and implementing systems using it for the past 10 years.

Last year, he gave up his freelance career and joined JDA Software, working in a technical role in their Service Industries division.

Alistair is also the author of the book *SpamAssassin: A practical guide to integration and configuration*, published by Packt.

I would like to thank my wife Louise for the support she has given me throughout the writing of all my books.

---

**Magnus Bäck** has been playing and working with computers since his childhood days. He is interested in everything in the computer field, from digital typography and compilers, to relational databases and UNIX. His interests also include e-mail services, and he is an active contributor to the Postfix mailing list. Besides computers, he enjoys photography, cars, and bicycling.

Magnus holds a Master's degree in Computer Science and Engineering from Lund Institute of Technology, Sweden, and currently works with software configuration management for mobile phone software at Sony Ericsson Mobile Communications.

---

**Ralf Hildebrandt** is an active and well-known figure in the Postfix community, working as a Systems Engineer for T-Systems, a German telecommunications company.

He speaks about Postfix at industry conferences and hacker conventions, and contributes regularly to a number of open source mailing lists. Ralf Hildebrandt is the co-author of *The Book of Postfix*.

---

Patrick Ben Koetter is an active and well-known figure in the Postfix community, working as an Information Architect. Patrick Koetter runs his own company, consulting and developing corporate communication for customers in Europe and Africa.

He speaks about Postfix at industry conferences and hacker conventions, and contributes regularly to a number of open source mailing lists. Patrick Koetter is the co-author of The Book of Postfix.

David Rusenko was born in Paris, France, and spent most of his childhood overseas. He began working as a freelance Web Designer in 1996 and had his first experience with open source, a box copy of Red Hat 5.2, shortly after in 1999. After six years and as many versions of Red Hat, he now creates appealing web pages and devises solutions implementing high availability through clustering and alternate security models.

He founded Aderes (http://www.aderes.net) in 2001, a company that provides e-mail and web-based security solutions. His search for an appropriate Webmail Platform for the company led him to SquirrelMail. Initially managing all aspects of the business—from the technical concerns to customer support—gave him the experience that he now contributes to the Webmail chapter of this book.

David has studied both, Information Sciences and Technology (IST) and Management Information Systems (MIS) at the Pennsylvania State University. He speaks English and French fluently, and is conversational in Arabic. During his free time and vacations, he enjoys scuba diving, backpacking, playing racquetball, and playing electronic music records.

Carl Taylor has worked over 20 years in the IT industry and has spent the majority of that time working on UNIX type systems, mainly communications or office automation projects. He was an early user of the UseNet network and taught himself to program in C through working on a variety of open source software. His experience covers roles including pre and post sales support, product development, end user training and management.

Carl now runs his own web solutions development company "Adepteo", where they specialize in intranet and workflow products building on the best open source applications available. Whilst not working or looking after his children, Carl is something of a dance addict and is currently learning Latin Ballroom and Salsa.
Linux E-mail Second Edition

Set up, maintain, and secure a small office e-mail server

Many businesses want to run their e-mail servers on Linux for greater control and flexibility of corporate communications, but getting started can be complicated. The attractiveness of a free-to-use and robust e-mail service running on Linux can be undermined by the apparent technical challenges involved. Some of the complexity arises from the fact that an e-mail server consists of several components that must be installed and configured separately, then integrated together.

This book gives you just what you need to know to set up and maintain an e-mail server. Unlike other approaches that deal with one component at a time, this book delivers a step-by-step approach across all the server components, leaving you with a complete working e-mail server for your small business network.

What This Book Covers

Chapter 1: Linux and E-mail Basics takes you through the essential elements of a Linux e-mail server and the network and mail protocols that make e-mail possible. Like it or not, running a Linux e-mail server does require some understanding of the underlying networking, and this chapter is where you will start to get that understanding. This chapter explains the benefits and disadvantages of running your own e-mail server and provides some guidance on hardware sizing for a typical organization.

Chapter 2: Setting Up Postfix speaks about basic Postfix setup. Postfix is our chosen Mail Transfer Agent (MTA), which forms the heart of any e-mail server. The MTA is responsible, among other things, for moving messages between the various mail servers on the Internet.

Chapter 3: Incoming mail with POP and IMAP covers what to do with incoming e-mails. It will show you how to set up IMAP and POP access to mailboxes. This means users will be able to send and receive messages using their familiar e-mail clients.

Chapter 4: Providing Webmail Access shows how to set up webmail access using SquirrelMail. This will give users an easy, out-of-office access to their e-mail.

Chapter 5: Securing Your Installation looks at how your installation can be secured to prevent misuse of your users' data and the e-mail facility itself.

Chapter 6: Getting Started with Procmail discusses the basics of Procmail and gets you familiar with the various files that Procmail uses to load recipes, the core principles of filtering, and the options available.

Chapter 7: Advanced Procmail explores Procmail and explains a large number of services and a large amount of functionality that it can provide in getting mail under control. It also discusses the advanced features of Procmail and their benefits.

Chapter 8: Busting Spam with SpamAssassin shows the use of SpamAssassin in conjunction with Procmail to filter out the wide range of spam that afflicts the modern e-mail user.

Chapter 9: Antivirus Protection shows another way to protect users from rogue e-mail—this time the spread of e-mail viruses. Using ClamAV you can scan mail for viruses and schedule tasks to maintain an up-to-date antivirus database.

Chapter 10: Backing up your System will show you how to protect all your hardwork by backing up not only the e-mail itself, but also all of the configuration options that make up your e-mail server. Examples are provided to create an automated backup schedule to minimize data loss. Of course, you'll also learn how to restore data from these backups.

Providing Webmail Access

You learned how to set up and configure an e-mail server in the previous chapters. Now that your e-mail server is ready to serve, how will your users access it? In this chapter, you will learn about the following:

- The benefits and disadvantages of a webmail access solution
- The SquirrelMail webmail package
- Setting up and configuring SquirrelMail
- What SquirrelMail plugins are and what they can do
- How to make SquirrelMail more secure

In the next section, we will introduce the SquirrelMail software package and examine the pros and cons of this and other webmail access solutions. After that, we will follow the installation and configuration of SquirrelMail step by step. Next, we will examine the installation of plugins and include a reference of useful plugins. Finally, we’ll include some tips on how to secure SquirrelMail.

The webmail solution

A webmail solution is a program or a series of scripts that is run on a server, is accessible over the web, and provides access to e-mail functions similar to a conventional mail client. It is used by Yahoo! Mail, Microsoft Hotmail, Microsoft Outlook Web Access, and Gmail as the primary interface to their e-mail solutions. You may already be familiar with various forms of webmail.

Though we will be examining the SquirrelMail webmail solution specifically, the benefits and drawbacks of SquirrelMail apply to most webmail systems in the market. From this point of view, we will approach the issue from a general perspective, and then in detail for the SquirrelMail package.

Providing Webmail Access

The benefits

This section will focus on the advantages offered by installing and maintaining a webmail solution. As with any list, it is not entirely comprehensive. Many benefits will be specific to a particular case; it is important to carefully examine and consider how the following qualities impact your individual situation.

The main benefits we will explore in this section are as follows:

- Easy and quick access with little or no setup
- Easy remote access
- No need to maintain client software or configuration
- Provision of a user interface to configure mail server options
- Possible security benefits

Easy and quick access

Although well suited to certain situations, traditional mail access solutions can often be difficult to set up and maintain. Generally, this involves installing software on a client's local computer and configuring it. This can be difficult, especially in cases where users need to set up the software themselves. Configuration can often be even more problematic as some users may not be competent enough to follow even a very detailed set of instructions. These instructions also need to be provided and maintained for many different mail clients on many different platforms.

However, a webmail solution does not have most of these problems. All of the user's settings can be configured on the server as the application itself resides on the server. This translates to almost zero set up time for the user. Once they have received their login credentials, they can visit the webmail site and instantly have access to all of their mail. The user is able to access the site instantly to send and receive e-mail.

As the Internet is so common now, many users will be familiar with webmail sites such as Google Mail and Windows Live Hotmail, which offer free e-mail services. However, the user interface provided by an open source package may be more primitive and lack some visual features. Squirrelmail provides access to e-mail, including the ability to send and receive attachments, and offers a good user interface.

It is also worth mentioning that a webmail solution can offer what certain traditional mail clients call groupware features. These features let groups communicate and coordinate in ways that complement e-mail communication. Examples of groupware components are private calendars, shared calendars, meeting scheduling, To-do lists, and other similar tools.

These applications can be preconfigured so that a user can instantly begin using them without having to configure them on their own. Several SquirrelMail plugins which implement these features are available from the SquirrelMail website.

**Easy remote access**

Another problem with traditional mail access software is that it is not portable, as an e-mail client needs to be installed and configured on a computer. Once it has been downloaded, installed, and configured on a particular computer, it is accessible only on that computer. Without webmail, users on the road will not be able to access e-mail from friends’ computers, mobile devices, or Internet booths at airports.

However, in a webmail solution, e-mail can be accessed from any location with an Internet connection. Employees can access their work e-mail from any computer with an Internet connection and a suitable browser.

As the administrator, you can choose to permit or deny users from accessing e-mail in insecure situations. By requiring the connection to be encrypted, you can ensure that when a user is in a remote location, their communication with the server is secure.

**No need to maintain clients**

Even if software mail clients have been installed and properly configured, they must be maintained. When a new version is released, all clients must be updated. This is not necessarily an easy task. Software that does not work as expected can result in a large number of support-desk calls.

Updating the software on each client can be a very large administrative burden. In fact, many expensive software packages are designed for the specific purpose of updating software on individual machines automatically. Despite this, problems specific to each local machine often arise and must be solved individually. It may also be difficult to convey instructions or notifications to remote branch locations or remote workers. With a webmail solution, this is not necessary.

In contrast to this, a webmail solution is centrally maintained and administered. The webmail application resides on the server. With webmail, only the web server and the webmail package need to be upgraded. Any exceptions or problems that arise can be dealt with before or during the upgrade. The software upgrade itself can be run through on a test system before it is deployed on a live system. Although changes in settings are rare with SquirrelMail, it is possible to update a user's settings to make them compatible with the changes introduced in an updated version.
Providing Webmail Access

Additionally, while upgrading or changing a mail server platform, testing effort can be greatly reduced as only supported browser versions need to be tested. It is advisable to mandate particular browser versions for corporate computers. In contrast with e-mail clients, there is no need to test on all of the possible clients and software platforms.

Configuring mail server interface via the user interface

Many traditional desktop e-mail clients provide only e-mail functionality and nothing more. Often there is no support for other essential tasks (such as changing the access password) that are performed on behalf of a mail user. Certain configuration options that reside on the server may require additional software applications or external solutions to provide for these needs. Examples of mail server options that may need to be configured include each user's password and junk mail filtering settings.

In the case of the SquirrelMail webmail application, many plugins have been developed that provide these features. For example, a user is able to change his/her password directly from the webmail interface. Also, there are plugins and systems that allow users to easily sign up without any direct human intervention. This may be useful if you are interested in providing a service where users can sign up without needing an administrative overhead.

Possible security benefits

This issue can be seen in two different ways—it is for this reason that the title is listed as "Possible" security benefits. Nonetheless, this is still an interesting point to examine.

In the software client access model, e-mail is traditionally downloaded onto the local user's computer, being stored in one or more personal folders. From a security perspective, this may be a bad thing. Users of the system may not be as conscientious or knowledgeable about computer security as a trained computer administrator might be. It is often much easier to gain unauthorized access to an end user's computer than a properly configured and secured server. The implication is that someone who stole a company laptop might be able to access all the e-mail stored on that computer.

There is one more disadvantage associated with the client access model. Even if an employee is terminated, he/she may still have access to all of the e-mail that resides on his/her local office computer. It may take a certain amount of time before important information may be secured. A disgruntled worker might easily connect an external storage source to their local office computer and download any data they desire.

It is also worth noting that in a webmail model, all e-mail is centrally stored. If an attacker were to gain access to the central e-mail server, he/she might access all the e-mail stored on that server. However, it is possible that an attacker will gain access to all the e-mail if the central mail server is compromised even if a webmail system is not used.

The disadvantages
This section focuses on the disadvantages resulting from providing and supporting a webmail solution. The warning given in the previous section applies: This list is not entirely comprehensive. Each situation is unique, and may bring its unique disadvantages.

We will go over the following disadvantages of a webmail solution:

- Performance issues
- Compatibility with large e-mail volumes
- Compatibility with e-mail attachments
- Security issues

Performance
The traditional e-mail client is designed in the client-server model. One mail server accepts and delivers e-mail to and from other mail servers. However, a desktop mail client can offer many additional productivity-enhancing features such as message sorting, searching, contact list management, attachment handling, along with more recent ones such as spam filtering and message encryption.

Each of these features may require a certain amount of processing power. The required level of processing power may be negligible when it comes to storing one user's e-mail on a desktop computer, but providing these features may be problematic when applied on a larger scale to a single server.

When examining the performance issue, it is important to consider the number of potential users that will access the webmail application and size a server accordingly. A single server may be able to easily handle something like 300 users, but if the number of users increases significantly, server load may become an issue.

For example, searching through several years' archived mail may take a few seconds on a client's computer. When one user performs this task using webmail, the load will be similar. However, if many clients request this operation at short intervals or concurrently, it may be difficult for the server to process all the requests in a timely manner. This may result in pages being served at a slower rate or, in extreme circumstances, the server failing to respond.
Providing Webmail Access

Optimally, load testing in the appropriate conditions should be performed if there is any concern that a server may not be able to handle a particular load of users.

Compatibility with large e-mail volumes

The webmail solution is not well suited to large mail volumes. This disadvantage is related to the previous one, but is more related to the amount of data sent. Even with a relatively low number of users, a large volume of e-mails may be difficult to manage in a webmail application. There are mainly the following two reasons for this:

- Firstly, every e-mail viewed and every folder listed must be sent from the server each time. With a traditional e-mail client, the client software can manage e-mail messages, creating lists and views to suit the user. However, with a webmail solution, this is performed on the server. So, if there are many users, this overhead may use a significant proportion of the server's resources.

- Secondly, each interaction with the webmail application requires a Hypertext Transfer Protocol (HTTP) request and response. These messages will typically be larger than those between an e-mail server and a desktop e-mail client. There may also be less parallelism when using a webmail client, in other words, fewer things going on at the same time. A desktop e-mail client may be able to check for new e-mails in several folders at the same time, but a webmail client will typically perform these tasks one after the other, if they occur automatically at all.

Compatibility with e-mail attachments

The webmail solution is not well suited to e-mail attachments. By virtue of the fact that a webmail application resides on a remote server, any and all e-mail attachments must first be uploaded onto that server. For a couple of reasons, it may be difficult or impossible to accomplish this operation with too many attachments or with attachments that are large in size.

Difficulties uploading large attachments may arise due to limited storage space on the webmail server. Large attachments may take a long time to upload over the HTTP protocol and even longer over HTTPS. Additionally, many file size limits may be imposed on uploaded files. PHP, the programming language used with SquirrelMail, imposes a 2MB limit on uploaded files in its default configuration.

The solution to the above problem may lie in the nature of the webmail access solution—e-mail and the mail access software reside on the server. In a traditional mail client, e-mail is often downloaded before the user is aware of the contents or size of the particular e-mail message. As opposed to this, in the case of webmail, the user is able to view e-mail with large attachments without downloading the attachments—a particular benefit to those without high-speed internet connections.

Finally, downloading and uploading large e-mail attachments from the server may cause a performance issue with the user interface. Many users are frustrated by an attachment's upload time in the webmail application, especially as the message cannot be sent until the attachment is uploaded. In a traditional mail client, the attachment is attached instantly, while the message takes time to send.

**Security issues**

The last issue we will examine is the potential for security shortcomings. One important feature of a webmail access solution also creates a potential problem. The benefit of remote access gives way to the potential insecurity of the local machines upon which the user accesses his/her mail.

A computer that is not directly under your control may be controlled by a third-party intent on accessing your information. Normally, a computer does not record a user's individual keystrokes. Internet cafes and kiosks, and even the home computers of employee's could be running malicious software. This malicious software may monitor keystrokes and websites visited. A user must type in his/her password or login credentials to gain access to the system. When these credentials are captured and stored on the computer with malicious software, they can be intercepted and used by third parties for unauthorized access.

Even if we take malicious intent out of the picture, there are still certain situations that may prove to pose security risks. For example, many modern web browsers offer the option of saving a password whenever it is entered. This password is stored on the local computer where the website is visited. If a user logs in to the webmail application and accidentally saves their password on the local computer, this password may be accessible to any user with access to that local computer.

Finally, users may inadvertently leave themselves logged in to the webmail application. Without logging out, any user with access to that specific computer might be able to gain access to the user's mail account.

The SquirrelMail webmail package

The following screenshot shows the SquirrelMail login screen:

SquirrelMail was chosen based on the combination of the following features it provides:

- It is a proven, stable, and mature webmail platform.
- It has been downloaded over two million times.
- It is standards-based and renders pages in pure HTML 4.0 without requiring the use of JavaScript.

SquirrelMail also includes the following features (and many more, via the flexible plugin system):

- Strong MIME support
- Address book functionality
- A spell checker
- Support for sending and receiving HTML e-mail
- Template and theme support
- Virtual host support

The following screenshot shows an inbox where you can see some of these features:

![SquirrelMail inbox screenshot]

**SquirrelMail installation and configuration**

SquirrelMail installation and configuration may seem daunting if you are not familiar with installing web applications. But by following the instructions to be discussed next, SquirrelMail can be installed without difficulty.

**Prerequisites to installation**

SquirrelMail requires both PHP and a web server that supports PHP scripts to be installed before proceeding. In our case, we will be using the Apache2 web server, although others will work as well.

First, we will go over the basic requirements, and what to do if you do not meet them. Then, we will go over some more advanced requirements that may impact on certain features within SquirrelMail.

---

Providing Webmail Access

Basic requirements
At the time of writing, the most current stable version of SquirrelMail available is 1.4.19. The following instructions apply to this version. There are two basic requirements for a SquirrelMail installation.

Installing Apache2
Any modern version of Apache that supports PHP, either the 1.x or 2.x series, will do the trick. Here we provide instructions for using Apache2. To query for an Apache installation on an RPM package management-based system, issue the following command at the prompt:

```
$ rpm -q apache
  apache-1.3.20-16
```

If, as in the example just seen, a version of Apache is returned, then the Apache web server is installed on your system.

To query for an Apache installation on a Debian package management-based system, issue the following command at the prompt:

```
$ apt-cache search --installed apache2 | grep HTTP
```

```
libapache2-mod-evasive - evasive module to minimize HTTP DoS or brute force attacks
libpoe-component-server-http-perl - foundation of a POE HTTP Daemon
libserf-0-0 - high-performance asynchronous HTTP client library
libserf-0-0-db - high-performance asynchronous HTTP client library debugging symbols
libserf-0-0-dev - high-performance asynchronous HTTP client library headers
nanoweb - HTTP server written in PHP
php-auth-http - HTTP authentication
apache2 - Apache HTTP Server metapackage
apache2-doc - Apache HTTP Server documentation
apache2-mpm-event - Apache HTTP Server - event driven model
apache2-mpm-prefork - Apache HTTP Server - traditional non-threaded model
apache2-mpm-worker - Apache HTTP Server - high speed threaded model
apache2.2-common - Apache HTTP Server common files
```

Similar commands are available for other distributions using other package management systems.

If you do not have an Apache installation present, it is best to first look into your distribution for a copy of Apache—such as on your operating system installation CDs or using an online package repository. Alternatively, you may visit the home page for the Apache foundation at http://www.apache.org.

**PHP**

The PHP programming language (version 4.1.0 or greater, including all PHP 5 versions) is required in order to install SquirrelMail. To check if your system has PHP installed, simply attempt to run it with the following command:

```bash
$ php -v
```

If the command succeeds, you will see a message describing the version of PHP that is installed. If PHP version 4.1.0 or higher is present, then your system has the required software. Otherwise, you will need to install or upgrade your current installation. As with Apache, it is best to look to your distribution for a copy to install. Alternatively, you may also visit http://www.php.net.

**Perl**

The Perl programming environment is not required for SquirrelMail, but having it available makes configuration of SquirrelMail much simpler. In this chapter, we assume that you will have Perl accessible to enable easy configuration of SquirrelMail.

To query for a Perl installation on an RPM-based system, simply attempt to run it with the following command:

```bash
$ perl -v
```

If the command succeeds, you will see a message describing the version of Perl that is installed.

If any version of Perl is present, your system has the required software. Otherwise, you will need to install or upgrade your current installation. As with Apache, it is best to look into your distribution for a copy to install. Alternatively, you may also visit http://www.perl.com/get.html.

**Review configuration**

You will need to review the PHP configuration file `php.ini` to ensure that settings are correct. On most Linux systems, this file may be found at `/etc/php.ini`.

---

Providing Webmail Access

php.ini is a text file and can be edited with a text editor such as Emacs or vi.
Firstly, if you want users to be able to upload attachments, make sure that the
option file_uploads is set to On:

    ; Whether to allow HTTP file uploads.
    file_uploads = On

The next option within the php.ini file you may want to change is
upload_max_filesize. This setting applies to uploaded attachments and
determines the maximum file size of an uploaded file. It may be helpful to
change this to something reasonable, such as 10M.

    ; Maximum allowed size for uploaded files.
    upload_max_filesize = 10M

Installing SquirrelMail

SquirrelMail may be installed either through a package or directly from source. While
no source code compilation takes place in either method, upgrades are made easier
using the packages.

Many of the various Linux and Unix distributions include the SquirrelMail package.
Install the appropriate package from your distribution to use the binary method. On
many Linux distributions, this may be an RPM file that begins with squirrelmail....

However, an updated version of SquirrelMail may not be included or available for
your specific distribution.

The following are the advantages of using the version of SquirrelMail provided with
a Linux distribution:

- It will be very simple to install SquirrelMail.
- It will require much less configuration as it will be configured to use the
  standard locations chosen by your Linux distributer.
- Updates will be very easy to apply, and migration issues may be dealt with
  by the package management system.

The following are the disadvantages of using the version of SquirrelMail provided
with a Linux distribution:

- It may not be the latest version. For example, a more recent version that may
  fix a security vulnerability may have been released, but Linux distributors
  may not have created a new package yet.

Sometimes Linux distributions alter packages by applying patches. These patches may affect the operation of the package, and may make getting support or help more difficult.

Source installation

If you do not install SquirrelMail through your distribution, you will need to obtain the appropriate tarball. To do so, visit the SquirrelMail website at http://www.squirrelmail.org, and click download it here. At the time of writing, this link is http://www.squirrelmail.org/download.php.

There are two versions available for download, a stable version and a development version. Unless you have specific reasons for choosing otherwise, it is generally best to choose the stable version. Download and save this file to an intermediate location.

$ cd /tmp

Next, unpack the tarball (.tar.gz) file. You may use the following command:

$ tar xzf squirrelmail-1.4.19.tar.gz

Move the folder just created to your web root folder. This is the directory from which Apache serves pages. In this case, we will assume that /var/www/html is your web root. We will also rename the clumsy squirrelmail-1.4.3a folder to a more simple mail folder. You will need to have superuser root privileges in order to do this on most systems.

# mv squirrelmail-1.4.19 /var/www/html/mail
# cd /var/www/html/mail

Here we have used the name mail, so the URL that users will use will be http://www.sitename.com/mail. You can choose another name, such as webmail, and use that directory name instead of mail in the commands that you enter.

It is also useful and secure to create a data directory for SquirrelMail that is outside the main web root, so that this folder will be inaccessible from the Web.

# mv /var/www/html/mail/data /var/www/sqmddata

It is important to make this newly created folder writable by the web server. To be able to do this, you must know the user and group that your web server runs under. This may be nobody and nobody, apache and apache, or something else. You will want to verify this; it will be listed in your httpd.conf file as the User and Group entries.

# chown -R nobody:nobody /var/www/sqmddata

Finally, we will create a directory to store attachments. This directory is special in that, although the web server should have write access to write the attachments, it should not have read access. We create this directory and assign the correct permissions with the following commands:

```
# mkdir /var/www/sqmdata/attachments
# chgrp -R nobody /var/www/sqmdata/attachments
# chmod 730 /var/www/sqmdata/attachments
```

SquirrelMail has now been properly installed. All of the folders have been set up with correct permissions that will secure intermediate files from prying eyes.

If a user aborts a message that contains an uploaded attachment, the attachment file on the web server will not be removed. It is a good practice to create a cron job on the server that erases excess files from the attachment directory. For example, create a file called `remove_orphaned_attachments` and place it in the `/etc/cron.daily` directory. Edit the file to have these lines:

```
#!/bin/sh

rm "find /var/www/sqmdata/attachments -atime +2 | grep -v "\."| grep -v _""
```

This will run daily and search the SquirrelMail attachments directory for files which are orphaned, and delete them.

### Configuring SquirrelMail

SquirrelMail is configured through the `config.php` file. To aid the configuration, a `conf.pl` Perl script has also been provided. These files are located within the `config/` directory in the base installation directory.

```
# cd /var/www/html/mail/config
# ./conf.pl
```
Once you have run this command, you should see the following menu:

To select an item from the menu, enter the appropriate letter or number, followed by the Enter key. As SquirrelMail has been developed, it has been noticed that IMAP servers don’t always behave in the same way. To get the most out of your setup, you should tell SquirrelMail which IMAP server you are using. To load a default configuration for your IMAP server, enter the D option and type the name of the IMAP server that you have installed. This book covers the Courier IMAP server, so you should choose that. Press Enter again, and you will return to the main menu.

We will be moving through the various subsections of the menu and configuring the appropriate options.

Type 1 and then press Enter to select the Organization Preferences. You will get a list of items you can change. You may wish to edit the Organization Name, Organization Logo, and Organization Title fields. Once you have modified these to your satisfaction, enter R to return to the main menu.

After this, type 2 to visit the Server Settings. This allows you to set the IMAP server settings. It is important that you update the Domain field to the proper value.

Providing Webmail Access

In our case, the **Update IMAP Settings** and **Update SMTP Settings** values should be correct. If you would like to use an IMAP or SMTP server that is located on a different machine, you may wish to update these values.

Press \( R \) followed by the \( Enter \) key to return to the main menu.

Next, type 4 to visit the **General Options**. You will need to modify two options in this section.

- Data Directory to be /var/www/sqmdata.
- Attachment Directory to be /var/www/sqmdata/attachments.
- Type in \( R \) followed by the \( Enter \) key to return to the main menu. Enter \( S \) followed by the \( Enter \) key twice to save the settings to the configuration file. Finally, enter \( Q \) followed by the \( Enter \) key to exit the configuration application.

We have finished configuring the SquirrelMail settings needed for basic operation. You may return to this script at any time to update any settings you have set. There are many other options to set, including those regarding themes and plugins.

**SquirrelMail plugins**

Plugins are pieces of software that extend or add functionality to a software package. SquirrelMail was designed from the ground up to be very extensible, and includes a powerful plugin system. Currently, there are over 200 different plugins available on the SquirrelMail website. They may be obtained at [http://www.squirrelmail.org/plugins.php](http://www.squirrelmail.org/plugins.php).

The functionality they provide includes administration tools, visual additions, user interface tweaks, security enhancements, and even weather forecasts. In the following section, we will first go over how to install and configure a plugin. After that, we'll go over some useful plugins, what they do, how to install them, and more.

**Installing plugins**

These SquirrelMail additions were designed to be simple to set up and configure. In fact, the majority of them follow exactly the same installation procedure. However, a few require custom setup instructions. For all plugins, the installation process is as follows:

1. Download and unpack the plugin.
2. Perform custom installation if needed.
3. Enable the plugin in `conf.pl`.

Example plugin installation

In this section, we will go over the installation of the Compatibility plugin. This plugin is required in order to install plugins created for older versions of SquirrelMail. No matter how bare-bones your installation, the Compatibility plugin will most likely be part of your setup.

Downloading and unpacking the plugin

All available plugins for SquirrelMail are listed on the SquirrelMail website at http://www.squirrelmail.org/plugins.php.

Certain plugins may require a specific version of SquirrelMail. Verify that you have this version installed. Once you have located a plugin, download it to the plugins/ directory within the SquirrelMail root folder.

You may locate the Compatibility plugin by clicking on the Miscellaneous category in the plugins page on the SquirrelMail plugins web page. This page has a list of plugins in the Miscellaneous category. Locate Compatibility and click on Details and downloads, and then download the latest version.

Providing Webmail Access

Download tarball to your SquirrelMail plugin directory.

# cd /var/www/mail/plugins

Once you have downloaded the plugin to the plugins directory, unpack it using the following command:

# tar zxvf compatibility-2.0.14-1.0.tar.gz

If a plugin of the same name has already been installed, its files may be overwritten. Verify that you either do not have a plugin of the same name, or save the files before you unpack the tarball.

Performing custom installation

The current version of the Compatibility plugin does not require any additional configuration. However, you should always check the documentation for a plugin, as certain other plugins may require custom installation. Once you have unpacked the plugin package, the installation instructions will be listed in the INSTALL file within the newly created plugin directory. It is advisable to check the installation instructions before enabling the plugin in the configuration manager, as some plugins may require custom configuration.

Enabling the plugin in conf.pl

Within the main menu of the configuration editor, option number 8 is used to configure and enable plugins. Start conf.pl and select option 8.

# cd /var/www/mail/plugins
# cd ../config
# ./conf.pl

SquirrelMail Configuration : Read: config_default.php (1.4.0)
------------------------------------------------------------------
Main Menu --
[...]
7. Message of the Day (MOTD)
8. Plugins
9. Database
[...]
Command >>

Chapter 4

You should get the following display when you select this option for the first time:

All the plugins that have been installed and enabled are listed under the **Installed Plugins** list. All the plugins that have been installed but not enabled are listed under the **Available Plugins** list.

Once you have unpacked a plugin within the `plugins/` directory, it will show up under Available Plugins. As you can see in the previous figure, there are a number of installed plugins, but none of them are enabled. As a malfunctioning or wrongly configured plugin can cause SquirrelMail to stop functioning properly, it is advisable to enable plugins one by one, and verify that SquirrelMail works after each one.

To enable the Compatibility plugin, locate it in the list **Available Plugins** (in this case, number 4) and press the `Enter` key. The Compatibility plugin is now installed. Plugins can be disabled by locating them in the **Installed Plugins** list and entering their number and pressing `Enter`.

Useful plugins

We’ll now see some useful SquirrelMail plugins that you may consider installing.

The information has been compiled to provide a helpful reference while deciding whether to install a plugin. Each plugin contains four specific categories:

- **Category**: The category in which the plugin is listed on the SquirrelMail site
- **Authors**: Authors who wrote the plugin, in chronological order
- **Description**: A short description of the plugin's functionality
- **Requirement**: A list of prerequisites for the plugin's successful installation

<table>
<thead>
<tr>
<th>Plugin name</th>
<th>Category</th>
<th>Author(s)</th>
<th>Description</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compatibility plugin</td>
<td>Miscellaneous</td>
<td>Paul Lesneiwski</td>
<td>This plugin allows any other plugin access to the functions and special variables needed to make it backward (and forward) compatible with most versions of SM in wide use. This eliminates the need for duplication of certain functions throughout many plugins. It also provides functionality that helps in checking whether the plugins have been installed and set up correctly.</td>
<td>Nothing</td>
</tr>
<tr>
<td>Secure login</td>
<td>Logging in</td>
<td>Graham Norbury, Paul Lesneiwski</td>
<td>This plugin automatically enables a secure HTTPS/SSL-encrypted connection for the SquirrelMail login page if it hasn't already been requested by the referring hyperlink or bookmark. Optionally, the secure connection can be turned off again after successful login.</td>
<td>SquirrelMail version 1.2.8 or above, HTTPS/SSL-capable web server with encryption already working on your SquirrelMail installation.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Plugin name</th>
<th>Category</th>
<th>Author(s)</th>
<th>Description</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP authentication</td>
<td>Logging in</td>
<td>Tyler Akins, Paul Lesniewski</td>
<td>If you keep SquirrelMail behind a password-protected directory on your web server and if PHP has access to the username and password used by the web server, this plugin will bypass the login screen and use that username/password pair.</td>
<td>SquirrelMail &gt;= 1.4.0</td>
</tr>
<tr>
<td>Password forget</td>
<td>Logging in</td>
<td>Tyler Akins, Paul Lesniewski</td>
<td>This plugin provides a workaround for the potential vulnerability of browsers, automatically storing usernames and passwords entered into web pages.</td>
<td>SquirrelMail &gt;= 1.0.1</td>
</tr>
<tr>
<td>HTML mail</td>
<td>Compose</td>
<td>Paul Lesniewski</td>
<td>This plugin allows users with IE 5.5 (and up) and newer Mozilla (Gecko-based browsers such as Firefox) browsers to compose and send their e-mail in HTML format.</td>
<td>SquirrelMail &gt;= 1.4.0</td>
</tr>
<tr>
<td>Quick save</td>
<td>Compose</td>
<td>Ray Black III, Paul Lesniewski</td>
<td>This plugin automatically saves messages as they are being composed, in order to prevent accidental loss of message content due to having browsed away from the compose screen or more serious problems such as browser or computer crashes.</td>
<td>SquirrelMail &gt;= 1.2.9, the Compatibility plugin, JavaScript-capable browser</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Plugin name</th>
<th>Category</th>
<th>Author(s)</th>
<th>Description</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check quota usage (v)</td>
<td>Visual additions</td>
<td>Kerem Erkan</td>
<td>This plugin will check and display users' mail quota status.</td>
<td>SquirrelMail 1.4.0+; Compatibility plugin, version 2.0.7+, UNIX, IMAP or cPanel quotas installed and configured</td>
</tr>
<tr>
<td>Sent confirmation</td>
<td>Miscellaneous</td>
<td>Paul Lesneiwski</td>
<td>Displays a confirmation message after a message is successfully sent, as well as other features.</td>
<td>SquirrelMail &gt;= 1.2.0, the Compatibility plugin</td>
</tr>
<tr>
<td>Timeout user</td>
<td>Miscellaneous</td>
<td>Ray Black III, Paul Lesneiwski</td>
<td>Automatically logs out a user if they are idle for a specified amount of time.</td>
<td>The Compatibility plugin</td>
</tr>
<tr>
<td>E-mail footer</td>
<td>Miscellaneous</td>
<td>Ray Black III, Paul Lesneiwski</td>
<td>This plugin automatically appends a custom footer onto the end of messages sent using SquirrelMail.</td>
<td>SquirrelMail &gt;= 1.4.2</td>
</tr>
<tr>
<td>Change password</td>
<td>Change password</td>
<td>Tyler Akins, Seth E. Randall</td>
<td>Allows a user to change their password using PAM or Courier authentication modules.</td>
<td>SquirrelMail &gt;= 1.4.0</td>
</tr>
<tr>
<td>Address book import-export</td>
<td>Address book</td>
<td>Lewis Bergman, Dustin Anders, Christian Sauer, Tomas Kuliavas</td>
<td>Allows the importing of address books from a CSV (comma separated values) file.</td>
<td>SquirrelMail &gt;= 1.4.4</td>
</tr>
<tr>
<td>Plugin updates (v0.7)</td>
<td>Administrator's Relief</td>
<td>Jimmy Conner</td>
<td>Checks for updates to your currently running plugins.</td>
<td>SquirrelMail &gt;= 1.4.2</td>
</tr>
</tbody>
</table>

Many other plugins exist that handle vacation messages, calendars, shared calendars, notes, to-do lists, exchange server integration, bookmarks, weather information, and much more. Check the Plugins section in the SquirrelMail website for all of the available plugins.
Securing SquirrelMail

The SquirrelMail package, in and of itself, is fairly secure. It is well written and does not require JavaScript to function. However, there are a few precautions that may be taken to allow SquirrelMail to run as a secured mail handling solution.

- **Have an SSL connection**: By using an SSL connection, you may be certain that all communications will be encrypted, and so usernames, passwords, and confidential data cannot be intercepted during transmission. This may be accomplished through the installation of the Secure Login plugin. Obviously a web server configured for secure SSL access will also be required; certificates will most likely need to be generated or acquired.

- **Time out inactive users**: Users may leave themselves logged in and neglect to log out once they are finished. To fight this, inactive users should be logged out after a certain amount of time. The Timeout User plug-in accomplishes this.

- **Fight "Remembered Passwords"**: Many modern-day browsers offer to remember a user's password. Although a convenience, this may be a large security vulnerability, especially if the user is located at a public terminal. To fight this, install the Password Forget plugin. This plugin will change the names in the username and password input fields, to make it more difficult for a browser to suggest them to future users.

- **Do not install security-compromising plugins**: Plugins such as Quick Save, HTML Mail, and View As HTML may compromise security.

Summary

Now that you've finished this chapter, you should have a working SquirrelMail installation as well as a greater understanding of the benefits and disadvantages of a webmail solution. You should be familiar with the benefits and drawbacks of a webmail solution. The benefits include remote access, a single central point to be maintained, and simpler testing; while disadvantages include potential performance problems and the security risk of allowing remote access from potentially compromised computers.

You are now aware of the main features of SquirrelMail, including its flexibility and the availability of plugins, along with what the prerequisites for installing SquirrelMail are, and how to identify if they are already installed.

You also have learned how to configure SquirrelMail, including locating, installing, and configuring plugins. You have been walked through the installation of a key plugin; the Compatibility plugin. Several other useful plugins have also been introduced. Finally, you have learned about some ways to improve the security of SquirrelMail, including web server configuration and some appropriate plugins.

Where to buy this book


Free shipping to the US, UK, Europe and selected Asian countries. For more information, please read our shipping policy.

Alternatively, you can buy the book from Amazon, BN.com, Computer Manuals and most internet book retailers.