Utilizing Open-Source Software to Build a (Relatively) Secure, Spam- and Virus-Free Mail Service

GIAC Security Essentials Certification
(GSEC) Practical Assignment
Assignment version 1.4b (amended August 29, 2002)
Option 1 – Research on Topic in Information Security

Prepared by: David R. Bailey

Date Submitted: April 16, 2004

Table of Contents

1.0 Introduction

Electronic mail (email) services have become critical to survival, whether a commercial business, non-profit organization, or government agency, in today's information-centric world. There are a myriad of solutions for providing email services, some are cost-effective and some are cost-prohibitive. Typically, the best solutions for providing email services are either cost-prohibitive or technically-complex or both, while the lower-cost or lower-complexity solutions are often lacking features that most will agree are critical in email services today such as effective virus and spam controls[1].

This document shows an example of a relatively easy-to-implement, professional-looking, reasonably-secure solution that offers several features that are very useful and are often out-of-reach, either due to cost or technical expertise required, for small to medium sized organizations.[2] This solution should work well for a few hundred, but not thousands, of users with only a small amount of tweaking to improve performance depending on how many users will be attaching at once and the performance of the hardware on which it is deployed.

This solution showcases several technologies that are relatively new (as standard features) and useful in Linux environments. This includes technologies such as- virus and spam blocking before a mail message has been received and acknowledged; requiring longer and more-complex user passwords which are encrypted using more-secure encryption technology; allowing users to change passwords securely from the web mail interface; utilizing ReiserFS with Logical Volume Manager with Exim and Courier-IMAP for high-performance message handling and processing; and fully-automated patch management for the operating system and most of the applications deployed.

[1] In this document, we're not talking about SPAM® or canned meat, we're talking about spam, the junk mail (otherwise known as "unsolicited commercial email") that you get in your email box. SPAM® is a registered trademark of Hormel Foods Corporation. For more about this strange nexus of meat and mail, check out this website. "SPAM and the Internet." http://www.spam.com/ci/ci_in.htm

[2] The other option that may be open to medium to small organizations is to subcontract email services out to a professional organization which can set up similar services utilizing your own Internet domain name, which can either forward the messages on to a local mail system, or simply host them entirely on their servers for you. The downside is the regular fees for doing so, which can increase depending on the number of accounts you wish to maintain.

2.0 Goals

This document gives an example of how to create an email system that provides the following features:

(Remember, you might want to add the word “relatively” to the beginning of most of the following items. Some of these problems have not been completely solved yet, and in some cases, might not be for a while.)

1. Not too difficult to set up. This means no downloading of code, patching, and compiling. There are still some configuration files to edit. (Until mail systems, TCP/IP, DNS, and SMTP are made easy or until they are replaced, no Internet email system will really be easy to set up.)
2. Easy to maintain. In fact, security patches should largely be applied automatically in a secure fashion.
3. Secure. Although perhaps not fully hardened, this server should provide a challenge to those who wish to use it without authorization.
4. Easy web access to email services so that no local client (other than a web browser) is required[3]. This avoids many security issues because there is very little chance of a virus impacting the client system.
5. Successful anti-spam services, including the ability for users to tag messages as spam or not, to improve spam recognition.
6. Virus-free email services that can compare with some of the better commercial services available and includes automatic update of the virus signature database.
7. Cost $0 in licensing or maintenance fees, although I highly recommend supporting these outstanding open-source initiatives if you find them to be useful.

[3] Additional software could be added to support remote clients such as secure IMAP clients.

3.0 Requirements

Some familiarity with Linux and mail servers is required. You should be somewhat familiar with the Linux installation process and how to utilize the command-line interface. You should understand some terminology for Linux and mail services.

This is a single-server solution for a small-to-medium sized organization. Any modern low-end server[4] available for $1,000-2,000 USD should be capable of running this solution for hundreds of users, depending on the workload, with a small amount of performance tuning. Older hardware can be used to the same effect but it would support fewer users. If you are going to use older hardware, the best upgrades are more memory, more memory, and perhaps a larger, faster hard drive. (Yes, more memory is in there twice.)

This solution should be set up and run behind a dedicated firewall on a network. Preferrably, the server would be placed on a DMZ network isolated from the rest of the internal network. This would limit damage to the internal network if the server was compromised. The details on how to configure the firewall for this purpose are beyond the scope of this document. Information on deploying a firewall can be found at the following footnote[5]. After configuration is complete, the server will require TCP ports 25, 80, and 443 to be opened to the Internet. Ports 22 and 10000 can be opened to allow remote management.

This solution should be run on a box that is not also performing other tasks. The more services that are running on a single box the more it increases the chances for security vulnerabilities through cross-application vulnerabilities. If you have a high-end box that you want to utilize for more than just a mail server, consider purchasing a copy of VMware GSX Server[6], which can allow you to run multiple virtual servers on a single system.

To run this server as a real Internet mail server, you will need a valid Internet domain name service (DNS) domain name and a valid host and mail record ('A' and 'MX' records). For the purposes of this example the Internet service provider (ISP) will maintain the DNS records for this mail server. How to set this up is beyond the scope of this document, but if you need more information, you can contact your ISP about this.

To follow these directions you will need an additional computer system with web browsing capabilities. It does not matter what operating system or browser it is running as long as it is capable of processing modern XHTML. I was successful working with Internet Explorer 6 and Mozilla 1.6.

[4] For example, a Dell PowerEdge 700 with a 2.8 GHz Pentium 4 processor, 512MB of 400MHz DDR RAM, and an 18GB 15,000 RPM Ultra 320 SCSI Hard Drive is found for $919 USD on www.dell.com.

[5] CERT Coordination Center. “Deploying Firewalls.” 20 Apr. 2001. URL: http://www.cert.org/security-improvement/modules/m08.html (03 Apr. 2004).

[6] Information on Vmware GSX Server can be found here- http://www.vmware.com/products/server/gsx_features.html

4.0 Software Selection

To an organization without dedicated computer systems security staff or without in-house, high-level Linux/UNIX expertise, simplicity is one of the best recipes for security. The simpler something is to install and maintain, the less chances for unknown security holes to wreak havoc. Many of the tools chosen have been chosen for their ease-of-configuration and ease-of-maintenance. These factors outweigh scalability, extensibility, and custom-configurability because this document assumes a relatively small implementation.

The major software packages selected for this solution are:

1. Linux as the operating system. Linux has shown itself to be inexpensive, compatible with most popular hardware, and relatively secure compared to many of the other alternatives available. In this case, we'll be using SUSE Linux[7] Professional 9.0. SUSE Linux is a GPL[8] product that comes with excellent automated update capabilities and a wide selection of included software and, therefore, built-in support for and updates for most of the tools listed below.

2. Exim[9] as the message transfer agent (MTA) software. Exim has proven to be secure[10], easy-to-configure, and can support thousands of accounts on modern hardware, although we will not be implementing that level of scalability in this solution.

3. Courier-IMAP[11] offers a secure, high-performance interface for our web-mail interface. Also, if you plan to attach IMAP clients (not covered in this document), Courier-IMAP can do secure IMAP (IMAPS) which is critical for remote mail clients.

4. Apache[12], SquirrelMail[13], and Webmin[14] as the web server, web-mail interface, and web administration software. All are widely accepted, offer good performance, and are kept up with timely patches.

[7] Information on SUSE Linux can be found here- http://www.suse.com

[8] SUSE's formerly proprietary configuration utility, YaST, has been (will soon be) licensed using the GPL license. Therefore, the entire SUSE Linux operating system is within the free license realm. See http://www.novell.com/news/press/archive/2004/03/pr04026.html for the YaST open-source announcement and http://www.gnu.org/copyleft/gpl.html for the GPL license.

[9] Information on the Exim MTA can be found here- http://www.exim.org/

[10] Searches on the CERT advisory database show only one Exim vulnerability listed. Other advisories for Exim are on software related to Exim such as OpenSSL. http://www.kb.cert.org/vuls/id/283723. Also Exim does not require additional software to utilize antivirus and antispam technology such as procmail.

[11] Information on Courier-IMAP can be found here- http://www.inter7.com/courierimap.html

[12] Information on the Apache can be found here- http://www.apache.org/

[13] Information on SquirrelMail can be found here- http://www.squirrelmail.org/

[14] Information on Webmin can be found here- http://www.webmin.com/

5. SpamAssassin[15] as the spam filter. SpamAssassin is an amazing tool that allows you to filter spam based on many different rules including Bayesian filtering, which has shown to be a highly effective method for controlling spam.
6. ClamAV[16] as the anti-virus software. Several large companies utilizing open-source desktops and servers use ClamAV as their anti-virus software. It has shown to be effective and its virus definitions database is kept current.
7. Exiscan-ACL[17] is a message handling interface between Exim and content checkers, such as SpamAssassin and ClamAV. Exiscan can block viruses and spam at the time the SMTP message is received, not as a response after the message has been received and the SMTP connection closed.
8. Poppassd[18] is a secure method for users to change their password from the web mail interface.
9. IPTables with Netfilter as our firewall[19] and Bastille Linux[20] hardening scripts will tighten up security on our server. We will not be fully exploiting all of these capabilities but will use them to show some of what is possible.

[15] Information on SpamAssassin can be found here- http://www.spamassassin.org/
[16] Information on ClamAV can be found here- http://www.clamav.net/
[17] Information on Exiscan-ACL can be found here- http://duncanthrax.net/exiscan-acl/
[18] Information on Poppassd can be found here- http://echelon.pl/pubs/poppassd.htmlhttp://echelon.pl/pubs/poppassd.html
[19] Information on IPTables/Netfilter can be found here- http://www.netfilter.org/
[20] Information on Bastille Linux can be found here- http://www.bastille-linux.org/

5.0 SUSE Linux

5.1 Introduction to SUSE Linux

SUSE Linux is widely respected as a professional distribution which is easy to install (from the CDs or *after* you get the network installation started) and one of the quickest to issue security patches.

Although SUSE Linux has long been well-known and respected in the European and Asian markets for some time, it has been rapidly increasing in marketshare in American markets[21].

5.2 Starting the Installation from the Internet

Although you *can't* currently download the SUSE Linux Professional 9 CD image for free, you *can* install it freely directly from the Internet. (Although it is easier if you purchase and install from the CDs, our goal is to do this cost free. If you use the CDs to install, skip ahead to the next section once the graphical installer comes up.)

- First, using the following URL, locate the fastest SUSE mirror to your location.

```
http://www.suse.com/us/private/download/ftp/int_mirrors.html
```

One easy way to do this, though probably not the most accurate way, is to ping each of these until you find one that responds quickly (less than 100ms). Another choice is to simply pick the nearest mirror, and hope it's not too busy.

- Click the link to your architecture, which is typically i386, then 9.0, and the "boot" directory. (IE- i386/9.0/boot)
- Write down the SUSE FTP mirror server and directory. We'll need it later in the installation process.
- Download the boot disk image and the required module disk images (bootdisk, modules1, and modules3), and utilizing the steps shown below, we'll create the boot disks[22]. If you are using a server that needs SCSI drivers, check the README to find out which disk may include all of the drivers you need and create these as well.
- In Windows, download and use the Rawrite utility to create the disk. You can use the graphical interface to open each disk image file and write it to the floppy disk.

[21] Recently, Novell acquired SUSE giving it a world-wide reach for sales and service. See http://www.novell.com/news/press/archive/2004/01/pr04003.html for details.

[22] If you prefer, you can download the boot CD image (boot.iso) and utilize your CD-burning software write it to a CD blank. Due to the different CD burning software available, instructions are not given in this document how to do this. Check your software documentation if you desire to use this approach.

- In Linux, use the dd command to create the disks in this fashion:

```
dd if=path/to/file of=/dev/fd0 bs=36b
```

- Start your system using the boot disk (or CD). Select Manual Installation. Insert the Modules1 disk when prompted. Select the language and keyboard.

Now before we start the installation, we’ve got to add the network driver because that’s where we’re getting the installation files.

- Select “System Information”, “PCI”, and scroll down to the network information and jot down the network adapter type and model. Press Enter and select “Back to the Main Menu”.
- Select “Kernel modules”, “Load network card modules”, insert a disk if requested, then select your network adapter from the list. You shouldn’t need to add any arguments. Just press enter. Go back to the Main Menu.

(If you need to load any SCSI drivers or any drivers for any other non-standard hardware, you can do it now from the Kernel modules menu.)

- Select “Start installation / system”, “Start installation / update”, “Network”, “FTP”, and “Yes” to automatic configuration (or enter manual information if you don’t have a DHCP server.)
- Enter the DNS name or IP address of the FTP mirror and the directory where SUSE was installed. Select “No” to use anonymous access, and “No” to use an HTTP proxy. If it gives an error, try it again, or select a different mirror site.
- Enter the path where the SUSE files are kept. Be sure to enter the path to the mirror you’ve chosen without the leading slash. For example[23]-

```
pub/linux/suse/i386/9.0/
```

After downloading the installation files, the easy to use YaST installer starts up.

5.3 Installing SUSE Linux 9

Now that we’re in the easy graphical installer, YaST, we can really begin the Linux installation process.

- Remove any floppy from the floppy disk drive (or CD from the CD-ROM drive).
- Select your preferred language.
- Allow any requested kernel modules to load to assist with hardware detection.

[23] I had some problems with the installer when using a leading slash for the path- the packages wouldn’t show properly. Apparently, depending on the FTP server, you might need to add or remove the leading slash. Having a closing slash worked for me. If the packages don’t show, abort the installation, restart, and try a different path or a different server.

5.3.1 Hard Disk Partitioning

WARNING: The following steps will erase all data on the disk system on the computer that is having Linux loaded on it. Do not follow these steps unless you do not mind losing all data on the disk(s) assigned to the system.

Hard disk partitioning is an important part of security on Linux. If certain parts of the disk fill up it will make it hard to recover without starting from another filesystem (such as a bootable CD) and removing files manually. One good way to both create partitions and manage your disk space effectively, is to use the Logical Volume Manager (LVM). While it is especially useful managing a SAN or RAID system, even when using a single IDE disk, it can be useful to assign additional storage to partitions if you need more space, and is especially easy to do with the Reiser filesystem[24]. SUSE Linux has excellent support for LVM and ReiserFS in its YaST installation utility.

Some of the benefits of this configuration include the ability to add available storage to a needed logical volume and even the ability to add more storage capacity and merge it into an existing logical volume. This gives us much more flexibility than physical partitions.

This setup assumes a single disk system- either a single physical disk or a single hardware RAID array. Other configurations will require different settings.

- Click “Partitioning”.
- Click “Create custom partition setup”, then “Next”.
- Click “Custom Partitioning”, then “Next”.
- Click on the hard disk (typically /dev/hda or /dev/sda) and click “Delete”.
- Click “Yes” to delete all hard disk partitions. (This removes all existing partitions.)
- Click “Create”, “Primary Partition”, then “OK”.
- Select “Format”
- Use the following settings-
  - Filesystem- Ext2
  - Start cylinder- 0
  - End- +50M
  - Mount point- /boot
- Click “OK”. (Now we have a boot partition to start the system from.)
- Click “Create”, “Primary Partition”, and “OK”.
- Click “Do not format”, change the “File System ID” to 0x8E, and Click “OK”.
- Click “LVM” and “OK”. (Now we can manage the rest of the drive as logical disk space.)
- Click “Add Volume” and “Next”.

[24] The Linux Documentation Project has an excellent “howto” for LVM at- http://tldp.org/HOWTO/LVM-HOWTO/index.html

- Now create the following volumes and partitions by clicking “Add”, then entering the following information and closing each with OK.
  (Format, Logical Volume Name, Size, Mount)
  - Swap[25], swap, (Twice physical memory, but no more than 2 GB), swap
  - Reiser, root, 1 GB, /
  - Reiser, usr, 2 GB, /usr
  - Reiser, var, 1 GB, /var
  - Reiser, tmp, 1 GB, /tmp
  - Reiser, home, (figure an amount per user), /home

Remember that you do not have to (nor do you necessarily want to) allocate all available storage to the Logical Volumes, because you can later add storage to your logical volumes if needed.

Once done with your logical volumes, click Next to continue and Next to return to the installation settings.

5.3.2 Boot Loader Configuration

If the boot loader is not already set to GRUB, we’ll configure the boot loader. GRUB is the modern Linux bootloader program.

- Click “Booting” at the installation settings menu.
- Edit the “Boot Loader type”, if necessary, to set it to “Grub”. If asked, tell it to propose a configuration.
- Click “Next” to go back to the installation settings menu.

5.3.3 Time Zone

If the time zone is not correct, we’ll configure the time zone.

- Click “Time zone” at the installation menu.
- Pick the region and appropriate time zone.
- Set the date and time and the hardware clock setting. Choose Local if you don’t understand what UTC is. Be careful to check the date format. It’s in the format Day-Month-Year.
- Click “Accept” when done.

5.3.4 Software Selection

Now we get to pick the software that will be installed.

- Click “Software” at the installation settings menu.
- Click “Minimum system”, then “Detailed Selection”.
- Select the “Filter Search”, then make sure that the only box checked is “Search In”.

[25] Yes, we could put the swap space into a raw partition and not manage it through LVM. However, being able to steal space from swap when you realize that you don’t need it can be useful in limited disk situations.

- Search for and add (only) the following packages (use plus to add them):
  - apache2, apache2-mod_php4, bastille, courier-imap, exim, ispell, ispell-american (or whatever dictionary you'd like), libnet, perl-DateManip, perl-Net_SSLeay, perl-Time-modules, perl-TimeDate, pico (unless you're comfortable with the vim editor), spamassassin, squirrelmail, squirrelmail-plugins, sudo, tnef, unzip, webmin, xinetd, xntp, yast2-http-server, zip
- Make sure the following packages are deselected / will not be installed (use minus to remove it):
  - portmap
- Once all these items have been checked, click "Accept".
- In the following screen-
  - Ignore conflict and install Apache2.
  - Do not install PostFix.
  - Install apache2-prefork
  - Click OK – try again
  - (Accept any automatic additions to meet dependencies)
- Click "Continue".
- Click "Accept", "Yes", and "Install" to begin the software installation.
- Depending on your connection to the SUSE installation files, the installation may take a few hours to download and install everything.

5.4 SUSE Linux Finishing the Installation

Now that the operating system is installed on the hard drive, we will configure it. Ignore the message about the graphical installation. We didn't install X, the graphical subsystem, because it is more secure and runs faster without it.

- Accept all prompts for detecting hardware, but skip any hardware you don't have or don't want to use. (IE- ISDN cards, modems)

5.4.1 Password Encryption and Root Password

By default, Linux uses DES encryption for passwords. This is insecure because of the speed by which DES passwords can be broken if the password hashes are obtained, and because standard Linux DES password encryption limits us to only eight character passwords. We will select the newer and more secure Blowfish algorythm which is much harder to crack and will allow us to have longer passwords.

- Select "Expert Options", "Blowfish", and "OK".

The next step is to select the root password. Selecting secure passwords is a topic unto itself. Some password guidelines from CERT[26]:

> [C]hoose passwords that are difficult to guess (for example, words that are not in any dictionary of any language; no proper nouns, including names of "famous" real or fictitious characters; no acronyms that are commonly used by computer professionals; no simple variations of first or last names.)
>
> A good heuristic for choosing a password is to choose an easy-to-remember phrase, such as "By The Dawn's Early Light", and use the first letters to form a password. Add some punctuation or mix case letters as well. For the phrase above, one example password might be: bt}DeL{. (DO NOT use this sample phrase for your password.)

In addition to these guidelines from CERT, many password strength papers now call for a minimum length of eight characters. The Blowfish encryption technology that we have selected allows us to create much longer passwords than that. (In fact, not that you'd want to, but we can create passwords up to 97 characters in length!) In an article from SecurityFocus[27], it is stated:

> Length means that the longer a password, the more difficult it is to crack. Simply put, longer is better. Probability dictates that the longer a password the more difficult it will be to crack. It is generally recommended that passwords be between six and nine characters. Greater length is acceptable, as long as the operating system allows for it and the user can remember the password. However, shorter passwords should be avoided.

So the best password uses a mixture of different types of characters, in a pattern easy to remember but hard to guess, and long enough to make brute-force attacks difficult to impossible, but not so long as to require your users to write them down. Also, because passwords can be cracked if given enough time, passwords should be required to be changed on a regular basis. We'll discuss this more in a later section.

- Enter your root password twice and select Next.
- Accept all prompts for detecting hardware- but skip any hardware you don't have or don't want to use.
- Select Next.

[26] CERT Coordination Center. "UNIX Configuration Guidelines." 04 Jun. 2003. URL: http://www.cert.org/tech_tips/unix_configuration_guidelines.html#A1 (2 Apr. 2004).

[27] Granger, Sarah. "The Simplest Security: A Guide To Better Password Practices." 17 Jan 2002. URL: http://www.securityfocus.com/infocus/1537 (05 Apr. 2004).

5.4.2 Downloading Updates and Configuring Automatic Update

SUSE Linux has a very useful and easy automatic update feature built into their product. YaST's Online Update (YOU) utilizes digital signing to ensure that fake/trojan updates do not get installed.

When the FBI and SANS jointly released the Top 20 vulnerabilities list[28], most of the issues could (and still can) be mitigated by simply making sure that the most-recent versions of software were running on the system.

- Allow the Internet test to occur and select "Next".
- Select the option to download the online updates and install them.
- Select "Installation Source" and "User-Defined Location".
- Enter the URL for your SUSE mirror into the Location field. (IE- "ftp://mirrors.usc.edu/pub/linux/distributions/suse", but use your closest/fastest mirror.)
- Uncheck "manually select patches" and check "Reload All Patches from Server".
- Select "Configure Fully Automatic Update".
- Enable the automatic update and select a time for them to occur. Select "OK".
- Select "Next".
- While the patches are downloading, select the "Remove Source Packages after Update" to free up the disk space consumed by the downloaded and no longer needed packages.

The latest patches will be retrieved from your local mirror and installed automatically. On a production system, you might want to review the patches before they are deployed, but that means that you must regularly review security patches when they are released, or you might miss a critical patch. Automatic deployment of patches means that there is a slight chance that something will break. However, it is relatively easy to roll-back a patch if you find something no longer works after a patch is deployed. Luckily, SUSE does an excellent job testing patches before deploying them.

- When the updates have been applied, select Finish.

[28] SANS. "SANS Top 20 Vulnerabilities." Version 4.0. 08 Oct 2003. URL: http://www.sans.org/top20/ (14 Apr. 2004).

5.4.3 Configuring Authentication and Users

We'll be using stand-alone authentication for this example. Another option if you already have your users and passwords in another directory is to use LDAP authentication for the various services. However, the complexities in configuring the various services to take advantage of these services is beyond the scope of this document.

- Select "Stand Alone Machine" and "Next".

At this next screen, we'll configure the default password settings and defaults.

- Select "User Management", "Expert Options", and "Defaults for New Users".
- Select "Secondary Groups" and delete all groups from the field.
- Select "Default Login Shell" and enter "/bin/false". This will keep our mail users from logging into the system for anything other than mail.
- Select "Next".

We'll create a local user account for your use. You should always use this account, and not the root account when administering the system. We'll cover some basic security-friendly administration techniques at a later point.

- Enter your full name, login name, password, and check the "Receive System Mail".
- Select "Details", "Additional Group Membership", and check wheel.
- Select "Login shell" and set it to "/bin/bash".
- Select "Next" and "Next".

5.4.4 Completing the SUSE Linux Installation

- Read the release notes, then select "Next".
- Select "Finish" to end the installation.

You're left at a login prompt. Congratulations on completing the first phase of installation!

5.5 Additional Operating System Configuration

Now that the operating system installation is completed, we'll be making some modifications to the default system configuration to finalize the installation and improve security.

5.5.1 Network Configuration and Host and Domain Names

Let's give the system its correct host name and network configuration.

- Login with your root account and password. (Normally, you don't want to do this, but as we're at the console and configuring our system for the first time, it's okay.)
- At the console to enter the system configuration utility, type:

```
yast
```

- Select "Network Devices", "Network Card", "Change", and "Edit".
- Set the appropriate static IP address and subnet mask.
- Select "Host name" and "Name server" and "Modify" (if asked).
- Enter the host name and domain name for the mail server.
- Check the "Name servers" and "domain search" for correct information.
- Select "OK".
- Select "Routing".
- Enter the appropriate default gateway and select "OK".
- Select "Next".

5.5.2 SUDO

A more secure way of doing things on a Linux system than logging in as root is to use the sudo command with an account which is a member of the wheel group. We'll configure your personal account (which we already made a member of the wheel group) to be able to use the sudo command to accomplish administrative tasks.

The nice thing about sudo is that it will log any commands run this way, whereas being at a root prompt does not. It also allows us to become an administrator only when we need to, instead of having that level of access all of the time.

- At the console, type in the following commands[29]:

```
export EDITOR=/usr/bin/pico
visudo
```

[29] You can skip the export EDITOR statement if you are comfortable with the vim editor.

- Comment (put a # symbol in front of) the following lines. (This will allow us to use our account’s password to use sudo.)

```
Defaults targetpw
%users ALL=(ALL) ALL
root  ALL=(ALL) ALL
```

- Uncomment (remove the # symbol in front of) the following lines. (This will allow all wheel group members to perform all administrative tasks with sudo[30].)

```
# %wheel        ALL=(ALL)       ALL
```

- Save the file and exit the editor. In Pico, press Control-X, 'Y' to save, and enter to save the file.
- Now let’s make sure that root is in the wheel group (your administrator account already is), so that root can also use the SUDO command. Type the following command:

```
usermod -G root,wheel root
```

5.5.3 Administrator Account Path

We’re going to put the administrative applications into our user’s binary executable path to make it easier to administer without logging in directly as root.

- Log out of Linux by typing the command:

```
exit
```

- Log back in using your administrator account (not the root account).
- Let’s test an administrative command. Let’s find out our network card configuration. Type in the following command:

```
ifconfig
```

- As you can see, we get a “bash: ifconfig: command not found” error because the ifconfig is not in our active path.

[30] You can customize the sudo command to allow only certain administrative commands to be processed. For more information, check this website- http://www.siliconvalleyccie.com/linux-hn/sudo.htm

- To fix this[31], type in the following command to add the appropriate administrator path to our path. (Be sure to use single quotes, not double.)

```
echo 'export PATH=/sbin:/usr/sbin:/usr/local/sbin:$PATH' >> ~/.profile
```

- Now log out using “exit” and log back in using your administrator account (not root.)
- Let’s try it again, type:

```
ifconfig
```

We should get the results now. Be aware that just because the administrator commands are in our path will not allow us to change settings or access information that we don’t have privileges for, that’s what the sudo command is for.

From now on, whenever you login to your server to do administrative work, use your administrative account. Don’t log in using the root account unless something “bad” happens to your administrative and you can’t use it, and then only fix your administrative account and go back to it.

5.5.4 Network Time Protocol (NTP)

It’s important for a mail server to keep proper time, so we’re going to set up the server as a time server client.

- Find a local, public secondary time server from the following website URL and write it down. (IE- clock.fmt.he.net)

```
http://twiki.ntp.org/bin/view/Servers/StratumTwoTimeServers
```

- From the server console, type in the following command:

```
sudo pico /etc/ntp.conf
```

- Add the following line in the server section of the file:

```
server ntp-server-name
```

- Close and save the file.
- Type the following command to start time services.

```
sudo /etc/init.d/xntpd start
```

[31] Some in the security community feel that typing in the full path of administrator commands is the safest way to make sure you’re running the *real* command and not a trojaned one. While this is true, it is also inconvenient, and you really shouldn’t ever be running administrative commands without being at a trusted system utilizing *your* administrator account. Never, ever, type in a “su” or “sudo” command while at an untrusted system or when someone else is logged in.

- Type the following command to automatically start time services whenever the server starts:

```
sudo chkconfig xntpd on
```

- Time services should now be active. You can view the time of the server by typing in the following command. Make certain it is correct. If it is exactly off by one or more hours you might have the timezone set incorrectly.

```
date
```

6.0 Webmin

So far, you've been exposed to two administrator interfaces for SUSE Linux, the command-line interface and the YaST toolset. Now we'll look at an even easier interface for administering servers- Webmin.

Webmin is an easy-to-use administration tool for Linux and UNIX-like operating systems. It's mostly ready-to-use, but we'll optimize the configuration for our use.

6.1 Webmin Configuration

- Using a web browser on a different system, connect to the server with the following URL or use the IP address if the DNS name is not currently set up. Make sure to use http**s** not http at the beginning. (Instead of *servername* use the server's DNS name or IP address that you are configuring.)

```
https://servername:10000
```

- Bookmark this page for later use.
- Login using your root user account and password.
- Click on "Webmin users".
- Click "Create a new Webmin user".
- Enter your administrator account name (not root) for the Username.
- Set the "Password" popup menu to "UNIX authentication".
- Check the following Modules:
  - Apache Webserver, Bootup and Shutdown, Change Passwords, Command Shell, Custom Commands, Disk and Network Filesystems, File Manager, Logical Volume Management, Network Configuration, PAM Authentication, Running Processes, SSH Server, SSH/Telnet Login, Scheduled Commands, Scheduled Cron Jobs, Software Packages, System Documentation, System Logs, System Time, System and Server Status, Users and Groups, Webmin Actions Log, Webmin Configuration, Webmin Servers Index, Webmin Users
- Click "Save".

Now you can use your administrator account to authenticate to Webmin and administer the system. Don't use the root account unless your administrator account ceases to function.

Now let's configure Webmin to be a bit more secure in its authentication attempts.

- Click the "Webmin icon" and "Webmin Configuration".

If you have one or more administrative workstations from which you want to limit Webmin access, follow the steps in the IP Access Control section. This will make the system much more secure. If not, skip ahead to the Authentication section.

- Click IP Access Control.
- Enter the following settings in the Webmin IP access control configuration: (leave the non-specified fields as they are set)
  - Click Only allow from listed addresses.
  - Enter the IP addresses of the systems you will be administrating from.
- Click Save.
- Click Authentication.
- Set the following settings in the Webmin authentication configuration (leave the non-specified fields as they are set):
  - Click "Enable Password Timeouts".
  - Set "Auto-logout After _10_ Minutes of Inactivity"
  - Uncheck "Offer to remember login permanently?"
  - Uncheck "Show hostname on login screen?"
  - Set "Show pre-login file" __/etc/motd__

Currently the message-of-the-day (MOTD) banner is set to "Have a lot of fun...", but we'll change it to something more secure later. Right now, it sounds like an invitation to hack the system!

- Click "Save".
- You will likely have to log back into Webmin.

Be aware that you may have to reload the login page to get past the pre-login message file now that one is set.

6.2 Webmin Upgrade

Because some modules require a current version of Webmin and SUSE Linux 9 doesn't necessarily deploy the most recent version, we'll use Webmin's built-in update capabilities to download and install the most recent version.

- Go to the Webmin interface.
- Click on "Webmin Configuration" and "Upgrade Webmin".
- Click the "Upgrade Webmin" button. The upload will be downloaded, the package signature checked, and installed.
- Once it's done, click on any links to download any minor updates to the Webmin release. Then click "Return to Webmin Configuration".

6.3 Additional Webmin Modules

There are many capabilities built into Webmin already, but in case you need some administrative functions that aren't already there, there is a large list of independently developed Webmin modules that you can find at the Webmin Add-On Modules website[32].

Let's add a module for configuring Extended Internet Services that we'll use later.

- At the Webmin interface, click on the "Webmin icon", "Webmin Configuration", and "Webmin Modules".
- Click the "Standard module from www.webmin.com" button.
- Type xinetd into the following field.
- Click "Grant access to all Webmin users".
- Click "Install Module".
- Now click on the "Networking icon" and "Extended Internet Services". (If Extended Internet Services is not there, you might have to go to the Webmin Users to add it your admin user.)
- Click "Module Config".
- Enter the following settings into the module configuration: (Do not change the settings for fields not listed below.)
    - Path to Xinetd PID file: Set to None (look for process instead)
    - Command to start xinetd:

      ```
      /etc/init.d/xinetd start
      ```

    - Click on the Add new services as file in directory button, then enter into the field:

      ```
      /etc/xinetd.d/
      ```

- Click Save.

[32] Webmin Add-On Modules. http://webadminmodules.sourceforge.net

7.0 Secure Shell (SSH)

Used with permission © 1999 Alexander Wilkie

SSH is a popular technology that allows for secure terminal sessions as well as the ability to send commands and files securely between systems. Although SSH has had a fair amount of security-related updates, it would be quite difficult to set up the same functionality using other tools. Also, the automatic updating will keep our copy of SSH up-to-date. Make certain that you are updating any other systems that use SSH.

By default SSH has some less-secure options set. We'll tighten these up by configuring SSHD better. We'll use the Webmin interface because its so easy.

- Go to the Webmin interface in a browser.
- Login using your administrator account and password (not root).
- Click the "Servers icon" and "SSH Server".

First, we'll set up the Webmin SSH module to work with the SUSE SSHD process ID file so it can restart SSHD to allow new settings to take effect.

- Click "Module Config".
- Change the "Full path to sshd PID file" to:

```
/var/run/sshd.init.pid
```

- Click "Save".

Now we'll block root from being able to log in directly over SSH.

- Click "Authentication".
- Set "Allow login by root?" to No.
- Set "Pre-login message file" to:

```
/etc/motd
```

- Click "Save".

Now we'll disable the insecure SSHv1 protocol.

- Click "Networking".
- Uncheck "SSH v1" and click "Save".

Now we'll enable SSH login only from wheel group members.

- Click "Access Control".
- Click the button for "Only allow members of groups".
- Click the "..." button for "Only allow members of groups".
- Click the "wheel" group and "OK".
- Click "Save".

And finally, we'll make all our changes active.

- Click the "Apply Changes" button.

8.0 ClamAV

ClamAV is an open-source antivirus solution that has excellent performance and an active community creating virus-detection signatures.

Unfortunately, SUSE Linux doesn’t include the RPMs for ClamAV, so we’ll have to get it from another source to install it.

Please help me convince SUSE that they need to include the outstanding ClamAV antivirus software package in their next release by suggesting it at their suggestions website, http://www.suse.de/cgi-bin/feedback.cgi , and hopefully we’ll have YaST-automated ClamAV updates in the next SUSE Linux release.

8.1 Installation

We will use a ClamAV package maintained for the Fedora project. It would be much better to have a package built specifically for SUSE Linux, but I couldn’t locate one. We’ll have to make some changes in some files after installation to get this package to work on SUSE Linux. Normally, to solve this you would do one of two things- download the source code and compile it locally (of course, we don’t have the software build tools installed for security reasons, and I promised this solution would not require compiling any code); or download the source code, compile it on a different, but similar system, and create an RPM package for our production system. However, this is beyond the scope of this document[33].

Because we are using a package not built for SUSE Linux, there are some things to be aware of.

1. Fedora uses different package names. So whenever you go into YaST to install software, you’ll get a warning stating that the ClamAV package requires a couple of packages that aren’t installed. You can safely ignore this, because the things the package is looking for are already installed, but are called something different.
2. The YaST automatic update tool won’t be automatically installing updates for the ClamAV software package, although the virus definitions database will automatically update. As a result, the ClamAV package will have to be monitored to make sure that if any critical security updates come out, they can be downloaded and installed manually using the same process by which we are going to install it. Therefore, you may wish to sign up for the “clamav-announce” mailing list[34] so you can keep yourself apprised of new ClamAV releases.

[33] If you know how to do this and want to maintain a SUSE Linux RPM package, please have it linked to the ClamAV website binary page. You may also want to contact SUSE to ask them to include the package in their official distribution.

[34] You can sign up for it here- http://lists.sourceforge.net/lists/listinfo/clamav-announce

Look at these websites to find pre-compiled RPM packages for ClamAV-

- http://crash.fce.vutbr.cz/crash-hat/1/clamav/ (I wrote the directions from this site.)
- http://dag.wieers.com/packages/clamav/ (Another possible source.)

Write down the URL for downloading this software.

To install this software, go to a server console screen (either locally or through SSH).

- Enter the following commands to download the latest ClamAV, create a link to the appropriate SUSE Linux services directory (so the Fedora package will work), and install the ClamAV software package (update the URL and ClamAV filename to match the current release on one of the websites above):

```
curl -O http://crash.fce.vutbr.cz/crash-hat/1/clamav/clamav-0.70rc-1.i386.rpm
sudo rm -r /etc/rc.d/init.d
sudo ln -s /etc/init.d /etc/rc.d/init.d
sudo rpm -Uvh --nodeps clamav-0.70rc-1.i386.rpm
```

- We need to fix the init (startup) scripts to work with SUSE Linux, so type in the following (each command is one line):

```
sudo curl -o /etc/init.d/clamd \
 http://davebailey.homeip.net/secure-mailserver/clamav/clamd
sudo curl -o /etc/init.d/freshclam \
 http://davebailey.homeip.net/secure-mailserver/clamav/freshclam
```

- We need to put the clamav user in the mail group. Type in the following command:

```
sudo usermod -G clamav,mail clamav
```

- Enter the following commands to update the antivirus signature database and start the ClamAV engine, and to set the services to start automatically when the server starts:

```
sudo /etc/init.d/clamd start
sudo /etc/init.d/freshclam start
sudo chkconfig clamd on
sudo chkconfig freshclam on
```

Freshclam will keep the antivirus signatures up to date, but you may want to check the /var/log/clamav/freshclam.log file to make sure it was successful. By default, Freshclam will check for updated antivirus signatures hourly.

8.2 Webmin Module Setup

Now we’ll add ClamAV support to Webmin.

- Go to the following webpage on your workstation and click on the link to “Download the webmin module of wbmclamav here”.

```
http://labs.libre-entreprise.org/project/showfiles.php?group_id=32
```

- Locate the latest release and jot down the URL.
- Now go to the web-browser on your workstation and go to the Webmin interface.
- Click “Webmin Configuration” and “Webmin Modules”.
- Enter the URL of the Webmin module in the “From ftp or http URL” field.
- Click “Grant access to all Webmin” users.
- Click the “Install Module From File” button.

Now let’s configure and try out the ClamAV module.

- Click the System icon and Module Config.
- Enter the following settings into the module configuration:
  - AMaViS
    - Path to quarantine repository:

      ```
      (leave blank)
      ```

  - ClamAV
    - Path to daemon init script:

      ```
      /etc/init.d/clamd
      ```

    - Daemon name:

      ```
      clamd
      ```

    - Path to logfile:

      ```
      /var/log/clamav/clamd.log
      ```

    - Path to configuration file:

      ```
      /etc/clamav.conf
      ```

    - Path to main virus signatures database:

      ```
      /var/lib/clamav/main.cvd
      ```

    - Path to daily virus signatures database:

      ```
      /var/lib/clamav/daily.cvd
      ```

- Freshclam
  - Path to configuration file:

    ```
    /etc/freshclam.conf
    ```

  - Method to use to refresh automatically viruses database?
    - Set to “Daemon”.
  - Path to logfile:

    ```
    /var/log/clamav/freshclam.log
    ```

  - Path to daemon init script:

    ```
    /etc/init.d/freshclam
    ```

- Click “Save”.
- You can now administer ClamAV in the System section of Webmin.

9.0 Exim

Exim is a fairly popular, though perhaps not as commonly adopted, message transfer agent (MTA). This may be because its configuration is quite different from Sendmail. Those that use Exim typically think that's a good thing, perhaps because Sendmail configuration is complex and bewildering to the unitiated. Some of the reasons we'll be using Exim here are that it has had a very good security record; has excellent documentation available; doesn't require additional software to install and configure to handle virus or spam filtering (such as Procmail); and the configuration is relatively easy (as MTAs go).

Exim normally uses UNIX-style mailbox (mbox) files for storage of messages. Because we'll be using Courier-IMAP in the next section to handle retrieving our mail and it uses maildirs instead of mbox files, we'll need to configure Exim to work this way. On a typical server with a typical load, maildirs are usually faster than mbox style mailbox files. This is especially true with the ReiserFS file system that we're using.

9.1 Securing Exim SMTP banner

By default the SMTP service announces what software it is running and what version. We'll change the SMTP banner to make it a little more difficult to probe for weaknesses.

- At the server console, type in the following command:

```
sudo pico /etc/exim/exim.conf
```

- Add the following line to the Main Configuration Settings section of the configuration file:

```
smtp_banner = $primary_hostname ESMTP $tod_full
```

- Save the file and exit the editor.

9.2 Creating the Maildir directories

We need to make certain each account has a proper Maildir directory.

- At the server console, type the following command (make sure to capitalize the 'M' in Maildir):

```
sudo maildirmake /etc/skel/Maildir
sudo maildirmake -f Drafts /etc/skel/Maildir
sudo maildirmake -f Sent /etc/skel/Maildir
sudo maildirmake -f Spam /etc/skel/Maildir
sudo maildirmake -f Trash /etc/skel/Maildir
```

- Type in the following command to create the subscribed folder list for the IMAP server (we'll be configuring it later):

```
sudo pico /etc/skel/Maildir/courierimapsubscribed
```

- Create the file with the following lines:

```
INBOX
INBOX.Drafts
INBOX.Sent
INBOX.Spam
INBOX.Trash
```

- Save the file and exit the editor.
- Also, let's set up the Maildir directories for our administrative account:

```
maildirmake ~/Maildir
maildirmake -f Drafts ~/Maildir
maildirmake -f Sent ~/Maildir
maildirmake -f Spam ~/Maildir
maildirmake -f Trash ~/Maildir
cp /etc/skel/Maildir/courierimapsubscribed ~/Maildir
```

9.3 Configuring Exim

This is probably the hardest part of the solution because we need to configure Exim to understand our environment and get it to work with the other software we have installed. I'd recommend copying and pasting, as opposed to typing in the commands, to avoid typing errors.

If you don't want to understand the configuration process and just get things working, you can try the command in this footnote to grab a pre-completed Exim configuration file[35].

9.3.1 Maildirs

Now we'll configure Exim to use Maildirs for local delivery.

- Type the following command at the server console:

```
sudo pico /etc/exim/exim.conf
```

[35] This command will download a pre-configured exim configuration file that should be able to be used immediately. You will be able to skip all exim.conf editing, but you will still have to follow the other steps including those that start Exim.
sudo curl -o /etc/exim/exim.conf \
http://davebailey.homeip.net/secure-mailserver/exim/exim.conf

- Find and comment out (put a # symbol in front of each line) the “userforward:” configuration in the Routers section of the file and put this in right after the commented lines:

```
userforward:
  driver = redirect
  check_local_user
  local_part_suffix = +*
  local_part_suffix_optional
  file = ${home}/.forward
  modemask = 002
  no_verify
  no_expn
  check_ancestor
  allow_filter
  file_transport = address_file
  pipe_transport = address_pipe
  reply_transport = address_reply
  directory_transport = address_directory
```

- Find and comment out the “local_delivery:” configuration in the Transports section of the file and put this in right after the commented lines:

```
# Local delivery to maildir style mailbox
local_delivery:
  driver = appendfile
  maildir_format
  create_directory
  directory = /home/${local_part}/Maildir/
  return_path_add
  delivery_date_add
  envelope_to_add
  group = mail
  mode = 0660
  no_mode_fail_narrower
```

- Add this section below the above section:

```
# This transport is used for handling deliveries to directories that are
# generated by aliasing or forwarding.
address_directory:
  driver = appendfile
  maildir_format
  delivery_date_add
  envelope_to_add
  return_path_add
```

- Add the following line to the “system_aliases:” Router configuration:

```
directory_transport = address_directory
```

- Save and Exit the file.
- Type the following commands to start Exim and set it up to start automatically upon system startup:

```
sudo /etc/init.d/exim start
sudo chkconfig exim on
```

9.3.2 Exiscan

This amazing software allows us to scan a message for viruses and spam content even before the SMTP connection is closed. The advantages for this are:

1. If we’re certain the message is spam, why even accept it? If we aren’t certain, we can still accept it and tag it to let the user review it as usual.
2. If we know there’s a virus in the message, we should block it so that we don’t have to resolve the issue of sending back virus warnings after closing the connection from the sending host. Many viruses today falsify the header information about where the infected message came from, so the wrong person gets notified that they sent infected mail. This way, either the infected person finds out, or at least the wrong person doesn’t get an invalid infection notice.

SUSE Linux includes Exim with the Exiscan patches already in place, so we just have to turn them on. This makes Exiscan both easier *and* more effective than the alternatives.

Many of these examples are taken or adapted from Tim Jackson’s very useful Exiscan HOWTO document[36].

- Type the following command to turn off writing the SpamAssassin report directly into the body of the message with the original attached. Exiscan doesn’t like this, so we’ll turn it off.

```
sudo pico /etc/mail/spamassassin/local.cf
```

- Add the following lines to the end of the file:

```
# Turn off writing reports directly into text body
# and attaching original
report_safe 0
```

- Save and exit the file.
- Now let’s set up the default settings for starting the SpamAssassin daemon correctly. Type the following:

```
sudo pico /etc/sysconfig/spamd
```

- Make sure the SPAMD_ARGS includes the -d and -a flags, but *not* -c.
- Save and exit the file.

[36] Jackson, Tim. “Spam and Virus Scanning with Exim 4 using Exiscan and/or SA-Exim Mini-HOWTO.” V1.0.10. 25 Feb 2004. URL: http://www.timj.co.uk/linux/Exim-SpamAndVirusScanning.pdf (10 Apr. 2004.)

- Type the following commands to set up storage for automatic whitelisting:

```
sudo mkdir /var/spool/clientmqueue/.spamassassin
sudo chown mail:mail /var/spool/clientmqueue/.spamassassin
sudo chmod 700 /var/spool/clientmqueue/.spamassassin
```

- Now let's start the spamd daemon and set it to start automatically:

```
sudo /etc/init.d/spamd start
sudo chkconfig spamd on
```

- Now let's add the commands into the Exim configuration to allow us to use Exiscan. Type the following at the server console:

```
sudo pico /etc/exim/exim.conf
```

- Find and uncomment the following lines in the Main section of the configuration file:

```
# acl_smtp_data = acl_check_content
# spamd_address = 127.0.0.1 783
```

- Find the following line:

```
# av_scanner = sophie:/var/run/sophie
```

- Change it to the following line:

```
av_scanner = clamd:127.0.0.1 3310
```

- Find the following lines:

```
# Reject virus infested messages.
deny  message = This message contains malware ($malware_name)
      malware = *
```

- Insert the following 'demime' line (new line shown in bold.) This will help ClamAV find viruses in the attachments.

```
# Reject virus infested messages.
deny  message = This message contains malware ($malware_name)
      demime  = *
      malware = *
```

- Save the file and exit the editor.

- For testing this configuration, rather than listing all of the steps here, the Exiscan HOWTO written by Tim Jackson in section 7.2 includes some very thorough and useful testing steps. (Reference included above and in References section.) I recommend that you follow them to check the functionality of the antivirus and antispam capabilities.

9.3.3 Quotas

One kind of denial-of-service (DoS) attack is to fill up hard drive storage with random information rendering it useless.

Creating partitions that restrict the amount of information written to storage is a step that will limit the damage this attack can do, for example, disabling a service instead of disabling the entire system. We have also created our filesystem using LVM and ReiserFS which allows us to add storage to any of our logical volumes on the fly without restarting to meet changing storage needs[37].

Now we want to restrict the amount of data a user can accept to their email account to keep someone from utilizing too much storage and filling up the logical volume.

Although we could set up quotas in the operating system[38], since this is an email-only server, we'll use the built-in quota support in Exim to perform this task.

- Let's create our Exim quota configuration file. At the server console, type the following command:

```
sudo pico /etc/exim/quotafile
```

- Now we can create storage limits for your administrative user, and all other users. In this example, the admin user would get 20MB and everyone else 10MB of storage. Enter the following content:

```
(admin-user)   20M
*              10M
```

- Save the file and exit the editor.
- Now type in the following command:

```
sudo pico /etc/exim/exim.conf
```

[37] Performing this kind of storage adjustment is beyond the scope of this document, but it is quite easy to do. Look in YaST under System, LVM. You can edit any logical volume and increase the size dynamically. I don't recommend trying to reduce the size of a volume dynamically. With the ReiserFS, you need to have the volume unmounted to safely do this. (Although it will let you try with it mounted, don't do it unless you look forward to running reiserfsck from a boot CD or rescue disk.)

[38] SUSE Linux 9 includes the patches to support filesystem quotas on ReiserFS with the 2.4.X kernel.

- Find the “local_delivery:” part in the Transports section of the configuration file. Add these lines to the bottom of this part:

```
  quota = ${lookup{$local_part}lsearch*{/etc/exim/quotafile}{$value}{4M}}
  maildir_tag = ,S=$message_size
  quota_size_regex = ,S=(\d+)
  quota_warn_threshold = 75%
```

- In the Retries section of the configuration add this quota rule as shown below:

```
# Domain                Error       Retries
# ------                -----       -------
*                       quota
*                       *           F,2h,15m; G,16h,1h,1.5; F,4d,6h
```

9.4 Exim Configuration Options

If your ISP requires all messages that are delivered from your hosts to go through their SMTP mail servers, as opposed to being delivered directly from your host to the remote host, you will need to do the following:

- Comment out the “dnslookup:” router (by placing # symbols in front of each line of the section).
- Enter the following in its place (change the *dns-name-of-remote-smtp-server* to the DNS name of the SMTP mail server you want to forward your outgoing mail to):

```
# forward all outgoing mail through a specific SMTP server
internet:
  driver = manualroute
  domains = ! +local_domains
  transport = remote_smtp
  route_list = * dns-name-of-remote-smtp-server
```

9.5 Exim Webmin Module

Let’s add Exim support to Webmin and make it easier to monitor Exim’s status.

9.5.1 Install Exim Webmin Module

- At a web browser, go to the following website:

```
http://mtlx.free.fr/webmin/exim/
```

- Find the most recent download and copy the link to it.
- Go to the Webmin interface, and main menu.
- Click on “Webmin Configuration” and “Webmin Modules”.
- Click on the button next to “From ftp or http URL”.
- Paste the Webmin module link you copied into the field “From ftp or http URL”.
- Click “Grant access to all Webmin users”.
- Click “Install Module From File”.

9.5.2 Configure Exim Webmin Module

- Click on the “Servers” icon, “Exim Monitor”, then “Module Config”.
- Enter the following settings into the Exim Webmin Module:
    - Exim executable-

    ```
    /usr/sbin/exim
    ```

    - Exim configuration file-

    ```
    /etc/exim/exim.conf
    ```

- Click “Save”.

10.0 SpamAssassin

SpamAssassin is the leading open-source antispam solution. It's mature, flexible, relatively fast, and easy to use once it's properly set up.

SpamAssassin utilizes multiple different methods to distinguish spam from good mail ("ham") and is very configurable in its scoring methods and rules. SpamAssassin can learn, using statistical analysis, from messages both good and bad. SpamAssassin can become more accurate over time, as the users help train SpamAssassin. SpamAssassin will learn spam from the Spam folder and "Ham" from read messages in the Inbox folder[39].

The configuration is easy because it's pretty much already configured. We just have to make sure it stays up-to-date and give the users an interface to utilize it properly so they can train it to better locate spam without tagging the mail they want to receive.

You need to train the users of this system to put their spam into the Spam folder, rather than deleting it, and to put/eave good messages after reading them in their Inbox for at least one night[40].

10.1 SpamAssassin Webmin Module

10.1.1 Install SpamAssassin Webmin Module

Let's put the SpamAssassin Webmin Module into Webmin. The SpamAssassin module is a standard (but not pre-installed) Webmin module, so it's really easy to install.

- Go to the Webmin interface.
- Click on "Webmin Configuration" and "Webmin Modules".
- Click on the "Standard module from Webmin" button.
- Type the following text in the "Standard module from Webmin" field: spam
- Click on the "Grant access to all Webmin users" button.
- Click on the "Install Module From File" button.

10.1.2 Configure SpamAssassin Webmin Module

- Click on the "Servers" icon, "SpamAssassin Mail Filter", and "Module Config".
- Enter the following settings into the module configuration (do not change the settings for fields not listed below):
  - SpamAssassin configuration file or directory:

```
/etc/mail/spamassassin/local.cf
```

  - Full path to SpamAssassin command:

[39] SpamAssassin's bayesian filtering takes a while to get started. It will take several hundred messages before the learning process starts significantly helping with spam detection.

[40] There are many other ways we could do this, such as creating spam and ham folders, but this is probably the least intrusive.

```
/usr/bin/spamassassin
```

  - Display warning if SpamAssassin is not set up? _no_
  - SpamAssassin daemon process names:

```
spamd
```

- Command to restart processes-

```
/etc/init.d/spamd restart
```

- Click the “Save” button.

10.2 Configure SpamAssassin

We can set how sensitive SpamAssassin is to spam. If we set the number higher, it is more tolerant of spam. If we set the number lower, it is less tolerant. The problem is that if we set it too low, we will get a lot of false positives. After the users have trained SpamAssassin, it will get even better at locating spam.

- Go to the Webmin interface.
- Click on the “Servers” icon and “SpamAssassin Mail Filter”.
- Click on “Spam Classification”.
- Enter the following settings into the spam classification configuration (do not change the settings for fields not listed below):
  - Hits above which a message is considered spam: _5_
- Click “Save”.
- Click “Apply Changes”.

10.3 Setting up Spam Redirection

Viruses and high-scoring spam are automatically blocked at SMTP receive time, but low-scoring spam is handed back to the user. To help the user keep track of what is spam and what is not, we’re going to file the spam to a different mail folder so that they can monitor it if they choose for possible non-spam messages, so they can move them into their inbox.

- Type in the following command:

```
sudo pico /etc/skel/.forward
```

- Create the file with the following commands. (You **must** start the file with the “# Exim filter” line to notify Exim that this is a special kind of .forward file.)

```
# Exim filter
if ( $h_X-Spam-Flag: is "YES" )
then
  # it's spam...
  save $home/Maildir/.Spam/
  finish
endif
```

- Save the file and exit the editor.
- Type in the following command to copy this .forward file into your admin user's home-

```
cp /etc/skel/.forward ~/
```

10.4 Automatic Spam Learning and Cleaning

It's great that all this spam sitting in these spam folders is separated out, but what happens if the user puts messages into the spam folder? SpamAssassin has no way of knowing that it is there, and it will keep treating manually added spam as though it's "ham" or valid mail.

We need a way to automatically scan the spam folders and learn new spam patterns from the messages.

The key to training SpamAssassin is to remember not to delete spam, but to put it into the Spam folder, so that SpamAssassin will learn that mail like that is spam, not "ham".

This script will also delete spam older than five days from your Spam folder, so that it doesn't pile up indefinitely until the user's storage is completely spent. But it does allow the user to view spam that is up to five days old to give them a chance to move it out of their Spam folder and into their Inbox.

10.4.1 Create the mail_learn Script

This script will find the read "ham" in the inbox and learn from it, find the spam in the Spam folder and learn from it, and then delete spam older than five days from the Spam folder.

- Type in the following command[41]:

```
sudo pico /usr/sbin/mail_learn
```

[41] To save yourself from creating this monster file, you can also use the following command to transfer this script to your system: sudo curl -o /usr/sbin/mail_learn \
http://davebailey.homeip.net/secure-mailserver/scripts/mail_learn

- Create the file with the following lines:

```
#!/bin/bash
# mail_learn
# Learn ham and spam from user mailboxes in Maildir format
# then clean older spam from .Spam directory

if [ -z "$1" ]; then
 echo "usage: mail_learn USER"
 exit 0
fi

if [ -f "/home/$1/.spamassassin/user_prefs" ]; then
 echo -n ""
else
 echo "Creating user_prefs file"
 USER=$1 sudo -u $1 mkdir "/home/$1/.spamassassin" &> /dev/null
 USER=$1 sudo -u $1 touch "/home/$1/.spamassassin/user_prefs"
fi

if [ -d "/home/$1/Maildir/cur" ]; then
 echo "Learning ham from read messages in /home/$1/Maildir/cur/"
 HOME="/home/$1" USER=$1 sudo -u $1 sa-learn --no-rebuild --ham \
   --dir "/home/$1/Maildir/cur/"
fi

if [ -d "/home/$1/Maildir/.Spam" ]; then
 echo "Learning spam from /home/$1/Maildir/.Spam/cur/"
 HOME="/home/$1" USER=$1 sudo -u $1 sa-learn --no-rebuild --spam \
   --dir "/home/$1/Maildir/.Spam/cur/"
 echo "Learning spam from /home/$1/Maildir/.Spam/new/"
 HOME="/home/$1" USER=$1 sudo -u $1 sa-learn --no-rebuild --spam \
   --dir "/home/$1/Maildir/.Spam/new/"
 echo "Cleaning old spam from /home/$1/Maildir/.Spam/"
 find "/home/$1/Maildir/.Spam/" -mount -type f -mtime +5 -exec rm -f "{}" \;
fi

echo "Rebuilding sa-learn database for $1"
HOME="/home/$1" USER=$1 sudo -u $1 sa-learn --rebuild

exit 0
```

- Save the file and exit the editor.

- Type in the following commands to set the privileges:

```
sudo chown root:wheel /usr/sbin/mail_learn
sudo chmod 770 /usr/sbin/mail_learn
```

10.4.2 Create the Cron Script

This script will be run daily to run the mail_learn script for each user in /home with a Maildir/.Spam folder.

- Type in the following command (this will create the cron script to run our learning scripts daily):

```
sudo pico /etc/cron.daily/mail_learn
```

- Create the file with the following lines:

```
#!/bin/bash
if [ -x /usr/sbin/mail_learn ]; then
  echo ""
  echo "Learning and cleaning spam:"
  cd /home
  for i in *; do
    if [ -d "${i}/Maildir/.Spam" ]; then
      echo "** Learning and cleaning for $i"
      /usr/sbin/mail_learn $i
    fi
  done
fi
```

- Save the file and exit the editor.
- Type in the following command to set the privileges:

```
sudo chown root:wheel /etc/cron.daily/mail_learn
sudo chmod 770 /etc/cron.daily/mail_learn
```

11.0 Courier-IMAP

Courier-IMAP is one of the most widely respected IMAP servers in existence. It has a smaller memory footprint and is typically considered more secure than the other widely used open-source IMAP server- UW-IMAP.

Now we've set up the Exim MTA to handle incoming messages plus it also has the ability to handle outgoing messages. What we don't have is an easy way for the users to check their mail!

The first step is to put in an IMAP server so that we can then provide an interface for our web mail client to get to the mail. For those wanting to use remote mail clients through the IMAP interface, this won't be covered here because it's beyond the scope of the document, but you will want to ensure that you use IMAP secure (IMAPS) instead of standard IMAP for a secure connection.

Luckily, Courier-IMAP is already configured for our use. We just need to turn it on.

11.1 Enabling Courier-IMAP

To start Courier-IMAP and set it up to start automatically when the server is started, follow this step:

- Type the following commands:

```
sudo /etc/init.d/courier-authdaemon start
sudo /etc/init.d/courier-imap start
sudo chkconfig courier-authdaemon on
sudo chkconfig courier-imap on
```

12.0 Apache

We'll be using Apache version 2.x (Apache2) as the web server for this solution. Apache is the most widely deployed web server in existence. Apache has a lot going for it including performance, stability, security, and configurability. Luckily, setting it up for our use is pretty easy because SUSE has done most of the work already.

12.1 Configure Apache Startup

We need to configure Apache so it always starts with SSL or HTTPS enabled.

- Type the following lines at the server console:

```
sudo pico /etc/sysconfig/apache2
```

- Find the following line:

```
APACHE_SERVER_FLAGS=""
```

- Edit it so that it looks like this:

```
APACHE_SERVER_FLAGS="SSL"
```

- Save the file and exit the editor.

12.2 Setting up the SSL Certificate for Apache

To make web services secure, we'll need to use the HTTPS (HTTP+SSL) protocol which requires a valid certificate. Although it is possible to get an SSL certificate which has been signed by an official Certification Authority (CA), we'll just be using a self-signed certificate for our own purposes. If this server will only be utilized by members of a single organization, and you can simply install the certificate in each browser, this is sufficient. If you will be having users from different organizations or you don't have control over the web browsers, it is recommended that you purchase a certificate signed by a CA. The process of setting up this kind of certificate is beyond the scope of this document, but you can find more information about the process in this footnote[42].

- Type the following commands in at the console to start the certificate creation process:

```
cd /usr/share/doc/packages/apache2/
sudo ./certificate.sh
```

- Step 1
  - Accept the default (R)SA Signature Algorithm.

[42] Look here for more information about SSL certificates- http://slacksite.com/apache/certificate.html

- Step 2
  - Enter the Country Name 2-letter code.
  - Enter the State or Province name.
  - Enter the City or Locality name.
  - Enter the Organization name (Just enter a space for blank).
  - Enter the Organization division or section (Just enter a space for blank).
  - Enter the DNS name/FQDN of the server.
  - Enter the mail address for the server (or enter a space for blank).
- Step 3
  - Accept the default version 3 certificate.
- Step 4
  - Press N to encrypt the private key (although it would be more secure to encrypt the key, we want the server to be able to start up without us typing in a password to do so).
- Now let's configure Apache's mod_ssl to use strong encryption ciphers. Type the following command:

```
sudo pico /etc/apache2/ssl-global.conf
```

- Now insert the following lines at the end of the file before the </IfModule> tag:

```
SSLProtocol all
SSLCipherSuite HIGH:MEDIUM
```

- Save the file and exit the editor.

12.3 Configure Apache Webmin Module

Unfortunately, the Apache Webmin module is not configured correctly. We need to set it up first.

- Go to the Webmin interface.
- Click on the "Servers" icon, "Apache Webserver", and "Module Config".
- Enter the following settings into the module configuration (do not change the settings for fields not listed below):
  - Configurable Options
    - Set File or directory to add virtual servers to- (click button next to the field)

    ```
    /etc/apache2/vhosts.d/
    ```

    - Test config file before applying changes? - Yes
    - Test config file after manual changes? - Yes
    - Test config file after other changes? - No
    - Show Apache directive names - No

  - System Configuration
    - Apache server root directory:

      ```
      /srv/www
      ```

    - Set Path to httpd executable:

      ```
      /usr/sbin/httpd2
      ```

    - Path to the apachectl command (click the button next to the field):

      ```
      /usr/sbin/apache2ctl
      ```

    - Command to start apache (click the button next to the field):

      ```
      /etc/init.d/apache2 start
      ```

    - Command to stop apache (click the button next to the field):

      ```
      /etc/init.d/apache2 stop
      ```

    - Command to apply configuration (click the button next to the field):

      ```
      /etc/init.d/apache2 restart
      ```

    - Path to httpd.conf:

      ```
      /etc/apache2/httpd.conf
      ```

    - Path to access.conf:

      ```
      (leave blank)
      ```

    - Path to srm.conf:

      ```
      (leave blank)
      ```

    - Path to mime.types:

      ```
      /etc/mime.types
      ```

    - Path to Apache PID file (click the button next to the field):

      ```
      /var/run/httpd2.pid
      ```

- Click “Save”.
- Click “Configure”.

12.4 Enabling Apache

Now that things are set up, let’s start the Apache web server.

- Type the following lines at the server console:

```
sudo /etc/init.d/apache2 start
sudo chkconfig apache2 on
```

12.5 Configure Apache Web Sites

Now that we’ve got Apache running, let’s set up some basic document settings and redirect default traffic to the secure website and the SquirrelMail web pages.

12.5.1 Securing Apache

Let’s configure the error messages not to give away more information than necessary.

- At the server console, type the following commands:

```
sudo pico /usr/share/apache2/error/include/top.html
```

- Find and delete the following line:

```
<link rev="made" href="mailto:<!--#echo encoding="url" var="SERVER_ADMIN" --
>" />
```

- Save the file and exit the editor.
- Now type the command:

```
sudo pico /usr/share/apache2/error/include/bottom.html
```

- Find and delete the following section:

```
<p>
<!--#include virtual="../contact.html.var" -->
</p>
```

- Also find and delete the following section:

```
<address>
  <a href="/"><!--#echo var="SERVER_NAME" --></a><br />
  <!--#config timefmt="%c" -->
  <span><!--#echo var="DATE_LOCAL" --><br />
  <!--#echo var="SERVER_SOFTWARE" --></span>
</address>
```

- Save the file and exit the editor.

- Now we need to secure the server tokens in messages (which we'll be using later). Type:

```
sudo pico /etc/sysconfig/apache2
```

- Find the line starting with "APACHE_SERVERTOKENS=" and make it look like:

```
APACHE_SERVERTOKENS="Prod"
```

- Save the file and exit the editor.

12.5.2 Setting up the secure web mail site

Now were going to create a virtual host named what our mail server's DNS name is so that requests directly to our mail host will be responded to in a correct fashion.

- Go to the Webmin interface.
- Click on the "Servers" icon and "Apache Webserver".
- Scroll down to the Virtual Servers, Create a New Virtual Server section.
- Enter the following settings for the new virtual server configuration (do not change the settings for fields not listed below):
  - Handle connections to address: Click Any address
  - Port: 443
  - Document Root:

    ```
    /srv/www/htdocs/squirrelmail
    ```

  - Server Name: Enter DNS name/fully-qualified domain name (FQDN) for server (ie- mailserver.domain.net)
- Click "Create".
- Click on the "Virtual Server" icon that you just created.
- Click on "SSL Options" and enter the following settings-
  - Enable SSL? - Yes
  - Certificate/private key file-

    ```
    /etc/apache2/ssl.crt/server.crt
    ```

  - Private key file-

    ```
    /etc/apache2/ssl.key/server.key
    ```

- Click "Save".

12.5.3 Redirecting default web page to the secure web mail login page

When our users type in the website URL (we'll say it's http://mail.our-domain.org) we want them to go not to the http default page, but the http***s*** SquirrelMail login page.

This will require us to redirect traffic from http://mail.our-domain.org to https://mail.our-domain.org.

Apache has the capability to do this as "redirection".

- Go to the Webmin interface.
- Click the "Servers" icon and "Apache Webserver".
- Scroll down to the Virtual Servers, Create a New Virtual Server section.
- Enter the following settings for the new virtual server configuration (do not change the settings for fields not listed below):
  - Handle connections to address- Click Any address
  - Port- 80
  - Document Root-

```
/srv/www/htdocs
```

  - Server Name- Enter DNS name/fully-qualified domain name (FQDN) for server (ie- mailserver.domain.net)
- Click "Create".
- Click on the "Virtual Server" icon that you just created (port 80).
- Click on "Aliases and Redirects".
- Enter the following settings for the aliases and redirects configuration (do not change the settings for fields not listed below):
  - URL redirects
    - From-

```
/
```

    - To-

```
https://servername.net
```

- Click "Save".
- Click "Apply Changes".

13.0 SquirrelMail

We've got the mail system and web server running. Now we just need a user interface to drop into place to give this solution a face to the users.

SquirrelMail is a PHP-based web mail solution that is very popular and considered to be relatively secure. It does this with a clean and easy interface that includes some nice features, both built-in and through modules that add additional features, like spell checking, attachment handling, user address books, user password changing, and even features like remote POP3 mail account retrieval.

13.1 SquirrelMail Login Custom Logo Graphic

If you like, you can transfer a moderately-sized (less than 100KB) logo graphic to the mail server to be shown at the SquirrelMail login. The file should be in PNG, JPEG, or GIF format. The easiest way to transfer the file is to simply download a logo from an existing web or FTP server[43].

- At a server console, type the following commands:

```
cd /srv/www/htdocs/squirrelmail/images/
sudo curl -O http://URL-to-graphic
```

- Note the name of the logo graphic file.

13.2 SquirrelMail Plugins

13.2.1 Change_Pass SquirrelMail Plugin

Let's get a plugin to allow the users of the system to change their own passwords.

- Go to the following website and write down the URL of the latest "change_pass" (not "change_passwd") plugin:

```
http://www.squirrelmail.org/plugins
```

[43] You can also use a floppy disk or USB flash disk to move the file to the server, but I won't be covering that procedure in this document. Look here for more information- "Using a USB flash drive with Linux." URL: http://www.extremetech.com/article2/0,3973,1256701,00.asp

- At the server console, type in the following commands (using the URL and file name you wrote down):

```
cd /srv/www/htdocs/squirrelmail/plugins/
sudo curl -O http://www.squirrelmail.org/plugins/change_pass-2.6-1.4.x.tar.gz
sudo tar xzf change_pass-2.6-1.4.x.tar.gz
sudo chown -R root:root change_pass
```

13.2.2 Install Poppassd

The change_pass SquirrelMail module requires a poppassd change password server. This is a simple interface that pipes requests from port 106 into a PAM module to allow password changes. Poppassd is not secure for remote use, but we'll only be using it from the local server, and we'll block remote access to it.

- Go to the Webmin interface.
- Click on the "System" icon and "Software Packages".
- In the "Install a New Package" area, use the following settings:
  - Click on "From ftp or http URL" button.
  - In the "From ftp or http URL" field, enter the following URL[44]:

```
http://ftp.silug.org/pub/kspei/add-ons/fedora/1/i386/poppassd-1.8.4-
0.i386.rpm
```

- Click "Install".
- Read the notice and click "Return to module index".
- Now click on the "Networking" icon and "Extended Internet Services".
- Click on poppassd.
- Set Service Enabled? to Yes
- Click "Save".
- Click "Apply Changes".

13.2.3 Administrator SquirrelMail Plugin

Let's set up the Administrator plugin to allow us to do SquirrelMail administration directly from within SquirrelMail, logged in as the admin user.

- Type the following (switch *adminuser* with your administrator user account name):

```
sudo chmod 660 /srv/www/htdocs/squirrelmail/config/config.php
sudo chown adminuser:www /srv/www/htdocs/squirrelmail/config/config.php
```

Once the SquirrelMail setup is done, you'll be able to administer many of the SquirrelMail settings directly from within your admin account's options.

[44] If this URL doesn't work, I have mirrored the file here- http://davebailey.homeip.net/secure-mailserver/poppassd-1.8.4-0.i386.rpm

13.3 SquirrelMail Configuration

We have already installed SquirrelMail, but we have some configuration we need to take care of before using it.

13.3.1 Securing SquirrelMail

We need to secure the SquirrelMail attachments directory.

- At the server console, type in the following commands:

```
sudo chown -R root:www /var/lib/squirrelmail/attach
sudo chmod -R 730 /var/lib/squirrelmail/attach
```

13.3.2 SquirrelMail Configuration Script

- At the server console, type in the following command:

```
sudo /srv/www/htdocs/squirrelmail/config/conf.pl
```

- If it asks about an old config.php, answer Y
- If it asks to stop warning you, answer Y
- Enter the following configuration information into the SquirrelMail configuration (do not change the settings for fields not listed below):
- 1. Organization Preferences
  - Organization Name- Enter the name of the server's organization
  - Organization Logo- ../images/*name-of-logo-graphic*
  - Org. Logo Width/Height- If desired, you can enter a logo width and height.
  - Organization Title- Enter the name of the website here (it will be at the top of each webpage).
  - Signout Page- If desired, you can specify a different logout URL than the login screen.
  - Default Language- if desired, enter a language code.
  - Enter 's' to save the settings, and 'r' to return to the main menu.
- 2. Server Settings
  - Domain- Enter the domain of your mail accounts. This is not the FQDN of the server. (IE- if your account would be- jdoe@our-domain.org, the domain is our-domain.org .)
  - Enter 'A' to update the IMAP settings:
    - Server software- courier
    - Delimiter- detect
  - Enter 's' to save the settings, and 'r' to return to the main menu.
- 3. Folder Defaults
  - Default Folder Prefix- none
  - Show Folder Prefix Option- n
  - Trash Folder- INBOX.Trash
  - Sent Folder- INBOX.Sent

  - Drafts Folder- INBOX.Drafts
  - Default Sub. of INBOX- y
  - Show 'Contain Sub.' Option- n
  - Enter ‘s’ to save the settings, and ‘r’ to return to the main menu.
- 4. General Options
  - Hide SM attributions- If desired, you can hide the SquirrelMail attributions.
  - Allow editing of identity- n and y (This will keep our users using their originally given email addresses. We don’t need people spoofing mail from our server.)
  - Enter ‘s’ to save the settings, and ‘r’ to return to the main menu.
- 5. Themes- We’ll leave these at their default settings.
- 6. Address Books (LDAP)- We’ll leave these at their default settings.
- 7. Message of the Day (MOTD)- We’ll leave these at their default settings.
- 8. Plugins
  - Type in the corresponding number for the following plugins and press enter for each one:
    - administrator
    - abook_import_export
    - attachment_tnef
    - change_pass
    - squirrelspell
    - translate
    - Enter ‘s’ to save the settings, and ‘r’ to return to the main menu.
- 9. Database (we’ll leave these at their default settings)
- Enter ‘s’ to save the settings, then ‘q’ to quit the configuration menu.

13.4 SquirrelMail Maintenance

We’re going to create a small script to clean-up any left-over attachment temporary files so they don’t build up over time.

- At the server console, type in the following command:

```
sudo pico /etc/cron.daily/clean_squirrelmail
```

- Create the file with the following lines:

```
#!/bin/sh
# Clean out leftovers in SquirrelMail attachments directory
find /var/lib/squirrelmail/attach/ -type f -mtime +1 -exec rm -f {} \;
```

- Save and exit the file.
- Type in the following commands to set the privileges on the file:

```
sudo chown root:wheel /etc/cron.daily/clean_squirrelmail
sudo chmod 770 /etc/cron.daily/clean_squirrelmail
```

13.5 Customizing SquirrelMail

Although outside of the scope of this document, the SquirrelMail interface can be themed and completely customized, if desired.

Check out the themes in the SquirrelMail configuration script, or look at this website for more information:
http://www.squirrelmail.org/wiki/en_US/CustomizingSquirrelMail

14.0 Hardening the Server

Bastille Linux logo. Used with permission.

We’ve already done several things to make the server more secure. This includes things like installing the least amount of software needed to perform the tasks the server needs to do; tightening up settings as we configure the server; and configuring the server to automatically download and install important system patches.

Now we’re going to go a step further and configure the security settings for the system.

While this is not a complete list of ways to harden the server, this example is a good start.

14.1 Enforcing Secure Password Requirements

Because passwords, along with the user accounts, are the only way users can prove themselves in most situations, password security is of paramount importance. Until an alternative unified, popular standard that includes most avenues of authentication, such as web logins, can be created, this will likely continue to be the case for a while.

As we reviewed earlier, “The best password uses a mixture of different types of characters, in a pattern easy to remember but hard to guess, and long enough to make brute-force attacks difficult to impossible, but not so long as to require your users to write them down. Also, because passwords can be cracked if given enough time, passwords should be required to be changed on a regular basis.”

We’ve discussed the various things that makes a password strong. However, your users may not be as informed as you about these things. Most users will choose simple and short rather than complex and long. This requires us to set some rules for selecting a password for our users, and enforcing those rules.

14.1.1 Using /etc/login.defs to enforce password aging

By changing a setting in the /etc/login.defs file, we can set up password aging.

- At the server console, type in the following command:

```
sudo pico /etc/login.defs
```

- Find the PASS_MAX_DAYS line and make it look like the line below (or choose after how many days the password must be changed):

```
PASS_MAX_DAYS   120
```

- Save the file and exit the editor.

These settings will take effect for new accounts when users are created using these rules.

14.1.2 Strengthening Passwords with PAM and pam_passwdqc

Many modern versions of Linux contain a technology which originated from Sun Microsystems called Pluggable-Authentication Modules (PAM). PAM allows us to set rules about settings or changing passwords.

SUSE Linux 9 includes an excellent password strength checking PAM module called pam_passwdqc[45]. While we can set a minimum password length with the existing PAM modules, pam_passwdqc allows us to require passwords of not only a certain length, but containing various combinations of different kinds of characters. The different classes of characters are lower-case letters, upper-case letters, digits, and symbols. Let's enable it to require better passwords from our users.

While we could edit the PAM files manually, this is error-prone, so we'll use Webmin to make this change.

- Go to the Webmin interface.
- Click on the "System" icon and "PAM Authentication".
- Click on "passwd" for the Password change module.
- Scroll down to the Password change steps section.
- Select pam_passwdqc.so in the popup menu and click Add step for:
- Enter the following settings in the "Add PAM Module" configuration (leave the non-specified fields as they are set):
  - Module arguments- (these settings require passwords with at least three different kinds of characters and a minimum length of 8 characters, or passphrases with at least 12 characters and the default of 3 words. Simpler passwords are disabled.[46]):

    ```
    min=disabled,disabled,12,8,8
    ```

  - Click "Save".
- On the "pam_passwdqc.so" line, click the up arrow twice to move it to the top of the Password change steps section.
- Click "Module Index".
- Make the same changes for the following PAM services-
  - login, poppassd[47], and other (put it after the pam_warn module)

[45] For information on pam_passwdqc look here- http://www.openwall.com/passwdqc/

[46] For more information about the module arguments for pam_passwdqc, read the man pages or this website- http://www.openwall.com/passwdqc/README.shtml

[47] Although not necessary, you can remove the pam_cracklib module. It's redundant.

14.2 Firewall (IPTables/Netfilter)

Now we’ll configure the firewall (IPTables/Netfilter) to allow access only to those services we want to allow access to from the outside.

- At the server console, type the following command:

```
sudo yast
```

- Select Security and Users and Firewall.
- At Firewall Configuration (Step 1 of 4): Basic Settings-
  - External Interface- eth0
- Select Next.
- At Firewall Configuration (Step 2 of 4): Services-
  - Enable the following services to be available-
    - HTTP
    - HTTPS
    - SMTP
    - Secure Shell (SSH)
  - Select Expert...
    - Additional Services- 10000
  - Select OK.
- Select Next.
- At Firewall Configuration (Step 3 of 4): Features-
  - Allow Traceroute- Uncheck
  - Forward Traffic and Do Masquerading- Uncheck
  - Protect All Running Services- Check
  - Protect from Internal Network- Check
- Select Next.
- At Firewall Configuration (Step 4 of 4): Logging Options-
  - Leave defaults (critical dropped and accepted packets logged, all others not logged)
- Select Next.
- At Save settings and activate firewall-
  - Select Continue.
- Select Quit.

14.3 Bastille Linux

Now, we’ll use the Bastille Linux scripts to configure some other system security settings to help tighten up security on the system. Bastille does not do much in this system configuration. Perhaps it is because much of what it secures is not installed. We’ll go through it just so you can see it.

- At the server console, type the following command:

```
sudo bastille
```

- Enter the following settings to Bastille to configure the security settings:
  - Type accept to accept license
  - Select “Next”.
  - Do you want to set a default umask? Yes
  - Would you like to display "Authorized Use" messages at log-in time? No
    - (We’ll create our own.)
  - Do you want to stop sendmail from running in daemon mode? No
    - (We aren’t running sendmail anyway, and we do want to leave Exim running as a daemon.)
  - Are you finished answering the questions? Yes
  - Tab to finish

14.4 Configuring the Login Banners

A large part of investigating or prosecuting computer crime is ensuring that our users (authorized or unauthorized) are warned of what rules are in force. By continuing to log in, they allow administrators to monitor their traffic and gather evidence for criminal actions, if necessary. Without these banners, admission of this evidence into court is difficult or impossible.

From Appendix A: Sample Network Banner Language of “Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations” written by the United States Department of Justice[48]:

> Network banners are electronic messages that provide notice of legal rights to users of computer networks. From a legal standpoint, banners have four primary functions. First, banners may be used to generate consent to real-time monitoring under Title III. Second, banners may be used to generate consent to the retrieval of stored files and records pursuant to ECPA. Third, in the case of government networks, banners may eliminate any Fourth Amendment "reasonable expectation of privacy" that government employees or other users might otherwise retain in their use of the government's network under O'Connor v. Ortega, 480 U.S. 709 (1987). Fourth, in the case of a non-government network, banners may establish a system administrator's "common authority" to consent to a law enforcement search pursuant to United States v. Matlock, 415 U.S. 164 (1974).

You will need to review these examples and decide what text you wish to display. You may want to get the advice of an attorney in your decision if this will be made into a production system. If you have a formal policy dictating what the login banners will display, you can put that text into the login banners.

[48] United States Department of Justice. “APPENDIX A: Sample Network Banner Language.” Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations. July 2002.
URL: http://www.usdoj.gov/criminal/cybercrime/s&sappendix2002.htm#_A_ (14 Apr. 2004).

14.4.1 Editing the Linux Banner Files

There are two banner files that we will be editing as part of this section.

- /etc/issue
- /etc/motd (message-of-the-day)

On this server, these banners are shown at the server console, after a console login, before and after an SSH login, prior to authenticating to Webmin, and after authenticating to SquirrelMail. We'll also be adding it to the SquirrelMail login screen.

Both files are edited in a similar manner.

- To edit a banner, enter the following commands:

```
sudo pico /etc/motd
sudo pico /etc/issue
```

- Then edit the file to contain the information that you want to be shown.
- For the purposes of this example, we can use the following banner text:

```
WARNING: Authorized users only. This system is monitored to ensure network
security and to track and possibly report any misuse or illegal activity to
the proper authorities. Use of this system shall constitute consent to
monitoring for such purposes.
```

- Save the file and exit the editor.
- Make certain to place the same text in both banner files.

Although we won't be doing it here, you can also place a banner in the SMTP server that will be seen by anyone connecting to the SMTP port on the server. We have already set a generic banner earlier. Usually, people don't see this message, only SMTP client software. However, if you'd like to place a similar banner to the ones we've already set up, check section 9.1.

14.4.2 Placing the MOTD Banner in the SquirrelMail Login

Even though SquirrelMail displays the MOTD banner *after* login, we also want it to be shown *prior* to login to ensure that the users are aware of our policy prior to login.

- To place a pre-login SquirrelMail banner, type in the following commands:

```
cd /srv/www/htdocs/squirrelmail/src
sudo cp login.php login.php.dist
sudo pico login.php
```

- Find the following block of text in the document (starting on line 118 for me):

```
html_tag( 'table',
    html_tag( 'tr',
        html_tag( 'td',
            _("Name:") ,
        'right', '', 'width="30%"' ) .
        html_tag( 'td',
```

- Insert the following lines so that it looks like this following (added lines in bold):

```
html_tag( 'table',
    html_tag( 'tr',
       html_tag( 'td',
          $motd , 'left', '', 'colspan=2'
       )
    ) .
    html_tag( 'tr',
        html_tag( 'td',
            _("Name:") ,
        'right', '', 'width="30%"' ) .
        html_tag( 'td',
```

- Save the file and exit the editor.

The /etc/motd file will now be displayed prior to the SquirrelMail login. It is already shown once in the main mail window immediately after logging in.

15.0 Next Steps

Now that we've got the server running, what are the next steps? Well, the sky is the limit, but we'll start off with some basics.

15.1 Creating Users

This server has two login users, root and the admin user. We need to create all of the users that will have accounts on this system. If this were a production server, there are some good ways to create users both singly and in bulk. We'll go over some different ways to create individual users. Bulk user creation is beyond the scope of this document.

15.1.1 Creating Users from the Shell

We can create users from the command-line shell.

- At a system console prompt, type the following commands to create a new user (change *username* to the account name of the user you wish to create):

```
sudo useradd -m username
sudo passwd username
(type in the password twice)
```

- Repeat for as many users as is needed.

15.1.2 Creating Users from YaST

We can create users within the SUSE Linux YaST utility.

- At a system console prompt, type the following command:

```
sudo yast
```

- Select Security and Users and Edit and create users.
- Select Add
- At the User Data section, type the following settings:
  - Full User Name- Enter the user's full name, if desired.
  - User Login- Enter the user account name.
  - Password- Enter the password.
  - Verify Password- Enter the password again.
- Select Create.
- Repeat as needed for as many users as is needed.

15.1.3 Creating Users from Webmin

We can create users from the Webmin interface.

- Go to the Webmin interface.
- Click on the System icon and Users and Groups.

- Click “Create a new user”.
- At the Create User screen, enter the following settings (do not change the settings for fields not listed below):
  - User Details
    - Username- Enter the user account name.
    - Real name- Enter the user’s full name, if desired.
    - Shell- Set to /bin/false .
    - Password- Click normal password and enter the password
  - Password Options
    - Maximum Days- 120 (or however long you would like until the password expires).
  - Group Membership
    - Use defaults.
  - Upon Creation
    - Use defaults.
- Click “Create”.
- Repeat for as many users as is needed.

15.2 Ongoing Maintenance

Regular maintenance should be done to a server while it is in production. While this document will not attempt to list all of these things, here are some things to consider-

15.2.1 Log files

SUSE Linux maintains many logs of things that happen on the server. These logs should be regularly checked for important information.

Here is a list of log files that this solution maintains that should be checked regularly for problems and signs of intrusion.

- Linux boot log- /var/log/boot.msg
- Linux general log- /var/log/messages
- Apache Logs- /var/log/apache2/*
- ClamAV Logs- /var/log/clamav/*
- Exim Logs- /var/log/exim/*
- Other Mail Logs- /var/log/mail
- SUSE YaST Logs- /var/log/YaST2/*

15.2.2 Backup

If the information on the server is important, it should be backed up. Data backup is a form of security in case the server is compromised.

While backup is beyond the scope of this document, there are many good resources for backing up a Linux server on the Internet. Here are a few-

The Linux System Administrator's Guide- Simple Backups
http://tldp.org/LDP/sag/html/x2570.html

Linux Complete Backup and Recovery HOWTO
http://tldp.org/HOWTO/Linux-Complete-Backup-and-Recovery-HOWTO/index.html

A BACKUP STRATEGY FOR LINUX VIA CD-R
http://www.bluehaze.com.au/unix/cdbkup.html

Easy Automated Snapshot-Style Backups with Linux and Rsync
http://www.mikerubel.org/computers/rsync_snapshots/

NSC: Secure Backups with Amanda and GPG
http://security.uchicago.edu/tools/gpg-amanda/

There are also, of course, plenty of good commercial backup solutions available.

15.3 Ways to Improve This Example

While this example may be the first step in a good direction, there are still many ways to improve on this solution. Listed below are some directions I think would be a good next step.

- Better Security Through Logging and Intrusion Detection/Avoidance
  - Regular analysis of various system logs utilizing log analysis tools.
  - Configure support for an external logging server.
  - Implement Tripwire and other IDS agents to track system changes and possible attacks.
  - Implement adaptive firewall policies to block attackers for a period of time[49].
- Better Anti-Spam Technologies
  - Investigate and possibly utilize other new and improved spam-filtering/blocking technologies in addition to, or replacing, SpamAssassin (Razor[50], CRM114[51], DSPAM[52]).
  - Shared spam and ham folders to increase the learning capabilities of SpamAssassin.
  - Investigate possibilities for challenge/response MTAs such as TMDA.[53]
- Include automated secure backup to remote server
  - Use of tar with gpg[54] (local secure backups)
  - Use of tar with ssh[55] (remote secure backups)
- Better Scalability
  - Configure the server with “virtual” users utilizing a MySQL back-end. This improves configurability and performance in large-scale deployments.
  - Performance tweaks, such as increasing the number of listening processes, to improve performance on a more heavily loaded system.
- Better User Interface to Improve Security
  - Store individual user’s mail filtering/antispam settings in MySQL and provide an interface to customize the user’s specific anti-spam/virus settings. SquirrelMail already has some capability to handle these kinds of settings, such as SpamAssassin, through plugin modules.
  - Send a reminder email to the user when the password is about to expire.
- More Hardening
  - Use security tools, such as Nessus[56], to determine how secure the server is in its current configuration.

[49] For more information about adaptive firewalls, check here- http://www.stearns.org/doc/adaptive-firewalls.current.html

[50] For more information about Razor, check here- http://razor.sourceforge.net/

[51] For more information about CRM114, check here- http://crm114.sourceforge.net/

[52] For more information about DSPAM, check here- http://www.nuclearelephant.com/projects/dspam/

[53] For more information about TMDA, check here- http://tmda.net/

[54] An article showing tar with gpg- http://www.webpronews.com/it/networksystems/wpn-21-20031009EncryptingBackups.html

[55] Some information showing an example of tar with ssh- (See the Piping Binary Data to a Remote Shell section.) http://www.linuxjournal.com/article.php?sid=6602

[56] For more information on Nessus, check here- http://www.nessus.org/

16.0 References

CERT Coordination Center. “Deploying Firewalls.” 20 Apr. 2001.
URL: http://www.cert.org/security-improvement/modules/m08.html (03 Apr. 2004).

CERT Coordination Center. “UNIX Configuration Guidelines.” 04 Jun. 2003.
URL: http://www.cert.org/tech_tips/unix_configuration_guidelines.html#A1
(02 Apr. 2004).

Granger, Sarah. “The Simplest Security: A Guide To Better Password Practices.” 17 Jan. 2002. URL: http://www.securityfocus.com/infocus/1537 (05 Apr. 2004).

SANS. “SANS Top 20 Vulnerabilities.” Version 4.0. 08 Oct. 2003.
URL: http://www.sans.org/top20/ (14 Apr. 2004).

“SuSE Linux Unofficial FAQ.” 04 Feb. 2004.
URL: http://susefaq.sourceforge.net/ (05 Apr. 2004.)

Exim.org. “Exim 4.30 Specification.”
URL: http://www.exim.org/exim-html-4.30/doc/html/spec.html (05 Apr. 2004).

Rennich, Joel. “Exim 4.10 with integrated spam and virus controls.”
19 Feb. 2003. URL: http://www.afp548.com/Articles/Jaguar/eximandamavis.html (02 Apr. 2004).

Merlin, Mark. “Exim4: Very detailled and featureful configuration example.”
03 Aug. 2003. URL: http://marc.merlins.org/linux/exim/files/exim4-conf/
(07 Apr. 2004).

Jackson, Tim. “Spam and Virus Scanning with Exim 4 using Exiscan and/or SA-Exim Mini-HOWTO.” V1.0.10. 25 Feb. 2004.
URL: http://www.timj.co.uk/linux/Exim-SpamAndVirusScanning.pdf (10 Apr. 2004).

Finch, Tony. “Runtime configuration file for Exim 4.” 14 Aug. 2003.
URL: http://www-uxsup.csx.cam.ac.uk/~fanf2/conf4.satellite (10 Apr. 2004).

Yoder, S. “Setting up Exim 4.x.” 04 Jan. 2004.
URL: http://www.flatmtn.com/computer/Linux-Exim4.html (12 Apr. 2004).

Pircher, Thomas. “Exim.” 01 Mar. 2004.
URL: http://www.tty1.net/exim_en.html (09 Apr. 2004).

“SquirrelMail Documentation.” 10 Apr. 2004.
URL: http://www.squirrelmail.org/wiki/SquirrelMail (13 Apr. 2004.)

“SquirrelMailAndCourierIMap.” 06 Mar. 2004.
URL: http://www.squirrelmail.org/wiki/en_US/SquirrelMailAndCourierIMap (13 Apr. 2004).

“SquirrelMail QuickAndDirty.” 12 Mar. 2004.
URL: http://www.squirrelmail.org/wiki/en_US/QuickAndDirty (14 Apr. 2004).

United States Department of Justice. “APPENDIX A: Sample Network Banner Language.” Searching and Seizing Computers and Obtaining Electronic Evidence in Criminal Investigations. July 2002.
URL: http://www.usdoj.gov/criminal/cybercrime/s&sappendix2002.htm#_A_ (14 Apr. 2004).

17.0 Image Credits

In order of use by section-

SUSE Linux- “SUSE Logo.” The SUSE logo is a registered trademark of SUSE and Novell, Inc. Used with permission. Neither Novell nor SUSE have reviewed this paper nor do they endorse the content of it.

Webmin- “Webmin Icon.” Images designed by John Smith. Public use allowed.
URL: http://www.webmin.com/graphics.html

Secure Shell- “A procedurally modeled shell with definitively non-natural colour.”
Copyright © 1999 Alexander Wilkie. Used with permission.
URL: http://www.cg.tuwien.ac.at/research/rendering/ART/Gallery/General/

ClamAV- “ClamAV Logo.” Used with permission.

Exim- “Exim Logo” by Jennifer Greenley. Used with permission.

SpamAssassin- “SpamAssassin Logo.” Used with permission.

Courier-IMAP- Original artwork by David R. Bailey.

Apache- “Apache Logo.” Used with permission.

SquirrelMail- “SquirrelMail Logo.” Used with permission.

Bastille Linux- “Bastille Linux Logo.” Used with permission.