Global Information Assurance Certification Paper

Securing A Samba File Server using Red Hat ES 2.1 on a Dell 2600

A Step by Step Guide

By Gary Long
System Administrator

12/2/2003

Contents **Page**

1. **Introduction** 4

   1.1. The existing Environment 5

   1.2. Overview 6

2. **Goals and Risk** 7

   2.1. Server Role and Function 7

   2.2. Cost 8

   2.3. Social and Legal Considerations 8

   2.4. Data Redundancy and Backup 9

   2.5. Network Risk and Security Strategy 10

   2.6. Goals and Risk Overview 11

3. **Why Red Hat ES and Dell?** 12

4. **Installing and Securing the Red Hat ES Operating System** 14

   4.1. Preparation 14

   4.2. Raid 15

   4.3. Create driver 15

   4.4. Install OS 17

   4.5. Firewall 20

   4.6. Register 21

   4.7. A View from the Network 21

   4.8. Automated Security 23

   4.9. Configuring Services 29

   4.10. *Passwords* 32

   4.11. ***sshd Configuration*** 34

4.12. X11 37

4.13. System Logs 38

4.14. Stack Protection 41

4.15. System Integrity 41

5. Samba Configuration 47

5.1. Server Function 47

5.2. SWAT 47

5.3. Setup smb.conf 49

5.4. Samba Security 54

5.5. Samba Configuration Summary 54

6. Backup 55

7. Maintenance 58

8. Checking the Configuration and Verifying System Security 61

8.1. Penetration Testing and Vulnerability Scanning 61

8.2. Don't just scan the box 63

8.3. Security works, can the workers? 68

9. Appendices 70

9.1. Ip Tables 70

9.2. Selections: Example output from IP tables script 73

9.3. Bastille Syslog.conf 74

9.4. Smb.conf 75

10. Bibliography 76

11. Webliography 77

1. Introduction

Securing a Samba File Server Using Red Hat ES 2.1 on a Dell Power Edge 2600 describes the set up, configuration and administration of the Red Hat operating system on a Dell Server using Samba for Security. This manual includes testing for vulnerability and maintenance. The Appendices contains documents that have been referenced throughout this paper.

This guide was written for system and network administrators who have a fundamental level of skill with Samba and the Red Hat operating system. The Appendices includes documents referenced throughout this guide.

Securing a Samba File Server Using Red Hat ES 2.1 on a Dell Power Edge 2600

This section provides an overview of the existing environment, hardware and operating system.

The purpose of this document is to demonstrate the steps taken to install Red Hat ES 2.1 on a Dell Power Edge 2600 server. This document will outline, step by step, the installation while also showing the configuration of Samba in a Windows environment. The steps to configure disks in a RAID 5 array are included. A section also contains information on how to run tests using freely available scanning security tools. These tools can be used to validate the security of a system, however, when using these tools it is important to obtain permission from the proper persons. The use of the tools may be considered hostile and cause problems that could have otherwise been avoided. Included is information on using automated tools, as well as manually changing settings that will enhance the security of this server. Software and driver file resources are available also in this paper.

This document starts with an explanation of the existing environment and gives a short risk assessment of the server to be installed. Then, the process of installing and securing the operating system is documented. Before proceeding in your own installation, be sure to evaluate the requirements of your system and network as demonstrated in this paper. Certain steps may or may not apply to your situation. Care has been taken to apply standard UNIX security practices in this document.

1.1 The Existing Environment

Determining the system design requires a brief explanation of the environment that the system will operate in. The customer has certain requirements that must be fulfilled. The objective is to meet the customer's needs satisfactorily, while providing a secure environment in which to work. Understanding the existing environment and the customer's desired outcome will determine what actions are necessary to provide an acceptable level of security.

The customer is responsible for overseeing clinical research at a reputable University. In a campus environment there are many factors to consider that may differ from a corporate environment. The network infrastructure is not maintained by the system administrators; therefore, some options available in other environments are not an option in the campus environment. One major downside of this is that the persons in charge would like to maintain control of every port. This policy makes it difficult to obtain permission to use a firewall appliance that places critical servers more out of reach of the internet. Another downside is that all servers on the network are internet facing servers. All machines on the campus network utilize public IP addresses. These reasons make security design at the system level absolutely critical. In all fairness to the university, there are some security measures in place network-wide. The campus does operate a switched network that may aide in the prevention of sniffing traffic across the wire. The university has also employed the use of Intrusion Detection and Intrusion Prevention devices. The campus security team has also blocked certain "troublesome" ports at the border router. To say that the campus network is "wide open" would not be entirely true. To develop a successful security design requires an understanding the existing environment.

1.2 Overview

To achieve an effective security strategy requires a layered approach to the design. This layered approach to security is sometimes compared to the layers of an onion. To get to the center of the onion requires penetrating many layers. Perhaps the intrusion prevention devices that are in place are extremely capable of stopping attacks however the devices are maintained by people and is a possibility for error. This must be taken into account when designing the system security. The system must be secured as if there is no other line of defense between it and the world.

A risk assessment is provided in this document that will explain in greater detail the needs of the customer. The risk assessment will also explain some of the possible security concerns and what will be done to prevent system compromise. A compromise of the system can be defined as intrusion by non-authorized persons. System failure would also fail to meet the needs of the customer. Assessing the existing environment is necessary to implement an effective security design.

2. Goals and Risks

This section covers the goals and risk that must be considered before implementation of the server and its security measures.

2.1 Server Role and Function

Before deciding exactly how to implement the new server, the function and role of the new server need to be defined. The needs and expectations of the customer must also be addressed.

To determine the implementation of the new server there are some questions that must first be answered.

- What type of data will be stored on this system?
- What sort of availability of the data is expected by the customer?
- What consequences would the customer suffer if the system were compromised?
- Would the customer face financial consequences or political consequences?
- Could there be legal consequences?

The customer is involved in clinical research studies. These studies involve the acquisition of patient data that may or may not have identifiers attached to it. It would be prudent to err on the side of caution and treat the data as if it absolutely had patient identifiers attached to it. While proper procedure is to strip the data of these identifiers, there always remains the possibility for error. With such sensitive data it is best to assume that the identifiers are there. This assumption helps to maintain the importance of system security.

The customer needs a file server to act as a storage facility for all of the acquired data. The data is processed and then stored for examination at a later date. The server needs to perform no other role in the customer's network design. The customer maintains standard operating hours of 7:00AM to 5:00PM Monday through Friday. During this time the customer expects access to the file server. It is important to understand the needs of the customer in order to define the server role and function.

2.2 Cost

One of the first considerations when assessing a new system or environment is the cost. What financial impact will the customer suffer if the new system is unavailable during the hours of operation? Currently the site has four employees responsible for maintaining the data. If the system is down during business hours, at least four employees may not be able to adequately perform their duties. If the employees are paid $20 an hour, the customer loses $80 an hour, for every hour the employees can not perform their duties owing to system failure. This loss of productivity is one aspect of cost to the customer. If there is hardware failure, the cost of the new hardware would be added to the cost of lost productivity. When considering the cost of hardware replacement, the worst case scenario would be the need to replace the entire server. In this case, obtaining a new server that has similar specifications may cost $6000.

Now that the cost of lost productivity and hardware replacement has been considered, the value of the data stored on the server must also be considered. The cost for hardware and software must also be accounted. A dollar figure could be placed on the data by counting the cost of each patient visit. A typical visit will cost $200. The hours of labor to process the information could be 10 hours per patient visit, so that adds another $200 per visit. The replacement cost of each data set could be estimated at $400. These studies are based on many visits over time. If the datasets become lost or corrupted, it would be very costly. Incremental patient data lost is priceless to the customer and patient. If the data were lost, the project would have to start over from scratch and then the project could outlive the principal investigator or lose funding.

2.3 Social and Legal Considerations

Aside from the possible financial loss and loss of productivity, there may be other ramifications if something were to happen to the data. A compromised machine with sensitive data could be quite a blow to an institution that maintains such a high reputation and good rapport with community. The politics involved in this dilemma can only begin to be addressed. The distrust over compromised data could cause problems recruiting new subjects for study or obtaining funding, thus slowing the entire research project. Researchers would have to spend more time hand holding and trying to convince and reassure the community that it will not happen again. Re-building trusts and community confidence would prevent researchers from doing the research that they are in the business to do. These potential problems can be prevented with proper security assessment and implementation.

Finally, the possible legal issues surrounding any unfortunate mishaps must be considered. This touches the issue of the stripped patient identifiers and possibility of human error. Certainly with the advent of HIPAA and the publicity behind it, there are attorneys, just “chomping at the bit” for some unfortunate mishap. Loss of money and productivity could easily be overshadowed by the threat of lawsuits.

The challenge is to design as much security as possible into this server. Even though the data is so valuable the customer only has a certain amount of money available to them for this project. The solution must fit within the framework of the budget and the security objectives. In addition to securing the operating system, there are certain security features that can be built into the server for protection.

In summary, the role and function of the new server has been defined. The possibility of loss of productivity was discussed. The value of the data was made clear. The possible political and legal consequences were also considered. Now that these issues have been addressed, there needs to be a description of steps taken to help prevent such consequences. The goals, risk and cost involved must be considered when implementing successful security strategy.

2.4 Data Redundancy and Back up

A new Dell Power Edge 2600 was purchased for this project. It was designed with enough hard disks to implement a RAID 5 configuration with an extra drive available as a hot spare. The RAID 5 configuration is a layer of security, in that it provides us with data redundancy in the event the drive fails. The hot spare will help protect the customer from loss of production while waiting for the arrival of a new hard disk. The system also features redundant power supplies. This will also protect the customer from loss in productivity. The server also has an onboard DLT tape drive to backup the data. Some of the tapes can then be stored offsite in a safe location. This process will help protect the customer from data loss from fire, flooding, or theft. The server is located in an environmentally controlled room that is kept locked. By restricting physical access to the server the possibility of console access is also reduced. Controlling physical access also helps protect from such unfortunate accidents as spilling a soda on the system. To add another layer of protection, the server is connected to an uninterruptible power supply.

2.5 Network Risk and Security Strategy

Network security is the next and greatest concern. The server can be locked in vault and sealed in iron, however if the network cable is attached there is a threat. By its very nature the fileserver must have certain services running. It is the possible compromise of these services that presents concerns regarding security.

As a Samba server there will be certain ports open for *smb* and *nmb* such as *port 137* and *139*. These ports have been the target for vulnerabilities in the past.
SWAT will also be used to aide in the management of Samba. SWAT is a web interface that allows a person to perform various Samba administrative tasks and SWAT runs on *port 901*.
Also *ssh* will be used to administer and manage the machine thus causing *port 22 to remain open*. In the past vulnerabilities were found in ssh and that lead to the development of ssh V.2. Some vulnerabilities have also been found in ssh2 so it is imperative that the system have the latest version or patched version of ssh.
The *Xserver* will also be running for ease of management tasks as well as ease of training beginning system administrators. The use of the Xserver creates the possibility of someone connecting to the *Xserver* via *port 6000* and remotely attacking the system.
Sendmail will also be used in order to email the administrators log files and alerts. *Sendmail* usually listens on *port 25*. This is not to be a sendmail server so there is no need to have the daemon listening for connections. The system only needs to send email to this site's email server.

Section eight contains a report describing how to verify that the services running on the ports above are configured to prevent intrusion and explains how to minimize this risk.

Security of the server must start at the very basic level of hardening the operating system. To harden an operating system means to make the system as difficult as possible for an unauthorized person to gain access. In other words if there were no firewall in place or tcp-wrappers in use this would still be a challenging machine to compromise. Further security can then be added by configuring a firewall. It will only allow certain hosts to access certain services. One key to the security of a server is to run only those services that are deemed necessary to do the job. File system data integrity checking tools as well as logging services will also be used. A secure system will not stay secure if left alone. Section 7 covers maintenance related tasks to help ensure the system remains secure.

2.6 Goals and Risk Overview

Possible consequences have been discussed and considered. The goals of the customer and risk involved reviewed. The next step is to explain to the customer the conclusions and decisions that have been made.

The next section will explain the system specifications, as well as the selection of Linux for an operating system. Considering the goals and risk will aide in the production of a successful plan to meet the customer's needs and budget while maintaining an acceptable level of security.

3. Why Red Hat ES and Dell?

This section provides in overview of why the Red Hat operating system and Dell hardware were chosen for this installation.

What is Red Hat ES? Red Hat ES is the Red Hat Enterprise Server edition of Red Hat Linux. Red Hat has different flavors of their enterprise edition of linux, each with certain advantages. The main difference among the server editions is the amount of support that is included in the purchase. Also Red Hat ES is only available for download while the more expensive server edition includes the media and hard copies of the documentation. Some of the other editions may include other enhancements not mentioned in this document. For further information regarding the different server editions that Red Hat has available please see www.redhat.com.

Why Red Hat ES for this project and not a free Red Hat distribution? Some may ask, "Why Red Hat was elected instead of a Microsoft file server?" These are good questions and certainly compelling arguments exist for any of these flavors of server operating systems. One might argue that if there are budget constraints, why not use one of the free distributions of Linux. The answer to this question is best answered on Red Hat's website. Thirty days of free installation support comes with the ES edition. A one year subscription to Red Hat Network also comes with the purchase of ES. The Red Hat Network is a web interface that can be used to manage systems and system updates. The Red Hat Network provides a facility that allows for an automated update procedure. There is also an automated mail notification that will alert you when updated packages are needed. Red Hat also proclaimed compatibility with the Dell server that is being used for the project. Another reason to use the ES edition is the "focused release cycle," the idea is to provide a stable platform for all phases of enterprise deployment. Major Red Hat releases will occur every 12 to 18 months, allowing customers to effectively plan migration and upgrade cycles.[1]

The other question might be, "Why not Microsoft?" One answer to that question is budget constraints. While the Red Hat ES software did cost $350, the Microsoft server editions are considerably more expensive. With Microsoft there is also the cost of additional client access licenses. The standard Microsoft server comes with 5 client access licenses. At the time of this writing, exactly five people from the customer's site will be accessing this server. In order to allow

[1] http://www.redhat.com/software/rhel/features

more connections, more client access licenses would need to be purchased. While the purpose of this paper is not to argue the security issues surrounding Microsoft, it is worth noting that the Red Hat operating system includes many facilities to secure the system on a more granular level than Microsoft servers.

The next topic for consideration is the choice of a Dell server. There are many servers available on the market; however, the decision to use Dell was motivated by three reasons.

1. Support. There are several other Dell servers in use at the customer's site, and whenever a problem has occurred Dell was very quick to respond.
2. Compatibility. Red Hat and Dell have maintained a good relationship and both were compatible with each other.
3. Price. Dell offers a good discount to educational institutions.

Hopefully this section has helped clarify why the decision was made to use Dell and Red Hat for this project. There are many arguments for using other products, and all are with merit; however, the combination of Red Hat and Dell is the best fit to meet the customer's needs. Choosing a fully supported OS and hardware configuration prevents productivity loss thus improving overall security risk recovery and success.

4. Installing and Securing the Red Hat ES Operating System

This section covers the installation of the Red Hat operating system and the implementation of a security strategy.

4.1 Preparation

Below are the hardware specifications of the server that is being used for this project. The server is a Dell Power Edge that contains the following components:

- Single Pentium Xeon 2.8Ghz Processor
- 512MB Memory
- 5 X 73GB Hard disk
- Dell Perc4 Raid Controller
- On Board 100Mbps Ethernet Card
- Internal DLT1 tape drive
- ATI 8MB Video
- CD-RW drive
- Floppy drive
- Redundant Power Supplies

The next chart shows the software that will run on the new server and which ports may be open in order for systems to connect to the server:

Software:	*Ports*:
Red Hat ES 2.1	-----
Samba 2.2.7	137,139,445
Sendmail 8.11	-----
Xserver	-----
OpenSSH 3.1	22
Legato Networker 7.0	

Once the necessary hardware has been obtained and setup, the installation and securing of the RedHat ES operating system can proceed.

Before trying to install the operating system:

1. The Perc4 controller will need an updated “megaraid” driver from Red Hat. See Creating Driver Disk

2. Do not attempt to use the Dell OpenManage Server Assistant CD that comes with the 2600. It does not support Red Hat ES. You must boot from CD 1 of 3 of the Red Hat ES media.

4.2 RAID

The first step in the installation is to configure the RAID controller to meet our requirements. Using a RAID 5 configuration is the first step in securing the server. The use of RAID 5 gives data redundancy should a hard drive fail. The default Dell installation includes all five drives in the RAID 5 configuration. For many this would be an adequate configuration, however the 5th drive was ordered with intention of being a hot spare. Having a drive as a hot spare would allow the immediate introduction of a new drive to rebuild the array, in the event of drive failure.

Step 1: configure RAID

Boot system

Ctl-m to bring up PERC configuration menu

From the Configure menu select **new config**

Select 1st 4 drives for RAID5

Create array(no span)

Use 5th drive for **hotspare**

If you receive the message, “Initialization background mode”, start over. A search of Dell’s website did not reveal anything regarding this message. However after searching Google, I found a post by [1]Josip Rodin. In his posting he concludes that the best answer is to simply delete the array that was just created and reconfigure the array again. His answer proved to be the simplest solution. Configuring the array a second time and then choosing to initialize proved to be successful.

4.3 Creating a driver disk:

This is how the driver disk scenario should work:

[1] Josip Rodin, http://kt.zork.net/debian/quotes/Josip_Rodin.html

1. Download driver from Red Hat
2. Use a utility (dd on linux, rawrite on windows) to create the disk
3. Boot to Red Hat Installation Disk 1 of 3
4. At the prompt type linux dd for installation type
5. Prompt you for installation disk
6. Observe drivers loading
7. System should proceed to normal installation process

This is the process I followed to create the disk:

The driver is available for download at:
[1]http://www.redhat.com/support/errata/rhel/21/qu2/driverdisks.html

Then you must gunzip the file, then use dd to create the diskette

For those that may not be familiar with the dd command do this:

```
#dd if=file.img of=/dev/fd0 bs=1440k
```

(do not issue mount floppy before performing this command)

My experience was not quite so straight forward. The Dell PowerEdge installation manual for Red Hat ES instructs us to go to[2] www.redhat.com and retrieve the driver. I followed the instructions and went to www.redhat.com and did not feel that it was obvious where to find the driver. Finally, I did a search for megaraid and found the link where the driver could be downloaded. I downloaded the driver, used dd to create the disk. I went back to the server and proceeded to install the driver. It appears to work, however when I reached the point of the installation where disk partitions are to be configured, I kept receiving an error message. There was no valid device to create a filesystem. I thought perhaps this could be a problem with the driver disk I created. I downloaded the driver again and this time on a windows machine. I also went to [3]http://uranus.it.swin.edu.au/~jn/linux/rawwrite.htm and downloaded the latest version of rawritewin-0.7.zip. I used this utility to create the driver disk. I went back to the server again and the driver appeared to take. When time to partition the disk it failed again. At this point my frustration level peaks, as it does when something that should be simple is not. It is at this point that I decide to call it a day. (It's Saturday, I should be at home anyway not here in my little cave fighting the digital goblins of raid drivers. Also important to note if you did not pay extra for Red Hat support, the ES installation support is only available during business hours during the week.)

[1] For driver download see http://www.redhat.com/support/errata/rhel/21/qu2/driverdisks.html
[2] www.redhat.com for Red Hat drivers

[3]rawritewin-0.7.zip can be found at http://uranus.it.swin.edu.au/~jn/linux/rawwrite.htm

At this point, wasting time working with the downloaded driver is not profitable. It is time to call Red Hat. After just a few minutes of going through phone trees and listening to Red Hat propaganda I am greeted by friendly support guy. I explain the situation and he first wants to know the version of ES that I am trying to install. We look at some of the files on the CD to determine that I am using the right distribution. He tells me I need a certain driver instead of the one that I had (instead of directing me to a site to download it?) he emails it. We go through the disk creation process once more. Again we install driver and it appears to take. We reach the partition section and we finally have a usable device. So my advice is this: When you purchased Red Hat Linux ES 2.1, you receive thirty days of free installation support. Use it. Call Red Hat and get that driver before trying to create the driver disk.

4.4 Installing the Operating System

Now that we have prepared our system and have created driver disk we can begin the operating system installation.

Insert Disc I of IV of Red Hat ES 2.1

Choose linux dd from the options

Be sure to have driver disk (see creating driver disk above)
System will prompt you for your driver disk
You will see it loading certain drivers, including the megaraid_2002 driver
Then the system will drop into a graphical install mode

Choose appropriate language
Choose appropriate keyboard Configuration
Choose appropriate mouse configuration
Then system prompts you for installation type:
choose custom
Then manually partition with Disk Druid
Partitioning our server:

Partitioning a server with static partitions helps prevent DoS attacks caused by the filling of static partitions. An example of this would be having one large root partition and a directory on the partition called /var. If /var were to fill up because of logging messages or other reason the system would be inaccessible to other users because this would mean that / is full.

Our partition scheme:

(We do not want to delete or overwrite /dev/sda1, this is the Dell utility partition. It is recognizable as type vfat)
/boot – 50MB – Fixed Size - This is recommended by Red Hat
/ - 1024MB –Fixed Size- This is our root partition
swap – 1024 – Fixed Size -This is our swap partition, Red Hat recommends 1.5 – 2 X physical memory
/tmp – 512MB - Fixed Size
/var – 512MB - Fixed Size
/usr – 2048MB – Fixed Size
/home – 512MB-Fixed Size
/data – Fill to max – The purpose of this machine is to be a file server, so the rest of the available space is devoted to data storage

Your file system layout may differ dramatically. Only you can determine what is suitable for your site.

Chose GRUB as boot loader (GRUB stands for the Grand Unified Boot Loader. GRUB has many features such as compatibility for booting FreeBSD, NetBSD, OpenBSD, and GNU/Linux. Proprietary OS's such as Windows 9x/NT/200/XP, and OS/2 are supported via a chain-loading function. GRUB supports multiple executable formats and supports non-Multiboot OS's. GRUB supports a human-readable configuration file, menu interface and a flexible command line interface. GRUB also has the ability to support multiple file system types, access data on any installed device and detect all installed RAM.[1])

Install boot loader on /dev/sda Master Boot Record

Partition: /dev/sda5 default boot image

Choose GRUB password to protect the system from someone altering system properties. GRUB will allow anyone to edit entries, run arbitrary commands and even edit /etc/passwd. As stated before the system is behind locked doors that few people have access to. Setting the GRUB password will help prevent someone, who may just be curious, from causing problems

Network configuration:

eth0 activate on boot

Network info: IP,netmask,network,broadcast,gateway,DNS

Hostname: testhost.secure.edu

Firewall: for installation purposes chose HIGH (At this point the network cable is not connected. The purpose of choosing high, is to give the added protection

[1] http://www.gnu.org/software/grub

incase the network cable is inadvertently plugged in before the custom firewall gets configured.)

Additional language support: Not necessary

Time Zone:
Eastern Time America/New York

Root password: (see choosing good passwords)

Added users:

Authentication Configuration:
Enabled MD% passwords
Enabled shadow passwords

NIS –no
LDAP –no
Kerberos –no
SMB –no

Packages Selection Screen:
Select the following choices:

Classic X
Gnome
Network Support
SMB windows file server
Network managed workstation
Utilities
Kernel Development
Server

After choosing the groups of software choose individual packages that will be unnecessary and unwanted. Items such as telnet server, yptools, ypserv, etc. There are certain packages that need to be selected to insure that they DO get installed. Choosing the flat view will list all the packages in alphabetical order. Packages that should be installed include Tripwire, SWAT, and iptables-ipv6. The system then proceeds to the unresolved dependencies screen. Choose install packages to resolve dependencies. Unnecessary services and daemons that may have been missed can be disabled or uninstalled later.

Agree to the option for creating a boot disk

Once the boot disk is created test the systems X settings and make any necessary changes.

Reboot the system

If the SMP kernel is selected when the system comes back up you may experience a kernel panic error. The Dell Installation manual recommends to enable the logical processor setting in the BIOS, however booting into the BIOS reveals that this setting is already enabled. According to the Dell Installation Manual the only other option are to purchase a second processor or only boot the UP kernel. Purchasing a second processor does not fit the design plan for the customer's needs. However when the system is updated this problem goes away. Until the updates are done select the UP kernel to boot.

It will take some time for the system to format and install the operating system.

Once the system has finished the installation, some adjustments will need to be made to various network settings. In previous Red Hat installations the format of the /etc/hosts file has had an entry resembling the following. This setting is incorrect and can cause some problems:

```
127.0.0.1       blahsystem.blah.domain localhost.local.domain
```

This can be corrected by editing the /etc/hosts file to look like the following:

```
127.0.0.1       localhost.local.domain          localhost
192.168.40.1    blahsystem.blah.domain          blahsystem
```

This issue is corrected in Red Hat ES. A quick look at the */etc/hosts* file reveals that it is properly configured.

Next examine the */etc/sysconfig/network-scripts/ifcfg-eth0* file to be sure the proper information is listed there. This reveals that something is missing. The following entries are not present and should be added. Missing entries may not adversely affect the operation of the system however these entries are required by the script that will be used to configure the firewall for this system.

```
BROADCAST=192.168.40.255
NETWORK=192.168.40.0
```

4.5 Firewall

The next step is to configure iptables so that the system has a safe firewall configuration to use during the hardening process. Thanks to the people at Network Partners Group, LLC, the process of setting up iptables was not very intensive. The Network Partners Group team has developed a very thorough

script that walks you through the process of configuring the firewall.[1] In just minutes you can have a fully customized firewall that offers a very granular level of protection. Other than the ease of configuring the firewall, another great feature of this product is a file called the selections file. The selections file is a plain text file produced according to your answers about the firewall configuration. The selections file is in plain English so there is no need to spend time trying to decipher iptable's rules syntax. One quick glance will allow you to tell if a certain address is allowed access to a service. I have included a copy of the "iptables formatted" rules, as well as the output of the selections file in the appendices. The system is now configured only to allow ssh and samba to and from specific hosts.

4.6 Register

The next step is to register the box with Red Hat Network. If the system fails to connect to the Red Hat Network, it may be that the system is in need of the updated ssl certificates. The original ssl certificates expired. The new packages that fix this problem can be downloaded at http://rhn.redhat.com Go to the appropriate Red Hat site and download the certificates. Then run the md5sum program on each ticket only to find out if the md5 checksums match. For the installation of this system the md5sums did not match. I chose not to install them and instead logged a phone call to Red Hat tech support once more. The person was not able to tell me why the checksums did not match, however he directed me to another location and that had instructions how to issue a wget command that presumably downloaded and installed a safe version of the certificates. Once the installation was complete, I ran **rhn_register** to register my system with Red Hat Network. Then the **up2date –uf** command was used to update all the packages including the kernel. Now all the packages on the system are updated and any packages that may have had some known vulnerabilities have been patched.

4.7 A View from the Network

Before proceeding to secure the system it is a good idea to see the system as it is before spending time locking it down. An **nmap** scan will give a quick picture of how the system appears to the rest of the world. **Nmap** is free port scanner available from www.insecure.org. Using a port scanner will give a good indication of services that may need some attention. **Nmap** was run with different options. The commands used and the output of the commands are included.

---

[1] If interested in ip tables script email info@dowhiletrue.com

For the nmap scan to succeed the firewall needs to be stopped.

Use the following command to disable the firewall:

```
#/etc/rc.d/init.d/iptables stop
```

Now nmap can be run and results obtained successfully.

The command used was:

```
#nmap –P0 –O  –oN  /tmp/nmap_file1 192.168.0.140

# nmap (V. 2.54BETA22) scan initiated Sun Nov 16 16:32:34 2003 as: nmap -P0 -O -oN
/tmp/nmap1 192.168.0.140
Interesting ports on  (192.168.0.140):

(The 1538 ports scanned but not shown below are in state: closed)
Port       State       Service
22/tcp     open        ssh
111/tcp    open        sunrpc
515/tcp    open        printer
6000/tcp   open        X11

No exact OS matches for host (If you know what OS is running on it, see
http://www.insecure.org/cgi-bin/nmap-submit.cgi).

TCP/IP fingerprint:

SInfo(V=2.54BETA22%P=i386-redhat-linux-
gnu%D=11/16%Time=3FB7ECF7%O=22%C=1)
TSeq(Class=RI%gcd=1%SI=1EF3E9%IPID=C%TS=100HZ)
TSeq(Class=RI%gcd=1%SI=1EF73C%IPID=C%TS=100HZ)
TSeq(Class=RI%gcd=1%SI=1EF768%IPID=C%TS=100HZ)
T1(Resp=Y%DF=Y%W=16A0%ACK=S++%Flags=AS%Ops=MNNTNW)
T2(Resp=N)
T3(Resp=Y%DF=Y%W=16A0%ACK=S++%Flags=AS%Ops=MNNTNW)
T4(Resp=Y%DF=Y%W=0%ACK=O%Flags=R%Ops=)
T5(Resp=Y%DF=Y%W=0%ACK=S++%Flags=AR%Ops=)
T6(Resp=Y%DF=Y%W=0%ACK=O%Flags=R%Ops=)
T7(Resp=Y%DF=Y%W=0%ACK=S++%Flags=AR%Ops=)
PU(Resp=Y%DF=N%TOS=C0%IPLEN=164%RIPTL=148%RID=E%RIPCK=E%UCK=E
%ULEN=134%DAT=E)



Uptime 0.018 days (since Sun Nov 16 16:06:02 2003)

# Nmap run completed at Sun Nov 16 16:32:39 2003 -- 1 IP address (1 host up) scanned
in 5 seconds
```

4.8 Automated Security

There is a tool that can assist in the process of securing the system by automating some of the steps. The tool is bastille-linux. It is available for download at[1] www.bastille-linux.org. Download the program and install it with the **rpm –ivh** command. Other packages needed for using Bastille-linux are **perl-Tk** and **perl-Curses**. Only one of these packages are needed depending on wether you prefer to use the GUI or CLI. Please consult the Bastille-linux webpage to determine what packages are required for your installation.

```
#rpm -ivh Bastille-2.1.1-1.0.i386.rpm
#rpm -ivh perl-Tk-800.022-11.i386.rpm
```

Once the proper packages are installed issue the **bastille** command to bring up the program interface.

```
#bastille
```

The bastille-linux program will ask if there are certain changes you would like to make to improve the security of your system. The following list shows the changes that were made to this system. It does not show all the options available only the options that were selected.

- Would you like to set more restrictive permissions on administration utilities?

  Yes. This option changes permissions on some common administration utilities so that they are not readable or excutable users other than root. These utilities (which include linuxconf, fsck, ifconfig, runlevel, and portmap) are ones that most users should never need.

The next section of questions deals with the issue of SUID permissions. A command that runs as SUID to root operates with the permissions of the root user. If vulnerabilities are discovered in a program that runs SUID an attacker may be able to exploit the vulnerability and obtain a root shell. In general it is better to have as few programs as possible that operate SUID.

- Would you like to disable SUID status for dump and restore?

[1] to download the bastille-linux tool go to www.bastille-linux.org

Dump and restore are used for backing up file systems and restoring them from disk. If used by an attacker, they could be used to construct an alternate file system in place. Further, anyone who backs up the machine and restores from backup should have authorization and special access granted by the system administrator.

- Would you like to disable SUID status for printing utilities?

Yes. This system is not going to perform any printing. Printing utilities have ahistory of security vulnerabilities. By disabling SUID status this will disallow local, non-root users from initiating, modifying and canceling print requests.

- Would you like to disable SUID for mount and umount commands?

Yes

- Would you like to disable SUID status for ping command?

Yes (SUID executes with root permission)

- Would you like to disable SUID status for at command?

Yes

- Would you like to disable SUID status for userntctl?

Yes

- Would you like to disable SUID status for traceroute?

Yes

- Would you like to enforce password aging?

Yes

- Would you like to restrict use of cron to admin accounts?

Yes

Root is the only user that needs access to cron on this system. Access to cron is restricted by setting up a /etc/cron.alllow file. If for some reason another user needs access to cron in the future the user's name can be add to the /etc/cron.allow file.

- Would you like to set a default umask?

Yes

The umask sets the default permissions of files that are created.

- What setting?

077

Here are the effects of different umask settings:

002 – Everyone can read your files and people in your group can alter them.
022 – Everyone can read your files, but no one can write to them.
027 – Only people in your group can read your files, no one can write to them.
077 – No one on the system can read or write your files

- Would you like to disable ctrl-alt-del rebooting?

Yes
Disabling ctrl-alt-del rebooting helps to prevent an attacker with access to the machine's keyboard from being able to reboot the machine. Obviously if the attacker can get to the keyboard he/she may have access to other ways of halting the system. This is just another layer in the security onion.

- Would you like to password protect single user mode?

Yes

Anyone who can physically interact with the boot loader can tell the boot loader to bring up the system in "single user mode", where he/she is given root privileges and everyone else is locked out of the system.

- Would you like to ensure that the telnet service does not run on this host?

Yes

Telnet is unencrypted communication. This means that all data transferred, including passwords, can be monitored by anyone else on your network. (even on a switched network) Telnet sessions are also susceptible to session hijacking. This is when another person use a tool such as Hunt or Ettercap to take over your telnet session and enter commands on your behalf and with your permissions.

- Would you like to ensure that inetd's ftp service does not run on this host?

Yes

FTP is also a plaintext protocol and therefore is susceptible to many of the issues that were listed for telnet. Firewalling ftp effectively can be difficult because ftp requires many ports to stay open. Also the FTP daemon has had along history of security vulnerability. FTP is easily replaced with scp and sftp.

- Would you like to display "Authorized Use" messages at log-in time?

Yes

These messages perform the function of warning unauthorized users that there may be consequences if they are caught on the system with out permission or if they have permission yet are committing acts that are contrary to the site's acceptable use policy. According to GIAC's bulletin these banners may also make it easier to prosecute intruders.[1]

- Would you like to disable the gcc compiler?

Yes

A common technique used by system crackers is to gain access to the system via a regular user account and use that access to compile exploits against the system. Disabling the gcc compiler may slow down or altogether prevent some attacks. This system is to be a dedicated file server therefore there is no reason for anyone to compile programs on it.

- Limit resource usage by changing /etc/security/limits.conf ?

Yes

By restricting the resources available available to users this action may prevent a Denial of Service attack. This will limit the amount of resources available to users. This setting will also set the number of allowed core files to zero. A core file is created by a program that unexpectedly crashes. The purpose of a core file is for use as a

[1] http://ciac.llnl.gov/ciac/bulletins/j-043.shtml

debugging tool. Core files can contain sensitive data such as passwords. These files can be analyzed to help pinpoint ways to crack the system. Generating core files purposely can be used as a form of Denial of Service attack to fill up the file system. The settings in this section can be edited again later if needed by editing the /etc/security/limits.conf file.

- Would you like to add additional logging?

Yes

Bastille adds some additional logging for the system and also gives the option to send messages to a site logging server. Logging is good.

- Would you like to disable apmd?

Yes

This is power management for laptops and not needed for a server.

- Would you like to disable sendmail daemon mode?

Yes

This system is not to be a mail server. To run sendmail in daemon mode means that sendmail will constantly be listening on port 25 for mail messages. The system is only required to sendmail out such as log files. There is no reason for this file server to receive any mail.

- Would you like sendmail to process the queue via cron?

Yes

Also Bastille adds the option to process the mail queue every so often to send mail that may not have sent right away. Cron is used to process the sendmail queue every 15 minutes. This makes the outbound mail more dependable.

Now Bastille asks if you are finished with the selections. If you are finished, answer yes. You will then be presented with three options:

Exit without saving
Save Configuration
Go Back and Change Configuration

(By choosing Save Configuration the changes are NOT automatically applied)

Bastille now prompts you again with three options:

Exit Without Changing System
Go Back and Change Configuration
Apply Configuration to System

A helpful feature of the bastille-linux program is, that for every change that it would like to make to the system, there is an explanation of what features the change will improve or what problems it may cause. The program is almost like having a set of linux security flash cards. Another advantage of using the bastille-linux program is the amount of time saved by not having to hand, edit all the changes. Bastille-Linux also includes a firewall configuration script; thus, this was not used because the firewall for this system has already been configured. The Bastille-Linux program performs well in securing this system; however, there are still some settings that will have to be changed by hand.

The next changes to make are to the */etc/sysctl.conf* file. This file tells the kernel how to handle different network requests. The following changes to the */etc/sysctl.conf* file can help increase the security.

net.ipv4.ip_forward = 0 (Prevents forwarding of IP packets)
net.ipv4.conf.all.accept_source_route = 0 (Disables IP source routing)
net.ipv4.tcp_max_syn_backlog = 4096 (Helps prevent SYN flooding)
net.ipv4.tcp_syncookies = 1 (Helps protect the system from SYN flooding)
net.ipv4.conf.all.accept_redirects = 0 (Disables acceptance of ICMP redirect messages)
net.ipv4.conf.all.send_redirects = 0 (Disables sending of ICMP redirect messages)
net.ipv4.icmp_echo_ignore_all = 1 (Do not respond to ICMP ping requests)
net.ipv4.icmp_echo_ignore_broadcasts = 1 (Do not respond to broadcasts request)
net.ipv4.conf.all.log_martians = 1 (Logs packets that have source address and no known route)
net.ipv4.conf.all.rp_filter = 1 (Enables IP spoofing protection by turning on source route verification)

Once these changes have been made, be sure the */etc/sysctl.conf* file is owned by root and has permissions set to 600. Also to make the change take effect either a reboot is needed or the network service needs to be restarted.

```
#chown root:root /etc/sysctl.conf
#chmod 0600 /etc/sysctl.conf

#/etc/rc.d/init.d/network start

or

#shutdown –r now
```

These tweaks to the kernel will help protect the system from various attacks such as SYN flooding and IP spoofing. These attacks *should* not go beyond the

firewall, however in order to maintain a layered defense strategy there should be a lot of deterents in the way of would be hackers in order to make the system less appealing.

While on the subject of networking, the */etc/hosts.deny* and */etc/hosts.allow* files need to be configured. The */etc/hosts.allow* and */etc/hosts.deny* files are used by *tcpd* the daemon responsible for tcp-wrappers. *Tcpd* uses the */etc/hosts.allow* and */etc/hosts.deny* file to determine if someone has the authority to connect to any services offered by *inetd*, such as ftp, ssh the rservices, etc. If by some chance the firewall is not configured properly or the firewall was disabled and someone forgot to re-initialize it, this will provide another means by which to deny access to the system.

First configure */etc/hosts.deny* to deny everybody.

```
#echo ALL: ALL >/etc/hosts.deny
```

Next the */etc/hosts.allow* file needs to be configured for the systems that should have access to the new server. The firewall has already been configured to allow certain hosts to connect to certain services. Since samba and ssh are the only two services anyone can connect to externally this makes our */etc/hosts.allow* short. The */etc/hosts.allow* and */etc/hosts.deny* files do not have any effect on the hosts that require access to the Samba server, that access is controlled via the firewall and */etc/samba/smb.conf* file that is covered later in this document. The */etc/hosts.allow* file only needs to be configured to allow ssh from specific hosts. There only two hosts that need to connect to the server via ssh. So this is how the entry in the file should look.

```
sshd: 192.168.40.2 192.168.40.3
```

4.9 Configuring Services

The next action is to shut down unnecessary services. As part of an in depth security strategy to harden the system, services that are not being used or that are not necessary should be turned off. The reason to turn these services off is, it is hard to crack a service that is not running. Red Hat has a nice utility, for monitoring services and run control levels, *chkconfig* can be used to show what run control level certain services are to be activated. The same command with proper arguments can be used to configure a service not start when the system boots.

Running the following command will show all the services that can be configured to start or stop and at which run-level.

```
chkconfig --list

rwhod           0:off   1:off   2:off   3:off   4:off   5:off   6:off
atd             0:off   1:off   2:off   3:on    4:on    5:on    6:off
keytable                0:off   1:on    2:on    3:on    4:on    5:on    6:off
syslog          0:off   1:off   2:on    3:on    4:on    5:on    6:off
gpm             0:off   1:off   2:on    3:on    4:on    5:on    6:off
sendmail                0:off   1:off   2:off   3:off   4:off   5:off   6:off
kudzu           0:off   1:off   2:off   3:on    4:on    5:on    6:off
netfs           0:off   1:off   2:off   3:on    4:on    5:on    6:off
network                 0:off   1:off   2:on    3:on    4:on    5:on    6:off
random                  0:off   1:off   2:on    3:on    4:on    5:on    6:off
rawdevices              0:off   1:off   2:off   3:on    4:on    5:on    6:off
apmd            0:off   1:off   2:on    3:on    4:on    5:on    6:off
ipchains                0:off   1:off   2:off   3:off   4:off   5:off   6:off
iptables        0:off   1:off   2:on    3:on    4:on    5:on    6:off
smb             0:off   1:off   2:off   3:off   4:off   5:off   6:off
crond           0:off   1:off   2:on    3:on    4:on    5:on    6:off
anacron                 0:off   1:off   2:on    3:on    4:on    5:on    6:off
xinetd          0:off   1:off   2:off   3:on    4:on    5:on    6:off
lpd             0:off   1:off   2:off   3:off   4:off   5:off   6:off
xfs             0:off   1:off   2:off   3:off   4:off   5:off   6:off
ntpd            0:off   1:off   2:off   3:off   4:off   5:off   6:off
portmap                 0:off   1:off   2:off   3:off   4:off   5:off   6:off
autofs          0:off   1:off   2:off   3:on    4:on    5:on    6:off
nfs             0:off   1:off   2:off   3:off   4:off   5:off   6:off
nfslock         0:off   1:off   2:off   3:off   4:off   5:off   6:off
nscd            0:off   1:off   2:off   3:off   4:off   5:off   6:off
identd          0:off   1:off   2:off   3:off   4:off   5:off   6:off
radvd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
snmpd                   0:off   1:off   2:off   3:off   4:off   5:off   6:off
snmptrapd               0:off   1:off   2:off   3:off   4:off   5:off   6:off
rhnsd           0:off   1:off   2:off   3:on    4:on    5:on    6:off
isdn            0:off   1:off   2:off   3:off   4:off   5:off   6:off
sshd            0:off   1:off   2:on    3:on    4:on    5:on    6:off
vncserver               0:off   1:off   2:off   3:off   4:off   5:off   6:off
ip6tables               0:off   1:off   2:on    3:on    4:on    5:on    6:off
rarpd           0:off   1:off   2:off   3:off   4:off   5:off   6:off
xinetd based services:
   chargen-udp:         off
   chargen:             off
   daytime-udp:         off
   daytime:             off
   echo-udp:            off
   echo:                off
   time-udp:            off
   time:                off
   swat:                on
   sgi_fam:             on
   rsync:               off
   services:            off
```

```
servers:                off
```

This view may be a little cumbersome so the following command can be used to show only the services that are actually configured to start.

```
chkconfig –list | grep –e “\(:.on\)”

atd             0:off  1:off  2:off  3:on  4:on  5:on  6:off
keytable        0:off  1:on   2:on   3:on  4:on  5:on  6:off
syslog          0:off  1:off  2:on   3:on  4:on  5:on  6:off
gpm             0:off  1:off  2:on   3:on  4:on  5:on  6:off
kudzu           0:off  1:off  2:off  3:on  4:on  5:on  6:off
netfs           0:off  1:off  2:off  3:on  4:on  5:on  6:off
random          0:off  1:off  2:on   3:on  4:on  5:on  6:off

rawdevices      0:off  1:off  2:off  3:on  4:on  5:on  6:off
network         0:off  1:off  2:on:  3:on  4:on  5:on  6:off
apmd            0:off  1:off  2:on   3:on  4:on  5:on  6:off
iptables        0:off  1:off  2:on   3:on  4:on  5:on  6:off
crond           0:off  1:off  2:on   3:on  4:on  5:on  6:off
anacron         0:off  1:off  2:on   3:on  4:on  5:on  6:off
xinetd          0:off  1:off  2:off  3:on  4:on  5:on  6:off
autofs          0:off  1:off  2:off  3:on  4:on  5:on  6:off
rhnsd           0:off  1:off  2:off  3:on  4:on  5:on  6:off
sshd            0:off  1:off  2:on   3:on  4:on  5:on  6:off
ip6tables       0:off  1:off  2:on   3:on  4:on  5:on  6:off
        swat:           on
        sgi_fam:        on
```

This gives a better view of only the services that are configured to start at certain run levels. The *chkconfig* command can be used to configure these services not to start during boot up. For example if *autofs* is not needed for the system to function properly, it should be turned off:

```
#/etc/rc.d/init.d/autofs stop
#chkconfig –level 345 autofs off
```

Which services you allow to run will vary depending on your environment. For this system, the services that were shut off were *portmap, lpd, isdn, nfslock* and *autofs*. If the service isn’t running it is not available to try and penetrate. Also important to note is that while *chkconfig* is a handy tool to use, it is not available on all Unix and Linux distributions. It is good to know that on any system there are corresponding start up scripts for services on each machine and that by changing the names of these startup scripts a service can be disabled from starting at boot time. If the *autofs* daemon should start at boot time on a Solaris machine, the following commands would cause the program not to start when the system boots.

```
#cd /etc/rc2.d
#mv /etc/rc2.d/s74autofs /etc/rc2.d/NOS74autofs
```

Now when rebooting, no need to worry about *autofs* starting. After stopping the services that are not needed, you may want to consider removing the package responsible for that service entirely from the system. To accomplish this simply run the following command.

```
#rpm –e packagename
```

So far the network settings and services have been configured for more secure settings. The goal of every security administrator is to make the host as impenetrable as possible. The amount of security in the system is designed to deter or frustrate the would-be bad guy. Now the system needs to be equipped with “No Trespassing” signs. If for some reason there were legal proceedings and the attacker had encountered a banner that read,” Welcome to our system, make yourself at home!” it would be hard to do a very convincing job explaining to the jury that only an authorized user was allowed to access the machine. So to battle this issue there must be clear statements on all login banners, “that if you are not authorized to be here, you shouldn’t be!” The banner that is commonly seen in many documents sufficiently covers most situations. If you are unsure about the contents of your banner, you should check with those responsible for making legal decisions in your company.

The following files need to be modified:

```
/etc/motd
/etc/issue
/etc/issue.net
```

Each of these should be modified to read:

```
This system is for authorized uses only! All access may be monitored and/or logged !
```

There is no question what this means.

4.10 Passwords

The next thing to consider is passwords. [1]Bastille-Linux made a change to */etc/login.defs* that set the maximum days for a password to a more reasonable setting of 180 days. This is much better that the default setting, however the settings that Bastille-Linux made do not coincide with the site’s password policy.

[1] Bastille Linux, see www.Bastille-linux.org

The customer's site password policy requires passwords to be changed every 90 days. The policy also states that once a password has been changed the user must wait a minimum of 15 days to change it again. There is also an eight character minimum requirement. So the changes to */etc/login.defs* would look like this:

```
PASS_MAX_DAYS     90
PASS_MIN_DAYS     15
PASS_MIN_LEN      8
PASS_WARN_AGE     15
```

Keep in mind that a password policy is important. It is also important to educate the users as to what constitutes a good password. There are many very good resources on the internet about choosing good passwords.[12]

The next file to look at for hardening the system is */etc/passwd.* Red Hat includes some unnecessary accounts in this file. These accounts should be removed. Before proceeding to remove the accounts make a backup copy of the files. It is a good habit to make backup copies of any important configuration files so that if a mistake is made the backed up copy of the file can be put back in place. This can be accomplished in the following manner:

```
#for file in /etc/{passwd,shadow,group}
>do
>cp –p $file $file.bk
>done
```

These files contain sensitive data so the permissions should only allow root to access the files.

```
# for file in /etc/{passwd.bk,shadow.bk,group.bk}
>do
>chown root:root $file
>chmod 400 $file
>done
```

Now that there is a backup, the unnecessary user accounts can be safely removed.

```
#for user in uucp operator gopher games
>do
>userdel $user
>done
```

[1] http://www.securitystats.com/tools/password.php
[2] http://www.mit.edu/afs/sipb/project/doc/passwords/passwords.html

Now do the same for unnecessary group accounts.

```
#for group in gopher games dip uucp
>do
>groupdel $group
>done
```

Some of the user account's shell should be changed to */dev/null*. With this invalid shell the account will be unable to log in to the system.

```
#for user in bin daemon adm ftp sync lp mail news nobody
>do
>usermod –L –s /dev/null
>done
```

After running these commands we can verify that things are still working properly and will allow the appropriate persons to log in. We use the next two commands to do this.

```
#pwck
#grpck
```

4.11 Sshd configuration

Ssh will be used to connect to the server for maintenance and management. Ssh is away of connecting to a remote host via an encrypted connection. This encryption aids in the prevention of someone reading or changing the information that is being exchanged between two systems. Although ssh is a great tool there are some entries in the configuration file that may not be desirable for running ssh securely. The */etc/ssh/sshd_config* file is responsible for how the daemon handles incoming ssh requests. First look at the file */etc/sshd_config*.

```
# cat /etc/sshd_conifg

# $OpenBSD: sshd_config,v 1.48 2002/02/19 02:50:59 deraadt Exp $

# This is the sshd server system-wide configuration file. See sshd(8) # for more
information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with # OpenSSH is to
specify options with their default value where # possible, but leave them commented.
Uncommented options change a # default value.

#Port 22
```

```
#Protocol 2,1
#ListenAddress 0.0.0.0
#ListenAddress ::

# HostKey for protocol version 1
#HostKey /etc/ssh/ssh_host_key
# HostKeys for protocol version 2
#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_dsa_key

# Lifetime and size of ephemeral version 1 server key
#KeyRegenerationInterval 3600
#ServerKeyBits 768

# Logging
#obsoletes QuietMode and FascistLogging
#SyslogFacility AUTH
SyslogFacility AUTHPRIV
#LogLevel INFO

# Authentication:

#LoginGraceTime 600
#PermitRootLogin yes
#StrictModes yes

#RSAAuthentication yes
#PubkeyAuthentication yes #AuthorizedKeysFile .ssh/authorized_keys

# rhosts authentication should not be used
#RhostsAuthentication no # Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes # For this to work you will also need host keys in
/etc/ssh/ssh_known_hosts
#RhostsRSAAuthentication no
# similar for protocol version 2
#HostbasedAuthentication no # Change to yes if you don't trust ~/.ssh/known_hosts for #
RhostsRSAAuthentication and HostbasedAuthentication #IgnoreUserKnownHosts no

# To disable tunneled clear text passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no

# Change to no to disable s/key passwords #ChallengeResponseAuthentication yes

# Kerberos options # KerberosAuthentication automatically enabled if keyfile exists
#KerberosAuthentication yes
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes

# AFSTokenPassing automatically enabled if k_hasafs() is true #AFSTokenPassing yes

# Kerberos TGT Passing only works with the AFS kaserver #KerberosTgtPassing no

# Set this to 'yes' to enable PAM keyboard-interactive authentication # Warning: enabling
this may bypass the setting of 'PasswordAuthentication' #PAMAuthenticationViaKbdInt
yes
```

```
#X11Forwarding no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PrintMotd yes
#PrintLastLog yes
#KeepAlive yes
#UseLogin no

#MaxStartups 10
# no default banner path
#Banner /some/path
#VerifyReverseMapping no

# override default of no subsystems Subsystem sftp /usr/libexec/openssh/sftp-server
```

Note the line that says:

```
#Protocol 2,1
```

This setting allows the ssh server to revert to Version 1 of the ssh protocol. SshVersion1 had some security vulnerabilities. It is best to only use Version 2. So add an entry that would only allow version 2.

```
#Protocol 2,1
Protocol 2
```

The next setting to pay attention to is:

```
#PermitRootLogin yes
```

It is a better practice not to allow root logins remotely. A user should be forced to log in as themselves and su to root. This practice maintains a better audit trail and promotes accountability among the system administrators. Add an entry that looks like the following.

```
#PermitRootLogin yes
PermitRootLogin no
```

Also it is good practice to display a banner message before the user logs in. Note that the default behavior is to show */etc/motd* after successful login, however the warning banner should appear before logging in. To do this make the following change.

```
#Banner /some/path
Banner /etc/issue.net
```

Be sure to enter the banner message that your site uses into the */etc/issue.net* file. After editing this file you must restart ssh in order to make the changes take affect. Either restart sshd or reboot the machine.

```
#/etc/init.d/sshd stop
#/etc/init.d/sshd start
```

4.12 X11

X11 will be used on this server for various management tasks. Since ssh has been configured with ability to tunnel X11 connections there is no reason to have the Xserver listening for network connections.

By running the following *netstat* command it is apparent that X11 is listening for connections. It listens on port 6000.

```
#netstat –l | grep tcp

tcp       0      0 *:x11                   *:*                     LISTEN
tcp       0      0 *:ssh                   *:*                     LISTEN
```

The firewall on this system should stop any unwanted x11 connections, however in the event the firewall fails or is turned off for some reason, it would be best to not have the service running. To stop X11 from listening for connections the *gdm.conf* file needs to be edited.

```
#cd /etc/X11/gdm
```

Open the *gdm.conf* file for editing. Find the following entry:

```
[servers]
0=/usr/bin/X11/X
```

edit the entry so that it reads:

```
0=/usr/bin/X11/X –nolisten tcp
```

Restart the system.
Now the netstat command reveals X11 is no longer listening for connections.

```
tcp       0      0 *:ssh                   *:*                     LISTEN
```

4.13 System Logs

The next subject is system logging.

Log files play an important role in system administration as well as security. Having a record of certain events will help put together a piece of the puzzle when trying to solve a problem that "suddenly" occurred. If one can get in the habit of watching log files on a daily basis, then the sudden problems can be forecast ahead of time and avoided. An example would be messages regarding problems with a hard disk. Often, these messages will appear in log files before complete failure. Log files are also an important item to have if a system is compromised. Examining log entries shows exactly how and when a compromise took place. Sometimes if a hacker can manage to get a rootkit on the system it may have built in software that will delete log entries remove evidence of their hack. However if there is a remote syslog server that the system is sending log entries to, then there is a good chance that the information that may have been deleted from the other system will appear on the remote system. If you selected to let Bastille increase the logging abilities of the system you can skip this section. However if Bastille was not used to modify the *syslog.conf* file please follow the next steps. Below is a new *syslog.conf* file that is recommended by the [1]Securing Linux Step by Step guide.

```
#syslog.conf.new

##########
# Section 1:  For all system (servers and workstations)
##########
# Log all ifno or higher messages, except facilities that use their own log
*.info;authpriv,auth,mail,cron,kern,local7.none     /var/log/messages

# authpriv is intended for messages related to authorizations
# (e.g.failed login attempts).authis deprecated, but included
# in case some older programs still use it.
Authpriv,auth.*

# Send mail messages to a separate file.
Mail.*                                              /var/log/cron

# Send kernel messages to a separate file.  Note that this will
# include messages generated by iptables about blocked network traffice.
Kern.*                                              /var/log/kernel


# Send boot messages to a separate file
local7.*                                            /var/log/boot.log
```

[1] Securing Linux, Step by Step Guide, see Bibliography

```
# Send emergency  messages of any type to all logged in users
*.emerg                                                 *


##########
#If you have a remote logging host, uncomment the lines corresponding to
# the types of messages you want to forward to it.  Replace the string
# loghost with the IP address of your central logging server.
##########

# kern.*                                        @loghost
# authpriv,auth.*                               @loghost
# mail.*                                        @loghost
```

Replace the default *syslog.conf* file with the new one. The new file will improve on the default logging parameters setup by the operating system.

```
#mv /etc/syslog.conf /etc/syslog.conf.orig
#cp syslog.conf.new /etc/syslog.conf
```

After changing the file we need to restart the *syslogd* daemon.

```
#/etc/init.d/syslogd restart
```

We should then use *chkconfig* to ensure that *syslogd* starts on system reboot.

```
#chkconfig –level 2345 syslog on
```

Once the new file is in place, then *logrotate.conf* and *logwatch* can be configured. These utilities will help to manage the system's log files. It does not do any good to have logging in place if the logs are not being monitored. One of the biggest reasons that system administrators neglect actively monitoring the logs is the overwhelming amount of information that is contained in them. Look at *logrotate.conf* first. The following is the default file for reference.

Logrotate.conf:

```
# see "man logrotate" for details
# rotate log files weekly
weekly

#keep 4 weeks worth of backlogs
rotate 4

#create new (empty) log files after rotating old ones
create

#uncomment this if you want your log files compressed
#compress
```

```
#RPM pachages drop log rotation information into this directory
include /etc/logrotate.d

#no packages own lastlog or wtmp – we'll rotate them here
/var/log/wtmp {
        monthly
        create 0664 root utmp
        rotate 1
}
```

The *logrotate.conf* file is almost self-explanatory. There are some changes that could be made help increase security of the system. Remember the logs can help forecast hardware failure as well as give some view of intrusion. To give us a better audit we may want to change the **weekly** parameter to monthly and **rotate** to twelve. This will give us more information if there is ever a need to audit the logs or a hacker goes un-noticed for more than a month. We can also choose to compress the logs to save space.

Red Hat also includes logwatch. Logwatch is an application that will parse through the logs and mail the person responsible a report based on the settings in the */etc/log.d/logwatch.conf* file. There are different log levels to choose from as well as ways to filter the sort of information that is required in the report. Logwatch should have been selected to be installed from the beginning. If you are unsure run the following command to verify that logwatch is installed.

```
#rpm –q logwatch
logwatch-2.6-1
```

If the package name and version is returned then logwatch is installed on the system. If it is not installed the RPM can be found at rhn.redhat.com. Download and install it. The configuration file is very straight forward.

4.14 Stack Protection

Now that the system logging is in order, there is another utility that needs to be installed that will also contribute to the security of the operating system. The program is called *libsafe.* With *libsafe* there is nothing to configure, it just needs to be downloaded and installed. Use the **rpm -ivh** option to install it. The reason to install a package like *libsafe* is to protect the system from stack attacks. It is not uncommon for the Unix/Linux operating systems to be attacked due to a vulnerability called a buffer overflow. A buffer overflow occurs when a memory stack is forced to fill to capacity then a piece of code is executed in the stack, causing some side effect potentially leading to a root shell. Once a root shell is obtained the intruder has control. These buffer overflows typically result from poor programming practices or even weakness in certain programming languages. While much more attention is given to this problem these days, it is still a problem. It is possible that your site uses a legacy piece of software that you were not aware was vulnerable. One way to deal with these attacks is to have the operating system deny buffer overflow attacks. *Libsafe* is a program that can accomplish this task. *Libsafe* monitors unsafe system calls. If an attack is detected *libsafe* will send the entire process group a **SIGKILL** signal and then it will log the attempt in */var/log/secure.* The *libsafe* software can be obtained from the following address.

[1]www.research.avayalabs.com/project/libsafe

4.15 System Integrity

The next item of importance is file system integrity checking tools. The log files may very well help to understand who and how someone penetrated the system but there needs to be a way to tell what was done when they were on the system. A product called Tripwire is used for the purpose of reporting what has changed on the system. Tripwire uses a set of rules that compares file sizes, access times, inode numbers, and ownership and some other configurable parameters to those settings recorded in the tripwire database when it was initialized. This is important to understand, because if a legitimate program is installed and the database is not updated tripwire will detect the change and report it as something that should not have happened.

[1] for more information on Libsafe see www.resarch.avayalabs.com/project/libsafe

The first step to configuring tripwire is to run the *twinstall.sh* script. Although the tripwire rpm is installed on the machine the *twinstall.sh* script must be run in order to setup tripwire for the system. The script will prompt you to set passphrases it will also create a set of cryptographic keys to protect the configuration and policy files.

```
#[root@testhost tripwire]# ./twinstall.sh

---------------------------------------------
The Tripwire site and local passphrases are used to
sign a variety of files, such as the configuration,
policy, and database files.

Passphrases should be at least 8 characters in length
and contain both letters and numbers.

See the Tripwire manual for more information.

---------------------------------------------
Creating key files...

(When selecting a passphrase, keep in mind that good passphrases typically
have upper and lower case letters, digits and punctuation marks, and are
at least 8 characters in length.)

Enter the site keyfile passphrase:
Verify the site keyfile passphrase:
Generating key (this may take several minutes)...Key generation complete.

(When selecting a passphrase, keep in mind that good passphrases typically
have upper and lower case letters, digits and punctuation marks, and are
at least 8 characters in length.)

Enter the local keyfile passphrase:
Verify the local keyfile passphrase:
Generating key (this may take several minutes)...Key generation complete.

---------------------------------------------
Signing configuration file...
Please enter your site passphrase:
Wrote configuration file: /etc/tripwire/tw.cfg

A clear-text version of the Tripwire configuration file
/etc/tripwire/twcfg.txt
has been preserved for your inspection.  It is recommended
that you delete this file manually after you have examined it.
---------------------------------------------
Signing policy file...
Please enter your site passphrase:
Wrote policy file: /etc/tripwire/tw.pol

A clear-text version of the Tripwire policy file
/etc/tripwire/twpol.txt
has been preserved for your inspection.  This implements
a minimal policy, intended only to test essential
```

```
Tripwire functionality.  You should edit the policy file
to describe your system, and then use twadmin to generate
a new signed copy of the Tripwire policy.
```

The policy file, tw.pol, which dictates how tripwire checks your system, is created. The configuration file, twcfg.txt, is also created.

Next initialize the tripwire database. This database is what serves as the baseline for integrity checks.

```
#/usr/sbin/tripwire -–init
```

After initializing the database run an integrity check using the following command.

```
#/sbin/tripwire –check –interactive
```

This command gives a long report of how the system looks to tripwire. At the end of the report is a list of directories that tripwire is checking that do not exist. Use this report to compare it to the entries in the */etc/tripwire/twpol.txt* file It is common to comment out these entries in the */etc/tripwire/twpol.txt* file.

```
Tripwire(R) 2.3.0 Integrity Check Report

Report generated by:        root
Report created on:          Mon 24 Nov 2003 09:09:07 PM EST
Database last updated on:   Never

================================================================================
Report Summary:
================================================================================

Host name:                  testhost.secure.edu
Host IP address:            192.168.0.139
Host ID:                    None
Policy file used:           /etc/tripwire/tw.pol
Configuration file used:    /etc/tripwire/tw.cfg
Database file used:         /var/lib/tripwire/testhost.secure.edu.twd
Command line used:          /usr/sbin/tripwire --check -I

================================================================================
Rule Summary:
================================================================================

--------------------------------------------------------------------------------
 Section: Unix File System
--------------------------------------------------------------------------------
```

```
  Rule Name                    Severity Level    Added    Removed  Modified
  ---------                    --------------    -----    -------  --------
  Invariant Directories          66              0        0        0
  Temporary directories          33              0        0        0
* Tripwire Data Files            100             1        0        0
  Critical devices               100             0        0        0
  User binaries                  66              0        0        0
  Tripwire Binaries              100             0        0        0
  Critical configuration files   100             0        0        0
  Libraries                      66              0        0        0
  Operating System Utilities     100             0        0        0
  Critical system boot files     100             0        0        0
  File System and Disk Administraton Programs
                                 100             0        0        0
  Kernel Administration Programs 100             0        0        0
  Networking Programs            100             0        0        0
  System Administration Programs 100             0        0        0
  Hardware and Device Control Programs
                                 100             0        0        0
  System Information Programs    100             0        0        0
  Application Information Programs
                                 100             0        0        0
  Shell Related Programs         100             0        0        0
  Critical Utility Sym-Links     100             0        0        0
  Shell Binaries                 100             0        0        0
  System boot changes            100             0        0        0
  OS executables and libraries   100             0        0        0
  Security Control               100             0        0        0
  Login Scripts                  100             0        0        0
  Root config files              100             0        0        0

Total objects scanned:  20014
Total violations found:  1

===============================================================================
============
Object Summary:
===============================================================================
============
-------------------------------------------------------------------------------
# Section: Unix File System
-------------------------------------------------------------------------------
-------------------------------------------------------------------------------
Rule Name: Tripwire Data Files (/var/lib/tripwire)
Severity Level: 100
-------------------------------------------------------------------------------

Added:
"/var/lib/tripwire/testhost.secure.edu.twd"

===============================================================================
============
Error Report:
===============================================================================
============
```

```
---------------------------------------------------------------------------
 Section: Unix File System
---------------------------------------------------------------------------

1.  File system error.
    Filename: /usr/sbin/fixrmtab
    No such file or directory
2.  File system error.
    Filename: /usr/bin/vimtutor
    No such file or directory
3.  File system error.
    Filename: /sbin/accton
    No such file or directory
4.  File system error.
    Filename: /sbin/busybox
    No such file or directory
5.  File system error.
    Filename: /sbin/busybox.anaconda
    No such file or directory
6.  File system error.
    Filename: /sbin/ftl_check
    No such file or directory
7.  File system error.
    Filename: /sbin/ftl_format
    No such file or directory
```

After editing the *twpol.txt* file, run the next command to re-read the *twpol.txt* and create the *tw.pol* file which is encrypted.

```
#/usr/sbin/twadmin –create-profile –S site.key /etc/tripwire/twpol.txt
```

After creating the new */etc/tripwire/tw.pol* file the database needs to be re-initialized. The recommended way to do this is to remove the existing database from /var/lib/tripwire.

```
#rm /var/lib/tripwire/testhost.secure.edu.twd
```

Then re-initialize the database with the following command.

```
#/usr/sbin/tripwire --init
```

Next, check the report again to make sure things looks acceptable. Once tripwire is reporting the way you want it to, save a copy of the database to a form of read only media and store it offline. This way if the machine is ever compromised you still have a pristine copy of the database. When the system installed tripwire it created */etc/cron.daily/tripwire-check*. This script will run tripwire daily, and it will mail the report to the person responsible for the system. RedHat's website has documentation about the configuration of Tripwire if you need more information.[1]

[1] http://www.redhat.com/docs/manuals/linux/RHL-7.2-Manual/ref-guide/s1-tripwire-install.html

In Summary, the process of configuring RAID on the Dell Power Edge 2600 server was documented. The process of installing Red Hat ES on the system was explained and the process of installing the operating system was explained. Securing the system, using tools such as Bastille-Linux and also hand editing certain files and setting for greater security has been covered. Applications that aide in the system security such as libsafe and tripwire are necessary security implementation tools. The system has services shut off that are not necessary and the firewall is configured. Other forms of access control are configured. The next item that needs to be configured is Samba. Samba is the purpose of this system.

5. Samba Configuration

This section includes Samba configuration for seamless file service.

5.1 Server Function

To meet the requirements of the customer Samba has to be configured on the server. Samba is an application that allows file serving in an environment of mixed platforms. Samba allows a Windows machine to share files with a Unix or Linux machine. Samba can also be configured to perform many other tasks. For the purpose of fulfilling the customer's needs Samba only needs to be configured as a file server. Fortunately in our particular situation we are only requiring this server to be file server. At the time of this writing the Samba team has released a new version of Samba. The new version may correct some problems and include some enhancements. It is generally good practice to give a product time in the field before installing it on a production server. The time in the field would allow time to work out any possible kinks in the new release. That being said the version Samba for the customer's server will be the one included in the distribution. Allowing Red Hat Network to update the system will apply any patches that may be necessary for the Samba package to be current.

5.2 SWAT

The first objective is to be sure that SWAT is enabled. SWAT is the forms based that editor that functions in your web browser. SWAT stands for Samba Web Administration Tool and runs as a daemon in xinetd Look at */etc/services* to see if an entry referring to SWAT is there.

```
#grep swat /etc/services
```

If there is not a reference to SWAT add the following entry to the end of the file.

```
swat      901/tcp
```

There is also a file in the */etc/xinetd.d* directory for SWAT.

```
#description : swat is the Samba Web Administration Tool, which
#          allows an administrator to configure Samba using a web
#          browser interface, with the URL http://localhost:901
```

```
service swat.
{
        socket_type             = stream
        wait                    = no
        protocol                =tcp
        only_from               = localhost
        user                    = root
        log_on_failure          += USERID
        server                  = /usr/local/samba/bin/swat
        port                    = 901
        disable                 = no
}
```

Be sure that there is an entry that reads:

```
disable = no
```

For some default installations of Samba this setting reads:

```
disable = yes
```

This will stop the system from being able to use SWAT. After editing this file the xinetd daemon needs to be restarted.

```
# /bin/kill –HUP –a xinetd
```

SWAT is now enabled and ready for use. To access SWAT, open your web browser and connect to *http://localhost:901* and login as root.

The next step is to configure the smb.conf file. If installed during the operating system installation this file can be found in */etc/samba*. If Samba was installed by compiling the source code the location of the *smb.conf* file may differ. The *smb.conf* file can be edited two different ways. The first way is to open the file in vi or another text editor and edit it manually line for line. The other option is to edit the file using SWAT. SWAT is the better solution. SWAT features help documents for each entry that explain why or why not make an entry. This is an extremely helpful tool and removes any guess work from the process of configuring the file. There are several other features available with samba that will not be discussed in this paper. There is a book that contains answers to most questions that may come up. The book also has instructions for the many possible uses of Samba. The book," Using Samba" is available to purchase. The second edition of the book was just released. The book can also be viewed online at the Samba website, [1]www.samba.org. The rest of this section will present the *smb.conf* file used for the customer's site and try to explain each item.

[1] for more information see www.samba.org

5.3 Setup smb.conf

The File

```
[global]
```

The global section refers to the server-wide settings

```
workgroup = Testgroup
```

This represents the NetBIOS name group that our systems belong to

```
netbios name = Samserv
```

The netbios name

```
server string = Not available
```

We have use the string not available, this is more or less security by obscurity, however it has been recommended by some other documents to help decrease the amount of information being advertised

```
encrypt passwords = yes
```

Encrypted passwords definitely needs to be used. This helps us to protect against someone grabbing plain text passwords off the wire by using some type of sniffer utility such as tcpdump. It should be noted that certain flavors of Microsft do not support encrypted passwords. At a site were security is a real concern, these flavors of Windows should no longer be in use.

```
unix password sync = no
```

Update encrypted synchronizes the unix passwd database with the smbpasswd database. The passwords are locked on the unix accounts, set this to no.

```
restrict anonymous = yes
```

This entry forces anonymous connections to be denied and will always require a username and password

```
log level = 2
```

We have chosen a log level of 2, according to the documentation a log level designation higher than 2 or 3 would mainly be beneficial to someone working on the development of Samba's source code, unless you are a programmer.

```
log file = /var/log/samba.log.%m
```

This entry defines where the log file can be found, also note that the %m variable substitution is used to append the client's NetBIOS name to the log file. This will be helpful when trying to isolate which client is having trouble.

```
max log size = 1000
```

This sets the log size not to exceed 1MB. If a log file exceeds the 1MB limit it is moved to *logfile.*old, if *logfile*.old already exist then it is overwritten..

```
debug timestamp = yes
```

adds timestamp to log entries

```
read only = no
```

We are not designating all shares as read only.

```
# Network configuration settings
hosts.allow = 192.168.0.100 192.168.0.101 192.168.0.102 192.168.0.3
hosts.deny = 10.0.0.
```

For the hosts.allow and hosts.deny entries there are some potential problems. If you do not understand how Samba parses the rules the desired outcome may not be achieved. If both options are used, **hosts.allow** and **hosts.deny**, then whatever is listed in hosts.deny can not appear in hosts.allow in any form. If a host is explicitly allowed, but the entire subnet that the host is on is denied access, deny wins and the host that you specifically allow is still denied.[1]Also it may be helpful to note that the **hosts.allow** and **hosts.deny** rules could be applied on a per share basis instead of defining them globally. Beware that the global definitions overrides the per share definition. The global definition is sufficient for this server.

```
Invalid users = root bin daemon adm sync shutdown pwrchute halt mail news uucp
operator
```

Invalid users option explicitly states who may not access the system. As the book says,"people use Samba not daemons."

```
[homes]
```

[1] See Using Samba, O'Reilly Books, see Bibliography

Homes is a special section. It is used as somewhat of a dynamic way to create shares for your users, this could have potentially undesirable results. In other words it may be possible to accidentally creat a share for root or uucp, this is why in the global section we specifically list invalid users.

```
browsable = no
```

Browsable for this section means that the “home” share will not show up in the browse list, however the *username* share will appear in the list

```
        writeable = yes
 [med-data]
        browasble = yes
        writeable = yes
        path = /data/med
        valid users = user1 user2 user3 user 5
 [med-image]
        browsable = yes
        writeable = yes
        path = /data/med-image
        valid users = user1 user3 user6
 [medware]
        browsable = yes
        writeable = yes
        path = /data/medware
         valid users = user2 user3 user5
```

From med-data to medware are the basic share configurations for the data shares. The shares are shared directories under the main data directory. They are writable because it would not do any good to have a file server that will not allow data to be written to it. Notice for each share there are **valid user** listings. There are certain people in the department who do not require the responsibility of access to certain shares. SWAT helps to maintain proper syntax. If SWAT was not used to create the file, *testparm* should be used on the file. It will go through the file and alert you to any misspellings or syntax issues.

NOTE: It will not catch poor configurations that could lead to compromise, ie: you shared /etc and made writable and said guest = yes.

Now that Samba is configured the smb daemon must be restarted. This can be done in the SWAT interface by choosing the status tab and then selecting the restart button.

Adding Users:

The first step is adding users to the system. This system serves no other purpose than a file server, so there is no reason for any user to login the server. Therefore the user accounts can be created with an invalid shell that will prevent them from

logging in directly. The following command will create the new use account with the invalid shell.

```
#useradd –u 505 –d /home/user1 –s /dev/null –m user1
```

Then the password for the user can be locked using the next command

```
passwd –l user1
```

To add a new user to Samba use the following command:

```
smbpasswd  –a user1
New SMB password:
Retype new SMB password:
Added user user1
```

If you receive a message that says the encrypted database does not exist. The program will oblige your request by creating it for you.

Restart the smb server.

It should be possible now to map a drive from a windows workstation as long as the following criteria are met:

- The host is allowed to access Samba through the firewall
- The host is listed as an allowed host in the smb.conf file
- The user trying to connect is a valid Samba user
- The user trying to connect has permission to access the share

On a windows 2000 workstation:

Right click on My Network Places

Select Map Network Drive

Select a drive letter to use

Then enter the share to connect to in this format \\server\share

Then Samba prompts for username and password:

If there are no incorrect spellings in the Samba configuration file, connection should be successful.

Although this document's focus is not on the security of the Windows clients that are connecting to the server, it is an important topic. Be sure the systems that will be connecting to the server are updated with the latest patches and virus definitions. Another step that is taken for the customer's site is to install Outpost Firewall on the clients. This is a great product and is available to download at [1]www.agnitum.com.

Once installed go to the Outpost Firewall-configuration.cfg and select the Options tab.

Then select system.

Put a check in the box beside Allow NetBios communication.

Click on Settings

You should see the NetBios Address screen

Highlight the IP address that is in the box

Click Remove. Now add the IP address of the Samba server

Click OK

Click Apply

All the windows clients at the customer's site are running this firewall. The key to a secure site is security in layers. It is just as important that the clients access the server be as secure as the server itself.

[1] Out post firewall download from www.agnitum.com

5.4 Samba Security

One security issue is that of enumeration. This is when someone uses a tool to obtain information from a source. In the case of Samba, one could use the *smbclient* tool.[1]

```
# smbclient –L sambaserver –I 192.168.42.2 –U ‘’
added interface ip=192.168.1.2 bcast=192.168.1.255 nmask=255.255.255.0
Password:[enter]
```

After pressing enter the system responds with a listing of information such as share names, workgroup name and version of Samba.

Another issue is the possibility of someone sniffing the wire for Samba password hashes, using a tools such an ettercap.

```
# ettercap –C –N –m
```

With this tool someone would try to capture the password hashes to a file and then employ some form of password cracking tool to obtain the unencrypted version of your password.

Another very important security weakness to consider is poor configuration. The only defense against this is taking your time. Some steps to take in order to defend against such an attack are to:

1) Change the Samba version string in the *smb.conf* file
2) Make use of the hosts.allow options in the *smb.conf* file
3) Be sure that our firewall is only allowing traffic from desired hosts
4) Also there are Intrusion Prevention Systems in place as well as certain blocked ports at the border router.

5.5 Samba Configuration Summary

This chapter covered the step-by-step details to configure Samba for the customer’s purpose. It pointed out that one should consult the documentation for more advanced uses of Samba. The chapter also showed how to add a new user account and restarted the smb daemons. Various possible threats were considered and possible solutions were shown that will help thwart attacks.

[1] Hack Notes, Linux and Unix Security Portable Reference, Nitesh Dhanjani, see Bibliography

6. Backup

This section includes the installation of Legato Networker for backing up data.

Now that the system is operational and configured to meet the customer's needs, the backup software needs to be installed and a full backup should be taken of the system. The backup up software of choice for this system is a product by Legato called Networker. There are many varieties of backup software available, both free and commercial. Legato was chosen because many servers on the customer's site are already running software by Legato. The decision to use the same software site wide helps in terms of system administration. The system administrator can spend more time maintaining the systems instead of spending time learning new software. A brief explanation of the installation of the Legato software is included; however, the configuration of Networker would require a paper unto itself. Please use the flavor of backup that meets your customers need and perform a full system backup before turning the system over for production.

This is a list of the Legato Networker Software and space requirements:

Software and Documentation Files NetWorker Server Space	Default Location
NetWorker Administrator program and NetWorker Client program files 17 MB	*/usr/bin*
• NetWorker daemon and utility command files 72 MB	*/usr/sbin*
• Device drivers 6 MB	*/usr/lib/nsr*
Online client file and server indexes; media database 4 MB	*/nsr*
License Manager files 2 MB	*/usr/sbin*
22 KB	*/usr/nsr/lic/res*
21 KB	*/nsr/lic/res*
NetWorker man pages 2 MB	*/usr/share/man*
PDF files varies	optional

The Networker server software requires that ksh is installed on the system. If this package was not selected during the operating system installation go to http://rhn.redhat.com and download the pdksh package. Pdksh is the public domain korn shell, a clone of the korn shell. This will meet the requirement for the Server software. The Legato software can be obtained by downloading it or it will be on cdrom. This installation is from the cdrom. It may also be worth noting that Legato uses the rpm utility for package installation. Put the disk in the drive.

```
#mount /mnt/cdrom
#cd /mnt/cdrom
#ls
#cd /
```

Also it is important to understand that the software must be installed in a certain order. The client must be installed first, then the device driver, storage node and then the server. Use the following commands to install each package.

```
#rpm –ivh lgtoclnt-7.0-1.i686.rpm
#rpm –ivh lgtodrvr-7.0-1.i686.rpm
#rpm –ivh lgtonode-7.0-1.i686.rpm
#rpm –ivh lgtoserv-7.0-1.i686.rpm
#rpm –ivh lgtolicm-7.0-1.i686.rpm
```

This is the license manager and can be installed any time after the client software.

```
#rpm –ivh  lgtoman-7.0-1.i686.rpm
```

This is the Legato man page package and it is optional.

Once the packages are installed verify that the Networker daemons are running.

```
# ps –ax | grep nsr
```

If the output does not list nsrexec and nsrd as current processes issue the following command to start the Networker daemons.

```
#/etc/init.d/networker start
```

To configure the server to perform a backup and schedule backups enter the following command:

```
#nwadmin &
```

You will be presented with a GUI screen that will allow you to configure Networker to perform to your wishes. The configuration of Networker requires its own documentation and Legato has made available the Legato Networker Administrator's Guide for Unix. You should become familiar with this guide before configuring the backup. Also you will need to decide what kind of backup schedule works best for your site. This customers' site does a full backup on the first Friday of every month and incremental backups rest of the time that record daily changes. This configuration allows for file recovery as well as saving on the cost of media. It also saves on the amount of time spent changing tapes. Your site may require full backups every day. Check your site's policy before you make that decision. (For those who may not be familiar with Legato, it is very easy to obtain software for 30 day evaluations.[1])

[1] http://www/legato.com

7. Maintenance

A secure system must be maintained in order to remain secure.

System security is not something that you can put in place and expect to stay secure. To keep a system secure requires continuous effort. A system administrator must be diligent in order to try to keep up with new vulnerabilities. It is a good idea to sign up for security notification lists. CERT offers an email list that offers very timely announcements of vulnerabilities when they are reported.[1] There are also several good websites to refer to for up to date information that may affect your systems, these include www.sans.org and www.securityfocus.com. When the ES operating system was purchased Red Hat signed up the customer with the red hat network, which is a good resource for errata specifically regarding this system.[2] A subscription to the Red Hat Network also includes an email list that Red Hat uses to notify the customer of potential threats and also of software updates.

System security and maintenance is continuous. It is imperative that patches be kept up to date. A patch is a piece of code that is applied to an already existing piece of code that may either enhance the code or fix a bug. As of this writing, a vulnerability was discovered in OpenSSH. The version of OpenSSH that contained the vulnerability shipped with Red Hat ES 2.1 (and many other operating systems), this is why one of the first steps in the installation process was to run **up2date** and have Red Hat patch the system with updates.

An important part of system maintenance is checking to make sure that nothing has been tampered with by some one without permissions. It is important to keep a check on the logs and watch for suspicious behavior such as unauthorized persons trying to connect to your server. The tripwire database that was stored on the cdrom should also be run occaisionally. Remember, if you do not keep your database on media that is read only that no one else has access to, it may be possible for an intruder to manipulate your database or Trojan the tripwire executable.

[1] http://www.cert.org/contact_cert/certmaillist.html
[2] http://rhn.redhat.com

It is also important to keep an eye on logfiles. Look for attempted connections from unknown sources. Also pay special attention to any hardware error messages.

Running a scan such as Nessus, once a month, can go along way in letting you know if anything has changed about the system. It may find a port that is not supposed to be open. This would give you a clue that something suspicious is going on. Ports don't just open up for no reason. For those administrators that may not have the time to perform system scans on a consistent basis, there are companies that offer scanning services for a small fee.[1]These services will setup automated scans and the output will arrive in you email at a frequency that you decide.

Another important aspect of ongoing maintenance is the integrity of the tape backups. The data is the reason for the system's existence. Without the data the system is just a hunk of metal and silicon. At least once a month it is a good idea to pull some of your backup tapes and restore the data to an area that you can inspect it to be sure that it actually got recorded by tape drive and also to be sure it did not become corrupted.

Again in order to maintain the system you must be proactive. Anticipate that vulnerabilities made known to the public will target you. Apply the proper patches. It is not uncommon for system administrators to think that they have time before their machine would be a target. This was made clear by the problems the blaster[2] worm caused for Windows machines. Microsoft issued a warning about the vulnerability many weeks before the blaster worm caused so many problems. Many people did nothing to patch their systems and when blaster was in full force on the internet many people suffered down time. Even administrators who had patched their machines suffered slow connectivity as a result of all the infected machines not been patched.

The use of cron can be a valuable asset for ongoing system maintainance. Cron can be used to execute scripts that are put into place to automate some the system administration. Cron will also send you an email to notify you that the job was completed. For a busy system administrator cron may be a good way to ensure the system stays updated with current system patches. Keep in mind that in a perfect world the system administrator would install new patches on a test machine before deploying them in a production environment. This may not be feasible in every environment so using cron to execute a script daily that will update the system may be a good solution. One way to help ensure that certain updates do not overwrite important configuration files is to carefully use RedHat Network to carefully choose packages that will not automatically update for the system's profile. The kernel is a package that you should choose not to update

[1] for information about internet vulnerability scanning service email info@dowhiletrue .com
[2] http://www.cert.org/advisories/CA-2003-20.html

automatically. Kernel updates require rebooting and if the new kernel is damaged the system would need to be booted into single user mode to make the change to boot into the old kernel. After carefully choosing packages a script could be created that would update the system nightly.

The following is an example of a script that could be used to update the system nightly.

```
#cat /root/update_daily.sh
/usr/bin/up2date –u
/usr/bin/up2date –p
```

The cron entry would look like the follwing:

```
0 21 * * * /root/update_daily.sh
```

Using cron will ensure that the system stays up to date with all the packages other than the one's selected not to be updated automatically. Cron never forgets and cron never gets "too busy to get around to it."

By implementing the practices above you may be able thwart many attacks, as well as, maintaining the system's availability. As a system administrator you want to keep the walls of your defense high enough to make your system less desirable attackers. There is also the responsibility of maintaining data integrity. By implementing the suggestions in this paper you may enjoy administering the system instead of re-installing the system.

8. Checking the Configuration and Verifying System Security

This section explains how to verify the changes that have been made to the system.

System integrity can be tested to find out what the system looks like to hackers. The use of certain hacker's tools needs to be employed to validate the security of the system. The next step of the document is penetration testing. The goal is to discover any vulnerability before someone else.

8.1 Penetration Testing and Vulnerability Scanning

Once the system has been installed and backed up, it is important to look at the host from the perspective of potential hackers. It is important to validate the security that has been implemented. It would be unfortunate to find out 2 days after the system has been put into production, that it had a wide open security breech. It is best search for the holes before the box is in production.

Scans of the system are made with nmap. Nmap scans the hosts for open ports and reports back. Nmap can be downloaded at www.insecure.org. For the next series of tests the firewall must be disabled. (Please do NOT stop your firewall if the system is connected to the internet. Only do this on a private network.) To disable the firewall use the following command:

```
#/etc/rc.d/init.d/iptables stop
```

Now perform the nmap scan

```
# nmap (V. 3.00) scan initiated Sun Nov 23 19:48:57 2003 as: nmap -P0 -O -oN
/tmp/nmap_final 192.168.0.139
Interesting ports on  (192.168.0.139):

The 1598 ports scanned but not shown below are in state: closed

Port       State       Service
22/tcp     open        ssh
139/tcp    open        netbios-ssn
901/tcp    open        samba-swat
```

```
Remote operating system guess: Linux Kernel 2.4.0 - 2.5.20
Uptime 0.534 days (since Sun Nov 23 06:59:39 2003)

# Nmap run completed at Sun Nov 23 19:49:03 2003 -- 1 IP address (1 host up) scanned
in 6 seconds
```

Next is the result of the nmap scan with the firewall active:

```
Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
All 1601 scanned ports on  (192.168.0.139) are: filtered

Nmap run completed -- 1 IP address (1 host up) scanned in 174 seconds
```

These results show that the firewall clearly makes a difference in the amount of security it adds to the machine. This is not a statement that a firewall is the panacea of security. The point is that the the firewall plays a critical role when it comes to the layered security approach.

Another scanning tool to employ is nbtscan. NBTscan is used to scan IP networks for NetBIOS name information. NBTscan will report back IPaddress, NetBIOS computer name, logged in user name and MAC address. This will help judge how the Samba server looks to the rest of the world. NBTscan can be downloaded from: www.inetcat.org/software/nbtscan.html

Here is the output of NBTscan without the firewall running:

```
Doing NBT name scan for addresses from 192.168.0.139


NetBIOS Name Table for Host 192.168.0.139:

Incomplete packet, 227 bytes long.
Name            Service         Type
----------------------------------------
SAMSERV      # <00>           UNIQUE
SAMSERV      # <03>           UNIQUE
SAMSERV      # <20>           UNIQUE
##__MSBROWSE__## <01>            GROUP
TMP          # <00>           GROUP
TMP          # <1d>           UNIQUE
TMP          # <1e>           GROUP

Adapter address: 00-00-00-00-00-00
----------------------------------------
```

Next are the results of NBTscan with the firewall up:

```
Doing NBT name scan for addresses from 192.168.0.139
```

By using port scanners and vulnerability scanners you can ascertain certain facts about the way your system may appear to others on the internet. There are many more flavors of scanners that could be run, however for our purposes these should suffice.

8.2 Don't just scan the box

Although employing the use of port scanners is good practice and will reveal information. It is a good idea to try to connect from an un-authorized host to the services that are running. The smbclient command can be used to try to enumerate Samba shares for the purpose of trying to mount them. Here is the output of the following command with the firewall down:

```
#smbclient –L smbserver –I 192.168.0.139 ''

added interface ip=192.168.0.89 bcast=192.168.0.255 nmask=255.255.255.0
Anonymous login successful
Domain=[TMP] OS=[Unix] Server=[Samba 2.2.7-security-rollup-fix]
tree connect failed: NT_STATUS_WRONG_PASSWORD
```

The command is not actually able to login to the Samba server. It also did not enumerate the shares; however, it was able to obtain the workgroup/domain name and the version of Samba that is running. This is not a terrible risk, though some may see the need o remove the version announcement.

Here is the output of the same command with the firewall in place:

```
added interface ip=192.168.0.89 bcast=192.168.0.255 nmask=255.255.255.0
timeout connecting to 192.168.0.139:139
Error connecting to 192.168.0.139 (Operation already in progress)
Connection to smbserver failed
```

While trying to connect to the system, certain log files can be monitored in real time to observe how the new system responds to the un-authorized connections.

To monitor the log file in real time:
Open a terminal window and use the tail command with –f option
To monitor the /var/log/secure enter the following command

```
#tail –f /var/log/secure
```

For the first connection test try to ssh as root from a system that is allowed. This will allow us to observe if the changes we made in the */etc/ssh/sshd_config* file are working properly.

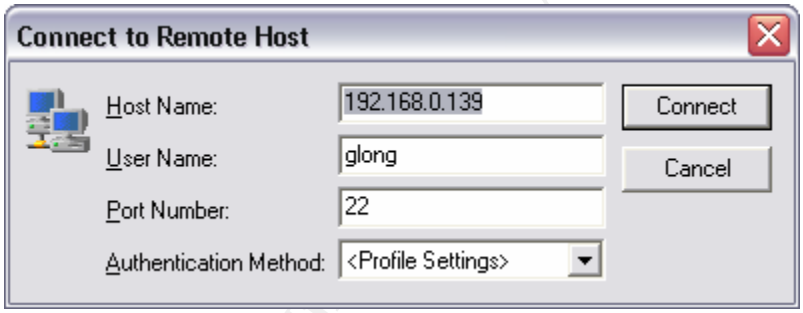
```
# ssh -l root 192.168.0.139
This system is for authorized uses only. All access may monitored and/or logged!
root@192.168.0.139's password:
Permission denied, please try again.
root@192.168.0.139's password:
Permission denied, please try again.
```

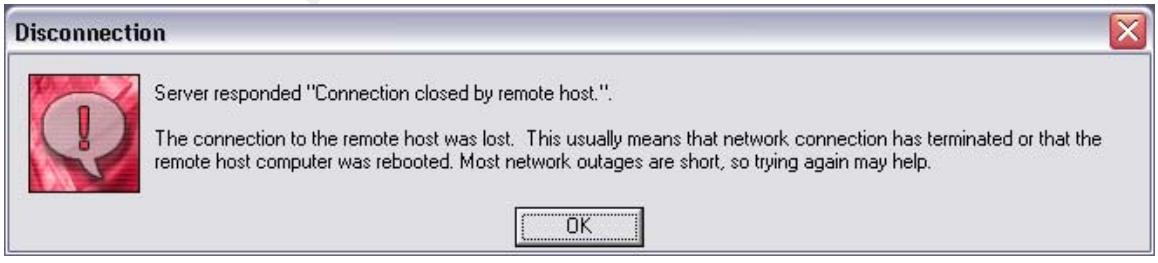
Here is the output observed in /var/log/secure file as we attempted to connect.

```
Nov 23 13:46:28 testhost sshd[2184]: ROOT LOGIN REFUSED FROM 192.168.0.124
Nov 23 13:46:28 testhost sshd[2184]: Failed password for root from 192.168.0.124 port
1014 ssh2
Nov 23 13:46:39 testhost sshd[2184]: ROOT LOGIN REFUSED FROM 192.168.0.124
Nov 23 13:46:39 testhost sshd[2184]: Failed password for root from 192.168.0.124 port
1014 ssh2
Nov 23 13:46:50 testhost sshd[2184]: Connection closed by 192.168.0.124
```

This confirms that sshd is behaving appropriately. We configured sshd to deny connecting as root and that is exactly what it does. Also notice that the warning banner appeared before authenticating. Try connecting from the same host as a legitimate user on the system and you will notice that you are able to connect with no trouble.

Next try to connect via ssh from a host that is not listed in */etc/hosts.allow*.

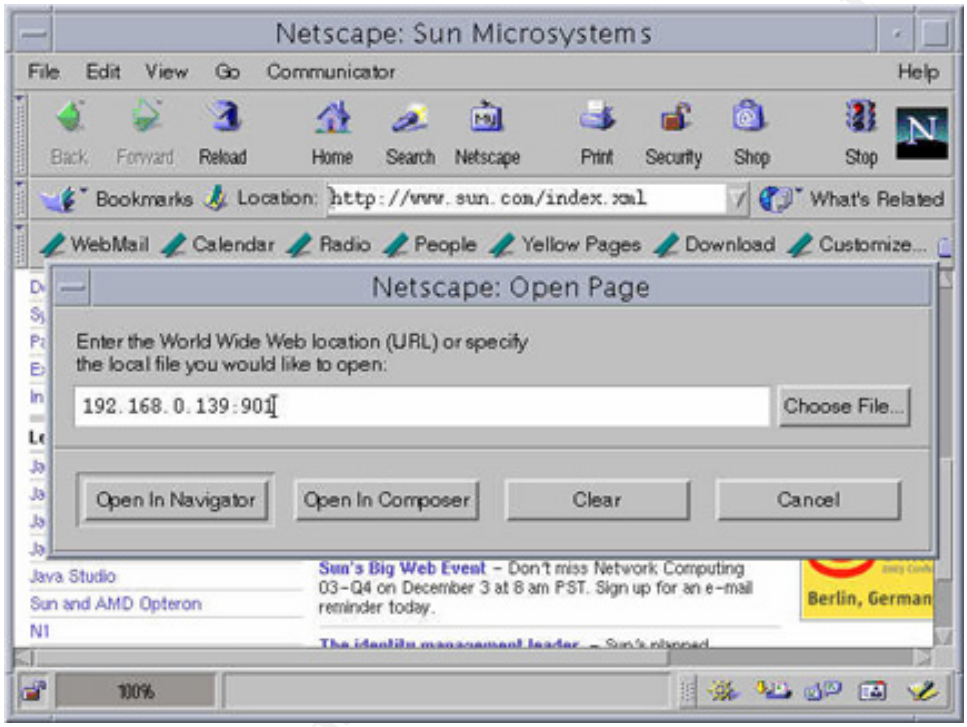


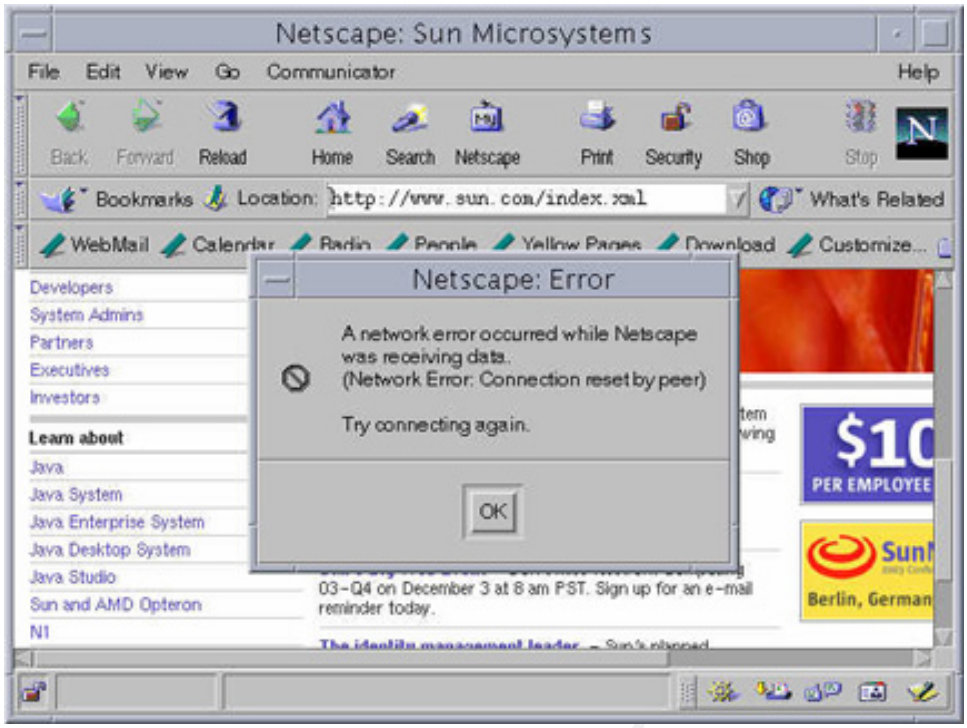


Even with a legitimate user name the connection is refused. Look at the output of the /var/log/secure file.

```
Nov 23 14:15:56 testhost sshd[2325]: refused connect from 192.168.0.1 (192.168.0.1)
Nov 23 14:17:17 testhost sshd[2326]: refused connect from 192.168.0.1 (192.168.0.1)
```

SWAT is running on port 901 and only the server should be able to access it. To verify this try to connect to SWAT from another host. Remember to leave */var/log/secure* open in order to observe the system's reaction to the attempted connection.

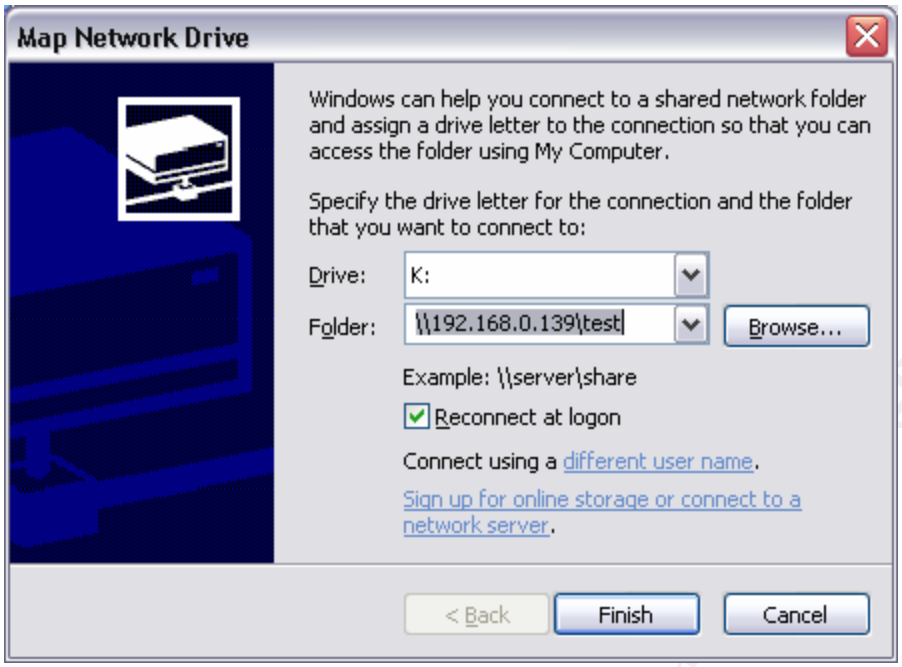


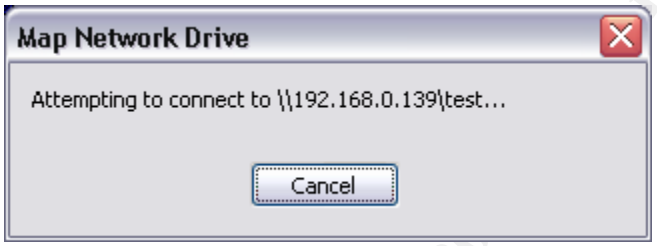


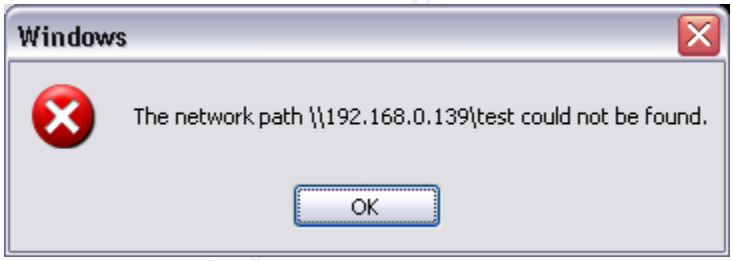
The connection is not successful. This is good news so far the system is responding as one would hope. See the logfile output.

```
results of SWAT connection from disallowed host
Nov 23 13:59:40 testhost xinetd[753]: START: swat pid=2243 from=192.168.0.124
Nov 23 13:59:40 testhost xinetd[2243]: FAIL: swat address from=192.168.0.124
```

Another good test to run is to try and connect to one of the Samba shares from an unauthorized host. Try to map a drive from a Windows host that is not authorized to connect to the shares.







The system is unable to connect to the share because it is not authorized in */etc/samba/smb.conf*. This time look at */var/log/samba/smbd.log* to see the system response.

```
[2003/11/23 14:29:37, 2] smbd/server.c:exit_server(461)
  Closing connections
[2003/11/23 14:29:37, 0] lib/access.c:check_access(331)
  Denied connection from  (192.168.0.1)
[2003/11/23 14:29:37, 1] smbd/process.c:process_smb(870)
  Connection denied from 192.168.0.1
[2003/11/23 14:29:37, 2] smbd/server.c:exit_server(461)
  Closing connection
```

NOTE: That was the last of the network access security test. Please re-enable the firewall now.

```
#/etc/rc.d/init.d/iptables start
```

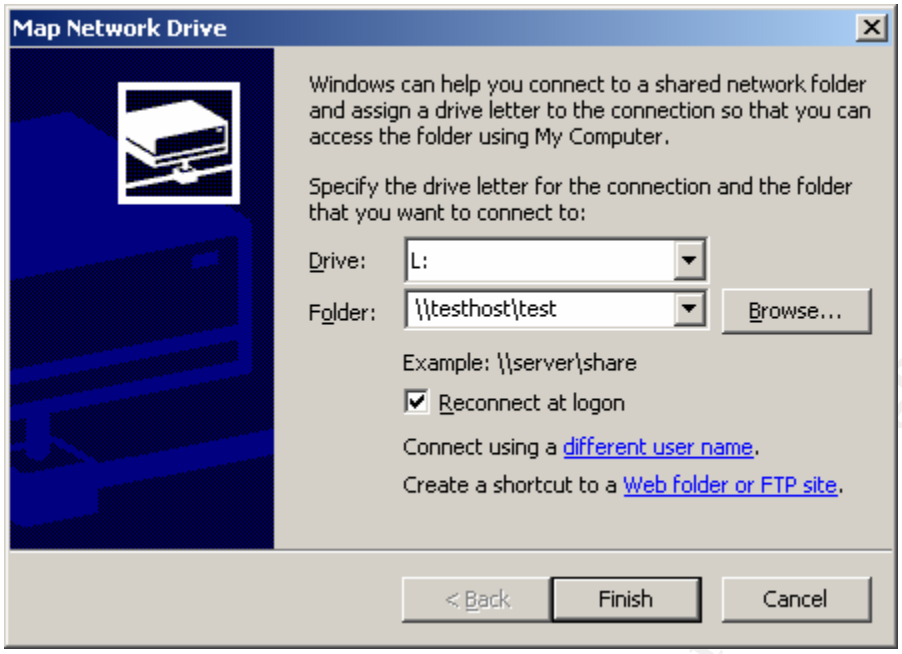
It is also good to test certain system features that were changed, to verify that the change is working. As an example log into the system as a user that is not root. Try to setup a cron job.

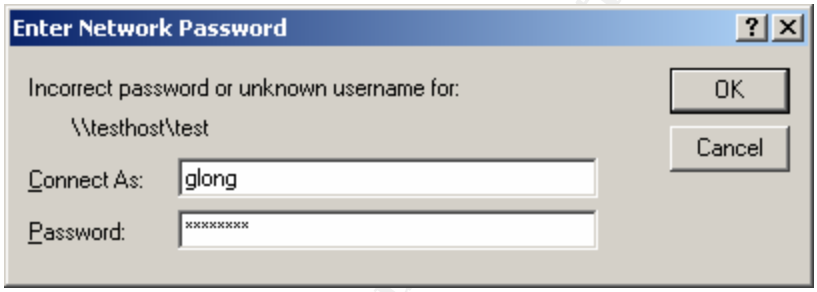
```
[glong@testhost glong]$ crontab -e
You (glong) are not allowed to use this program (crontab)
See crontab(1) for more information
```

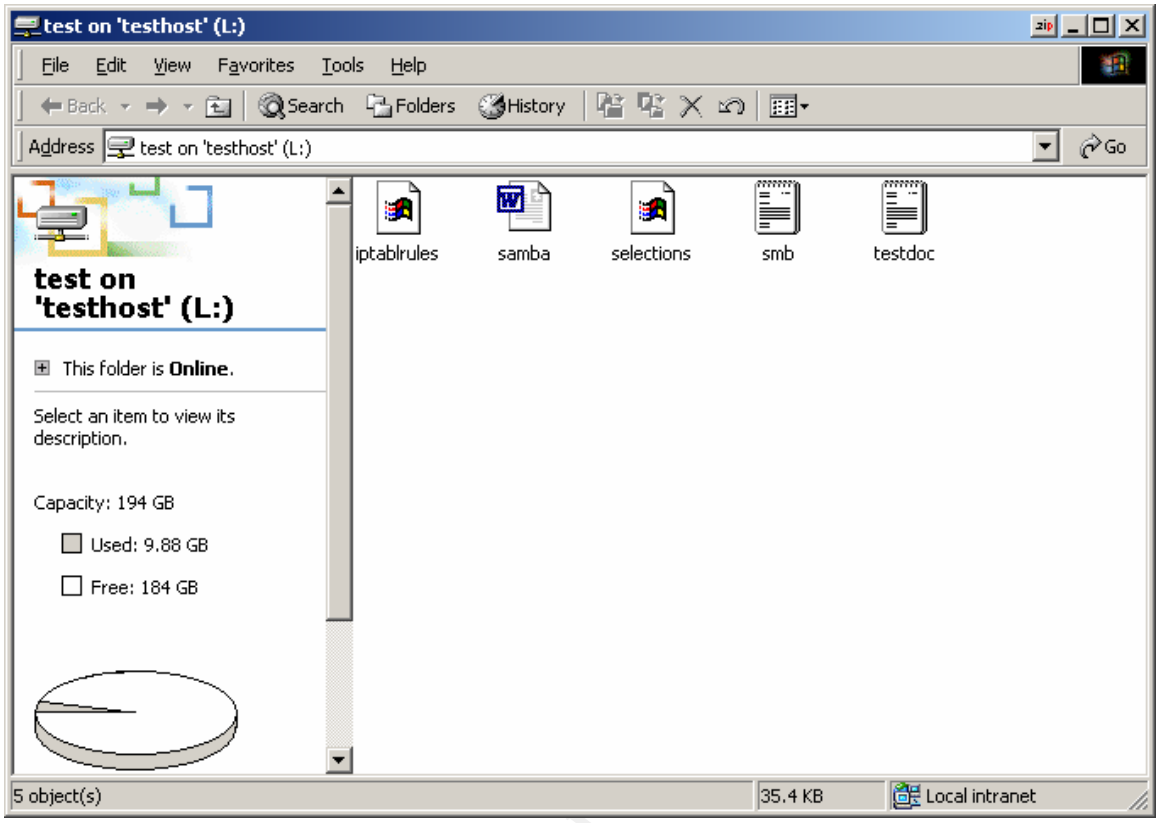
/etc/cron.allow was configured to only allow root the ability to use cron.

8.3 Security works, can the workers?

Now that the security has been verified sufficiently it is equally as important to verify that those who are authorized to use the server still have access. The security that has been put in place in worthless if the system is not usable by the people that are authorized to use it. To perform this test try accessing the server from a host that is authorized to access the server.







The correct user has access and is able to create files and transfer files to the server. The server implementation is successful.

These are examples of the steps that check the system's configuration and verify that the security measures that you put in place are in fact working. While no system connected the internet is safe, there are many precautions that can be taken to make a system a less desirable target. If you are ever faced with a security question that you are uncertain about the SANS reading room is a great place to search. After implementing all of the steps in this paper you should have a secure server. After running the various tests to verify the steps taken to secure the box, the system can now be placed into production with confidence.

.

9. Appendices

The Appendices contain documents that have been referenced throughout this paper.

9.1 IPTables

```
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
BADIP      all  --  anywhere             anywhere
BANNED     all  --  anywhere             anywhere
IN         !icmp --  anywhere            anywhere
IN_ICMP    icmp --  anywhere             anywhere
LOG_DROP   all  --  anywhere             anywhere

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
BADIP      all  --  anywhere             anywhere
BANNED     all  --  anywhere             anywhere
OUT        !icmp --  anywhere            anywhere
OUT_ICMP   icmp --  anywhere             anywhere
LOG_DROP   all  --  anywhere             anywhere

Chain BADIP (2 references)
target     prot opt source               destination
LOG_BADIP  all  --  samserv.secure.com   samserv.secure.com
LOG_BADIP  all  --  127.0.0.0/8          samserv.secure.com
LOG_BADIP  all  --  10.0.0.0/8           samserv.secure.com
LOG_BADIP  all  --  172.16.0.0/12        samserv.secure.com
LOG_BADIP  all  --  224.0.0.0/4          samserv.secure.com
LOG_BADIP !udp  --  224.0.0.0/4          samserv.secure.com
ACCEPT     udp  --  224.0.0.0/4          samserv.secure.com
LOG_BADIP  all  --  240.0.0.0/5          samserv.secure.com
LOG_BADIP  all  --  0.0.0.0/8            samserv.secure.com
LOG_BADIP  all  --  169.254.0.0/16       samserv.secure.com
LOG_BADIP  all  --  192.0.2.0/24         samserv.secure.com
LOG_BADIP  all  --  255.255.255.255      samserv.secure.com
LOG_BADIP  all  --  0.0.0.0              samserv.secure.com

Chain BANNED (2 references)
target     prot opt source               destination

Chain FLAGS (2 references)
```

```
target     prot opt source               destination
LOG_FLAGS  tcp  --  anywhere             anywhere            tcp
flags:FIN,SYN,RST,PSH,ACK,URG/NONE
LOG_FLAGS  tcp  --  anywhere             anywhere            tcp
flags:FIN,SYN,RST,PSH,ACK,URG/FIN,SYN,RST,PSH,ACK,URG
LOG_FLAGS  tcp  --  anywhere             anywhere            tcp flags:FIN,SYN/FIN,SYN
LOG_FLAGS  tcp  --  anywhere             anywhere            tcp flags:SYN,RST/SYN,RST
LOG_FLAGS  tcp  --  anywhere             anywhere            tcp flags:FIN,RST/FIN,RST
LOG_FLAGS  tcp  --  anywhere             anywhere            tcp flags:FIN,ACK/FIN
LOG_FLAGS  tcp  --  anywhere             anywhere            tcp flags:PSH,ACK/PSH
LOG_FLAGS  tcp  --  anywhere             anywhere            tcp flags:ACK,URG/URG

Chain FLOOD (1 references)
target     prot opt source               destination
RETURN     all  --  anywhere             anywhere            limit: avg 5/sec burst 10
LOG_FLOOD  all  --  anywhere             anywhere

Chain IN (1 references)
target     prot opt source               destination
LOG_DROP   all  --  anywhere             anywhere            state INVALID
FLOOD      tcp  --  anywhere             anywhere            tcp flags:SYN,RST,ACK/SYN
FLAGS      tcp  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere            state
RELATED,ESTABLISHED
LOG_DROP   all  --  samserv.secure.com   anywhere
ACCEPT     tcp  --  192.168.0.137        anywhere            tcp dpt:ssh state NEW
ACCEPT     tcp  --  192.168.0.125        anywhere            tcp dpt:ssh state NEW
ACCEPT     udp  --  192.168.0.232        anywhere            state NEW udp dpt:netbios-dgm
ACCEPT     tcp  --  192.168.0.232        anywhere            tcp dpt:netbios-ssn state NEW
ACCEPT     tcp  --  192.168.0.232        anywhere            tcp dpt:microsoft-ds state NEW
ACCEPT     udp  --  192.168.0.36         anywhere            state NEW udp dpt:netbios-dgm
ACCEPT     tcp  --  192.168.0.36         anywhere            tcp dpt:netbios-ssn state NEW
ACCEPT     tcp  --  192.168.0.36         anywhere            tcp dpt:microsoft-ds
state NEW

Chain IN_ICMP (1 references)
target     prot opt source               destination
ACCEPT     icmp --  192.168.0.0/24       samserv.secure.comicmp echo-request
ACCEPT     icmp --  192.168.0.0/24       samserv.secure.comicmp echo-reply
ACCEPT     icmp --  anywhere             anywhere            icmp destination-unreachable
ACCEPT     icmp --  anywhere             anywhere            icmp source-quench
ACCEPT     icmp --  anywhere             anywhere            icmp time-exceeded
ACCEPT     icmp --  anywhere             anywhere            icmp parameter-problem

Chain LOG_BADIP (12 references)
target     prot opt source               destination
LOG        all  --  anywhere             anywhere            limit: avg 1/sec burst 10 LOG level
error prefix `IPT BAD: '
DROP       all  --  anywhere             anywhere

Chain LOG_BAN (0 references)
target     prot opt source               destination
LOG        all  --  anywhere             anywhere            limit: avg 1/sec burst 10 LOG level
error prefix `IPT BANNED: '
DROP       all  --  anywhere             anywhere
```

```
Chain LOG_DROP (5 references)
target     prot opt source               destination
LOG        all  --  anywhere             anywhere            limit: avg 1/sec burst 10 LOG level
error prefix `IPT DROP: '
DROP       all  --  anywhere             anywhere

Chain LOG_FLAGS (8 references)
target     prot opt source               destination
LOG        all  --  anywhere             anywhere            limit: avg 1/sec burst 10 LOG level
error prefix `IPT FLAGS: '
DROP       all  --  anywhere             anywhere

Chain LOG_FLOOD (1 references)
target     prot opt source               destination
LOG        all  --  anywhere             anywhere            limit: avg 1/sec burst 10 LOG level
error prefix `IPT FLOOD: '
DROP       all  --  anywhere             anywhere

Chain OUT (1 references)
target     prot opt source               destination
FLAGS      tcp  --  anywhere             anywhere
LOG_DROP   all  -- !samserv.secure.com  anywhere
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:ssh state NEW
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:smtp state NEW
ACCEPT     udp  --  anywhere             anywhere            state NEW udp dpt:netbios-dgm
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:netbios-ssn state NEW
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:microsoft-ds state NEW

Chain OUT_ICMP (1 references)
target     prot opt source               destination
ACCEPT     icmp --  anywhere             192.168.0.0/24      icmp echo-request
ACCEPT     icmp --  anywhere             192.168.0.0/24      icmp echo-reply
ACCEPT     icmp --  anywhere             anywhere            icmp destination-unreachable
ACCEPT     icmp --  anywhere             anywhere            icmp source-quench
ACCEPT     icmp --  anywhere             anywhere            icmp fragmentation-needed
ACCEPT     icmp --  anywhere             anywhere            icmp parameter-problem
```

9.2 Selections: Example output from IPtables script

```
IPADDR=192.168.0.139
NETWORK=192.168.0.0/255.255.255.0
SSH=BOTH SSH_CLIENTS="192.168.0.137 192.168.0.125"
HTTP=OFF
HTTP_CLIENTS=
HTTPS=OFF
HTTPS_CLIENTS=
DNS=OFF
DNS_CLIENTS=
SMTP=OUT
SMTP_CLIENTS=
POP3=OFF
POP3_CLIENTS=
FTP=OFF
FTP_CLIENTS=
IMAP=OFF
IMAP_CLIENTS=
SMB=BOTH SMB_CLIENTS="192.168.0.232 192.168.0.36"
TRACEROUTE=OFF
TRACEROUTE_CLIENTS=
NTP=OFF
NTP_CLIENTS=
SYSLOG=OFF
SYSLOG_CLIENTS=
SYSLOGD_SOURCE_PORT=
SYSLOGD_DEST_PORT=
AUTH=OFF
AUTH_CLIENTS=
LPD=OFF
LPD_CLIENTS=
WHOIS=OFF
WHOIS_CLIENTS=
FINGER=OFF
FINGER_CLIENTS=
NNTP=OFF
NNTP_CLIENTS=
TELNET=OFF
TELNET_CLIENTS=
TFTP=OFF
TFTP_CLIENTS=
XSERVER=OFF
XSERVER_CLIENTS=
XDM=OFF
XDM_CLIENTS=
XFS=OFF
XFS_CLIENTS=
NFS=OFF
NFS_CLIENTS=
PORTMAP_PORT=111
NFS_PORT=2049
MOUNTD_PORT=32767
LOCKD_PORT=32768
NIS=OFF
NIS_CLIENTS=
PORTMAP_PORT=111
NIS_UDP_PORT=792
NIS_TCP_PORT=792
NESSUSD=OFF
NESSUSD_CLIENTS=
NESSUSD_PORT=1241
```

9.3 Bastille Syslog.conf

```
# Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.*                                                 /dev/console

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none        /var/log/messages

# The authpriv file has restricted access.
authpriv.*                                              /var/log/secure

# Log all the mail messages in one place.
mail.*                                                  /var/log/maillog


# Log cron stuff
cron.*                                                  /var/log/cron

# Everybody gets emergency messages
*.emerg                                                 *

# Save news errors of level crit and higher in a special file.
uucp,news.crit                                          /var/log/spooler

# Save boot messages also to boot.log
local7.*                                                /var/log/boot.log
############ BASTILLE ADDITIONS BELOW : ##################
# Log warning and errors to the new file /var/log/syslog
*.warn;*.err       /var/log/syslog

# Log all kernel messages to the new file /var/log/kernel
kern.*    /var/log/kernel

# Log all logins to /var/log/loginlog
auth.*;user.*;daemon.none        /var/log/loginlog

# Log additional data to the Alt-F7 and Alt-F8 screens (Pseudo TTY 7 and 8)

*.info;mail.none;authpriv.none    /dev/tty7
authpriv.*        /dev/tty7
*.warn;*.err       /dev/tty7
kern.*    /dev/tty7
mail.*    /dev/tty8

*.*        /dev/tty12
########## BASTILLE ADDITIONS CONCLUDED : ################
```

9.4 smb.conf

```
# Samba config file created using SWAT # from localhost.localdomain (127.0.0.1) # Date:
2003/10/07 11:55:15

# Global parameters
[global]
        workgroup = TMP
        netbios name = SAMSERV
        server string = Not Available
        encrypt passwords = Yes
        allow trusted domains = No
        min passwd length = 8
        obey pam restrictions = Yes
        pam password change = Yes
                passwd program = /usr/bin/passwd %u passwd chat = *New*password*
        %n\n *Retype*new*password* %n\n
        *passwd:*all*authentication*tokens*updated*successfully*
        restrict anonymous = Yes
        log level = 2
        log file = /var/log/samba/%m.log
                max log size = 1000 socket options = TCP_NODELAY
        SO_RCVBUF=8192 SO_SNDBUF=8192
                dns proxy = No invalid users = root bin daemon adm lp sync halt
        shutdown mail news uucp operator games gopher ftp nobody vcsa mailnull ntp
        rpc xfs gdm rpcuser nfsnobody nscd pcap @wheel
        create mask = 0644
        hosts allow = 192.168.0.
        printing = lprng

[homes]
        comment = Home Directories
        valid users = %S
        read only = No
        create mask = 0664
        directory mask = 0775
        browseable = No

[printers]
        comment = All Printers
        path = /var/spool/samba
        printable = Yes
        browseable = No


[med-data]
        path = /data/medware
        valid users = user2 user3 user5
```

10.0 Bibliography

Firewalls, A Complete Guide by Marcus Goncalves, ISBN 0-07-135639-8

Hacking Exposed, Network Security Secrets and Solutions by Stuart McClure & Joel Scambray, George Kurtz, ISBN 0-07-818187-0

Hack Notes, Linux and Unix Security by Nitesh Dhanjani, ISBN 0-07-222786-9

O' Reilly, Using Samba by Rober Eckstein, David Collier-Brown and Peter Kelly, ISBN 1-56592-449-5

Red Hat Linux Security and Optimization by Mohammed J. Kabir, ISBN0-7645-4754-2

Red Hat Linux Networking and System Administration by Terry Collings and Kurt Wall, ISBN 0-7645-3632-X

Red Hat Enterpris Linux ES (Version 2.1) Installation Instructions and Important Information by Dell August 2003 P/N Rev. A00

Red Hat Enterprise Linux ES 2, Red Hat Enterprise Linux ES Installation Guide by Red Hat 2003

RHCE, Red Hat Certified Engineer Linux Study Guide (Exam RH302) by Michael Jang, ISBN 0-07-222485-1

Securing Linux, A Survival Guid for Linux Security Version 1.0 by David Koconis, Jim Murray, Jos Purvis and Darrin Wassom, ISBN 0-9724273-5x

11.0 Webliography

Agnitum www.agnitum.com

CERT http://www.cert.org/advisories/CA-2003-20.html
http://www.cert.org/contact_cert/certmaillist.html

Bastille-Linux www.bastille-linux.org

GRUB http://www.gnu.org/software/grub

Libsafe www.research.avayalabs.com/project/libsafe

NBTscan www.inetcat.org/software/nbtscan.html

NMAP www.insecure.org

Nessus www.nessus.org

Passwords http://www.securitystats.com/tools/password.php
http://www.mit.edu/afs/sipb/project/doc/passwords/passwords.html

Red Hat http://www.redhat.com/software/rhel/features
http://www.redhat.com/support/errata/rhel/21/qu2/driverdisks.html
http://rhn.redhat.com

Security Info http://www.securityfocus.com

Samba www.samba.org

Uranus http://uranus.it.swin.edu.au/~jn/linux/rawwrite.htm

Last Updated: May 28th, 2017

Upcoming Training

SANS London July 2017	**London, United Kingdom**	**Jul 03, 2017 - Jul 08, 2017**	**Live Event**
SANSFIRE 2017	**Washington, DC**	**Jul 22, 2017 - Jul 29, 2017**	**Live Event**
SANS Network Security 2017	**Las Vegas, NV**	**Sep 10, 2017 - Sep 17, 2017**	**Live Event**
SANS OnDemand	**Online**	**Anytime**	**Self Paced**
SANS SelfStudy	**Books & MP3s Only**	**Anytime**	**Self Paced**