GIAC Systems and Network Auditor(GSNA)
Practical Assignment
Version 3.0, Option 1

Auditing a Fedora Core 1 Linux

Audit you Fedora Home System

Jorge D. Ortiz-Fuentes

February 17, 2004

Abstract

This document explains how to conduct a security audit to a computer with the Fedora Core 1 as the only operating system installed. Although this document addresses threats, vulnerabilities and risks of computers in a home environment, the aim of this suggestions are to enhance the security of the corporate network they connect to.

The document contains a vulnerability analysis for these computers, a check list compiling the best practices regarding Fedora Core 1 security, some examples of the results of the checks and a report to management explaining the findings and giving some recommendations.

Contents

List of Tables

Part I
Research in Audit, Measurement Practice, and Control

1 Identify the system to be audited

Virtual Company, Inc. is a medium size development company. As part of our social benefits program, Virtual Company, Inc. has been offering special discounts for personal computer acquisition and broadband connection lease. The double purpose of this program is to contribute to the employee satisfaction and get more computer awareness among the employees.

These computers are desktop systems with the following hardware:

CPU: IA-32 based at 2.4 GHz.

RAM: 512 MB.

Graphics: 3D accelerated card.

Hard disk: 120 GB IDE ATA-100.

CD/DVD: combo CD-RW/DVD-ROM.

Preinstaled O.S.: Fedora Core 1.

Monitor: 17” TFT.

Network: 10/100BT to connect to the DSL router.

The computers obtained through the program, as well as some others, are often used to connect to Virtual Company’s internal network using virtual private networks (VPNs). That has caused serious security problems in the past mainly due to viruses and worms that had infected these computers —for example, through email attachments or simply while surfing the web— and got around the company’s firewall while connecting remotely.

Although, the computers are in no way officially supported by the company’s IT personnel, those previous security incidents had made Virtual Company’s CIO, to invest a small budget in developing a policy for remote connections and to help the users by providing them with a baseline for the security of their home computers. The success of this document has been providing the users with a clear and concise document which really helps them to have less problems with their own systems.

These systems are used by their owners for several purposes, but an initial effort was conducted while subscribing to the computer discount plan to identify the top five most usual activities, and the majority agreed on the following ones:

- Internet surfing.
- Read and write email.
- Instant messaging.
- Playing games.
- Remote connections to Virtual Company.

All of this tasks are successfully performed with Linux so Fedora Core 1 was chosen for its functionality, hardware compatibility and free (as in beer) license.

Fedora Core 1 is a Linux distribution that constitutes the evolution of Red Hat Linux. After the communication from Red Hat, Inc. by the middle of 2003 committing to their enterprise Linux version, they decided to change the product into a project. They opened the development of the Red Hat Linux product to the community while still having some participation in the project. They still participate in the project, have a steering committee that controls the release schedule and the global technical decisions but dropped support for it[1].

The list of packages available as part of the distribution for Fedora Core 1 can be found here: http://fedora.redhat.com/projects/package-list/.

2 Evaluate the most significant risk to the system

In this section I will provide definitions of the basic terms (risk, threat and vulnerability), explain the methodology used, and enumerate the results.

2.1 Definitions

In the rest of this document I will use the terms threat, vulnerability, and risk with the following meanings:

Threat The security issues represented by a natural aggression or an attack conducted by an individual or a group.

Vulnerability The specific exposure of the system or network under the influence of a threat.

Risk The probability of having a security problem considering the current conditions and the degree of exposure of the system to the different vulnerabilities.

2.2 Methodology

Thomas R. Peltier identifies[2] the following five basic steps that constitute the structure of every risk analysis method:

1. Identify the asset to be reviewed.

2. Ascertain the threats, risks, concerns or issues to that asset.
3. Prioritize the risk or determine the vulnerability of the threat to the asset.
4. Implement corrective measures, controls, safeguards, or accept the risk.
5. Monitor the effectiveness of the controls and assess their effectiveness.

The methodology employed in Virtual Company for performing a risk analysis is strongly based on the information provided as the first method of chapter 2 in his book “Information Security Risk Analysis.”[2]

The steps to do the risk analysis are:

1. Define the scope and identify the information assets to protect.

2. Identify the most significant threats. Information security threats are related to at least one of the following aspects of information:

   Confidentiality: The information is only accessed by those allowed to do it.

   Integrity: Only authorized modifications are allowed, other than that the information remains complete and unaltered.

   Availability: The information can be used when need.

3. Identify the main vulnerabilities.

4. For each of them evaluate the exposure of the system being analyzed. Assign a number to this exposure in the range 1 to 5 as detailed in the Table 1.

Probability	Meaning
1	Really exceptional or rare.
2	Low.
3	Medium.
4	High.
5	Almost certain.

Table 1: Meaning of the vulnerability exposure

5. For each of the vulnerabilities evaluate the impact of it happening. Assign a number to this impact in the range 1 to 5 with the meaning described in the Table 2.

6. Calculate the risk factor for each of the vulnerabilities as the addition of the exposure and the impact. Vulnerabilities with a risk factor of 6 (medium) or higher (high) should be considered together with the controls in place to study if other measures are required. Threats with a risk factor lower than 6 will be ignored.

   As we can see in the Table 3 of possible risk factors the values resulting from both a low probability and a low impact will be ignored.

Impact	Meaning
1	Very low.
2	Low.
3	Medium.
4	High.
5	Very high.

Table 2: Meaning of the vulnerability probabilities

		Exposure				
		1	2	3	4	5
Impact	1	2	3	4	5	6
	2	3	4	5	6	7
	3	4	5	6	7	8
	4	5	6	7	8	9
	5	6	7	8	9	10

Table 3: Resultant risk factors

There are other methods available, like the one suggested by Gareth Davies or the Facilitated Risk Analysis Process[2]. But I consider the one used here very simple to implement while still providing meaningful results.

This is a qualitative method. Instead of calculating real probabilities based on existent data, a subjective estimation is done based on experience. Although this produces subjective results which cannot be used for cost analysis, calculations are much simpler and provides a more flexible method to implement.

2.3 Results

The definition of scope for this study takes into account both the information assets directly related to the home computers, which are important only to the owner of the computer, and the ones that belong to Virtual Company that are accessible when a remote connection is made from those computers. These assets are the ones included in Table 4.

Table 4: Information assets

Asset	Examples
Data stored in the remote computer.	Private documents, pictures, email.
Data accessed from the computer.	Visited websites, products browsed, financial data.
System resources availability.	Be able to use your computer when you want to.
Source code and data.	The source code or data of each of the projects of Virtual Company.
Knowledge bases.	The support knowledge base.

continued on next page . . .

Table 4: Information assets (cont.)

Asset	Examples
Strategic data.	Plans for future products and releases.
Customer data.	Addresses, bills
Employee data.	Personal data, address, salaries.

The most significant threats to the information assets described in Table 4 are enumerated in Table 5. Each of the threats has a different capacity to inflict damage which is commented in the table.

Table 5: Threats to the information assets

Threat	Description	Damage
Fire, flood or similar natural disaster	Destruction of home computer due to a natural threat.	Computer and data unusable.
Physical theft	An intruder getting inside of the house and stealing the computer	Computer and data loss.
Power outage	Electric company fails to provide you with the energy to power your computer.	Computer temporally unusable.
Communications outage	The connections to the Internet and to Virtual Company provided by a communications company fail.	No connection to the Internet nor to Virtual Company.
Subsystem hardware failure	One of the subsystems of the computer (video card, memory, hard disk. . .) fails.	Computer temporally unusable, possible data loss for some cases (hard disk).
Untrained administration	The computers are administered by people with insufficient knowledge about the tasks they need to perform.	Computer temporally unusable, data loss.
File corruption	Files are corrupted by a hardware or software failure.	Data loss.
File deletion	Files are intentionally or unintentionally deleted.	Data loss.
Denial of Service	The computer cannot be used as expected as a result of an attack.	Computer temporally or permanently unusable.
Viruses, trojans and other malware	Software introduced the computer that performs an evil action (deleting files, attack other computers. . .)	Computer unusable and possible data and confidentiality loss.

continued on next page . . .

Table 5: Threats to the information assets (cont.)

Threat	Description	Damage
System *owned* by intruder	An intruder has unauthorized control of the computer.	Data loss, confidentiality loss, identity theft.

The main vulnerabilities of the system are enumerated and described in Table 6. For each vulnerability I have estimated the exposure —in the *Exp.* column— and the impact — in the *Imp.* column— and calculated the risk factor —in the *R. Factor* column.

The exposure and impact are estimated taking in account the target system of this study. Although a commercial web site would have a low to medium (2–3) exposure to a denial of service attack and its impact would be high or very high (4–5), a home computer is less exposed to such attack and the impact would lower in any case. But life is not always easier for the home computer owners: since the system and the data contained in it are not consider too critical, not so many preventive and proactive actions are taken. Also, taking care of these computers is more of a hobby than a job.

Table 6: Mayor vulnerabilities

#	Vulnerability	Description	Exp.	Imp.	R. Fact.
1	Lack of BIOS boot password	System can be booted without a password.	1	4	6
2	Lack of BIOS setup password	Boot order can be modified without a password.	1	5	6
3	Lack of O.S. loader boot password	The operating system can be loaded with other options without a password.	1	5	6
4	Weak, trivial, default or blank passwords	The password used for authentication is easy to guess.	4	5	9
5	Password hashes can be retrieved	The encrypted passwords (one way hashes) can be read by an unauthorized user.	2	4	6
6	Passwords last too long	There is no maximum age for the passwords.	3	3	6
7	Root can do login.	Root user can be used to login in some services.	3	4	7
8	Many root users	There are more than one privileged user accounts.	2	5	7

continued on next page . . .

Table 6: Mayor vulnerabilities (cont.)

#	Vulnerability	Description	Exp.	Imp.	R. Fact.
9	Start-up files with weak permissions	Files that get executed automatically at the beginning of every session can be modified.	3	5	8
10	Current working directory in PATH	A file in the current working directory can be executed without explicit indication of its directory path.	2	4	6
11	System accounts with shell	System accounts which are not directly used for login into the system that have a shell assigned can be used to execute commands.	2	4	6
12	Weak permissions in default execution paths	The permissions of the directories of the default execution path or their files allow to modify or change them.	2	4	6
13	Weak permissions or wrong owner of the home directories	The permissions or owner of the home directory of some user allows for reading from or writing into it.	3	3	6
14	Unnecessary group membership	Some user belongs to more groups than it needs to.	2	4	6
15	Binaries with unnecessary privileges	There exists some binary which has the SETUID or the SETGID privileges.	4	5	9
16	System binaries with weak permissions or wrong owners.	There exist system binaries with permissions or owners that allow for unauthorized modification or deletion.	3	4	7
17	System configuration files with weak permissions or wrong owners.	There exist system configuration files with permissions or owners that allow for unauthorized modification or deletion.	3	4	7
18	System libraries with weak permissions or wrong owners.	There exist system libraries with permissions or owners that allow for unauthorized modification or deletion.	3	4	7

continued on next page . . .

Table 6: Mayor vulnerabilities (cont.)

#	Vulnerability	Description	Exp.	Imp.	R. Fact.
19	System device files with weak permissions or wrong owners.	There exist system device files with permissions or owners that allow for unauthorized modification or deletion.	3	4	7
20	Public temporal directories with excessive permissions	Users can delete or modify any files in the public temporal directory without having to own them.	3	3	6
21	Misplaced or uncontrolled device files	There are device files outside of the `/dev` directory.	2	4	6
22	Unnecessary services running	There are services running in the computer with the potential to attend clients that have no specific purpose and that can be stopped safely.	4	5	9
23	Services running with unnecessary privileges	There are services run by a privileged user which could run without any loss in functionality by a regular user.	4	5	9
24	RPC services attending connections	There are RPC services running in the computer with the potential to attend clients that have no specific purpose and that can be stooped safely.	4	5	9
25	Incoming network connections are accepted	The computer accepts incoming network connections.	5	4	9
26	Corrupted or badly installed software	There are programs installed that don't correspond to their required image.	2	4	6
27	Unpatched known software bug	There is a software bug exploitable local or remotely which has been published through a security bulletin or similar and that has a software patch available.	5	4	9

continued on next page . . .

Table 6: Mayor vulnerabilities (cont.)

#	Vulnerability	Description	Exp.	Imp.	R. Fact.
28	Unknown system or application bug	There is a software bug exploitable local or remotely which hasn't been published and without a software patch available.	5	4	9
29	Security event not registered	Some event related to security has occurred but it hasn't been registered anywhere.	5	2	7
30	Web browser, mail client or gaim allow for execution of arbitrary code	The applications used for surfing in Internet, manage email, and communicate with instant messages allow for execution of arbitrary code.	3	5	8

3 What is the current state of practice?

Several documents have been written about Red Hat Linux security —the basis for Fedora Core— both by Red Hat, Inc. and by independent sources. While many of the recommendations and explanation remain applicable to Fedora Core, some things have changed and some aspects are not covered by any of them.

The most significant security guides for Red Hat Linux, with a description extracted from their introductions, are included by order of relevance in the Table 7 with a short description. Further references to them will be done using the bibliography reference listed at the end of this document.

Table 7: Security resources

Title	Description
Red Hat Linux Security Guide[4]	Designed by Red Hat to assist users of Red Hat Linux 9 in securing their systems against local and remote intrusion, exploitation, and malicious activity. Details the planning and the tools involved in creating a secured computing environment.
Securing and Optimizing Linux: The Ultimate Solution[5]	Covers secure and clean installation of Red Hat Linux, how to tighten the security of the configured system, how to build a fortress around the system by securing the network, and recommended programs to keep communications secure.

continued on next page . . .

Table 7: Security resources (cont.)

Title	Description
Security Quick Start HOWTO for Red Hat Linux[6]	An introduction to the most basic concepts of security as they relate to Red Hat Linux, and as a starting point only.
Linux Security Quick Reference Guide[7]	A starting point for improving the security of the system, to serve as a pointer to more in-depth security information, and to increase security awareness and methods that can be used to improve security.

There are other documents which are not particular to neither Fedora Core 1 nor Red Hat Linux.[8], but rather referred to general aspects of information security. They are listed in Table 8.

Table 8: Security resources

Title	Description
Home Network Security[8]	An overview for home users of the security risks and countermeasures associated with Internet connectivity, especially focused in permanent connections (such as cable modems and DSL).
ISO-17799/BS-7799[9]	It is a comprehensive, current, internationally recognized, and auditable security management standard, originally developed as a result of industry and government demand for a common framework for effective security management practice and inter-company trading and is used by a wide range of organizations in industry and commerce.

Part II
Create an Audit Checklist

4 Checklist

In this section I will offer a checklist for auditing a Fedora Core 1 used as a home computer with the purposes mentioned in section 1, namely: Internet surfing, read and write email, instant messaging, playing games, and remote connections to Virtual Company.

Unless otherwise stated, all the checks must be run login as root first.

4.1 Physical security

#	1	Title:	BIOS Boot Password
Ref.	Risk	Nature	
[4][5]	1	O	

Check

A computer without a BIOS boot password can be booted by anybody. Setting a BIOS boot password disallows this.

Switch on the computer and check that it asks for a password before booting up the system.

✓	The computer asks for a password to boot. A wrong password keeps the computer from booting.
✗	No password is asked to boot the machine.

#	2	Title:	BIOS Setup Password
Ref.	Risk	Nature	
[4][5]	2	O	

Check

A computer without a BIOS setup password allows for booting from a different device —another hard disk, a floppy or a CD/DVD— thus bypassing the operating system. Setting a BIOS setup password disallows this.

Switch on the computer. Try to modify the boot order or any other BIOS setting and check that it asks for a password before allowing you to do so.

✓	The BIOS asks for a password to modify its settings.
✗	No password is asked to change the BIOS settings.

#	3	Title:	Boot Loader Password
Ref.	Risk	Nature	
[4][7]	3	O	

Check

If GRUB, the default boot loader for Fedora Core 1, doesn't have password set, the booting parameters can be modified at boot time. Setting a password in GRUB's configuration disallows this.

Switch on the computer. Wait for the GRUB —the boot loader— to load. Press the E key to edit the boot sequence.

✓	GRUB asks for a password to modify the boot parameters.
✗	No password is asked when trying to modify GRUB's boot parameters.

4.2 Authentication

#	4	Title:	Password Strength
Ref.	Risk	Nature	
[4][5] [6][7]	4	O	

Check

Password strength can be audited using a password cracker. That is a tool which uses a dictionary of words and modifications of them to try to guess the user passwords.

1. Download John the Ripper from the following location ftp://mirrors.kernel.org/fedora.us/fedora/fedora/1/i386/SRPMS.stable/john-1.6-0.fdr.2.1.src.rpm

2. Install it by executing the following commands as root:

```
# rpm -ivh /home/jorge/john-1.6-0.fdr.2.1.src.rpm
   1:john                   ### ... ### [100%]
# cd /usr/src/redhat/
# rpmbuild -bb SPECS/john.spec --target i686
  .... Lots of output ...
# rpm -ivh RPMS/i686/john-1.6-0.fdr.2.1.i686.rpm
Preparing...                ### ... ### [100%]
   1:john                   ### ... ### [100%]
```

3. Run it against the computer passwords for 1 hour with the following commands:

```
# mkdir /root/JOHN
# cp /etc/shadow /root/JOHN
# cd /root/JOHN
# john shadow
Loaded x passwords with y different salts (FreeBSD MD5 [32/32])
... Passwords in clear text ...
```

✓	No password can be cracked with John the Ripper running for 1 hour.
✗	Any cracked password.

#	5	Title: Password Hashes
Ref.	Risk	Nature
[4][5] [6][7]	5	O

Check

The user file /etc/passwd used to be the password database in traditional UNIX systems. However, since /etc/passwd must be world readable —it is also used for converting UIDs and GIDs to user and group names,— everybody could read the password hashes and try to decrypt them by automated guessing with a password cracker. Modern UNIX-like systems do not use this file to store password hashes anymore. The shadow or password file —/etc/shadow is used in Linux instead.

Read the user file /etc/passwd and check that there is no password file included in it.

With a regular user —a non privileged user— try to read the password file /etc/shadow

✓	No password are found in /etc/passwd and a regular user gets a *Permission denied* error message when trying to read /etc/shadow.
✗	Any password can be read as a regular user.

#	6	Title:	Password Age
Ref.	Risk	Nature	
[4][5]	6	O	

Check

Passwords should be changed periodically to make harder the task of guessing them. This is controlled with the maximum password age parameter.

The maximum password age of a user can be checked and modified using the `chage` command, but this command has to be run for each user.

For a faster way to check the maximum password age, run the following command:

```
# awk  -F: ’{if($2!˜/ˆ*/ && $2 !˜ /ˆ!/){print $1 ":"$5}}’ /etc/shadow
```

This command prints a list of the users and their maximum password age. It only shows the users with a password assigned.

✓	No users with a maximum password age bigger than 360 (one year).
✗	Any user with a maximum password age bigger than 300.

4.3 Users

#	7	Title:	Root login
Ref.	Risk	Nature	
[4][5] [6][7]	7	O	

Check

Using the privileged accounts for regular use is a bad practice that can result in many problems.

The `login` program used as for signing onto the computer through telnet or through a virtual terminal. When Fedora Core 1 is configured to allow graphical logins the program used is `gdm`. Both `login` and `gdm` honor the configuration of the `/etc/securetty` that indicates which tty lines or pseudo-ttys can be used by root to login. Only virtual consoles should be included in this file so the administrator can login when there is a serious problem.

You can verify the contents of the `/etc/securetty` with the `less` command.

```
# less /etc/securetty
```

✓	A small number of virtual consoles can be used for root login or none at all.
✗	Root can login using many tty lines.

#	8	Title:	One privileged user
Ref.	Risk	Nature	
[4][5] [6][7]	8	O	

Check

In Linux —like in any other UNIX-like operating system— the users with unlimited privileges are the ones whose user identifier (UID) is equal to zero. Many exceptions to the security checks are hardcoded in the kernel of the operating system allowing disabling the security restrictions for users with UID 0.

This can be easily check by running the following command that prints all the users with UID equal to 0:

```
# awk  -F: ’{if($3 == 0) {print $1}}’ /etc/passwd
```

✓	Only root has UID 0.
✗	There are more users with UID 0.

#	9	Title:	Permissions of startup files
Ref.	Risk	Nature	
[4][5] [6][7]	9	O	

Check

The permissions and owners of each startup file should only allow modification by the user that executes it (and root).

To check this run the following command:

```
# for i in \
> $(awk -F: ’{if ($7 != "/sbin/nologin"){print $6}}’ /etc/passwd)
> do
>   ls -ld $i/.[A-z]*
> done
```

✓	Startup files should not be writable by others and should be owned by the user.
✗	Writable startup files or not owned by user.

#	10	Title:	Execution PATH
Ref.	Risk	Nature	
[4][5] [6][7]	10	O	

Check

The PATH environment variable indicates where to look for the binaries to execute them. The current directory might not be a controlled directory and trojanized versions of system commands can be executed instead of the legitimate ones.

Execute the command `echo $PATH` running as each of the computer users.

✓	Current working directory is not included.
✗	A dot between two colons or two consecutive colons.

#	11	Title:	System Accounts Shell
Ref.	Risk	Nature	
[4][5] [6][7]	11	O	

Check

System accounts are never used for login into the system so no valid shell should be configured for them.

The following command prints a list of the users and their configured shells:

```
# awk -F: ’{print $1": "$7}’ /etc/passwd | grep -v "/sbin/nologin"
```

✓	Only regular users and root can have a shell different to `/sbin/nologin`. Special purpose users may have other shell programs.
✗	System accounts with a valid shell.

#	12	Title: Execution PATH Permissions
Ref.	Risk	Nature
[4][5] [6][7]	12	O

Check

The permissions of the directories included in the default PATH should protect the binaries from unauthorized deletion.

The default PATH after installation of Fedora Core 1 contains the following directories:

- `/bin`
- `/sbin`
- `/usr/bin`
- `/usr/sbin`
- `/usr/kerberos/bin`
- `/usr/local/bin`
- `/usr/local/sbin`
- `/usr/X11R6/bin`

To look for directories that allow deleting files or files that can be modified, run the following command (**Note:** Don't forget the parenthesis):

```
# (IFS=:; for i in $PATH;
> do find $i -maxdepth 0 -type d -perm -0002 ;
> find $i -maxdepth 1 -type f -perm -0002; done)
```

✓	No file or directory is printed.
✗	Some file or directory is printed.

#	13	Title: HOME Permissions and Owners
Ref.	Risk	Nature
[4][5] [6][7]	13	O

Check

Only the user should be able to create or delete files in his home directory. This can be checked with the following command:

```
# for i in $(awk -F: '{print $6}' /etc/passwd | sort -u);
> do  ls -ld $i; done
```

✓	All the directories should exist, belong to the user and its main group and not have the write permission for others.
✗	Any directory which does not exist, belongs a different user or group, or has the write permission for others.

#	14	Title:	Group Membership
Ref.	Risk	Nature	
[4][5] [6][7]	14	O	

Check

A user should only be belong to minimal set of groups which are strictly needed. Special tasks should be done only with the administrator user.

The group membership of each user is described in `/etc/group` and can be explored with the following command.

```
# for i in $(awk -F: ’{print $1}’ /etc/passwd); \
> do  GRP=""; \
>   for j in $(grep $i /etc/group | awk -F: ’{print $1}’); \
>   do \
>     GRP="$GRP $j"; \
>   done; \
> echo "$i =$GRP"; \
> done
```

✓	Regular users should belong to their User Private Group (UPG) and root should belong to `root`, `bin`, `daemon`, `sys`, `adm`, `disk`, and `wheel`.
✗	Any other group that is not required for tasks that must be done by the user.

4.4 Files

#	15	Title:	SUID/SGID Binaries
Ref.	Risk	Nature	
[4][5] [6][7]	15	O	

Check

The SUID and SGID bits provide a program the special privilege to run as the user or group owning it. If such a program can be tricked into running arbitrary code, the code would run as the user or group owning it.

```
# find / -type f -perm +06000 2> /dev/null
```

The following files have this privilege in the default installation. The system will have problems to run completly if the privilege is removed form the ones that have a star at the end.

```
/usr/X11R6/bin/XFree86 *
/usr/sbin/userhelper *
/usr/sbin/utempter *
/usr/sbin/sendmail.sendmail
/usr/sbin/usernetctl
/usr/sbin/lockdev *
/usr/sbin/userisdnctl
/usr/sbin/gnome-pty-helper *
```

/usr/bin/chage
/usr/bin/wall
/usr/bin/chsh
/usr/bin/write
/usr/bin/lockfile *
/usr/bin/rlogin
/usr/bin/slocate
/usr/bin/sudo
/usr/bin/lppasswd
/usr/libexec/openssh/ssh-keysign *
/bin/ping6 *
/bin/mount
/bin/su *
/sbin/pam_timestamp_check
/sbin/unix_chkpwd
/usr/bin/gpasswd
/usr/bin/chfn
/usr/bin/newgrp
/usr/bin/passwd *
/usr/bin/rcp
/usr/bin/rsh
/usr/bin/at
/usr/bin/crontab
/usr/lib/vte/gnome-pty-helper *
/bin/ping *
/bin/traceroute6 *
/bin/umount
/bin/traceroute *
/sbin/pwdb_chkpwd
/sbin/netreport

✓	Only files listed above are printed.
✗	Additional files have been printed.

#	16	Title:	System Binaries Permissions and Owners
Ref.	Risk	Nature	
[4][5] [6][7]	16	O	

Check

System binaries should not have write permissions for group or others and should belong to user root and group root with some exceptions. The exceptions are:

File	User	Group
/bin/rpm	rpm	rpm
/bin/mail	root	mail
/usr/sbin/lockdev	root	lock
/usr/sbin/utempter	root	utmp
/usr/sbin/sendmail.sendmail	root	smmsp
/usr/sbin/gnome-pty-helper	root	utmp

The following two commands can be used to check the permissions of the binaries and their ownership respectively.

```
# for i in /bin /sbin /usr /usr/sbin; \
> do
>   find $i -maxdepth 1 -type f -perm +0022; \
> done
# for i in /bin /sbin /usr /usr/sbin; \
> do
>   find $i -maxdepth 1 -type f -a ! \( -user root -a -group root \) ; \
> done
```

✓	No output from the first command and only the exceptions of the table for the second.
✗	Any file from the first command or files not listed above for the second.

#	17	Title:	System Configuration Permissions and Owners
Ref.	Risk	Nature	
[4][5] [6][7]	17	O	

Check

System configuration files contained in `/etc` should not have write permissions for group or others and should belong to user root and group root.

The following two commands can be used to check the permissions of the system configuration files and their ownership respectively.

```
# find /etc -type f -perm +0022
# find /etc -type f -a ! \( -user root -a -group root \)
```

✓	No files printed with the first command nor the second one, except for `/etc/dumpdates` that will appear on both.
✗	Files other that `/etc/dumpdates` printed as a result of the first or the second command.

#	18	Title:	System Libraries Permissions and Owners
Ref.	Risk	Nature	
[4][5] [6][7]	18	O	

Check

System libraries should not have write permissions for group or others and should belong to user root and group root.

The following two commands can be used to check the permissions of the libraries and their ownership respectively.

```
# for i in /lib /usr/lib; \
> do
>   find $i -maxdepth 1 -type f -perm +0022; \
> done
# for i in /bin /sbin /usr /usr/sbin; \
> do
>   find $i -maxdepth 1 -type f ! -a \( -user root -a -group root \) ; \
> done
```

✓	No file gets printed.
✗	Any file printed as a result of executing the first command or the second one.

#	19	Title:	System Devices Permissions
Ref.	Risk	Nature	
[4][5] [6][7]	19	O	

Check

The permissions and ownership of the device files are a fundamental way to restrict access to hardware and kernel services.

The package manager application can verify different properties of the files installed with each package against the intended values. This is probably the best way to verify the permissions and ownership of the device files. However, it must be taken into account that the PAM console module changes the permissions of certain devices. Therefore, the check has to be run from a remote system with no user logged in or ignore all differences reported by it that are also specified in the console module configuration file (`/etc/security/console.perms`).

The verification of the dev package —the one that installs all the device files— can be done with this command:

```
rpm -V dev
```

✓	No differences reported by rpm besides the ones included in `/etc/security/console.perms`.
✗	Any differences reported by rpm that are not included in `/etc/security/console.perms`.

#	20	Title:	Public Temporal Directories Permissions
Ref.	Risk	Nature	
[4][5] [6][7]	20	O	

Check

The sticky bit should be set for all the public temporal directories.

This can be checked with the command:

```
# for i in /tmp /var/tmp ; \
> do \
>   find $i -maxdepth 0 -perm +01000 ; \
> done
```

✓	No output is printed after executing the suggested command.
✗	Either `/tmp` or `/var/tmp`.

#	21	Title:	System Devices Placement
Ref.	Risk	Nature	
[4][5] [6][7]	21	O	

Check

The standard location for device files in a UNIX-like system is under `/dev` and files outside this path should be disallowed since they are uncontrolled by the administrator.

The device files that are out of the standard path can be found using the following command:

```
find / -type c -o -type b 2> /dev/null | grep -v "^/dev"
```

✓	No device files are printed after executing the command.
✗	Any files printed as a result of the command.

4.5 Services

#	22	Title:	Services Running
Ref.	Risk	Nature	
[4][5] [6][7]	22	S	

Check

Many Fedora Core 1 services behave as network servers —a daemon is listening for connections on one or more network ports— and can be used to attack the computer.

Fedora Core 1 comes with most services switched off by default. The run level —which can be obtained with the command `runlevel`— defines which services are activated. A graphical desktop runs by default in run level 5.

The command used to verify which services will run in every run level is `chkconfig --list`. Read the output and switch off any services that you don't need.

✓	Only necessary services switched on.
✗	Unnecessary services switched on.

4.6 Connections

#	23	Title:	Incoming Connections
Ref.	Risk	Nature	
[4][5] [6][7]	25	O	

Check

Fedora Core 1 comes with a built-in firewall that is even able to do stateful inspection of network packets.

This firewalls should be enabled. Services should only be trusted when they are actually needed and secure services used rather than the insecure ones that offer similar functionality. For example, ssh should always be used instead of telnet or ftp.

To check that the firewall is enabled execute the following command:

```
# redhat-config-securitylevel
```

✓	Firewall is enabled as in Figure 1.
✗	Firewall is disabled.

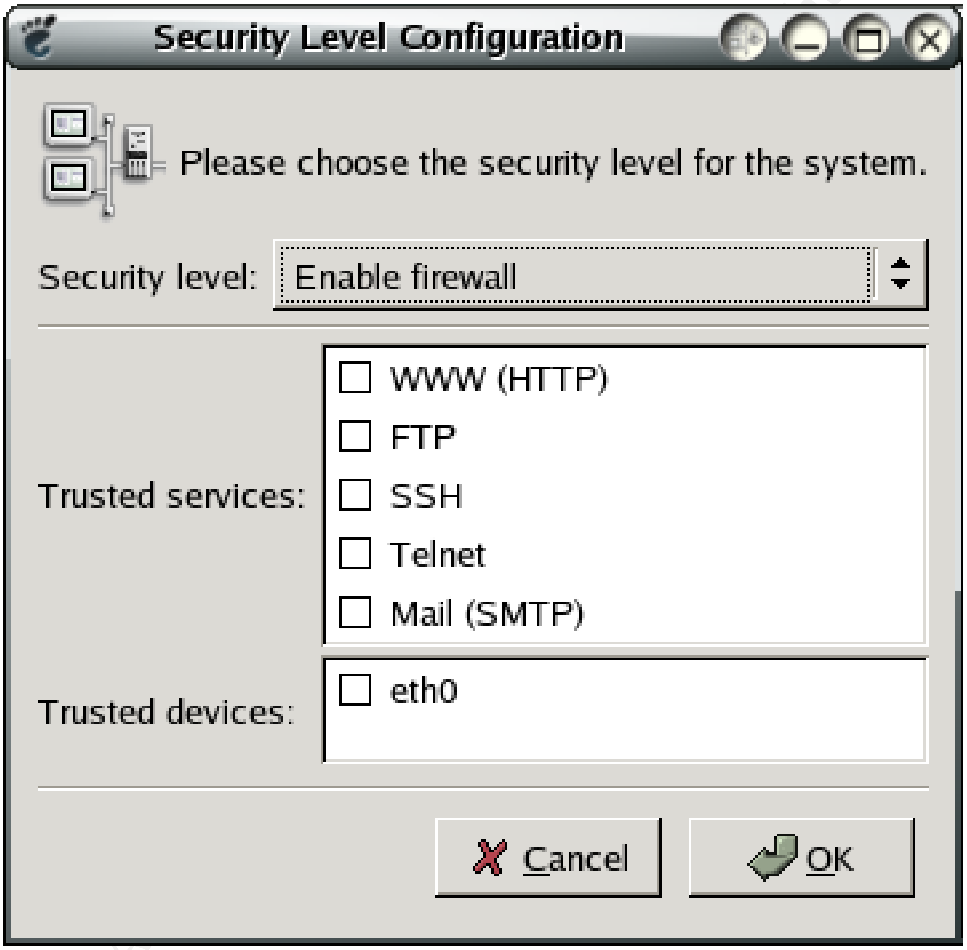
4.7 Software

#	24	Title:	Package Status
Ref.	Risk	Nature	
[4][5] [6][7]	26	O	

Check

Software installation in Fedora Core 1 is done using the RPM package management system. The list of installed packages can be obtained with the command `rpm -qa`. Many properties —user owner, group owner, size, MD5 hash, . . . — of all the files installed as part of any package can be checked against the expected values of those properties using the command `rpm -V` *`package`* or `rpm -Va` if all packages are to be changed.

It is considered normal behavior of the system to get reported many changes as the result of the verification process. But it is very significant to get a different MD5 sum for a binary or library file. Other results must be studied carefully.



Please choose the security level for the system.

Security level: Enable firewall

Trusted services:
- □ WWW (HTTP)
- □ FTP
- □ SSH
- □ Telnet
- □ Mail (SMTP)

Trusted devices:
- □ eth0

Figure 1: Firewall configuration

✓	No MD5 sum is modified for binaries or libraries (no 5 appears preceding the file name).
✗	Binaries or libraries with MD5 sum modified (appears a 5 preceding the file name).

#	25	Title:	Software Patches
Ref.	Risk	Nature	
[4][5] [6][7]	27	O	

Check

Software must be kept up to date so known software vulnerabilities cannot be used to attack the computer.

Fedora Core 1, as its predecessors, comes with the `up2date` utility that helps to update the computer software and shows if there are new versions of any package available.

To get the list of available packages that are not yet installed in the computer, execute:

```
up2date-nox --dry-run
```

✓	No new packages should be available for install.
✗	There are new packages available for install.

Part III

Conduct the Audit - Testing, Evidence and Findings

5 Results of the Audit

For this section I have chosen the ten checks from the previous section that I consider more relevant to show an example here. The criteria for choosing them have been the risk factor of the vulnerability related to that check, the complexity of the check and the significance of the results.

#	1	Checking:	3	Boot Loader Password

Evidence

The machine is switched on and The first interactive screen displays a blue background and the Fedora Core logo at the bottom of the screen. There should be at least one line to choose to boot the Fedora Core 1, although more lines can be present if the kernel has been updated. I choose any of them and press the E key. The screen changes and no password is asked. Three lines are displayed:

1. The first one indicates in which disk and partition can be found the kernel.

2. The second one selects one the many possible kernels present in the partition and provides it with boot parameters.

3. The third one is a file containing kernel modules that must be loaded before the root file system is available.

If I edit the third one adding the word `single` at the end of the line, Linux is booted in single user mode. Then you are logged in as root and no password has been asked.

Findings

✗ The boot loader entries can be edited without password. The direct result of this non-compliance is that the computer is exposed to vulnerability 3.

#	2	Checking:	4	Password Strength

Evidence

These home computers come with only two users configured that can do login: a regular user and root. They come with preset passwords that should be changed as soon as possible. The two accounts in this computer are `jorge` —the regular user— that has the password *user* and `root` that has a more complicated password (*virtual*).

I copy the shadow file (`/etc/shadow` and start running john with this shadow file for 1 hour.

Findings

✗ Both passwords were successfully guessed. The direct result of this non-compliance is that the computer is exposed to vulnerability 4.

#	3	Checking:	8	One Privileged User

Evidence

I run the suggested command to check if more than one privileged user exists.

```
# awk -F: ’{if ($3 == 0) {print $1}}’ /etc/passwd
root
```

Findings

✓ There is only one privileged user.

#	4	Checking:	10	Execution PATH

Evidence

I start as root and run the command `echo $PATH`. Then I login as each of the users —one in this case— and run the same command. Notice the `–` used with `su` that returns a login shell, thus modifying the PATH.

```
# echo $PATH
/usr/kerberos/sbin:/usr/kerberos/bin:/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/u
su - jorge
$ echo $PATH
/usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin:/home/jorge/bin
```

Findings

✓ None of the execution paths contains the current working directory expressed as a dot or two consecutive colons.

# 5	Checking:	11	System Accounts Shell

Evidence

I run the sugested command to check if more than one privileged user exists.

```
# awk -F: ’{print $1": "$7}’ /etc/passwd | grep -v "/sbin/nologin"
root: /bin/bash
sync: /bin/sync
shutdown: /sbin/shutdown
halt: /sbin/halt
news:
jorge: /bin/bash
```

The users `root` and `jorge` aren’t system accounts and must have a shell to be userd. The users `sync`, `shutdown`, and `halt` are special purpose users and can have this programs as their shell. However, the user `news` has no shell assigned and the manual page of the `passwd` file says that:

> *shell, the program to run at login (if empty, use /bin/sh).*

Findings

✗ The user `news` has the `/bin/sh` shell assigned. The direct result of this non-compliance is that the computer is exposed to vulnerability 11.

# 6	Checking:	17	System Configuration Permissions and Owners

Evidence

I run the two suggested commands:

```
# find /etc -type f -perm +0022
/etc/dumpdates
# find /etc -type f -a ! \( -user root -a -group root \)
/etc/dumpdates
```

Findings

✓ Only the file `/etc/dumpdates` is printed and this one is considered valid in the check.

# 7	Checking:	19	System Devices Permissions

Evidence

I run the verification utility of RPM with the following results:

```
# rpm -V dev
.....U..   /dev/apm_bios
.....U..   /dev/audio
```

```
.....U..   /dev/audio1
.....U..   /dev/audioctl
.....U..   /dev/beep
.....U..   /dev/console
.....U..   /dev/dsp
.....U..   /dev/dsp1
.....U..   /dev/dsp56k
.....U..   /dev/fb0
.....U..   /dev/fb1
.....U..   /dev/fb10
.....U..   /dev/fb11
.....U..   /dev/fb12
.....U..   /dev/fb13
.....U..   /dev/fb14
.....U..   /dev/fb15
.....U..   /dev/fb16
.....U..   /dev/fb17
.....U..   /dev/fb18
.....U..   /dev/fb19
.....U..   /dev/fb2
.....U..   /dev/fb20
.....U..   /dev/fb21
.....U..   /dev/fb22
.....U..   /dev/fb23
.....U..   /dev/fb24
.....U..   /dev/fb25
.....U..   /dev/fb26
.....U..   /dev/fb27
.....U..   /dev/fb28
.....U..   /dev/fb29
.....U..   /dev/fb3
.....U..   /dev/fb30
.....U..   /dev/fb31
.....U..   /dev/fb4
.....U..   /dev/fb5
.....U..   /dev/fb6
.....U..   /dev/fb7
.....U..   /dev/fb8
.....U..   /dev/fb9
.....U..   /dev/fd0
.....U..   /dev/fd0CompaQ
.....U..   /dev/fd0D360
.....U..   /dev/fd0D720
.....U..   /dev/fd0H1440
.....U..   /dev/fd0H360
.....U..   /dev/fd0H720
.....U..   /dev/fd0d360
```

```
.....U..   /dev/fd0h1200
.....U..   /dev/fd0h1440
.....U..   /dev/fd0h1476
.....U..   /dev/fd0h1494
.....U..   /dev/fd0h1660
.....U..   /dev/fd0h360
.....U..   /dev/fd0h410
.....U..   /dev/fd0h420
.....U..   /dev/fd0h720
.....U..   /dev/fd0h880
.....U..   /dev/fd0u1040
.....U..   /dev/fd0u1120
.....U..   /dev/fd0u1440
.....U..   /dev/fd0u1660
.....U..   /dev/fd0u1680
.....U..   /dev/fd0u1722
.....U..   /dev/fd0u1743
.....U..   /dev/fd0u1760
.....U..   /dev/fd0u1840
.....U..   /dev/fd0u1920
.....U..   /dev/fd0u2880
.....U..   /dev/fd0u3200
.....U..   /dev/fd0u3520
.....U..   /dev/fd0u360
.....U..   /dev/fd0u3840
.....U..   /dev/fd0u720
.....U..   /dev/fd0u800
.....U..   /dev/fd0u820
.....U..   /dev/fd0u830
.....U..   /dev/fd1
.....U..   /dev/fd1CompaQ
.....U..   /dev/fd1D360
.....U..   /dev/fd1D720
.....U..   /dev/fd1H1440
.....U..   /dev/fd1H360
.....U..   /dev/fd1H720
.....U..   /dev/fd1d360
.....U..   /dev/fd1h1200
.....U..   /dev/fd1h1440
.....U..   /dev/fd1h1476
.....U..   /dev/fd1h1494
.....U..   /dev/fd1h1660
.....U..   /dev/fd1h360
.....U..   /dev/fd1h410
.....U..   /dev/fd1h420
.....U..   /dev/fd1h720
.....U..   /dev/fd1h880
```

```
.....U..   /dev/fd1u1040
.....U..   /dev/fd1u1120
.....U..   /dev/fd1u1440
.....U..   /dev/fd1u1660
.....U..   /dev/fd1u1680
.....U..   /dev/fd1u1722
.....U..   /dev/fd1u1743
.....U..   /dev/fd1u1760
.....U..   /dev/fd1u1840
.....U..   /dev/fd1u1920
.....U..   /dev/fd1u2880
.....U..   /dev/fd1u3200
.....U..   /dev/fd1u3520
.....U..   /dev/fd1u360
.....U..   /dev/fd1u3840
.....U..   /dev/fd1u720
.....U..   /dev/fd1u800
.....U..   /dev/fd1u820
.....U..   /dev/fd1u830
.M...U..   /dev/hdc
.....U..   /dev/input/js0
.....U..   /dev/input/js1
.....U..   /dev/input/js2
.....U..   /dev/input/js3
.....U..   /dev/midi0
.....U..   /dev/midi00
.....U..   /dev/midi01
.....U..   /dev/midi02
.....U..   /dev/midi03
.....U..   /dev/midi1
.....U..   /dev/midi2
.....U..   /dev/midi3
.....U..   /dev/mixer
.....U..   /dev/mixer1
.....U..   /dev/radio0
.....U..   /dev/radio1
.....U..   /dev/radio2
.....U..   /dev/radio3
.M...U..   /dev/scd0
.M...UG.   /dev/sda1
.M...U..   /dev/sdb1
.....U..   /dev/sequencer
.M...U..   /dev/sg1
.M......   /dev/shm
......G.   /dev/tty0
.M....G.   /dev/tty1
.M....G.   /dev/tty2
```

```
.M....G.   /dev/tty3
.M....G.   /dev/tty4
.M....G.   /dev/tty5
.M....G.   /dev/tty6
......G.   /dev/tty7
.....U..   /dev/usb/dc2xx0
.....U..   /dev/usb/dc2xx1
.....U..   /dev/usb/dc2xx10
.....U..   /dev/usb/dc2xx11
.....U..   /dev/usb/dc2xx12
.....U..   /dev/usb/dc2xx13
.....U..   /dev/usb/dc2xx14
.....U..   /dev/usb/dc2xx15
.....U..   /dev/usb/dc2xx2
.....U..   /dev/usb/dc2xx3
.....U..   /dev/usb/dc2xx4
.....U..   /dev/usb/dc2xx5
.....U..   /dev/usb/dc2xx6
.....U..   /dev/usb/dc2xx7
.....U..   /dev/usb/dc2xx8
.....U..   /dev/usb/dc2xx9
.....U..   /dev/usb/mdc8000
.....U..   /dev/usb/mdc8001
.....U..   /dev/usb/mdc80010
.....U..   /dev/usb/mdc80011
.....U..   /dev/usb/mdc80012
.....U..   /dev/usb/mdc80013
.....U..   /dev/usb/mdc80014
.....U..   /dev/usb/mdc80015
.....U..   /dev/usb/mdc8002
.....U..   /dev/usb/mdc8003
.....U..   /dev/usb/mdc8004
.....U..   /dev/usb/mdc8005
.....U..   /dev/usb/mdc8006
.....U..   /dev/usb/mdc8007
.....U..   /dev/usb/mdc8008
.....U..   /dev/usb/mdc8009
.....U..   /dev/usb/rio500
.....U..   /dev/usb/scanner0
.....U..   /dev/usb/scanner1
.....U..   /dev/usb/scanner10
.....U..   /dev/usb/scanner11
.....U..   /dev/usb/scanner12
.....U..   /dev/usb/scanner13
.....U..   /dev/usb/scanner14
.....U..   /dev/usb/scanner15
.....U..   /dev/usb/scanner2
```

```
.....U..   /dev/usb/scanner3
.....U..   /dev/usb/scanner4
.....U..   /dev/usb/scanner5
.....U..   /dev/usb/scanner6
.....U..   /dev/usb/scanner7
.....U..   /dev/usb/scanner8
.....U..   /dev/usb/scanner9
.....U..   /dev/vbi0
.....U..   /dev/vbi1
.....U..   /dev/vbi2
.....U..   /dev/vbi3
.....U..   /dev/video/em8300
.....U..   /dev/video/em8300_ma
.....U..   /dev/video/em8300_mv
.....U..   /dev/video/em8300_sp
.....U..   /dev/video0
.....U..   /dev/video1
.....U..   /dev/video2
.....U..   /dev/video3
.....U..   /dev/vtx
.....U..   /dev/vtx0
.....U..   /dev/vtx1
.....U..   /dev/vtx2
.....U..   /dev/vtx3
.....U..   /dev/winradio0
.....U..   /dev/winradio1
.....U..   /dev/winradio2
.....U..   /dev/winradio3
```

There are a lot of changes. But the check states that the ones included in the file /etc/security/console.perms should be ignored, so I read this file that has the following contents:

```
# /etc/security/console.perms
#
# This file determines the permissions that will be given to priviledged
# users of the console at login time, and the permissions to which to
# revert when the users log out.

# format is:
#   <class>=list of regexps specifying consoles or globs specifying files
#   file-glob|<class> perm dev-regex|<dev-class> \
#     revert-mode revert-owner[.revert-group]
# the revert-mode, revert-owner, and revert-group are optional, and default
# to 0600, root, and root, respectively.
#
# For more information:
# man 5 console.perms
# file classes -- these are regular expressions
<console>=tty[0-9][0-9]* vc/[0-9][0-9]* :[0-9]\.[0-9] :[0-9]
<xconsole>=:[0-9]\.[0-9] :[0-9]

# device classes -- these are shell-style globs
<floppy>=/dev/fd[0-1]* \
         /dev/floppy/* /mnt/floppy*
<sound>=/dev/dsp* /dev/audio* /dev/midi* \
        /dev/mixer* /dev/sequencer \
        /dev/sound/* /dev/beep
<cdrom>=/dev/cdrom* /dev/cdroms/* /dev/cdwriter* /mnt/cdrom*
<pilot>=/dev/pilot
<jaz>=/mnt/jaz*
<zip>=/mnt/pocketzip* /mnt/zip*
<ls120>=/dev/ls120 /mnt/ls120*
<scanner>=/dev/scanner /dev/usb/scanner*
<rio500>=/dev/usb/rio500
<camera>=/mnt/camera* /dev/usb/dc2xx* /dev/usb/mdc800*
<memstick>=/mnt/memstick*
<flash>=/mnt/flash*
<diskonkey>=/mnt/diskonkey*
<rem_ide>=/mnt/microdrive*
<fb>=/dev/fb /dev/fb[0-9]* \
     /dev/fb/*
<kbd>=/dev/kbd
<joystick>=/dev/js[0-9]*
<v4l>=/dev/video* /dev/radio* /dev/winradio* /dev/vtx* /dev/vbi* \
      /dev/video/*
<gpm>=/dev/gpmctl
<dri>=/dev/nvidia* /dev/3dfx*
<mainboard>=/dev/apm_bios

# permission definitions
<console>  0660 <floppy>     0660 root.floppy
<console>  0600 <sound>      0600 root
<console>  0600 <cdrom>      0660 root.disk
<console>  0600 <pilot>      0660 root.uucp
<console>  0600 <jaz>        0660 root.disk
<console>  0600 <zip>        0660 root.disk
<console>  0600 <ls120>      0660 root.disk
<console>  0600 <scanner>    0600 root
<console>  0600 <camera>     0600 root
<console>  0600 <memstick>   0600 root
<console>  0600 <flash>      0600 root
<console>  0600 <diskonkey>  0660 root.disk
<console>  0600 <rem_ide>    0660 root.disk
```

```
<console>  0600 <fb>          0600 root
<console>  0600 <kbd>         0600 root
<console>  0600 <joystick>    0600 root
<console>  0600 <v4l>         0600 root
<console>  0700 <gpm>         0700 root
<console>  0600 <mainboard>   0600 root
<console>  0600 <rio500>      0600 root

<xconsole> 0600 /dev/console 0600 root.root
<xconsole> 0600 <dri>         0600 root
```

The classes `console`, `floppy`, `sound`, `cdrom`, `scanner`, `rio500`, `camera`, `fb` `v4l`, and `mainboard` explain all the changes found with the RPM verification.

Findings

✓ No differences have been found besides the changes covered by `/etc/security/console.perms`

# 8	Checking: 22 Services

Evidence

To list the services that are enabled for the default run level, I execute the following commands:

```
# runlevel
N 5
# chkconfig --list
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
syslog          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
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
saslauthd       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
anacron         0:off   1:off   2:on    3:on    4:on    5:on    6:off
atd             0:off   1:off   2:off   3:on    4:on    5:on    6:off
irda            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
acpid           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
irqbalance      0:off   1:off   2:off   3:on    4:on    5:on    6:off
pcmcia          0:off   1:off   2:on    3:on    4:on    5:on    6:off
nfslock         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
microcode_ctl   0:off   1:off   2:on    3:on    4:on    5:on    6:off
smartd          0:off   1:off   2:on    3:on    4:on    5:on    6:off
isdn            0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

```
autofs          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
portmap         0:off   1:off   2:off   3:on    4:on    5:on    6:off
sendmail        0:off   1:off   2:on    3:on    4:on    5:on    6:off
rhnsd           0:off   1:off   2:off   3:on    4:on    5:on    6:off
crond           0:off   1:off   2:on    3:on    4:on    5:on    6:off
yum             0:off   1:off   2:off   3:off   4:off   5:off   6:off
winbind         0:off   1:off   2:off   3:off   4:off   5:off   6:off
messagebus      0:off   1:off   2:off   3:on    4:on    5:on    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
xfs             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
cups            0:off   1:off   2:on    3:on    4:on    5:on    6:off
ntpd            0:off   1:off   2:off   3:off   4:off   5:off   6:off
xinetd based services:
        chargen-udp:    off
        rsync:  off
        chargen:        off
        daytime-udp:    off
        daytime:        off
        echo-udp:       off
        echo:   off
        services:       off
        time:   off
        time-udp:       off
        cups-lpd:       off
        sgi_fam:        on
```

For each of the services that is enable for run level 5 —which is the default run level— that I do not recognize, I run try to get a manual —man— page.

At least the following services are considered unnecessary for this environment: netfs, isdn, portmap and sendmail.

Findings

✗ There are unnecessary services enabled. The direct result of this non-compliance is that the computer is exposed to vulnerability 22.

# 9	Checking: 23 Connection

Evidence

I run the redhat-config-securitylevel utility and the window fully matches the one presented in Figure 1.

Findings

✓ Firewall is enabled.

# 10	Checking: 25 Software Patches

Evidence

I run the utility that comes with Fedora Core 1 to keep the system updated:

```
up2date-nox --dry-run

Fetching package list for channel: fedora-core-1...

Fetching http://fedora.redhat.com/releases/fedora-core-1/headers/header.info...
########################################

Fetching package list for channel: updates-released...

Fetching http://fedora.redhat.com/updates/released/fedora-core-1/headers/header.info...
########################################

Fetching Obsoletes list for channel: fedora-core-1...

Fetching Obsoletes list for channel: updates-released...

Fetching rpm headers...
########################################

Name                      Version        Rel
----------------------------------------------------------
XFree86                   4.3.0          55                  i386
XFree86-100dpi-fonts      4.3.0          55                  i386
XFree86-75dpi-fonts       4.3.0          55                  i386
XFree86-Mesa-libGL        4.3.0          55                  i386
XFree86-Mesa-libGLU       4.3.0          55                  i386
XFree86-base-fonts        4.3.0          55                  i386
XFree86-devel             4.3.0          55                  i386
XFree86-doc               4.3.0          55                  i386
XFree86-font-utils        4.3.0          55                  i386
XFree86-libs              4.3.0          55                  i386
XFree86-libs-data         4.3.0          55                  i386
XFree86-tools             4.3.0          55                  i386
XFree86-truetype-fonts    4.3.0          55                  i386
XFree86-xauth             4.3.0          55                  i386
XFree86-xfs               4.3.0          55                  i386
foomatic                  3.0.0          21.3                i386
gaim                      0.75           1.3.0               i386
ghostscript               7.07           15.1                i386
gimp-print                4.2.6          4                   i386
gimp-print-plugin         4.2.6          4                   i386
gimp-print-utils          4.2.6          4                   i386
gnome-libs                1.4.1.2.90     36                  i386
gnome-libs-devel          1.4.1.2.90     36                  i386
hpijs                     1.5            4.1                 i386
mutt                      1.4.1          5                   i386
nss_ldap                  207            6                   i386
```

```
pam_krb5                  2.0.5          1                   i386
pango                     1.2.5          4                   i386
pango-devel               1.2.5          4                   i386
redhat-config-printer     0.6.79.5       1                   i386
redhat-config-printer-gui 0.6.79.5       1                   i386
samba-client              3.0.2          7.FC1               i386
samba-common              3.0.2          7.FC1               i386


Testing package set / solving RPM inter-dependencies...
########################################

Name                                    Version        Rel
----------------------------------------------------------
XFree86                   4.3.0          55                  i386
XFree86-100dpi-fonts      4.3.0          55                  i386
XFree86-75dpi-fonts       4.3.0          55                  i386
XFree86-Mesa-libGL        4.3.0          55                  i386
XFree86-Mesa-libGLU       4.3.0          55                  i386
XFree86-base-fonts        4.3.0          55                  i386
XFree86-devel             4.3.0          55                  i386
XFree86-doc               4.3.0          55                  i386
XFree86-font-utils        4.3.0          55                  i386
XFree86-libs              4.3.0          55                  i386
XFree86-libs-data         4.3.0          55                  i386
XFree86-tools             4.3.0          55                  i386
XFree86-truetype-fonts    4.3.0          55                  i386
XFree86-xauth             4.3.0          55                  i386
XFree86-xfs               4.3.0          55                  i386
foomatic                  3.0.0          21.3                i386
gaim                      0.75           1.3.0               i386
ghostscript               7.07           15.1                i386
gimp-print                4.2.6          4                   i386
gimp-print-plugin         4.2.6          4                   i386
gimp-print-utils          4.2.6          4                   i386
gnome-libs                1.4.1.2.90     36                  i386
gnome-libs-devel          1.4.1.2.90     36                  i386
hpijs                     1.5            4.1                 i386
mutt                      1.4.1          5                   i386
nss_ldap                  207            6                   i386
pam_krb5                  2.0.5          1                   i386
pango                     1.2.5          4                   i386
pango-devel               1.2.5          4                   i386
redhat-config-printer     0.6.79.5       1                   i386
redhat-config-printer-gui 0.6.79.5       1                   i386
samba-client              3.0.2          7.FC1               i386
samba-common              3.0.2          7.FC1               i386
```

```
The following Packages were marked to be skipped by your configuration:

Name                                    Version        Rel  Reason
-------------------------------------------------------------------------------
kernel                    2.4.22         1.2166.nptlPkg name/pattern
kernel-source             2.4.22         1.2166.nptlPkg name/pattern
```

Findings

✗ There are several packages available for install. The direct result of this non-compliance is that the computer is exposed to vulnerability 27.

Part IV

Audit Report or Risk Assessment

6 Audit Report

6.1 Executive Summary

As a result of the investment of the Information Security Office, we have been designing a security checklist for the computers offered to the employees of this company.

This computers come with Fedora Core 1 preinstalled and are used among other things to connect to Virtual Company's internal network. These aspects have been taken into account to choose the more relevant security checks that will help both the owners of the computers and Virtual Company.

We have conducted an audit with the resultant checklist to one of those computers as it comes from the vendor. From the results of the audit, it must be remarked that:

- The computers come with two configured users whose passwords are easy to obtain.
- There are several security patches available that have not been installed in the computer.
- There are services enabled that are not necessary for the regular user of these computers and keep them running is a security risk.

We suggest to distribute a very short document with the best administration practices (changing passwords, software updates, . . .) together with the checklist that will be available for any employee.

6.2 Audit Findings

All the audit findings have been completely documented in Part III of this document.

6.3 Audit Recommendations

Some of the vulnerabilities described in Table 6 are not addressed in the check list. The two main ones are the unknown bug (28) and the arbitrary execution of code (30). They are so important because they are the *engine* for virus and worm transmission. Although viruses and worms are not so frequent in Linux, they do exist.

Apart from having the system updated frequently, it would be a good idea to run an anti-virus. Panda offers a freeware product[11] and Central Command sells his product[10] for individual workstations for $34.95. I would suggest to install and test each of them (there is a trial version for Central Command's product) and include the conclusions in another document available for the employees. This would take a week of work of one of the Information Security Office Engineers. Since this job profile has a documented cost of $75 per hour, this means a total cost of $3000.

If this cost is considered too high, there are some controls already in place. The main ones are stopping unnecessary services and updating the software frequently.

References

[1] Red Hat, Inc. “Fedora Project” 19 Jan. 2004.
URL: http://fedora.redhat.com/ 7 Feb. 2004.

[2] Peltier, Thomas R. “Information Security Risk Analysis.”, Boca Raton: Auerbach, 2001.

[3] SANS Institute. “The Twenty Most Critical Internet Security Vulnerabilities. The Experts Consensus.” v. 4.0. 8 Oct. 2003.
URL: http://www.sans.org/top20/ (13 Feb. 2004).

[4] Red Hat, Inc. “Red Hat Linux Security Guide”, 20 Feb. 2003.
URL: http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/pdf/rhl-sap-en-9.pdf (02 Feb. 2004)

[5] Mourani, Gerhard. “Securing and Optimizing Linux: The Ultimate Solution” v.2.0. 10 Jun. 2001.
URL: http://www.openna.com/products/books/sol/solus.php?e=0,1,4 (02 Feb. 2004)

[6] Burgiss, Hal. “Security Quick Start HOWTO for Red Hat Linux” v.1.2, 21 Jul. 2002.
URL: http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/pdf/Security-Quickstart-Redhat-HOWTO.pdf

[7] Wreski, Dave; Thomas, Benjamin. “Linux Security Quick Reference Guide” v.1.1. 2000.
URL: http://www.tldp.org/REF/ls_quickref/QuickRefCard-A4.pdf (02 Feb. 2004)

[8] CERT Coordination Center, “Home Network Security.” Tech Tips. 5 Dec. 2001.
URL: http://www.cert.org/tech_tips/home_networks.html (13 Feb. 2004).

[9] International Organization for Standardization. “ISO-17799:2000 Information technology – Code of practice for information security management.” 21 Sept. 2001.
URL: http://www.iso.ch/iso/en/CatalogueDetailPage.CatalogueDetail?
CSNUMBER=33441&ICS1=35 (12 Feb. 2004)

[10] Central Command. “Vexira Antivirus for Linux.”
URL: http://www.centralcommand.com/linux_workstation.html (17 Feb. 2004)

[11] Panda Software. “Panda Antivirus for Linux.” Version 7.0-1 Feb. 2004.
URL: http://www.pandasoftware.com/download/linux/linux.asp (17 Feb. 2004)