Unit No-6 : Security

- What is Computer Security?
- Hacking, Physical threats, Local security,
- **Monitoring Security:** Networking services.
- Configuration files

6.1 What is Computer Security?

Computer security is about the protection of computing assets against threats such as theft, accidental loss, unauthorized access by a third party, and "denial of service" (whereby legitimate users of computing assets are prevented from doing so). Computing assets include things such as:

➢Computer hardware

➢Information stored on a computer (personal details, financial & customer information, intellectual property, etc.)

➢ CPU, memory, storage, and network resources

➢Computing services (Web sites, FTP, and Print servers)

The concepts involved in computer security are straightforward;

1. **we need to identify what we want to protect and**
2. **The threats we want to protect them from,**
3. **Figure out how much effort to put into our protection schemes, and,**
4. **Once we've implemented them, regularly test the effectiveness of our protection.**

Identifying What Is Valuable

➢The first step in planning the security of your Red Hat Linux system is to **identify what you have on that system that is of value to us** . We'll probably want to include the **computer hardware**

➢Think about the **confidentiality of your data**, if unauthorized changes to our data.

➢So, **availability of your data is important too**. Even if your data is secure, and hasn't been modified by unauthorized people

➢In some environments, **performance of computer systems is critical**. Transaction processing systems need to Web servers need to respond quickly to requests from Web browsers. Sometimes the performance may be unacceptably low, so we'd better add performance to your list of things we want to protect.

➢ **Taking Back-up of Information**

6.2 Potential Threats

Next step in planning our system security is to identify the potential threats.

Hacking

- A hacker typically starts their activities by attempting to connect to your system using various TCP port numbers to try and find out where your system is vulnerable. This process is known as **probing.**
- If the hacker is somewhere out there on the Internet, they can probe your system only when you are connected, which is why people with "always on" Internet connections, *need to pay* particular attention to security. A secure system will simply ignore incoming probes so the potential hacker has no indication that there is anything using the IP address they picked to probe.
- If your system does respond to a hacker's probes, their next step will be to identify what software is listening on the ports (for example, a Web server, FTP server, Telnet, and so on) and try to exploit any security vulnerability in those programs.
- If the hacker is successful, they will be able to gain **root access** to your system, and may install modified versions of standard software that will allow them back in, or to use your system to launch attacks on other systems.

Physical Threats

Theft of computer equipment can be a big problem – especially for laptop users, and certain areas may suffer from environmental problems, that can damage computer equipment.

So, in summary our list of possible threats could look something like this:

- Operator error
- Hardware failure (particularly disk drives)
- Theft of equipment
- Unauthorized local access
- Unauthorized remote access
- Denial of service (DoS) attack
- Eavesdropping of network communications
- Environmental conditions damage

Denial of Service (DoS)

- Denial of Service attacks are designed to disrupt legitimate use of computer systems, rather than gain unauthorized access to information. These may be implemented by exploiting bugs in network services so that
- these services fail (for example, by sending malformed requests to a Web server, causing it to crash), or by4 overloading networks so that legitimate network traffic is unable to pass.
- Hackers sometimes attempt Denial of Service attacks by sending the target system TCP/IP packets with the SYN flag set, as if they were starting up a TCP connection, but never completing the connection. The target system allocates some resources for each incoming connection, so if the hacker sends enough of these SYN packets the target system will eventually exhaust its network resources. This is sometimes called a **SYN flood.**
- If your system is compromised, a hacker may use it to launch a Denial of Service attack on some other system without your knowledge. Coordinated attacks launched against a target system from a number of compromised systems are called **distributed denial of service attacks**, and these have been known to cause serious problems for the target system,

Transferring Data

Transferring information over a network also opens up the possibility of eavesdropping − someone other than the intended recipient may intercept the information on route.

Local Security

we've sorted out the physical security of our Red Hat system. We've enabled the BIOS password so that only those who know it can start the machine up in the first place, and we've taken care with the physical location So it won't get stolen and the disk drives will stay nice and cool.

Using the Root Account

The trick is always to work with the lowest level of authority that we need to perform a particular task. Usually, this is not the root account.

Passwords

The password that we use to authenticate ourselves to the operating system when we log in is the only way we have of proving our identity. Instead, a non−reversible mathematical function called a **hash is** used to combine your password with a random value and the hashed password is stored by the system.

Choosing a Strong Password

Choosing a strong password is really better phrased as avoiding weak passwords. Weak passwords are ones that can be guessed, either by humans or by password cracking programs.

So, avoid passwords that can be found in a dictionary (in any language), or variations of these, passwords that someone who knows you may guess (name of family member or pet), or anything generated in a systematic way. Include punctuation and other symbols, and allow passwords to be up to 15 characters long.

For example, here are some weak passwords:

- **Fred123**
- **Asdfghj (simple pattern on keyboard)**
- **B10nlc (trivial modification of dictionary word)**
- **Nitsob (a pet's name spelled backward)**

These would have been strong passwords until they were published in this book:

- **Xz%!q)_2+!**
- **3#&–Aa%yty?**
- **ap^Lj+~rZxp**]

However, make sure you can remember your password so you don't have to write it down. As soon as you do that, you're weakening it.

6.3 Monitoring Security

There are several ways of monitoring the security of your Red Hat system, allowing you to spot security breaches and other potential problems.,

System Logs: /var/log

The Red Hat Linux system maintains several log files that record system activity. Most of the interesting ones reside in **/var/log**. Here's a table describing the most important log files:

File in /var/log	Contents	View with
btmp	Record of all bad logins attempts. Updated by login program if it exists.	lastb command
cron	Messages sent to syslogd from the cron daemon (which schedules jobs on Unix systems).	Normal text viewing tools
dmesg	Kernel messages (from boot)	Normal text viewing tools
lastlog	Last login times for all users.	lastlog command
messages	Messages sent to syslogd with level of info or higher, except those from mail, cron or authentication related messages.	Normal text viewing tools
secure	Messages sent to syslogd from authpriv (i.e. authentication and security messages that should only be visible to privileged users).	Normal text viewing tools
wtmp	Record of all logins and logouts.	last command

Network Services

- One way a hacker may try to gain unauthorized access to our system is by exploiting weaknesses in the network services that we are running on our Red Hat Linux system.
- These are programs − often run in the background with no controlling terminal (called "daemons" in Unix, and "services" in Microsoft Windows) − that provide services to other computers. Examples include file transfer protocol (ftp), Web, Network File System (NFS), and print servers.

Enabling and disabling services

In particular, we need to be very careful about older services that send sensitive information (such as user names and passwords) across the network without any form of encryption (in **plain text).** Also, unreasonably hand out information about our system should be avoided where possible.

Service	TCP/UDP Port number	Description	Red Hat Package	Security Level
echo	7	Sends received characters back to sender.	xinetd	None.
daytime	13	Sends current date and time as ASCII string back to sender.	xinetd	None.
chargen	19	Generates continuous stream of ASCII characters for testing terminals.	xinetd	None.
chargen	20 (data) 21 (control) Random ports >1023	File Transfer Protocol. Allows transfer of files to and from remote systems.	vsftp or wu−ftpd	Weak. user names and passwords sent in plain text. "Anonymous FTP" allows access with no password.
ssh	22	Secure shell. Allows remote system to	Openssh	Good. Data is encrypted and

telnet	23	Allows remote system to access command line shell on local machine.	telnet–server	Weak. User names and passwords sent in plain text.
SMTP	25	Simple Mail Transfer Protocol. Used to transfer mail between systems.	sendmail	Weak. Mail tranferred in plain text.
time	37	Sends current time (in seconds since 00:00 1st Jan 1900) back to sender.	xinetd	None.
finger	79	Gives information about local system or users to remote machine.	finger–server	None.
http	80	Web server.	Httpd	Depends on server configuration.
auth (ident)	113	Indentification protocol. Allows remote system to determine indentity of a user of a particular TCP/IP connection.	pident	Supports DES encryption of returned information.
sftp	115	Secure File Tranfer Protocol. FTP–like data transfers over secure SSH connection.	openssh	Good.
nntp	119	Network News Transfer Protocol. Used to transfer USENET news groups.	inn	Depends on server configuration.
smb	137 138	Server Message Block. Allows Microsoft Windows	Samba	Weak. Information passed over network without encryption.

https	443	Secure Web server.	Httpd	Depends on server configuration.
lpd	515	Print Daemon. Allows remote machines to send print jobs to our printers.	LPRng or cups	Weak. Information passed over network without encryption.
rsh/rlogin	514	Allows remote system to access command line shell on local machine without supplying a password.	rsh−server	Weak. Relies on DNS (Domain Name Service) to identify remote system, so vulnerable if DNS compromised.
nfs	2049 (requires portmapper to be listening on port 111)	Network File System. Allows other systems to access file systems remotely.	nfs−utils	Weak. Information passed over network without encryption.

6.4 System Configuration & Configuration files

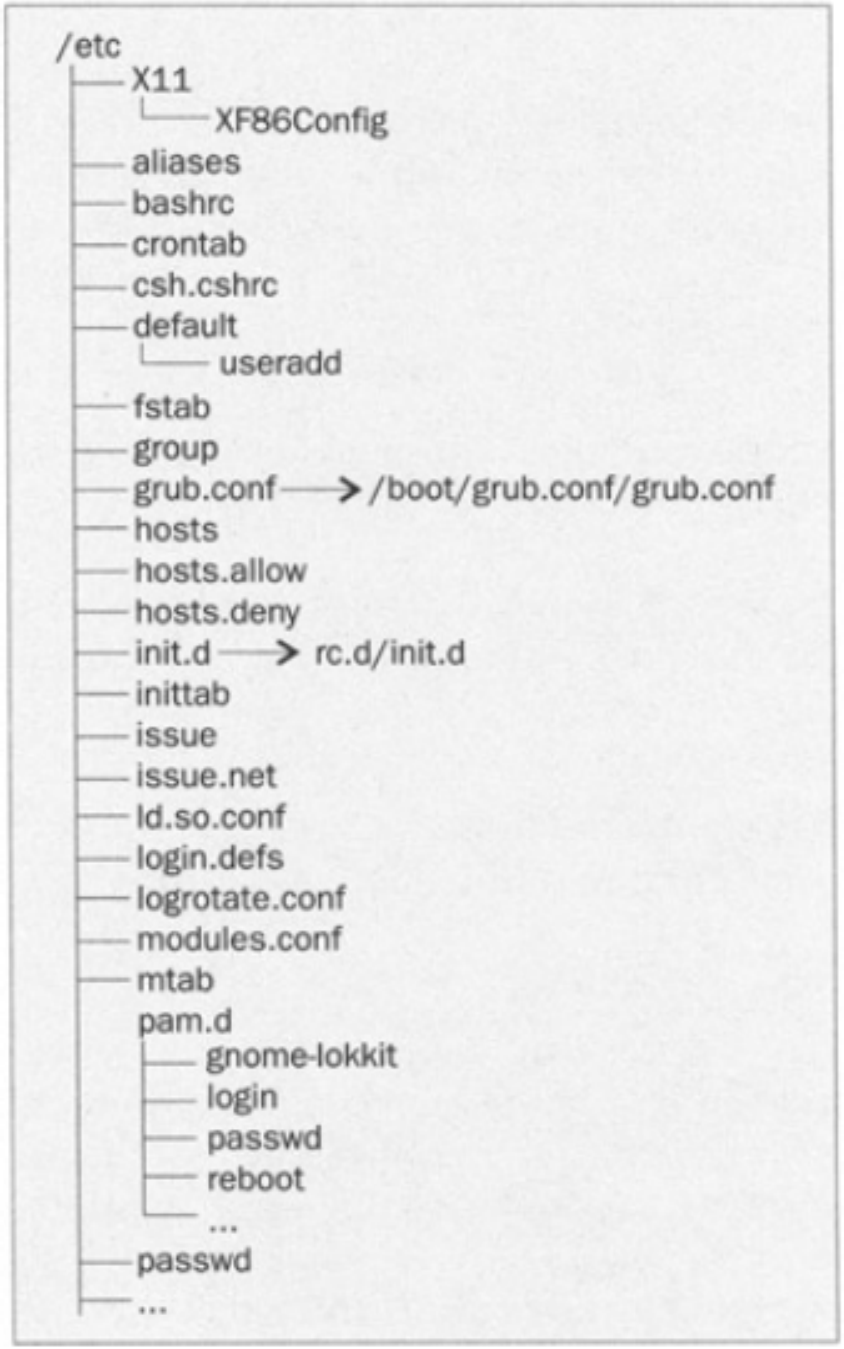
- System configuration encompasses such a broad area of system administration that it is very difficult to cover all aspects. And so, we will concentrate on configuration related to users and login, hardware, booting up and startup services, networking, and security.
- Depending on the operating system, the configuration information is stored in different locations.
- For example, Microsoft Windows stores most configuration data in the Registry, while the Mac OS stores it in separate binary files in a special Preferences folder.

How about Linux? Where does it store the configuration details? For the most part, Linux, the core components as well as individual applications, stores the information in *plain text files in the **/etc** directory or in one of its subdirectories. This gives us a* number of advantages over the other operating systems, namely:

- ➢ **We can read and edit the information easily with an text editor.**
- ➢ **We can back up the files consistently.**
- ➢ **We can maintain version control, thereby keeping track of all changes.**

Configuration Files

We mentioned that there are a large number of configuration files contained in the /etc directory hierarchy. They're stored in a tree structure;



etc/XF86Config

The XF86Config configuration file controls specific aspects of the X Window System (X11) server, from keyboard and mouse to monitor. This file is essential if X11 is to work properly on your system.

/etc/aliases

The aliases configuration file contains aliases for mail users and is used by the sendmail application. For example, you can set up an alias such that all mail sent to the alias mickeymouse is forwarded to the system administrator.

/etc/bashrc and /etc/csh.cshrc

These two configuration files set the defaults (file creation masks/ permissions, shell prompts, and so on) that are used by all **bash** and **csh** shell users upon starting a new shell.

/etc/crontab

This file is a configuration file for the **cron daemon, crond, which allows us to execute automated tasks –** tasks that run unattended at specified times.

Once a minute, the cron daemon checks for changes in the crontab file and reloads them into memory as necessary.

/etc/default/useradd

This file sets the default parameters that are used whenever a new user is created. For example, if you want all new users to have the C shell by default, then you would change the SHELL directive in the **useradd** configuration file.

etc/fstab

The fstab file contains the file system table, which is a table of all disk partitions, and their mount points and default mount options. You can use this file to tell Linux about any and all file systems to which the machine has access.

etc/group

This configuration file lists the group names and group IDs (GIDs) of all the groups of users known to the system. In Red Hat Linux, every user must be associated with at least one group.

/etc/grub.conf

The grub.conf configuration file is used at the time you start your system. When you start your system, the first program that runs is the **grand unified bootloader (GRUB).**

The GRUB is responsible for transferring control to the Linux kernel. The grub.conf file found in the /etc directory is, in fact, a symbolic link to the file **/boot/grub/grub.conf** – which in turn specifies the path to the kernel and the root partition.

/etc/hosts

The hosts file allows us to set up aliases for local and remote hosts. This is a very powerful feature that can simplify host name lookups. For example, if you wanted to force all of your users to go to www.google.com when they enter google, simply add this record to the hosts file:

216.239.57.101 google

/etc/init.d

This is a symbolic link **to a directory that contains a number of startup scripts. The startup scripts perform a** number of functions, including initialization of network connections and startup of server daemon processes.

/etc/Inittab

The inittab configuration file is probably the single most important file in the system − it controls the initialization process that occurs when you start the system. It is responsible for starting the init process; it contains a line to set the default run level to be used:

id:3:initdefault:

/etc/passwd

The passwd configuration file stores the account information (including the user name, full name, and path to the home directory and default shell) for every user on the system