ICSI MODULE 7:

Configuring Windows

7.1 Configuring Windows

Properly configuring Windows (Windows 7, 8, 10 and Server Editions) consists of many facets. You must disable unnecessary services, properly configure the registry, enable the firewall, properly configure the browser, and more.

Previously we have discussed the firewall concepts and the processes of both stateful packet inspection and stateless packet inspection, and a later section of this chapter discusses browser security. For now, let's go over the other important factors in Windows security configuration.

7.1.1 Accounts, Users, Groups and Passwords

Any Windows system comes with certain default user accounts and groups. These can frequently be a starting point for intruders who want to crack passwords for those accounts and gain entrance onto a server or network. Simply renaming or disabling some of these default accounts can improve your security.

Note: Windows has an affinity to move things in the control panel with each version. Your version (7, 8, 8.1, 10, etc.) might have things in a different location. If you have not already done so, take some time to familiarize yourself with the location of utilities in your version of Windows.

In Windows 7 or Windows 8, you find user accounts by going to Start, Settings, Control Panel, Users and Groups. In Windows 10 go to Start, Settings, and Accounts.

7.1.1.1 Administrator Accounts

The default administrator account has administrative privileges, and hackers frequently seek to obtain logon information for an administrator account. Guessing a logon is a two-step process of first identifying the username, and then the password. Default accounts allow the hacker to bypass the first half of this process. Administrators should disable this account.

Having an account with administrative privileges is necessary for maintaining your server. The next step is adding a new account, one with an innocuous name and giving that account administrative privileges. Doing so makes a hacker's task more difficult, as he must first discover what account actually has administrative privileges before he can even attempt to compromise that account.

Some experts suggest simply renaming the administrator account, or using an administrator account that has a username that indicates its purpose. That is not a recommendation for the following reasons:

- The whole point is that a hacker should not be able to identify which username has administrative privileges.
- Simply renaming the administrator account to a different name, but one that still indicates its administrative rights will not help this situation.

7.1.1.2 Other Accounts

The administrator account is the one most often targeted by hackers, but Windows also includes other default user accounts. Applying an equally demanding behaviour to all default accounts is a good idea. Any default account can be a gateway for a hacker to compromise a system. A few accounts that you should pay particular attention are:

- **IUSR_Machine name:** When you are running IIS, a default user account is created for IIS. Its name is IUSR_ and the name of your machine. This is a common account for a hacker to attempt to compromise. Altering this one in the manner suggested for the administrator account is advisable.
- **ASP.NET:** If your machine is running ASP.NET, a default account is created for web applications. A hacker that is familiar with .NET could target this account.
- **Database accounts:** Many relational database management systems, such as SQL Server, create default user accounts. An intruder, particularly one who wants to get at your data, could target these accounts.

When adding any new account, always give the new account's user or group the least number and type of privileges needed to perform their job, even accounts for IT staff members. Below are some examples:

- A PC technician does not need administrative rights on the database server. Even though belongs to the IT department, does not need access to everything in that department.
- Managers may use applications that reside on a web server, but they certainly should not have rights on that server.
- Just because a programmer develops applications that run on a server does not mean that should have full rights on that server.

These are just a few examples of things to consider when setting up user rights.

Remember: Always give the least access necessary for that person to do her job. This concept is often called least privileges, and is a cornerstone of security.

7.1.2 Setting Security Policies

Setting appropriate security policies is the next step in hardening a Windows server. This does not refer to written policies an organisation might have regarding security standards and procedures. In this case, the term security policies refers to the individual machines' policies.

The first matter of concern is setting secure password policies. The default settings for Windows passwords are not secure. The table

below shows the default password policies. Maximum password age refers to how long a password is effective before the user is forced to change that password.

Enforce password history refers to how many previous passwords the system remembers, thus preventing the user from reusing passwords. Minimum password length defines the minimum number of characters allowed in a password.

Password complexity means that the user must use a password that combines numbers, letters, and other characters. These are the default security settings for all Windows versions from Windows NT 4.0 forward. If your system is protected within a business environment, the settings at Local Security will be greyed out, indicating you do not have permissions to make changes.

Policy	Recommendation
Enforce password history	1 password remembered
Maximum password age	42 days
Minimum password age	0 days
Minimum password length	0 characters
Passwords must meet complexity requirements	Disabled
Store password using reversible encryption for all users in the domain	Disabled

The default password policies are not secure enough, but what policies should you use instead? Different experts answer that question differently. The table below shows the recommendations of Microsoft and the National Security Agency.

Policy	Microsoft	NSA
Enforce password history	3 passwords	5 passwords
Maximum password age	42 days	42 days
Minimum password age	2 days	2 days
Minimum password length	8 characters	12 characters
Passwords must meet complexity requirements	No recommendation	Yes
Store password using reversible encryption for all users in the domain	No recommendation	No recommendation

Developing appropriate password policies depends largely on the requirements of your network environment. If your network stores and processes highly sensitive data and is an attractive target to hackers, you must always skew your policies and settings toward greater security. However, bear in mind that if security measures are too complex, your users will find complying difficult. For example, very long, complex passwords (such as $%Tbx38T@_FgR$$) make your network quite secure, but such passwords are virtually impossible for users to remember.

7.1.3 Account Lockout Policies

When you open the Local Security Settings dialog, your options are not limited to setting password policies. You can also set account lockout policies. These policies determine how many times a user can attempt to log in before being locked out, and for how long to lock them out. The default Windows settings are shown in the table below.

Policy	Default Settings
Account lockout duration	Not defined
Account lockout threshold	0 invalid logon attempts
Reset account lockout counter after	Not defined

These default policies are not secure. Essentially, they allow for an infinite number of log-in attempts, making the use of password crackers very easy and virtually guaranteeing that someone will eventually crack one or more passwords and gain access to your system. The table below provides the recommendations from Microsoft and National Security Agency.

Policy	Microsoft	NSA
Account lockout duration	0, indefinite	15 hours
Account lockout threshold	5 attempts	3 attempts
Reset account after	15 minutes	30 minutes

7.1.4 Registry Settings

The Windows Registry is a database used to store settings and options for Microsoft Windows operating systems. This database contains critical information and settings for all the hardware, software, users, and preferences on a particular computer. Whenever users are added, software is installed or any other change is made to the system (including security policies), that information is stored in the registry.

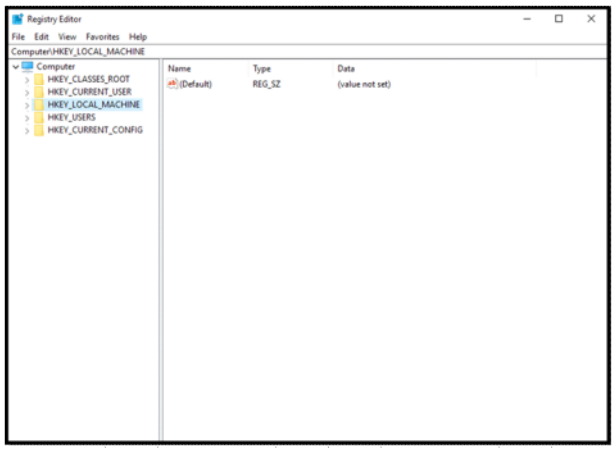
Secure registry settings are critical to securing a network. Unfortunately, that area is often overlooked. One thing to keep in mind is that if you do not know what you are doing in the registry, you can cause serious problems. So, if you are not very comfortable with the registry, do not touch it. Even if you are comfortable making registry changes, always back up the registry before any change.

7.1.5 Registry Basics

The physical files that make up the registry are stored differently depending on which version of Windows you are using. Older versions of Windows (that is, Windows 95 and 98) kept the registry in two hidden files in your Windows directory, called USER.DAT and SYSTEM.DAT. In all versions of Windows since XP, the physical files that make up the registry are stored in %SystemRoot% \System32\Config. Since Windows 8, the file has been named ntuser.dat.

Regardless of the version of Windows you are using, you cannot edit the registry directly by opening and editing these files. Instead you must use a tool, regedit.exe, to make any changes. There are newer tools like regedit32. However, many users find that the older regedit has a more user friendly "find" option for searching the registry. Either one will work.

Although the registry is referred to as a "database," it does not actually have a relational database structure (like a table in MS SQL Server or Oracle). The registry has a hierarchical structure similar to the directory structure on the hard disk. In fact, when you use regedit, you will note it is organized like Windows Explorer. To view the registry, go to Start, Run, and type regedit. You should see the Registry Editor dialog box as shown below. Some of the folders in your dialog box might be expanded.



Your Registry Editor dialog box will likely have the same five main folders as the one shown above in the screenshot. Each of these main branches of the registry is briefly described in the following list. These five main folders are the core registry folders. A system might have additions, but these are the primary folders containing information necessary for your system to run.

- **HKEY_CLASSES_ROOT:** This branch contains all of your file association types, OLE information, and shortcut data.
- **HKEY_CURRENT_USER:** This branch links to the section of HKEY_USERS appropriate for the user currently logged on to the PC.
- **HKEY_LOCAL_MACHINE:** This branch contains computer-specific information about the type of hardware, software, and other preferences on a given PC.
- **HKEY_USERS:** This branch contains individual preferences for each user of the computer.
- **HKEY_CURRENT_CONFIG:** This branch links to the section of HKEY_LOCAL_MACHINE appropriate for the current hardware configuration.

If you expand a branch, you will see its subfolders. Many of these have, in turn, more subfolders, possibly as many as four or more before you get to a specific entry. A specific entry in the Windows Registry is referred to as a key. A key is an entry that contains settings for some particular aspect of your system. If you alter the registry, you are actually changing the settings of particular keys.

7.1.6 Restrict Null Session Access

Null sessions are a significant weakness that can be exploited through the various shares that are on the computer. A null session is Windows' way of designating anonymous connections. Any time you allow anonymous connections to any server, you are inviting significant security risks. Modify null session access to shares on the computer by adding **RestrictNullSessAccess**, a registry value that toggles null session shares on or off to determine whether the Server service restricts access to clients logged on to the system account without username and password authentication. Setting the value to **"1"** restricts null session access for unauthenticated users to all server pipes and shares except those listed in the **NullSessionPipes** and **NullSessionShares** entries.

Key Path: HKLM\SYSTEM\CurrentControlSet\Services\LanmanServer

Action: Ensure that it is set to: Value = 1

7.1.7 Restrict Null Session Access Over Named Pipes

The null session access over named pipes registry setting should be changed for much the same reason as the preceding null session registry setting. Restricting such access helps to prevent unauthorised access over the network. To restrict null session access over named pipes and shared directories, edit the registry and delete the values, as shown below.

Key Path: HKLM\SYSTEM\CurrentControlSet\Services\LanmanServer

Action: Delete all values

7.1.8 Restrict Anonymous Access

The anonymous access registry setting allows anonymous users to list domain user names and enumerate share names. It should be shut off. The possible settings for this key are:

- 0—Allow anonymous users
- 1—Restrict anonymous users

- 2—Allow users with explicit anonymous permissions

Key Path: HKLM\SYSTEM\CurrentControlSet\Control\Lsa

Action: Set Value = 2

7.1.9 Remote Access to the Registry

Remote access to the registry is another potential opening for hackers. The Windows XP registry editing tools support remote access by default, but only administrators should have remote access to the registry. Fortunately, later versions of Windows turned this off by default. In fact, some experts advise that there should be no remote access to the registry for any person. This point is certainly debatable. If your administrators frequently need to remotely alter registry settings, then completely blocking remote access to them will cause a reduction in productivity of those administrators. However, completely blocking remote access to the registry is certainly more secure. To restrict network access to the registry:

1. Add the following key to the registry: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurePipeServers \winreg.

2. Select winreg, click the Security menu, and then click Permissions.

3. Set the Administrator's permission to Full Control, make sure no other users or groups are listed, and then click OK.

Recommended Value = 0

7.1.10 Services

A service is a program that runs without direct intervention by the computer user. In Unix/Linux environments, these are referred to as daemons. Many items on your computer are run as services. Internet Information Services, FTP Service, and many system services are good examples. Any running service is a potential starting point for a hacker. Obviously, you must have some services running for your computer to perform its required functions. However, there are services your machine does not use. If you are not using a service, it should be shut down.

7.1.11 Encrypting File System

Beginning with Windows 2000, the Windows operating system has offered the Encrypting File System (EFS), which is based on public key encryption and takes advantage of the CryptoAPI architecture in Windows 2000.

This still exists in Windows 7, 8, and 10; however, with the later versions of Windows, EFS is only available in the upper-end editions of Windows such as Windows Professional. With this system, each file is encrypted using a randomly generated file encryption key, which is independent of a user's public/private key pair; this method makes the encryption resistant to many forms of cryptoanalysis-based attacks. For our purposes the exact details of how EFS encryption works are not as important as the practical aspects of using it.

7.1.12 Security Templates

We have been discussing a number of ways for making a Windows system more secure, but exploring services, password settings, registry keys, and other tools can be a daunting task for the administrator who is new to security. Applying such settings to a host of machines can be a tedious task for even the most experienced administrator.

The best way to simplify this aspect of operating system hardening is to use security templates. A security template contains hundreds of possible settings that can control a single or multiple computers. Security templates can control areas such as user rights, permissions, and password policies, and they enable administrators to deploy these settings centrally by means of Group Policy Objects (GPOs).

Security templates can be customized to include almost any security setting on a target computer. A number of security templates are built into Windows. These templates are categorized for domain controllers, servers, and workstations. These security templates have default settings designed by Microsoft. All of these templates are located in the C:\Windows\Security\Templates folder. The following is a partial list of the security templates that you will find in this folder:

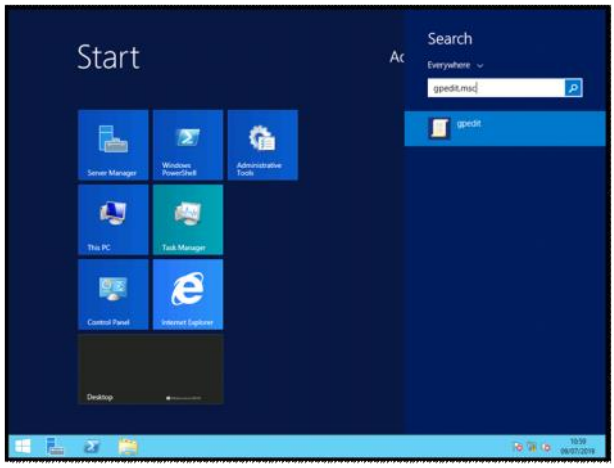
- **Hisecdc.inf:** This template is designed to increase the security and communications with domain controllers.
- **Hisecws.inf:** This template is designed to increase security and communications for client computers and member servers.
- **Securedc.inf:** This template is designed to increase the security and communications with domain controllers, but not to the level of the High Security DC security template.
- **Securews.inf:** This template is designed to increase security and communications for client computers and member servers.
- **Setup security.inf:** This template is designed to reapply the default security settings of a freshly installed computer. It can also be used to return a system that has been misconfigured to the default configuration.

Guided Exercise: Password Policies

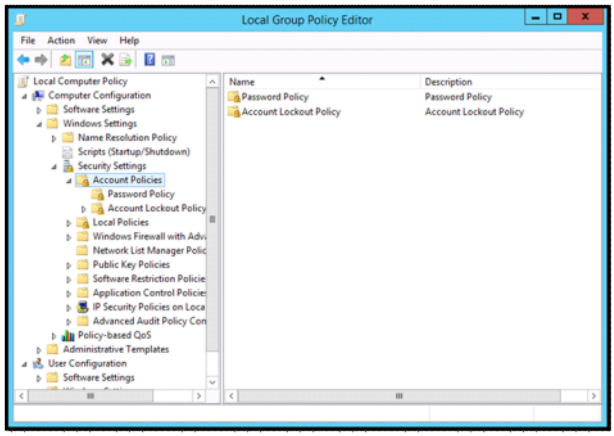
7.2 Guided Exercise: Password Policies

Resources	
Files	None
Machines	Windows Server

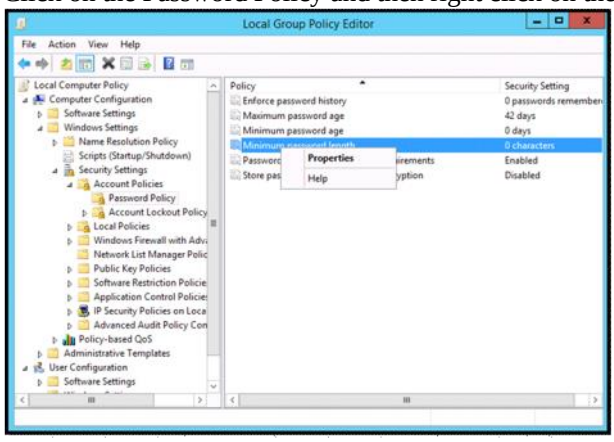
In this exercise you will create a new user account and set its password policies.
Login to Windows Server, click the Start button, write gpedit.msc and press Enter.



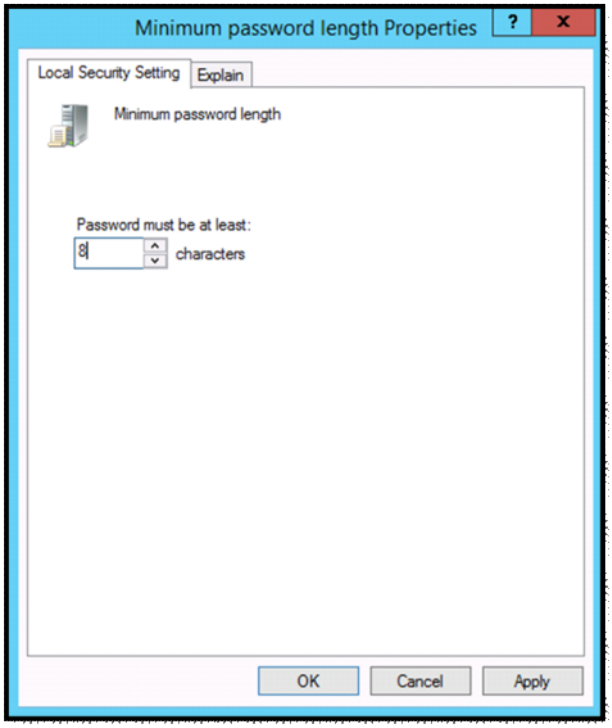
The Local Group Policy Editor window will open. Below Computer Configuration exand the folders Windows Setiings -> Account Policies.



Click on the Password Policy and then right click on the Minimu password length and select properties.



On the Minimum password length Properties window change the value 0 to 8, click Apply and then OK. Then close the Local Group Policy Editor.



Configuring Linux

7.3 Configuring Linux

An in-depth review of Linux security would be a lengthy task indeed. One reason is the diversity of Linux setups. Users could be using Debian, Red Hat, Ubuntu, or other Linux distributions. Some might be working from the shell, while others work from some graphical user interfaces such as KDE or GNOME. Fortunately, many of the same security concepts that apply to Windows can be applied to Linux. The only differences lie in the implementation, as explained in the following list:

- User and account policies should be set up the same in Linux as they are in Windows, with only a few minor differences. These differences are more a matter of using different names in Linux than in Windows. For example, Linux does not have an administrator account; it has a root account.
- All services (called daemons in Linux) not in use should be shut down.
- The browser must be configured securely.
- You must routinely patch the operating system.

In addition to some tactics that are common to Windows and Linux, a few approaches are different for the two operating systems:

- No application should run as the root user unless absolutely necessary. Remember that the root user is equivalent to the administrator account in Windows. Also, remember that all applications in Linux run as if started by a particular user, and therefore having an application run as root user would give it all administrative privileges.
- The root password must be complex and must be changed frequently. This is the same as with Windows administrator passwords.
- Disable all console-equivalent access for regular users. This means blocking access to programs such as shutdown, reboot, and halt for regular users on your server.
- Hide your system information. When you log in to a Linux box, it displays by default the Linux distribution name, version, kernel version, and the name of the server. This information can be a starting point for intruders. You should just prompt users with a "Login:" prompt.

7.3.1 Disable Services

Every service (daemon) that runs is executing code on the server. If there is a vulnerability within that code, it is a potential weakness that can be leveraged by an attacker; it is also consuming resources in the form of RAM and CPU cycles.

Many operating systems ship with a number of services enabled by default, many of which you may not use. These services should be disabled to reduce the attack surface on your servers. Of course you should not just start disabling services with reckless abandon—before disabling a service, it is prudent to ascertain exactly what it does and determine if you require it.

There are a number of ways to ascertain which services are running on a UNIX system, the easiest of which is to use the **"ps"** command to list running services. Exact argument syntax can vary between versions, but the **"ps ax"** syntax works on most systems and will list all currently running processes. For minor variations in syntax on your operating system, check the manual page for **"ps"** using the command **"man ps"**.

Services should be disabled in start-up scripts (**"rc"** or **"init"**, depending on operating system) unless your system uses **"systemd"**, in which case you can refer to the following discussion on **"systemd"**. Using the **"kill"** command will merely stop the currently running service, which will start once more during a reboot. On Linux the commands are typically one of: **"rc-update"**, **"update-rc.d"**, or

“service”. On BSD-based systems, you typically edit the file /etc/rc.conf.

For example, on several flavours of Linux the service command can be used to stop the sshd service: **service sshd stop**

To start sshd (one time): **service start sshd**

And to disable it from starting after a reboot:**update-rc.d -f sshd remove**

Some Linux distributions have moved toward using **“systemd”** as opposed to SysV startup scripts to manage services. **“systemd”** can be used to perform other administrative functions with regards to services, such as reloading configuration and displaying dependency information.

To stop sshd (one time): **systemctl stop sshd**

To enable sshd upon every reboot: **systemctl enable sshd**

And to disable sshd upon further reboots: **systemctl disable sshd**

Older Unix/Linux operating systems may use inetd or xinetd to manage services rather than rc or init scripts. (x)inetd is used to preserve system resources by being almost the only service running and starting other services on demand, rather than leaving them all running all of the time. If this is the case, services can be disabled by editing the inetd.conf or xinetd.conf files, typically located in the /etc/ directory.

7.3.2 File Permissions

Most Unix/Linux file systems have a concept of permissions—that is, files which users and groups can read, write, or execute. Most also have the SETUID (set user ID upon execution) permission, which allows a nonroot user to execute a file with the permission of the owning user, typically root. This is because the normal operation of that command, even to a nonroot user, requires root privileges, such as su or sudo.

Typically, an operating system will set adequate file permissions on the system files during installation. However, as you create files and directories, permissions will be created according to your umask settings. As a general rule, the umask on a system should only be made more restrictive than the default. Cases where a less restrictive umask is required should be infrequent enough that chmod can be used to resolve the issue. Your umask settings can be viewed and edited using the umask command. See man umask1 for further detail on this topic.

Incorrect file permissions can leave files readable by users other than whom it is intended for. Many people wrongly believe that because a user has to be authenticated to log in to a host, leaving world or group readable files on disk is not a problem. However, they do not consider that services also run using their own user accounts.

Take, for example, a system running a web server such as Apache, nginx, or lighttpd; these web servers typically run under a user ID of their own such as “www-data.” If files you create are readable by “www-data”, then, if configured to do so, accidentally or otherwise, the web server has permission to read that file and to potentially serve it to a browser. By restricting file system-level access, we can prevent this from happening—even if the web server is configured to do so, as it will no longer have permission to open the file.

As an example, the file test can be read and written to by the owner _www, it can be read and executed by the group staff, and can be read by anybody. This is denoted by the rw-, r-x, and r-- permissions in the directory listing:

$ ls -al test

-rw-r-xr-- 1 _wwwstaff 1228 16 Apr 05:22 test

In the Unix/Linux file system listing, there are 10 hyphens (-), the last 9 of which correspond to read, write, and execute permissions for owner, group and other (everyone). A hyphen indicates the permission is not set; a letter indicates that it is set. Other special characters appear less often; for example, an S signifies that the SETUID flag has been set.

If we wish to ensure that others can no longer see this file, then we can modify the permissions. We can alter them using the chmod command (o= sets the other permissions to nothing):

$ sudo chmod o= test

$ ls -la test

-rw-r-x--- 1 _wwwstaff 1228 16 Apr 05:22 test

7.3.3 File Integrity

File Integrity Management tools monitor key files on the file system and alert the administrator in the event that they change.These tools can be used to ensure that key system files are not tampered with, as in the case with a rootkit, and that files are not added to directories without the administrator’s permission, or configuration files modified, as can be the case with backdoors in web applications, for example.

There are both commercial tools and free/open source tools available through your preferred package management tool. Examples of open source tools that perform file integrity monitoring include Samhain and OSSEC. If you are looking to spend money to obtain extra features like providing integration with your existing management systems, there are also a number of commercial tools available.

Alternatively, if you cannot for whatever reason install file integrity monitoring tools, many configuration management tools can be configured to report on modified configuration files on the file system as part of their normal operation. This is not their primary function and does not offer the same level of coverage, and so is not as robust as a dedicated tool. However, if you are in a situation where you cannot deploy security tools but do have configuration management in place, this may be of some use.

7.3.4 Separate Disk Partitions

Disk partitions within Unix/Linux can be used not only to distribute the file system across several physical or logical partitions, but also to restrict certain types of action depending on which partition they are taking place on. Options can be placed on each mount point in /etc/fstab.

There are some minor differences between different flavours of Unix/Linux with regards to the options, and so consulting the system manual page—using man mount—before using options is recommended.

Some of the most useful and common mount point options, from a security perspective, are:

nodev

Do not interpret any special dev devices. If no special dev devices are expected, this option should be used. Typically only the /dev/ mount point would contain special dev devices.

nosuid

Do not allow setuid execution. Certain core system functions, such as su and sudo will require setuid execution, thus this option should be used carefully. Attackers can use setuid binaries as a method of backdooring a system to quickly obtain root privileges from a standard user account. Setuid execution is probably not required outside of the system-installed bin and sbin directories. You can check for the location of setuid binaries using the following command:

$ sudo find / -perm -4000

Binaries that are specifically setuid root, as opposed to any setuid binary, can be located using the following variant:

$ sudo find / -user root -perm -4000

ro

Mount the file system read-only. If data does not need to be written or updated, this option may be used to prevent modification. This removes the ability for an attacker to modify files stored in this location such as config files and static website content.

noexec

Prevents execution, of any type, from that particular mount point. This can be set on mount points used exclusively for data and document storage. It prevents an attacker from using this as a location to execute tools he may load onto a system and it can defeat certain classes of exploit.

7.3.5 Chroot

chroot alters the apparent root directory of a running process and any children processes. The most important aspect of this is that the process inside the chroot jail cannot access files outside of its new apparent root directory, which is particularly useful in the case of ensuring that a poorly configured or exploited service cannot access anything more than it needs to.

There are two ways in which chroot can be initiated:

The process in question can use the chroot system call and chroot itself voluntarily. Typically, these processes will contain chroot options within their configuration files, most notably allowing the user to set the new apparent root directory.

The chroot wrapper can be used on the command line when executing the command. Typically this would look something like:

sudo chroot /chroot/dir/ /chroot/dir/bin/binary -args

For details of specific chroot syntax for your flavor of Unix, consult man chroot.3

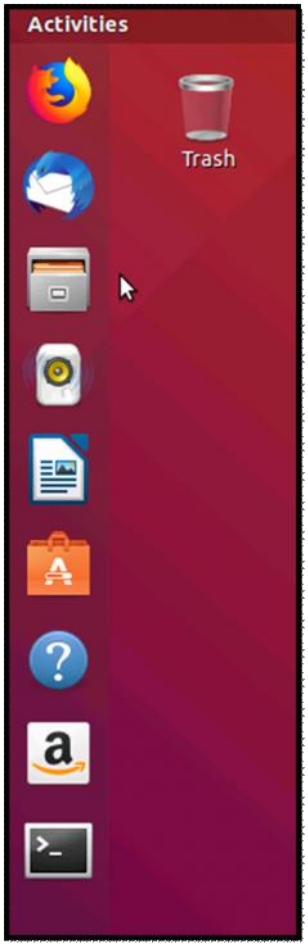
It should be noted, however, that there is a common misconception that chroot offers some security features that it simply does not. Chroot jails are not impossible to break out of, especially if the process within the chroot jail is running with root privileges. Typically processes that are specifically designed to use chroot will drop their root privileges as soon as possible so as to mitigate this risk. Additionally, chroot does not offer the process any protection from privileged users outside of the chroot on the same system.

Neither of these are reasons to abandon chroot, but should be considered when designing use cases as it is not an impenetrable fortress, but more a method of further restricting file system access.

Guided Exercise: Linux File Permissions
7.4 Guided Exercise: Linux File Permissions

Resources	
Files	None
Machines	Ubuntu Server

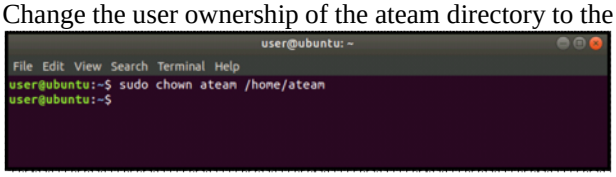
Login to Ubuntu Server with the following credentials:
Username: user
Password: Pa$$w0rd
Once logged in click on the terminal icon (last icon) on the left side menu.



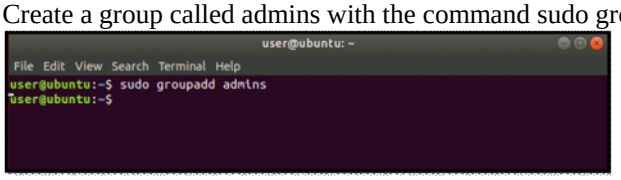
Create a directory in /home called ateam with the command sudo mkdir /home/ateam. When prompted enter the user password and the press the enter button

Create a user called ateam with the command sudo useradd ateam.

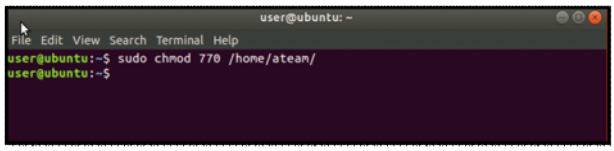
Change the user ownership of the ateam directory to the user ateam with the command sudo chown ateam /home/ateam



Create a group called admins with the command sudo groupadd admins

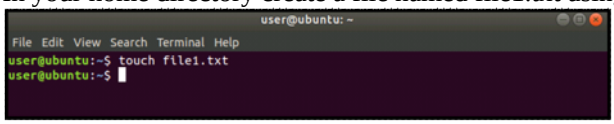


Ensure on the ateam directory the user ateam and group admins to have full permissions. All other users of the system should not have permissions. Use the command sudo chmod 770 /home/ateam.

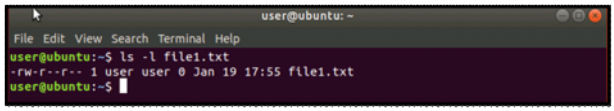


Confirm that you have set the correct permissions using the command sudo ls –ld /home/ateam

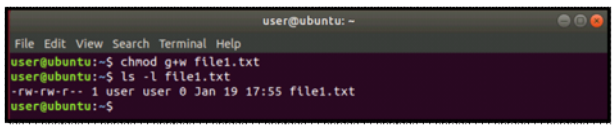
In your home directory create a file named file1.txt using the command touch file1.txt



View the permissions of the file using the command ls –l file1.txt



Give to the group the write permission using the command chmod g+w file1.txt. Then confirm it using the command ls –l file1.txt



Guided Exercise: Disabling Linux Services

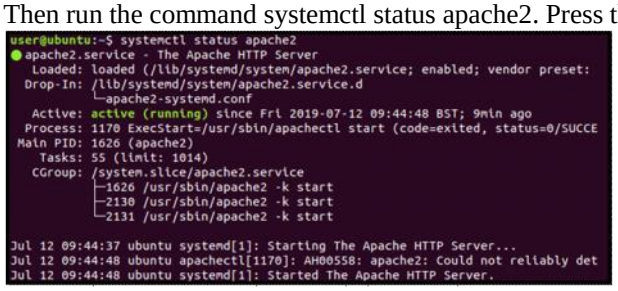
7.5 Guided Exercise: Disabling Linux Services

Resources	
Files	None
Machines	Ubuntu Server

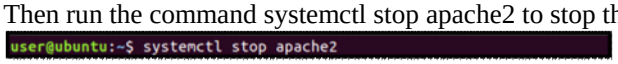
In this exercise, you are required to disable the apache web server service.

On the Ubuntu Server run the command netstat –tulpn to confirm that the webserver is listening on port 80.

Then run the command systemctl status apache2. Press the q button after the command shows the result.



Then run the command systemctl stop apache2 to stop the service. Authentication is required so you need to enter the user password.



Run one more time systemctl status apache2 to confirm that the apache2 service has been stopped.

```
user@ubuntu:~$ systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset:
  Drop-In: /lib/systemd/system/apache2.service.d
           └─apache2-systemd.conf
   Active: inactive (dead) since Fri 2019-07-12 09:55:00 BST; 30s ago
  Process: 2589 ExecStop=/usr/sbin/apachectl stop (code=exited, status=0/SUCCESS
  Process: 1170 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCE
 Main PID: 1626 (code=exited, status=0/SUCCESS)

Jul 12 09:44:37 ubuntu systemd[1]: Starting The Apache HTTP Server...
Jul 12 09:44:48 ubuntu apachectl[1170]: AH00558: apache2: Could not reliably det
Jul 12 09:44:48 ubuntu systemd[1]: Started The Apache HTTP Server.
Jul 12 09:54:50 ubuntu systemd[1]: Stopping The Apache HTTP Server...
Jul 12 09:55:00 ubuntu apachectl[2589]: AH00558: apache2: Could not reliably det
Jul 12 09:55:00 ubuntu systemd[1]: Stopped The Apache HTTP Server.
```

Run the command sudo systemctl disable apache2 to disable the webserver. The command requires the user password and enter Pa $$w0rd. That command will prevent the webserver from starting during boot time.

```
user@ubuntu:~$ sudo systemctl disable apache2
Synchronizing state of apache2.service with SysV service script with /lib/system
d/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable apache2
```

Run the command netstat –tulpn to confirm that port 80 is not listening for connections.

```
user@ubuntu:~$ netstat -tulpn
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN
-
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
-
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
-
tcp6       0      0 :::22                   :::*                    LISTEN
-
tcp6       0      0 ::1:631                 :::*                    LISTEN
```

Operating System Patches

7.6 Operating System Patches

From time to time, security flaws are found in operating systems. As software vendors become aware of flaws, they usually write corrections to their code, known as patches or updates. Whatever operating system you use, you must apply these patches as a matter of routine.

Windows patches are probably the most well-known, but patches can be released for any operating system. You should patch your system any time a critical patch is released. You might consider scheduling a specific time simply to update patches. Some organisations find that updating once per quarter or even once per month is necessary.

7.6.1 Applying Patches

Applying patches means that the operating system, database management systems, development tools, Internet browsers, and so on are all checked for patches. In a Microsoft environment this should be easy because the Microsoft website has a utility that scans your system for any required patches to the browser, operating system, or office products. It is a very basic tenet of security to ensure that all patches are up-to-date.

This should be one of your first tasks when assessing a system. Regardless of the operating system or application vendor, you should be able to go to its website and find information regarding how to download and install the latest patches. But remember that everything must be patched—the operating system, applications, drivers, network equipment (switches, routers, etc.), literally everything.

Once you have ensured that all patches are up to date, the next step is to set up a system to ensure that they are kept up to date. One simple method is to initiate a periodic patch review where, at a scheduled time, all machines are checked for patches. There are also automated solutions that will patch all systems in your organisation. It is imperative that all machines be patched, not just the servers.

7.6.2 Automated Patch Systems

Manually patching machines can be quite cumbersome, and in larger networks, simply impractical. However, there are automated solutions that will patch all systems on your network. These solutions scan your systems at pre-set times and update any required patches.

7.6.3 Windows Update

For systems running Microsoft Windows, you can set up Windows to automatically patch your system. Recent versions of Windows have this turned on automatically. If your system is older, simply go to https://support.microsoft.com/en-us/help/12373/windows-update-faq and follow the instructions to keep your system updated. This will give that individual machine routing updates for the Windows operating system.

This approach does have a few shortcomings, the first being that it will only update Windows and not any other applications on your machine. The second drawback is that it does not provide any way to check patches on a test machine before deploying them to the entire network. Its main advantages are that it is free, and integrated with the Windows operating system.

Surprisingly, another commonly overlooked protection is some type of software update platform. Windows Server Update Services (WSUS), System Centre Configuration Manager (SCCM), and other third-party applications can keep the endpoints up-to-date with the latest security patches. Not only should you worry about regular Windows system patches, but there should also be a focus on outdated versions of commonly exploited software such as Java, Adobe Reader, Firefox, and others that are currently in use.

7.6.4 Unix/Linux Software Updates

Unlike Microsoft environments, Unix-based environments typically use a system of package management to install the majority of third-party applications.

Package management and update tools vary depending not only on which flavor of Unix you are running, but also differ depending on distribution you use. For example, Debian Linux and SUSE Linux use two different package management systems, and FreeBSD uses another.

Despite the differences, there are common themes surrounding the package management systems. Typically, each host will hold a repository of packages that are available to install on the system via local tools. The system administrator issues commands to the package management system to indicate that she wishes to install, update, or remove packages. The package management system will, depending on configuration, either download and compile, or download a binary of the desired package and its dependencies (libraries and other applications required to run the desired application), and install them on the system.

The various package management systems are so comprehensive in a modern distribution that for many environments it would be unusual to require anything further. Deploying software via package management, as opposed to downloading from elsewhere, is the preference unless there is a compelling reason to do otherwise. This greatly simplifies the issue of staying up-to-date and tracking dependencies.

The same package management system can be used to perform upgrades. As the repository of available packages is updated, new versions of already installed packages appear in the package database. These new version numbers can be compared against the installed version numbers and a list of applications due for an upgrade to a new version can determined automatically, typically via a single command line.

This ease of upgrade using package management means that unless a robust system of checking for and applying changes is in place for installed applications, the package management system should be used to provide an easy, automated method of updating all packages on UNIX application servers.

Not only does this remove the need to manually track each application installed on the application servers, along with all their associated dependencies, but it (typically) means that it has already been tested and confirmed to work on that distribution. Of course, individual quirks between systems mean that you cannot be sure that everything will always work smoothly, and so the testing process should remain. However, the testing process may be entered with a good degree of confidence.

7.6.4.1 Core Operating System Updates

Many, but not all, UNIX systems have a delineation between the operating system and applications that are installed on it. As such, the method of keeping the operating system itself up-to-date will often differ from that of the applications. The method of upgrading will vary from operating system to operating system, but the upgrade methods fall into two broad buckets:

Binary update

Commercial operating systems particularly favour the method of applying a binary update; that is, distributing precompiled binary executables and libraries that are copied to disk, replacing the previous versions. Binary updates cannot make use of custom compiler options and make assumptions about dependencies, but they require less work in general and are fast to install.

Update from source

Many open source operating systems favour updates from source, meaning that they are compiled locally from a copy of the source code and previous versions on disk are replaced by these binaries. Updating from source takes more time and is more complex, however the operating system can include custom compiler optimizations and patches.

There are many debates over which system is better, and each has its pros and cons. For the purposes of this book, however, we will assume that you are sticking with the default of your operating system as the majority of arguments centre on topics unrelated to security.

Updates to the operating system are typically less frequent than updates to third-party software. Additionally, they are more disruptive, as they typically require a reboot because they often involve an update to the kernel or other subsystems that only load at startup, unlike application updates, which can be instantiated via the restart of the appropriate daemon. Core operating updates are advisable, though as vulnerabilities are often found within both operating systems and applications.

As with any other patch of this nature, it is advisable to have a rollback plan in place for any large update such as one for an operating system. In the case of virtualized infrastructure, this can be achieved simply by taking a snapshot of the file system prior to upgrade; thus a failed upgrade can be simply rolled back by reverting to the last snapshot. In physical infrastructure this can be more problematic, but most operating systems have mechanisms to cope with this issue, typically by storing a copy of the old binaries and replacing them if required.

Nevertheless, patches to the operating system are often required in order to close security gaps, so you should have a process defined to cope with this. As with applications, the effort to upgrade the operating system is lower the more up-to-date a system already is, so we recommend remaining as current as is reasonable, leaving only small increments to update at any one time.

Quiz:

- 1.What level of privileges all users must have?
- 2.A Linux system has a repository of packages available to be installed on the system
- 3.What account lockout threshold does the NSA recommends?
- 4.What is the rule for unused services on any computer?
- 5.What operating system require periodic patches?
- 6.What type of encryption does EFS utilize?
- 7.What maximum password age does Microsoft recommends?
- 8.Which of the following best describes the registry
- 9.What minimum password length does the NSA recommends?
- 10.The command sudo find / -perm -4000 checks for the location of suid binaries