OS Chapter – 8

"Security & Protection"

1. **Introduction**
2. **Security Goals**
3. **Intruders/Attackers**
4. **User Authentication**
5. **Protection Mechanism:** **(Protection Domain, Access Control List)**
6. **Malware**

1)Introduction

- Many companies possess valuable information they want to guard closely.
- Among many things, this information can be technical (e.g., a new chip design or software), commercial (e.g., studies of the competition or marketing plans), financial (e.g., plans for a stock offering) or legal.
- Most of this information is stored on computers. Home computers increasingly have valuable data on them, too.
- Hard disks these days are full of important financial information, including tax returns and credit card numbers, photos, videos, and movies on their computer.
- The need to protect this information is becoming increasingly important.
- Guarding the information against unauthorized usage is therefore a major concern of all operating systems.

2)Security Goals

- Security can have three different goals:
    1) Confidentiality
    2) Integrity
    3) Availability

1) Confidentiality

- **Confidentiality** is concerned with having **"secret data remain secret"**.
- More specifically, if the owner of some data has decided that these data are to be made available only to certain people and no others, the system should guarantee that the data should not be exposed to some unauthorized people.
- The owner should be able to specify who can see what, and the system should enforce these specifications.
- Threat to confidentiality : Exposure of data

2) Integrity

- **Integrity** means Data should not be modified by someone without owner's permission.
- In short, data must remain unchanged until owner wants to change it.
- Data modification includes not only changing the data, but also removing data and adding false data.
- No unauthorized change should be allowed on data.
- Threat to Integrity : Tampering / Modifying data

3) Availability

- **Availability**, means System should be available whenever required.
- One attack possible on availability is DoS (Denial of Service)
- For example, if a computer is sending so many fake requests(flood of requests) to a server, then server will be busy in handling the fake requests and at a point the server will be saturated.
- As a result it will not be able to serve genuine clients.
- So, actual / genuine clients will be denied from getting the services, so it is called Denial of Services.
- Threat to Availability : DoS

3) Intruders/Attacker

- A person who breaks any security goals and harms the system is known as intruder.
- Intruder/Attackers are of two types:
- 1) Passive Attacker
- 2) Active Attacker

1) Passive Attacker

- The attacker who only observes the system and doesn't perform any malicious activity is known as passive attacker.
- Passive attack means unauthorized party monitors system with the only purpose to gain information about the system.
- Passive attackers do not change the data.
- The attacks caused by passive attacker are known as passive attacks.

2) Active Attacker

- The attacker who doesn't only observe the system but also perform some malicious activities in known as active attacker.
- Active attackers change the data.

- The attacks caused by active attacker are known as active attacks.

4) User Authentication

- Proving the identity means user authentication.
- Identity of the user can be given at the time of login in machine.
- There are 3 ways of user authentication :
  1. Something the user knows.
  2. Something the user has.
  3. Something the user is.

1) Something the user knows (Password)

- Password is the first and the simplest strategy for authentication.
- Password should have following desirable properties:
  1) It should have minimum few characters in length.
  2) It should be the combination of alphabets as well as special characters.
  3) It should not be guessable password.
  4) It should not be related with personal details of any person.
- One suggestion to keep Password:
- Ex. Decide a sentence: I love OS subject.
- Possible Password: IlOs.
- OTP : One Time Password
- So many e-banking sites use OTPs for securing transaction.
- OTP, as the name indicates is used only once.
- Once it is used for one transaction, it can't be used for another transaction.
- It is used to achieve security.
- How the passwords are stored in OS?
- OS doesn't store the password in original form.
- It stores the password in encryption form.

2) Something the user has (Cards)

- Cards can also be used as user authentication.
- Advantages :

  1) Simple Authentication
- Disadvantages :

  1) Card can be stolen.

  2) Using card needs special hardware (Card Reared).

3) Something the user is (Biometrics)

- This is a widely used approach nowadays.
- Examples :

  1) Finger Print Reorganization

  2) Retina Reorganization

  3) Face Reorganization

  4) Hand writing etc.
- Advantages :

  1) The most accurate technique for authentication.
- Disadvantages :

  1) This is the costliest technique.

  2) Requires special hardware for implementation.

5) Protection Mechanism

- In some systems protection is enforced by the program called "reference monitor".
- Every time an access to a resource is attempted, the system first asks the reference monitor to check the validity for it.
- The reference monitor then looks at its policy and makes a decision.
- There are two protection mechanisms.

  1) Protection Domain

  2) Access Control List (ACL)

1) Protection Domain

- A **domain** is a set of pairs (object, rights).
- Each pair specifies an object and some subset of the operations that can be performed on it.
- Example of a domain :

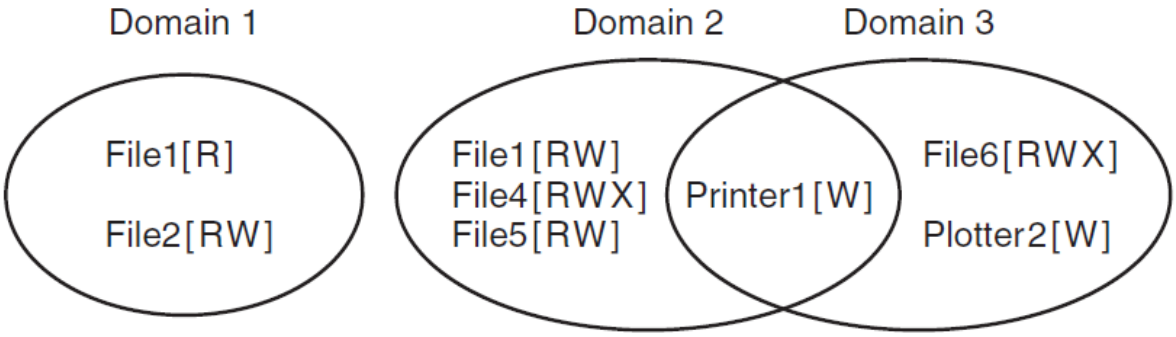

Figure: Protection Domain

- Figure shows three domains, showing the objects in each domain and the rights (Read, Write, eXecute) available on each object.
- Note that *Printer1* is in two domains at the same time, with the same rights in each.
- *File1* is also in two domains, with different rights in each one.
- Domain1 represents the rights given to all the users in Domain 1.
- Domain2 represents the rights given to all the users in Domain 2.
- Domain3 represents the rights given to all the users in Domain 3.
- Now we can implement domain concept using protection matrix as follows :

Domain	Object							
	File1	File2	File3	File4	File5	File6	Printer1	Plotter2
1	Read	Read Write						
2			Read	Read Write Execute	Read Write		Write	
3						Read Write Execute	Write	Write

Figure : Protection Matrix

- The matrix has **n * m** dimensions ,
  where **n** is no. of domains and
  m is no. of files.
- Problem with this matrix is that majority of the places in the matrix will be empty because normal users have limited rights on the limited files.
- Thus, memory is wasted.

2) Access Control List (ACL)

- This technique overcomes the problem occurring in the previous method.
- Let's assume that users A and B are associated with domain 1 & User C is associated with domain 2.
- "The technique consists of associating with each object an ordered list containing all the domains that may access the object, and how. This list is called the **ACL** (**Access Control List**)."

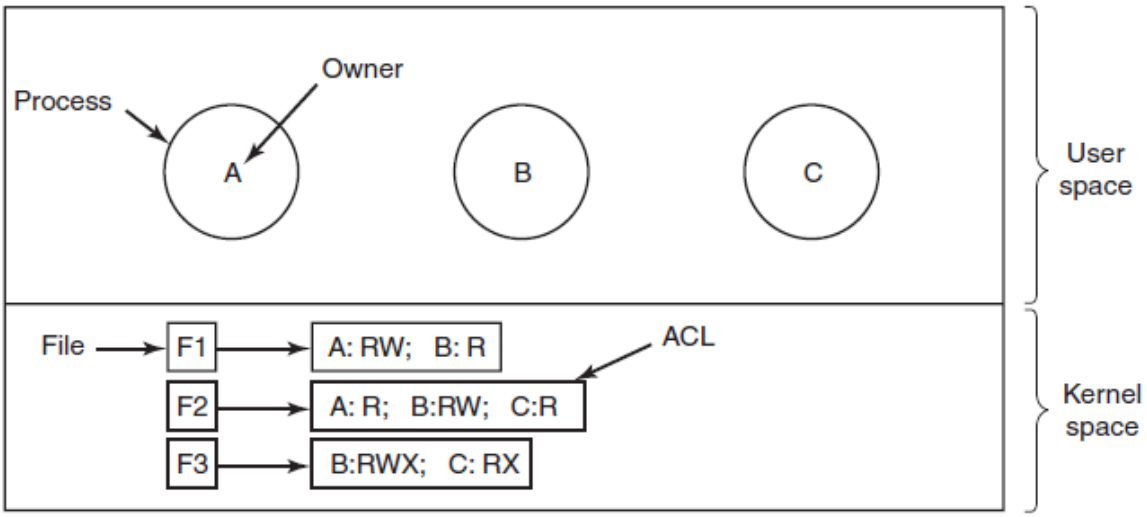

Figure: Use of access control lists to manage file access.

- This example illustrates the most basic form of protection with ACLs.
- Here, ACL contains 3 lists.
- In general it contains **n** no. of lists, where n is no. of files.
- This approach doesn't waste memory because it contains only entries which exists, so no memory wastage.
- More sophisticated systems are often used in practice.
- We have shown only three rights so far: **read, write, and execute**. There may be additional rights as well.
- Some of these may be generic, that is, applicable to all objects, and some may be object specific, that is, applicable to only few objects.
- Examples of generic rights are **destroying object** and **copy object**.
- Object-specific rights might include **append message** for a mailbox object and **sort alphabetically** for a directory object.

6) Malware

- In ancient times (say, before 2000), bored (but clever) teenagers would sometimes fill their idle hours by writing malicious software that they would then release into the world for the heck of it.
- **"**This software, which included Trojan horses, viruses, and worms are collectively called **malware"**.
- Malwares are often quickly spread around the world.

Trojan Horse

- The beauty of the Trojan horse attack is that it does not require the author of the Trojan horse to break into the victim's computer. The victim does all the work.
- The Trojan horse, at first glance will appear to be useful software but will actually do damage once installed or run on your computer.
- Unlike viruses and worms, Trojans do not reproduce by infecting other files nor do they self-replicate. Means Trojan horse viruses differ from other computer viruses in that they are not designed to spread themselves.
- For example, A user *smit* on a particular system might consist of the following
  directories:
  :/usr/smit/bin
- When the user types **prog** to the shell, the shell first checks to see if there is a program at the location */usr/smit/bin/prog*.
- **If there is, it is executed**. **If it is not there, the shell tries**
- ***/usr/local/bin/prog*,**
- ***/usr/bin/prog*,**
- ***/bin/prog*, and so on,**
- **trying many directories in turn before giving up.**
- Suppose that just one of these directories was left **unprotected** and a cracker put a program there.

- If this is the first occurrence of the program in the list, it will be executed and the Trojan horse will run.
- Most common programs are in */bin* or */usr/bin*, so putting a Trojan horse in */usr/bin/X11/ls* does not work for a common program because the real one will be found first.
- However, suppose the cracker inserts ***la*** into */usr/bin/X11*.
- If a user mistypes *la* instead of *ls* (the directory listing program), the Trojan horse will run, do its dirty work, and then issue the correct message that *la* does not exist.
- By inserting Trojan horses into complicated directories and giving them names that could represent common typing errors, there is a fair chance that someone will invoke one of them sooner or later.
- One of the most destructive type of Trojan horse program claims to remove virus from your machine but instead of removing it installs virus in the machine itself.
- Trojan Horse is not easily detectable.
- Trojan Horse may cause the computer to run slowly.
- Trojan Horse may give hacker, remote access to a targeted computer system.
- Following operations may take place :
    1) Crashing the Machine
    2) Data corruption
    3) Destroying all contents
    4) Electronic money theft
    5) Data theft
    6) Modification or deletion of the files
    7) Controlling computer system remotely.
    8) Modification of registry.

Trap door:

- It is also known as backdoor.
- The designer of a program or system might leave a hole in the software that only the designer is capable of using. This type of security breach is known as **trap door.**
- For instance, the code might check for a specific user ID or password, and it might avoid normal security procedures.
- Programmers have been arrested for cheating banks by including **rounding errors** in their code and having the **occasional half-cent credited** to their accounts.
- A clever trap door could be included in a compiler. The compiler could generate standard object code as well as a trap door, regardless of the source code being compiled.
- Trap doors are difficult to detect because we have to analyze all the source code for all components of a system.
- A trap door is a secret entry point into a program that allows someone to gain access without going through the usual security access procedures.
- We can also say that it is a method of bypassing normal authentication methods.
- Trap doors have been used legally for many years by programmers to debug and test programs.
- Trapdoors are written in the programs. Once written they can be used to perform various malicious activities.
- Trapdoor might be a hidden program which makes the protection system ineffective.
- Trapdoors are not always used for wrong purposes.
- Trap doors become threats when they are used by dishonest programmers to gain unauthorized access.

Virus:

- "A **virus** is a program that can reproduce itself by attaching its code to another program, analogous to how biological viruses reproduce".
- **"**A computer virus is a program or piece of code that is loaded onto your computer without your knowledge and runs against your wishes**"**.
- Viruses can increase their chances of spreading to other computers by **infecting files** on a network file system or a file system that is accessed by other computers.
- Viruses can also replicate themselves.
- All computer viruses are manmade.
- A simple virus that can make a copy of itself over and over again is relatively easy to produce. Even such a simple virus is dangerous because it will quickly use all available memory and bring the system to a halt. An even more dangerous type of virus is one capable of transmitting itself across networks and bypassing security systems.

Worm:

- **"**A worm is a program which spreads usually over network connections.**"**
- Unlike a virus which attaches itself to a host program, worms always need a host program to spread.
- In practice, worms are not normally associated with one personal computer systems.
- They are mostly found in multi-user systems such as UNIX environments.