Protection and Security

Protection

- A **protection-oriented system** provides a **means to distinguish between authorized and unauthorized usage**.
- **Protection** is any **mechanism for controlling the access of processes** or users **to the resources defined by a computer system**. This mechanism must provide means to specify the controls to be imposed and to enforce the controls.
- If a computer system has **multiple users** and allows the **concurrent execution of multiple processes**, then **access to data must be regulated**.
- For that purpose, mechanisms ensure that **files, memory segments, CPU**, and other resources can be **operated on by only those processes** that have **gained proper authorization from the operating system**.
- **Memory-addressing hardware** ensures that a **process can execute only within its own address space.**
- The **timer ensures** that **no process can gain control of the CPU** without eventually relinquishing control.
- **Device-control registers** are **not accessible to users**, so the integrity of the various peripheral devices is protected.

Security

- A system can have **adequate protection but still be prone to failure** and allow inappropriate access.
- Consider a user whose **authentication information** (her means of identifying herself to the system) is **stolen**. Her data could be copied or deleted, even though file and memory protection are working.
- It is the job of **security to defend a system** from **external and internal attacks**.
- Such **attacks** spread across a huge range and **include viruses** and **worms**, **denial-ofservice attacks** (which use all of a system's resources and so keep legitimate users out of the system), **identity theft**, and **theft of service** (unauthorized use of a system).
- **Prevention** of some of **these attacks** is considered an **operating-system function on some systems**, while **other systems leave it to** policy or **additional software**.

- **Protection and security** require the system to be able to distinguish among all its users.
- Most **operating systems** maintain a **list of user names** and associated user identifiers (**user IDs**).
- These numerical IDs are unique, one per user.
- When a user logs in to the system, the authentication stage determines the appropriate user ID for the user.
- That **user ID is associated** with all of the **user's processes** and **threads**.
- When an ID needs to be readable by a user, it is translated back to the user name via the user name list.
- In some circumstances, we wish to distinguish among sets of users rather than individual users.
- For example, the owner of a file on a UNIX system may be allowed to issue all operations on that file, whereas a selected set of users may be allowed only to read the file.
- To accomplish this, we need to define a group name and the set of users belonging to that group.
- Group functionality can be implemented as a system-wide list of group names and group identifiers.

Protection and Security Mechanisms

1. Dual Mode Operation
2. Registers to limit process in own address space
3. I/O Instructions are kept in supervisory mode
4. Timer to protect monopolising the system