Protection & Security

Multiple processes coexist in single system but they must not interfere in each others address or activities or resources

Protection Problem Goals

- Computers consists of a collection of Objects which can be h/w or s/w.
- Each object has a unique name & is accessed through well defined set of operation
- Protection Problem ensures that each object is accessed correctly & only by those process which are allowed/ authorized to do so

Categories of Protection Violation

→ Breach of Confidentiality → unauthorized read
→ " " Integrity → " write
→ " " Availability → " destruction/ deletion

→ Theft of service → " use of resources
→ Denial of service → Prevention of legitimate use

(Confidentiality, Integrity, Availability — (CIA triad)

Principle of Least Priviledge :-

- Programs, users, systems - all are given just enough / sufficient priviledges so that their tasks are performed smoothly
(Eg ous SAP, director, faculty, student each have limited yet sufficient for them rights)

- This reduces damage in case of failures or attacks

- Priviledges granted can be static or constant ~~and~~ i.e remain same throughout the lifetime of a ~~system~~ system or process
It can also be dynamic i.e changed as per process needs.
↳ This is called domain switching / priviledge escalation.

Domain Structure

Domain is basically a set of access rights.
↳ i.e a set of all valid operations that can be performed in a particular object.

Format : < object name, set of rights >
Eg: < O₁ , { read, write } >
↳ ∴ O₁ can be read & write
< O₂ , { execute, read } >
↳ ∴ O₂ can't be written.

Access matrix & its use

- It provides a view into protection scenario as a matrix.
- Rows = domains
- Columns = Objects
- Access (i, j) i.e an element in the matrix is the set of operations that a process executing in Domain i can invoke / operate / perform on Objects j.

Object / domain	O_1	O_2	printer
D_1	read	write	print
D_2			print
D_3		execute	

∴ Process executing in Domain 2 can only print whereas in Domain 1 it can read Object 1, write Object 2 & also print.

∴ If a process in D_i want to do some operation X on Object j, thus this must be mentioned in the access matrix. If its not mentioned, the operation can't be done.

Disadv: Infeasible if no. of process or objects increase

Security Problem :-

- Security is maintained if resources are used and accessed as intended under all circumstances.
- Intruders / crackers try to breach security — Attack
- Threat is a potential security violation
- Attacks are accidental / malicious & are easy to be protected from. Diagnosis of Threat may be difficult. Malicious / Planned Attacks are most dangerous.

Security Violation

→ Masquerading / Acting / Pretending to be someone else to escalate priviledges. → it ∴ breaches authenticati

→ Replay Attack — Resend exact / modified messages

→ Man-in-the-middle Attack — Overhear conversation by sitting on data flow & acting as sender / receiver

→ Session hijacking — Intercept an established connection to bypass from it.

Security Measure Levels

Increase cost of penetration to deter most intruders.

Security must occur at 4 levels

→ Physical : Data centres, servers etc

→ Human : Avoid phishing

→ OS : Protection mechanism / debugging

→ N/W : Interrupt communication must be avoided

(Hopefully you know what is a virus, worm, trojan horse, firewall.)