Lecture Notes

There are two ways to address security issues:
- Full Disclosure:- Announce to the public e.g. Treatises on Locks To force the developers to take action
- Inform the Developer Only:- Inform the developers and wait for them to take action

- Different levels of trust decides implementation level of security
  - e.g Sony BMG CD Scandal

- Sometimes trust is “transitive”

“If you delete an edge in a trust diagram, you ruin the trust chain”

Need Trust ?
- Write code yourself. (non-practical in large projects)
- Check code by multiple people, peer review
- Cross - check with other code, replicate if needed.
- Develop a Trusted Authority for software
- Use multiple compilers to compile same code (Somewhat Complex for Big projects)
- Minimize the TCB size, simple code is easier to debug and review
Core secure system Design Principles

- Keep it small (a small TCB is preferred)

- “Economy of Mechanism” - Keep it simple. A line of code deleted is risk reduced.

- Introduce Fail-Safe Defaults in the system
  Default DENY - Fails loudly
  Default ALLOW - Fails Silently
  - e.g C:\ had RW access to * in Win 3.1. Fixed

- Have Least Privilege settings
  Customer Web Server (Less) ----> Amazon DB <------- Employee Web Server (High )

- Introduce Complete Mediation
  - To reduce risk, reduces access paths.

- Encourage separation of privileges
  - Using two compilers, have modular architecture
  - Have two people to initiate missile launch
  - Example: Mail Server

Issues:-
- If there is a vulnerability, the complete mail server will be compromised

Solution:-
- Split the functionality

- The Validator is the TCB
Alternatives
- What if we move the validator to the Receiver?
- The old Validator Receiver now can work as a router
- The Validator now also serves the function of receiver

Privileges of
Validator (Listen, Queue)
Router (De-queue, Queue, Deliver)
Forwarder (De-queue, Send)

- Least shared Mechanism
  - Least shared state: Have many instances of the same software and have them monitor different states. We can have many instances of oracle having different Database i.e Customer, Employees etc
  - Separate account systems of user registration systems

- Psychological Acceptability
  - Sometimes, making access restricted encourages people to experiment. E.g Firewalls in corporate networks.

Tip - “Criticize papers based on above principles”