Lecture notes for 5.1-5.3

Trusted software
- Functional correctness [does what it was designed to do]
- Maintains data integrity [even for bad input]
- Minimizes access to secure data [does not pass to untrusted software]
- Confidence [experts analyze program & give trust]

Secure vs. Trusted
- Secure: boolean property, almost never true
- Trusted: scalar property

Trusted computing base (TCB)
- Set of all hardware and software trusted to operate securely
- Required for all other trust in the system

Security policy
- Statement giving security we expect system to enforce

Military policy
- Compartment
  - Separation of data into categories
- Sensitivity level
  - TS > SEC > CONF > RESTR > UNCLASS
  - Division within each compartment (has same meaning in every compartment)
- To access, must first be member of compartment, and second be at or above sensitivity level

Separation of duty
- Critical steps (ordering goods, receiving goods, paying for goods) done by separate people
- Prevents corruption

Chinese Wall security policy

- Objects assigned to companies
- Companies assigned to conflict classes
  - Companies in same class are competitors
- Temporal access policy
  - Access to object allowed only if user has never accessed object for different company in same conflict class
  - Once access object, all objects for conflicting companies can never be accessed

Lattice

- Sensitivities + ordering relation
- [Use military sensitivity example from above] [straight-line lattice]

Bell - La Padula confidentiality model

- [draw high confid above low confid with line between]
- [No read up] Subject s can read object o only if C(s) \geq C(o)
- [No write down] Subject s reading object o can write object p only if C(p) \geq C(o)

Biba integrity model

- [continue drawing]
- [No write up] Subject s can write object o only if I(s) \geq I(o)
- [No read down] Subject s reading object o can write object p only if I(o) \geq I(p)
  - [This statement is actually about allowing read down only if writing down... no read down and writing at higher integrity level]

Harrison-Ruzzo-Ullman

- Defines primitive operations
  - Object creation
  - Insertion of access right into matrix
  - [4 more]
- Commands are one or more operations
- Q: Can a subject S ever gain right R to object O?
  - Decidable IFF every command is a single operation
- Unix requires more than one operation per command, cannot generally determine if subject can gain access to object