

Introduction to Computer Security

Instructor:

Mahadevan Gomathisankaran

mgomathi@unt.edu

- Identifying you are who you claim your are
- Basic issue: How do you identify subjects?
 - Absolute authentication vs. continuity of authentication
 - Identity typically relative to a security *domain*
- Authentication typically based on:
 - Something you know– password, etc.
 - Something you have – badge, key, etc.
 - Something you are – biometrics
- Multi-factor authentication combines multiple techniques
- Policy issues:
 - Employees/staff must understand authentication issues
 - Awareness activities to remind regularly

- Physical security: password/passphrase
- Passwords or PINS
 - Either assigned by the user or by the system
 - Widespread (no special hardware/readers)
 - Important to have strong password policy

Password Threats and Controls

- Initial passwords (defaults bad!)
 - Use random initial passwords and/or 1st-time use change
- Discovered/sniffed passwords
 - Expire passwords (force change) and use encryption
 - Use different passwords for different systems!
- Forgotten passwords
 - “Security question” or e-mail to valid address
- Small search space (PINs or easily guessed)
 - “Complexity requirements” on password (brute force attack – but also dictionary as in John the Ripper, SATAN, etc.)
- Trojan horse login programs!
 - “Trusted path” to mitigate

Problems with Passwords

- Loss
 - Deactivation
 - Replacing the old password
- Use
 - Each time you access a file you use a password – impractical
- Disclosure: If password disclosed to unauthorized 3rd parties, file protection is no longer valid. Change the password and inform other users a long process
- Revocation: Involves changing the password. SO, same problems with disclosure.

Attacks on Passwords

- Try all possible
- Try many probable
- Try likely passwords
- Search system lists
- Ask users

Exhaustive Attacks

- Brute force
- An 8 character password known to use alphabetic characters will take

$$26^1 + 26^2 + \dots + 26^8 = 5 \text{ million million trials}$$

- 150 year attack!!
- Alternatively
 - Half of the search space will be sufficient to reach the correct password
 - Furthermore, users tend to pick easy to remember passwords, which narrows the search space more

- People pick predictable passwords!
 - Dictionary words
 - Family names, pet names, etc.
- 86% of passwords are uncovered within a week's time brute force trial

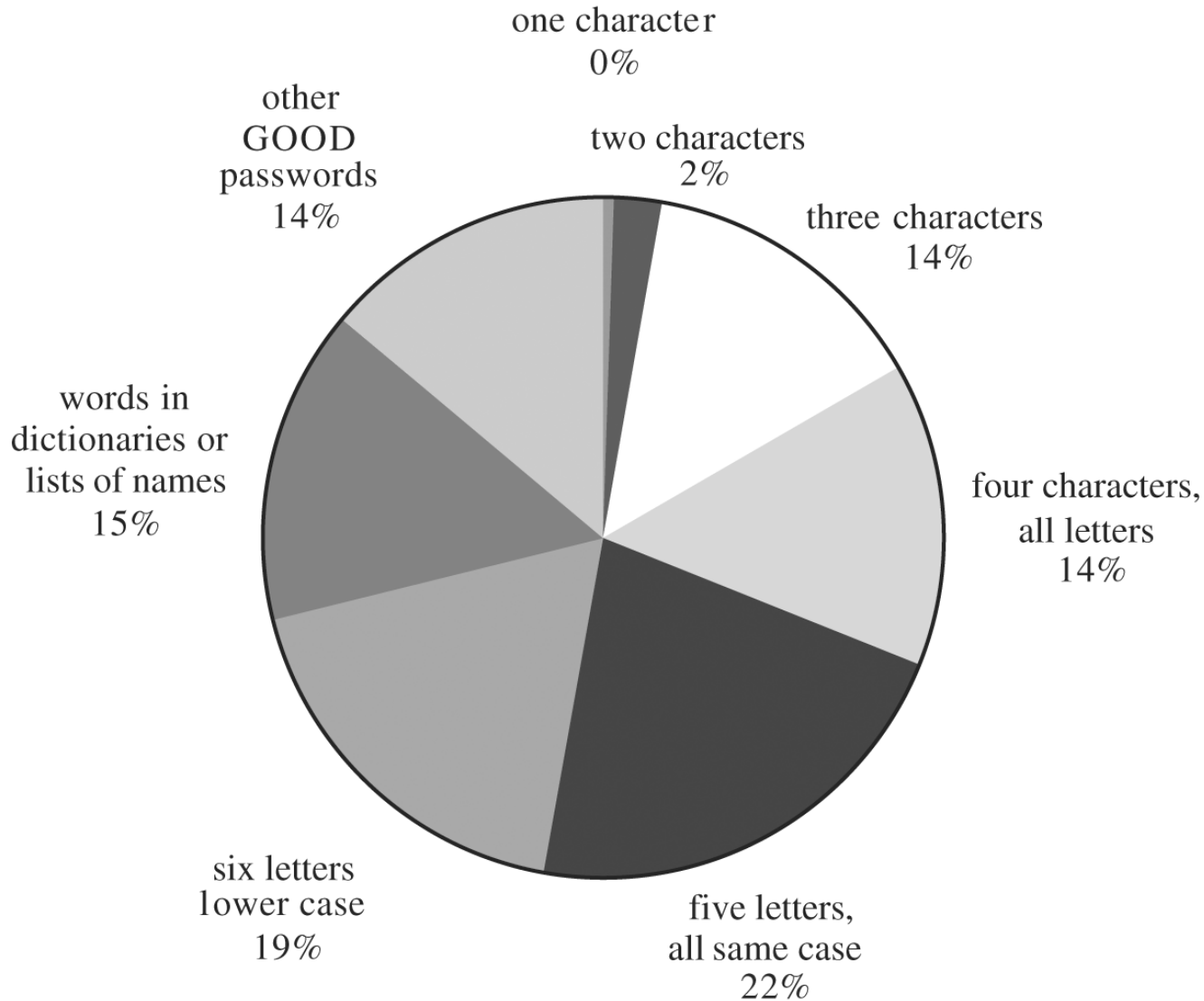
Passwords Likely for a User

- No password at all!!
- Same as user ID
- Derived from user name
- Common word list (“password”, “asdfg”, etc.)
- Dictionary words
- Complete English word list
- Common non-English language dictionaries
- Short college dictionary with capitalization (PaSsWorD) and substitutions (o for O)
- Complete English with capitalization and substitutions
- Common non-English dictionaries with capitalization and substitutions
- Brute force, lowercase alphabetic characters
- Brute force, full character set

Password Protection

- Encrypted passwords
 - Still, password is available as plaintext in memory!
 - Use one-way encryption: When user enters password, it is encrypted, then stored
- Unix salt value:
 - 12-bit number generated using sys time and pid
 - Store $E(\text{password}_{\text{userA}} || \text{salt}_{\text{userA}})$

User's Password Choices



Password Selection Criteria

- Use characters other than just A – Z
- Choose long passwords
- Avoid actual names or words
- Choose an unlikely password
- Change password regularly
- Don't write it down
- Don't tell anyone else

- Limit # of attempts
- Fixing flaws
 - Challenge-Response: Ask a question to be replied before allowing login in attempt
 - Impersonation of login:
 - A login scenario is very easy to pretend! Trojan horse attack
 - Use Ctrl+Alt+Delete for each login
 - Check last login time/date

Something You Have

- Physical: Keys, cards, etc.
- Electronic: Mag stripe card, smart card, tokens, ...
 - Capabilities range from simple storage to executing cryptographic protocols
- Note: Authenticating token, not person
- Multifactor authentication important!

Example 1: Dallas Semiconductor iButton

- Many models
 - Simple storage
 - 64-bit unique ID
 - Read-only or R/W
 - SHA-based
 - Challenge-response
 - Java/Cryptographic
 - 1024-bit RSA in <1 sec
 - Private key in tamper-resistant iButton only



Example 2: RSA SecureID

- Based on a token assigned to a user in the type of
 - h/w (e.g. a token or USB) or
 - s/w (e.g. a "soft token" for a PDA or cell phone)
 - generates an authentication code at fixed intervals (e.g. 30- 60 sec.) using a built-in clock and a 128 bits long seed
 - The seed is different for each token, and is loaded into the corresponding RSA SecureID server as the tokens are purchased.

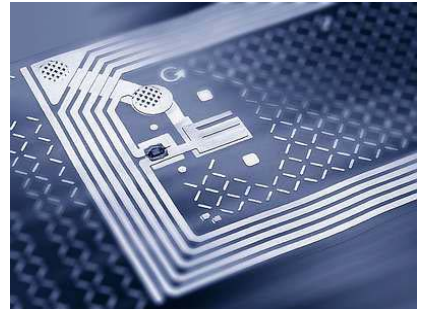


Example 2: RSA SecureID – cont'd

- No special reader needed
 - (but must type numbers)
- Values come from hash-function pseudo-random generator
- Top model hashes PIN with changing code

Example 3: RFID

- Relies on storing and remotely retrieving data using devices called RFID tags (transponders).
- Contains 2 parts
 - An IC
 - for storing and processing information,
 - modulating and demodulating an RF signal
 - for other specialized functions
 - An antenna for receiving and transmitting the signal.



Example 3: RFID

- 3 types:
 - Passive: require no internal power source. Active only when a reader is nearby to power them
 - Active: require a power source, usually a small battery.
 - Semi-passive (battery-assisted)



- Physical: Personal knowledge, handwriting/signature, apparent age, ...
- Biometrics: Image, fingerprints, retina scans, voice prints, DNA, ...
- Adoption getting widespread:
 - In 2000 companies spend \$127 million on biometric systems – 40% on fingerprint IDs
- Problems:
 - Ways around many techniques
 - False positives
 - False negatives

Biometric Examples



Fingerprint scanner



Hand-Geometry



Retinal scanner

Trusted System

- A system is trusted if it “meets the intended security requirements, is of high enough quality, and justifies the user's confidence.”
- Reflects user's perception, not the developer's view
- Different from security

Secure	Trusted
Either-or: Secure	Graded- degrees of “trustworthiness”
Property of presenter	Property of receiver
Asserted based on product characteristics	Judged based on evidence and analysis
Absolute: not qualified as to how, where, when or by whom	Relative: viewed in context of use
A goal	A characteristic

Security Policy and Models

- A security policy is a clear, consistent, and effective “set of rules that lay out what is to be secured and why.”
 - Policies can be formal specifications or English
 - Guidelines or enforced by the system
 - Example: Password policies

- A security model is a formal representation of policy.

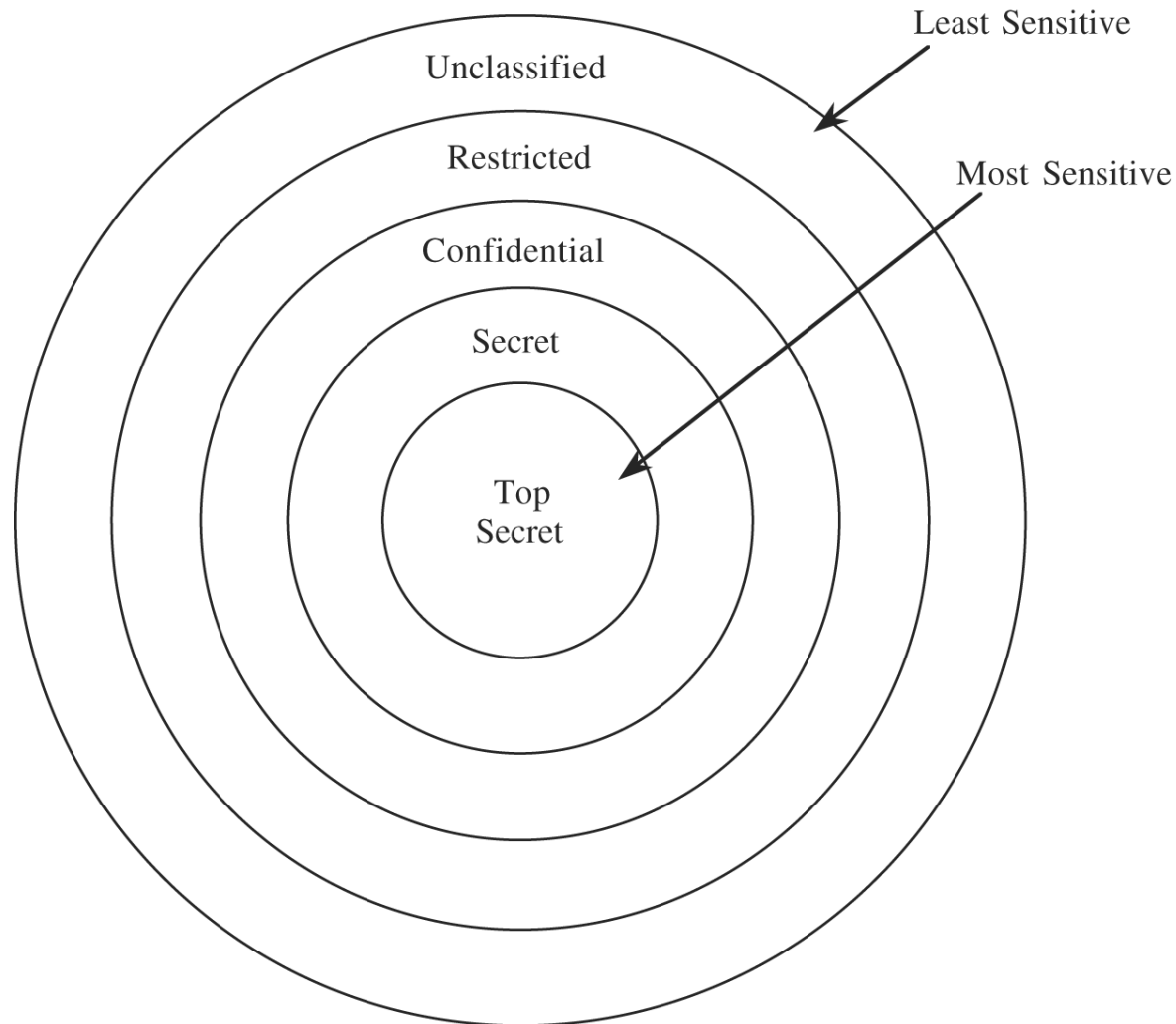
- Why use models?
 - They give a concrete mental picture of policy
 - Should be formal enough to allow analysis, proofs, and automated tools
 - Ask questions like “Can Subject S obtain access to Object O?”
- Interaction of implementation, model, policy:
 - System policy defined in terms of concrete model
 - Policy on data takes priority over wishes of users
 - Mandatory Access Control (MAC): Based on policy, over-rides access control/permission decisions of users
 - Discretionary Access Control (DAC): User-specified rights (as in Windows and Unix)

- 4 major underpinnings of trusted systems:
 - Policy: Requirements
 - Model: Concrete, verifiable specification
 - Design: Implementation of model (both functionality and construction methods)
 - Trust:
 - Features (system does what it's supposed to do)
 - Assurance (confidence that it enforces the security policy)

- Some examples:
 - Gemini Multiprocessing Secure Operating System (GEMSOS) Security Kernel
 - Trusted Solaris
 - Trusted IRIX
 - MLS+ (as Ultrix MLS+ and DEC Unix MLS+)
- Technicalities aside, “trusted” generally means
 - Developed with great care toward proper security enforcement
 - Supports data labeling and system-enforced policies based on data labels (MAC)
 - Has undergone independent evaluation

- Confidentiality vital to military operations
 - Ideas come from centuries of experience
 - Individual wishes less important than institutional policies
- Foundations of military security policies:
 - Sensitivity levels
 - Information can be unclassified, restricted, confidential, secret, or top secret
 - Subjects cleared for certain sensitivity levels

Military Security Policies – cont'd



Hierarchy of Sensitivities

Military Policies – Cont'd

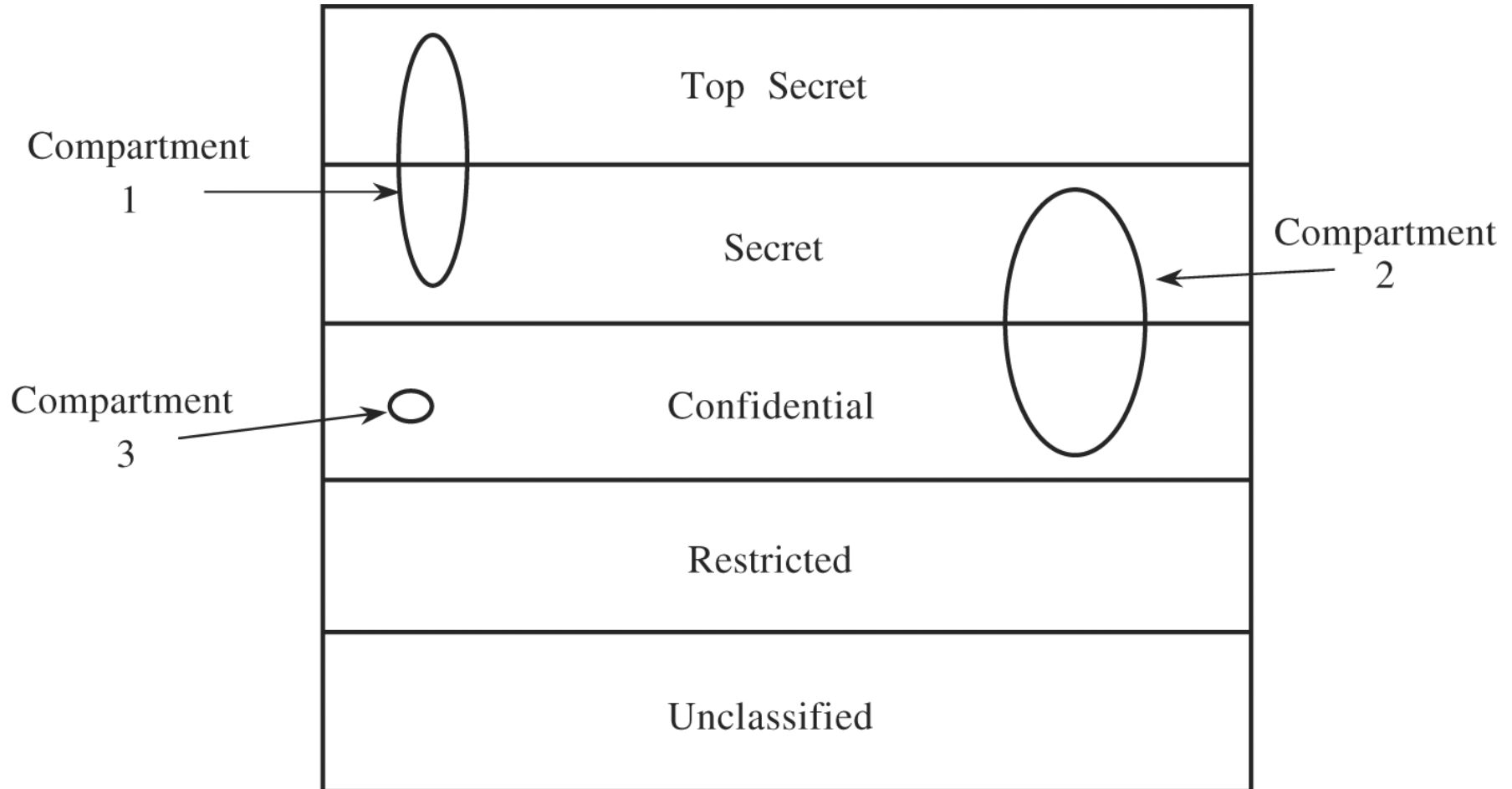
- Examples:
 - A low-level state department staffer cleared for confidential information can't get access to secret diplomatic information
 - In World War II, a top-secret cleared officer in the Pacific didn't get access to top-secret information about European operations

- Important issue: Security clearances and information sensitivity determined by a security officer and rules, not users!
 - MAC a necessity!

Military Policies – Cont'd

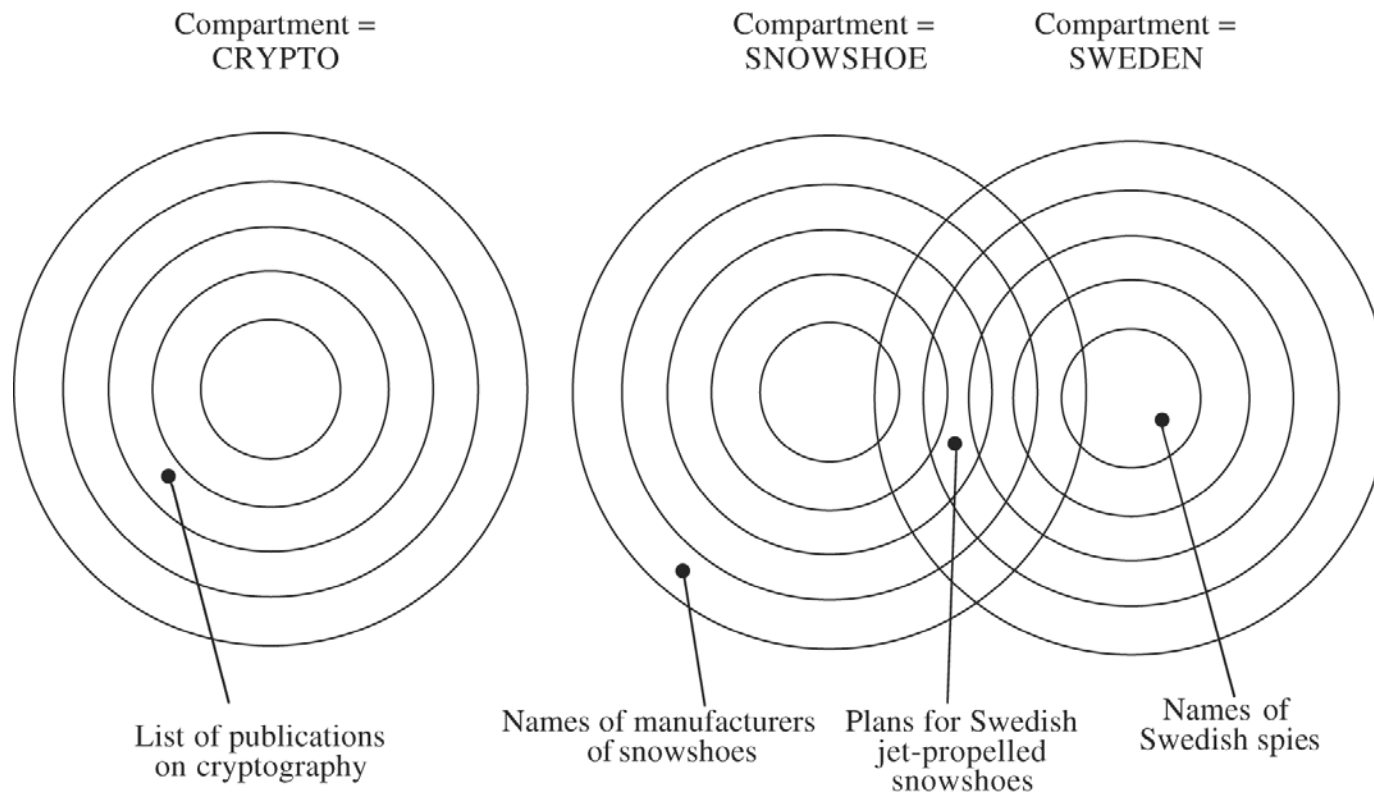
- **Need-to-know rule:** Access to sensitive data is allowed only to subjects who need to know those data to perform their jobs.
- Each piece of classified information may be associated with one or more projects, called **compartments**, describing the subject matter of the information.

Compartment and Sensitivity Levels



Military Policies – Cont'd

- Names are assigned to identify compartments
- A single piece of information can be coded with 0, 1, 2, .. N compartment names, based on # of categories to which it relates.



- **Ranks:** Preassigned sensitivity levels
 - Unclassified
 - Restricted
 - Confidential
 - Secret
 - Top secret
- **Class (classification):** <rank; compartments>
- **Clearance:** Is an indication that a person is trusted to access information up to a certain level of sensitivity and that the person needs to know certain categories of sensitive information.

Military Policies – Cont'd

- Formally:
 - Sensitivity level denoted as “rank”
 - Level of information object O denoted $rank_O$
 - Clearance of subject S denoted by $rank_S$
 - Need-to-know compartments
 - For object O : $compartments_O$
 - For subject S : $compartments_S$

- Dominance relation:
 - $O \leq S$ (“ S dominates O ”) if and only if
 - $rank_O \leq rank_S$ and
 - $compartments_O \subseteq compartments_S$
 - Sample restriction: S can read O only if $S \geq O$

Military Policies – Cont'd

- Example for dominance relation:
- Assume information classified as $\langle \textit{secret}; \{\textit{Sweedden}\} \rangle$
- It could be read by
 - $\langle \textit{topsecret}; \{\textit{Sweedden}\} \rangle$
 - $\langle \textit{secret}; \{\textit{Sweedden}, \textit{crypto}\} \rangle$
- Cannot be read by
 - $\langle \textit{top secret}; \{\textit{crypto}\} \rangle$
 - $\langle \textit{confidential}; \{\textit{Sweedden}\} \rangle$
 - $\langle \textit{secret}; \{\textit{France}\} \rangle$

Military Policies – Example

- User MacArthur cleared at $\langle \text{topsecret}, \{\text{Pacific}, \text{Japan}, \text{Phillipines}\} \rangle$
- User Patton cleared at $\langle \text{topsecret}, \{\text{Atlantic}, \text{Africa}, \text{Europe}\} \rangle$
- User Miller cleared at $\langle \text{secret}, \{\text{Atlantic}, \text{Europe}\} \rangle$
- Data MidwayPlans cleared at $\langle \text{topsecret}, \{\text{Pacific}\} \rangle$
- Data NormandyPlans cleared at $\langle \text{topsecret}, \{\text{Europe}\} \rangle$
- Data EuropeUnits cleared at $\langle \text{secret}, \{\text{Europe}\} \rangle$
- MacArthur can read MidwayPlans only
- Patton can read NormandyPlays and EuropeUnits
- Miller can read EuropeUnits only

- Important differences in commercial world:
 - Usually no “security officer” and clearances
 - Integrity often more important than confidentiality (which was the basis for military security models)
 - Prevent unauthorized modification, fraud, and errors

Commercial Policies

- Purchasing Dept.:
 - Purchasing creates order for a supply,
 - sends copies to supplier and receiving department
- Supplier:
 - ships the items to the receiving dept. Receiving clerk checks the delivery, quantity, item type. Delivery form and original order go to accounting dept.
- Receiving Dept.:
 - Receiving clerk checks the delivery, quantity, item type.
 - Delivery form and original order go to accounting dept.
- Supplier:
 - sends an invoice to the accounting dept. Accounting clerk compares the invoice with original order and the delivery order. Issues a check to supplier.
- Accounting Dept.:
 - Accounting clerk compares the invoice with original order and the delivery order. Issues a check to supplier.