

Chapter 5 CONFIDENTIALITY POLICIES

- Confidentiality policies emphasize the protection of confidentiality.
- In this chapter, we explore one such policy (the Bell-LaPadula Model) and the controversy it generated.

5.1 Goals of Confidentiality Policies

- A *confidentiality policy* (information flow policy) prevents unauthorized disclosure of information.
- Example: The Navy must keep confidential the date on which a troop ship will sail.
 - If the date is changed, the redundancy in the systems and paper work should notice that change.
 - If the enemy knows the date of sailing, they can sink the ship!
 - In military communications channels, there is extensive redundancy.
- In the US, the Privacy Act requires that certain personal data be kept confidential.
 - Income tax returns are legally confidential.
 - They are available to the IRS and to legal authorities with a court order.

5.2 The Bell-LaPadula Model (1973 and 1975)

- Corresponds to military style classifications.
- Has influenced the development of many other models, including computer security technologies.
- Consider Figure 5-1.

| | | |
|-------------------|------------------|-----------------------|
| TOP SECRET (TS) | Tamara, Thomas | Personnel Files |
| SECRET (S) | Sally, Samuel | Electronic Mail Files |
| CONFIDENTIAL (C) | Claire, Clarence | Activity Log Files |
| UNCLASSIFIED (UC) | Ulaley, Ursula | Telephone List Files |

Figure 5-1 Simplest type of confidentiality classification

Chapter 5 CONFIDENTIALITY POLICIES

- In Figure 5-1:
 - The four security levels are arranged with the most sensitive at the top and the least sensitive at the bottom.
 - The higher the security clearance, the more sensitive the information.
 - A subject has a *security clearance*.
 - Claire’s security clearance is C, and Thomas’ is TS.
 - An object has a *security classification*.
 - The security classification of the Electronic Mail Files is S.
 - The security classification of the Telephone List Files is UC.
- The goal of the Bell-LaPadula security model is to prevent read access to objects at a security classification higher than the subject’s clearance.
- The Bell-LaPadula security model combines mandatory and discretionary access controls.
- “S has discretionary read (write) access to O” means that the access control matrix for S and O corresponding to the discretionary access control component contains a read (write) right!
 - If mandatory controls are not present, S would be able to read (write) O.

Chapter 5 CONFIDENTIALITY POLICIES

- Let $L(S) = I_s$ be the security clearance of subject S , and let $L(O) = I_o$ be the security classification of object O .
 - Simple Security Condition, Preliminary Version: S can read O if and only if $I_o \leq I_s$, and S has discretionary read access to O .
 - In Figure 5-1, Claire & Clarence cannot read personnel files but Tamara & Sally can read the activity log files.
 - If Tamara decides to copy the contents of the personnel files into the activity log files and set the discretionary access permissions appropriately, Claire could then read the personnel files.
 - Thus, Claire is able to read the files at a higher level of security.
 - However, a second property prevents this!
 - *-Property, Preliminary Version: S can write O if and only if $I_s \leq I_o$, and S has discretionary write access to O .
 - Because the activity log files are classified as C, and Tamara has a clearance of TS, she is not able to write to the activity log files.
- A security systems needs to be defined as one in which both the *simple security condition, preliminary version*, and the **-property, preliminary version*, hold.
- Theorem 5-1. *Basic Security Theorem, Preliminary Version*: Let Σ be a system with a secure initial state σ_0 , and let T be a set of transformations. If every element of T preserves the simple security condition, preliminary version, and the *-property, preliminary version, then every state σ_i , $i \geq 0$, is secure.

Chapter 5 CONFIDENTIALITY POLICIES

- The model can be expanded by adding a set of categories to each security classification.
 - Each category describes a kind of information.
 - Objects placed in multiple categories have the kinds of information in all of those categories.
 - These categories arise from the “*need to know*” principle.
 - The set of categories to which a person may have access is the power set of the set of categories.
 - Example: The categories are NUC, EUR, and US.
 - Someone can have access to any of the following sets of categories:
 \emptyset (null set), {NUC}, {EUR}, {US}, {NUC, EUR}, {NUC, US}, {EUR, US}, and {NUC, EUR, US}.
 - These sets of categories form a lattice under the operation \subseteq .
- Definition of a lattice:
 - The combination of a set of elements S and a relation R meeting the following criteria:
 - R is reflexive, antisymmetric, and transitive on the elements of S .
 - For every $s, t \in S$, there exists a greatest lower bound under R .
 - For every $s, t \in S$, there exists a least upper bound under R .
 - A relation R defined over a set S is reflexive if aRa for all $a \in S$.
 - A relation R defined over a set S is antisymmetric if aRb and bRa imply $a=b$ for all $a, b \in S$.
 - A relation R defined over a set S is transitive if aRb and bRc imply aRc for all $a, b, c \in S$.
 - Example: $S = \{0, 1, 2\}$; R is \leq .
 - The relation is reflexive, antisymmetric, and transitive.
 - The greatest lower bound of any two integers is the smaller.
 - The least upper bound of any two integers is the larger.

Chapter 5 CONFIDENTIALITY POLICIES

- Consider Figure 5-2.

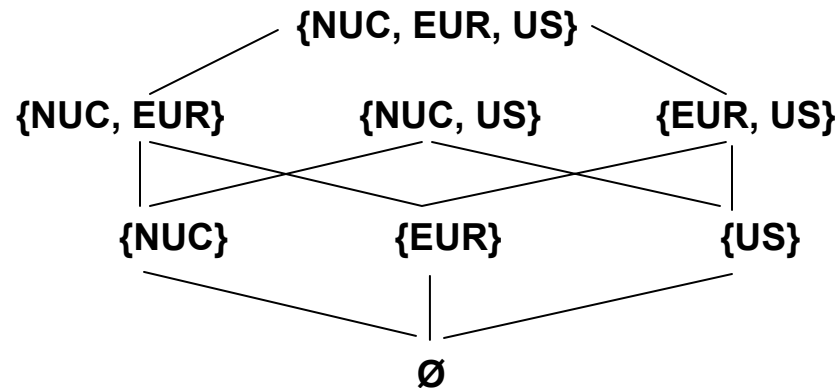


Figure 5-2. Lattice generated by the categories NUC, EUR, and US

- Each security level and category form a security level.
 - Security levels are also called “compartments” or “categories.”
 - Subjects have clearance at a security level.
 - Objects are at the level of a security level.
 - Example:
 - William may be cleared into the level (SECRET, {EUR})
 - George may be cleared into the level (TOP SECRET, {NUC, US})
 - A document may be classified as (CONFIDENTIAL, {EUR})
 - Security levels change access.
 - Someone with access to the category set {NUC, US} presumably has no need to access items in the category EUR.
 - Read access should be denied even if the security clearance of the subject is higher than the security classification of the object.
 - If the desired object is in any of the security levels \emptyset , {NUC}, {US}, or {NUC, US}, and the subject’s security clearance is no less than the document’s security classification, access should be granted.

Chapter 5 CONFIDENTIALITY POLICIES

- A new relationship is needed for capturing the combination of security classification and category set.
- Define the relation *dom*.
- Definition 5-1. The security level (L,C) *dominates* the security level (L',C') if and only if $L' \leq L$ and $C' \subseteq C$.
 - When (L,C) *dom* (L',C') is false, the notation is $(L,C) \neg\text{dom} (L',C')$.
 - Example:
 - George is cleared into security level $(\text{SECRET}, \{\text{NUC}, \text{EUR}\})$.
 - DocA is classified as $(\text{CONFIDENTIAL}, \{\text{NUC}\})$.
 - DocB is classified as $(\text{SECRET}, \{\text{EUR}, \text{US}\})$.
 - DocC is classified as $(\text{SECRET}, \{\text{EUR}\})$.

George *dom* DocA as $\text{CONFIDENTIAL} \leq \text{SECRET}$ and $\{\text{NUC}\} \subseteq \{\text{NUC}, \text{EUR}\}$.

George $\neg\text{dom}$ DocB as $\{\text{EUR}, \text{US}\} \not\subseteq \{\text{NUC}, \text{EUR}\}$.

George *dom* DocC as $\text{SECRET} \leq \text{SECRET}$ and $\{\text{EUR}\} \subseteq \{\text{NUC}, \text{EUR}\}$.
- Let $C(S)$ be the category set of subjects S , and let $C(O)$ be the category set of objects O .
- The simple security condition, preliminary version, is modified in the obvious way.

Chapter 5 CONFIDENTIALITY POLICIES

- **Simple Security Condition:** S can read O if and only if $S \text{ dom } O$ and S has discretionary read access to O .
 - Example:
 - George can read DocA and DocC but not DocB (assuming that the discretionary access controls allow such access) .
 - Suppose Paul is cleared into security level (SECRET, {EUR, US, NUC}) and has discretionary read access to DocB. Paul can read DocB. If he would copy its contents to DocA and set its access permissions accordingly, George could then read DocB.
- The modified *-property prevents this!
- *-Property: S can write to O if and only if $O \text{ dom } S$ and S has discretionary write access to O .
 - DocA dom Paul is false ($C(\text{Paul}) = \{\text{EUR}, \text{US}, \text{NUC}\}$ and $C(\text{DocA}) = \{\text{NUC}\}$). Hence, $C(\text{Paul}) \not\subseteq C(\text{DocA})$, so Paul cannot write to DocA.
- The simple security condition is often described as “no reads up” and the *-Property as “no writes down.”
- Now a *secure system* needs to be redefined as one in which both the simple security property and *-property hold.
- Theorem 5-2. **Basic Security Theorem:** Let Σ be a system with a secure initial state σ_0 , and let T be a set of transformations. If every element of T preserves the simple security condition and the *-property, then every state σ_i , $i \geq 0$, is secure.
- **EXAMPLE:**
 - A colonel with (SECRET, {NUC, EUR}) clearance needs to send a message to a major with (SECRET, {EUR}) clearance.
 - The colonel must write a document that has at most the (SECRET, {EUR}) classification.
 - This violates the *-property because (SECRET, {NUC, EUR}) dom (SECRET, {EUR}). ⁷

Chapter 5 CONFIDENTIALITY POLICIES

- The model provides a mechanism for allowing this type of communications.
 - A subject has a *maximum security level* and a *current security level*.
 - The maximum security level must dominate the current security level.
 - A subject may effectively decrease its security level from the maximum in order to communicate with entities at lower security levels.
 - **EXAMPLE:** The colonel's maximum security level is (SECRET, {NUC, EUR}). She changes her current security level to (SECRET, {EUR}). She can then create the document at the major's clearance level and send it to him.
 - How this policy is instantiated in different environments depends on the requirements of each environment. The conventional use to define "read" and "write" are as follows:
 - "read": is defined as "allowing information to flow from the object being read to the subject reading."
 - "write": is defined as "allowing information to flow from the subject writing to the object being written."
- The Data General B2 UNIX (DG/UX) System
 - Provides mandatory access controls (MACs).
 - Only the default labels are described; the system enables other labels to be created.
- Assigning MAC labels
 - When a process (subject) begins, it is assigned the MAC label of its parent.
 - The initial label (which is assigned at login time) is the label assigned to the user in a database called the *Authorization and Authentication (A & A) Database*.
 - Objects are assigned labels at creation but the labels may either be *explicit* or *implicit*.
 - The system stores explicit labels as part of the object's attributes.
 - The system determines implicit labels from the parent directory of the object.
 - The least upper bound of all compartments in the DG/UX lattice has the label IMPL_HI.
 - The greatest lower bound of all compartments in the DG/UX lattice has the label IMPL_LO.
 - The lattice is divided into 3 regions as shown in Figure 5-3.

Chapter 5 CONFIDENTIALITY POLICIES

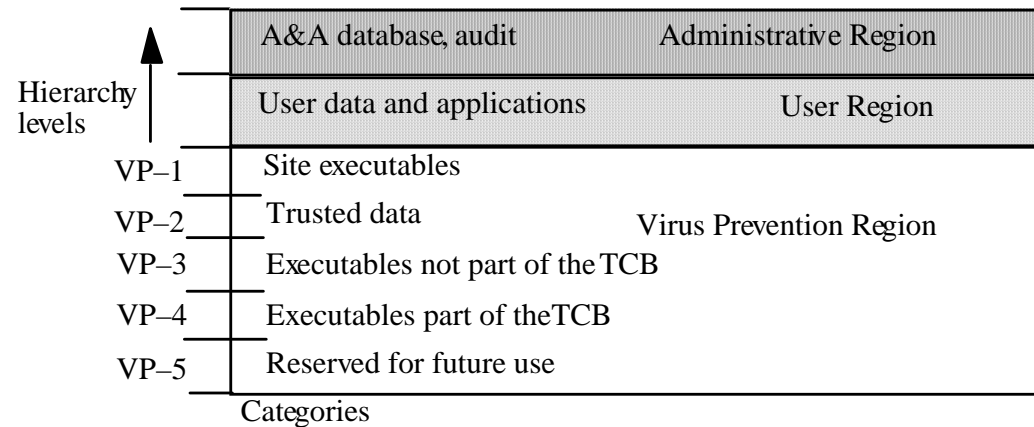


Figure 5-3. The three MAC regions in the MAC lattice

- In Figure 5-3:
 - The highest region (administration region) is reserved for data that users cannot access (e.g., logs, MAC label definitions, etc.).
 - Reading up and writing down are disallowed. Hence, users can neither read nor alter data in this region.
 - System programs are in the lowest region (virus prevention region).
 - No user process can alter them.
 - Users, however, can execute programs.
 - The name of this region comes from the fact that viruses and other forms of malicious logic involve alterations of trusted executables.

Chapter 5 CONFIDENTIALITY POLICIES

- Problems arise when programs of different levels access the same directory.
 - If a program with label *MAC_A* tries to create a file, and a file of that name but with label *MAC_B* exists, the creation will fail.
 - To prevent this leakage of information, only programs with the same MAC label as the directory can create files in that directory.
 - For the */tmp* directory, and the mail spoof directory */var/mail*, this restriction will prevent standard operations (e.g., compiling and delivering mail) .
 - DG/UX introduces a “multilevel directory” to solve this problem!
 - A *multilevel directory* is a directory with a set of subdirectories, one for each label.
 - These hidden directories normally are not visible to the user.
 - However, if a process with label *MAC_A* tries to create a file in */tmp*, it actually creates a file in the hidden directory under */tmp* with label *MAC_A*.
 - The file can have the same name as one in the hidden directory corresponding to label *MAC_A*.
 - The parent directory of a file in */tmp* is the hidden directory.
 - **EXAMPLE:**
 - » A process with label *MAC_A* creates a directory */tmp/a*.
 - » Another process with label *MAC_B* creates a directory */tmp/a*.
 - » The processes then change the correct working directory to */tmp/a* and then to .. (the parent directory).
 - » Both processes will appear to have */tmp* as the current working directory.
 - » However, the system call *stat (“.”, &stat_buffer)* returns a different inode number for each process because it returns the inode number of the current working directory – the hidden directory.
 - » The system call *dg_mstat (“.”, &stat_buffer)* translates the notion of current working directory to the multilevel directory when the current working directory is a hidden directory.

Chapter 5 CONFIDENTIALITY POLICIES

- Using MAC Labels
 - The DG/UX B2 system uses the Bell-LaPadula notion of dominance with one change.
 - The system obeys the simple security condition (reading down is permitted).
 - However, the implementation of the *-property requires that the process MAC label and the object MAC label be equal. Writing up is not permitted but writing is permitted in the same compartment.
 - Because of this restriction on writing, the DG/UX system provides processes and objects with a range of labels called a *MAC tuple*.
 - A range is a set of labels expressed by a *lower bound* and an *upper bound*.
 - A MAC tuple consists of up to three ranges (one for each region in Figure 5-3).
 - EXAMPLE: A system has two security levels, TS and S (TS dominates S).
 - The categories are COMP, NUC, and ASIA.
 - Examples of ranges
 - » [(S, {COMP}), (TS, {COMP})]
 - » [(S, ∅), (TS, {COMP, NUC, ASIA})]
 - » [(S, {ASIA}), (TS, {ASIA, NUC})]
 - » The label (TS, {COMP}) is in the first two ranges.
 - » The label (S, {NUC, ASIA}) is in the last two ranges.
 - » [(S, {ASIA}), (TS, {COMP, NUC})] is not a valid range because *not* (TS, {COMP, NUC}) *dom* (S, {ASIA}).

Chapter 5 CONFIDENTIALITY POLICIES

- An object can have a MAC tuple as well as the required MAC label.
 - If both are present, the tuple overrides the label.
 - A process has read access when its MAC label grants read access to the upper bound of the range.
 - A process has write access when its MAC label grants write access to any label in the MAC tuple range.
 - EXAMPLE: Suppose an object's MAC tuple is the single range $[(S, \{ASIA\}), (TS, \{ASIA, COMP\})]$.
 - A subject with MAC label $(S, \{ASIA\})$ cannot read the object because $(TS, \{ASIA, COMP\}) \text{ dom } (S, \{ASIA\})$.
 - The subject can write to the object because $(S, \{ASIA\})$ dominates the lower bound and is dominated by the upper bound.
 - A subject with MAC label $(TS, \{ASIA, COMP, NUC\})$ can read the object but cannot write the object.
 - A subject with MAC label $(TS, \{ASIA, COMP\})$ can both read and write the object.
 - A subject with MAC label $(TS, \{EUR\})$ can neither read nor write the object because its label is incomparable to that of the object, and the *dom* relation does not hold.
- A process has both a MAC label and a MAC tuple.
 - The label always lies within the range of the region in which the process is executing.
 - Initially, the subject's accesses are restricted by its MAC label.
 - However, the process may extend its read and write capabilities to within the bounds of the MAC tuple.