Security Engineering
Part II – Access Control

Juan E. Tapiador

`jestevez@inf.uc3m.es`

Department of Computer Science, UC3M

Security Engineering – 4th year BSc in Computer Science, Fall 2011

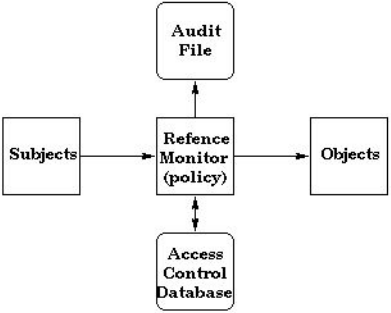
Overview

- One major security task is to *prevent* and *detect* unauthorised actions over information.
- Measures depending on how the attacker accesses the information:
  - direct access → cryptography
  - access through a software layer → access control
- Access control techniques implement a security policy in the software used to access information:
  - Operating systems
  - Databases
  - Web servers
  - Network traffic
- Access control can (must!) be implemented at several layers

Definitions

- Object: Entity that can be accessed (a file, a program, ...)
- Access operations: Actions executed over objects (read, write, delete, copy, execute, ...)
- Subject: Entity who executes access operations over objects.
- Access control policy: Rules establishing what is permitted and what is forbidden.
- Access control model: Formalism that allows to write down and process an access control policy.
- Access control mechanism: System/Technique that implements and enforces an access control policy.

System model



The access matrix

- A simple way of representing access policies
- $\mathcal{A}[s_i, o_j] =$ Access operations subject s_i is authorised to perform over object o_j
- Example:

	$\mathbf{o_1}$	$\mathbf{o_2}$	$\cdots$	$\mathbf{o_m}$
$\mathbf{s_1}$	rwx	rw	$\cdots$	r
$\mathbf{s_2}$	r	–	$\cdots$	rwx
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$\mathbf{s_n}$	x	rw	$\cdots$	w

The access matrix (cont.)

Remarks:

- The access matrix is a conceptual vehicle, not a proper data structure
- Scalability problems: In large systems the AM is
  - huge
  - sparse
- Common implementations:
  - columns → ACLs
  - rows → capabilities
  - hybrid → authorisation relationships

The access matrix (cont.)

Access Control List (ACL)

ACL	s_1	s_2	$\cdots$	s_n
$\mathbf{o_1}$	rwx	r	$\cdots$	x

- Each object stores its access control information
- Advantages – It's easy to...
  - check what privileges a user has over an object
  - revoke all privileges over an object
- Drawbacks – It's costly to...
  - determine a user's privileges (over all objects)
  - revoke a user's privileges (over all objects)

The access matrix (cont.)

Capabilities

Capability	o_1	o_2	$\cdots$	o_n
$\mathbf{s_2}$	r	-	$\cdots$	rwx

- Each user carries with him his access privileges
- Advantages / Drawbacks

  → Dual to ACLs
- More appropriate than ACLs for distributed systems

  → Slightly related to attribute certificates

The access matrix (cont.)

Authorisation relationships

Subject	Object	Access Op.
s_1	o_1	rwx
s_1	o_2	rw
s_2	o_1	x
$\vdots$	$\vdots$	$\vdots$

- Hybrid approach – store non-empty entries of the AM
- Combines advantages of ACLs and capabilities
- Generally implemented in a relational DB → very efficient queries

Extensions: Groups and Roles

- Common goals:
  - Facilitate design and maintenance of access control policies
  - Capture structure and dynamics of the organisation
- A group is a set of subjects
- A role is a set of access privileges that one or more subjects can assume
- **Role-Based Access Control (RBAC)**
  - Users assigned to roles (UA, many-to-many relation)
  - Permissions assigned to roles (PA, many-to-many relation)
  - *User session* → subset of roles are *activated* by a user
  - Role hierarchies (generally a POSET)
  - Constraints (e.g. separation of duty)

Discretionary Access Control (DAC)

Model: Basic access matrix + 2 additional elements:

1. Every object is associated with a specially designated subject → **Owner**
   - Generally the creator, though not necessarily
2. Only the owner can assign access privileges over an object.
   - Goal: Access controlled by the owner of the object

Extensions:

- Ownership transfer
- Delegation

Discretionary Access Control (DAC)

The Trojan Horse Problem

- By themselves, DAC policies cannot enforce information flow policies.

Mandatory Access Control (MAC)

- Emerged from confidentiality requirements of the military but has broad applications for integrity and separation objectives.
- Goal is **to confine the flow of information to one direction in a lattice of security labels**
  - Information flow is controlled by assigning each object a security class (or security label)
  - Whenever information flows from object x to object y, there is an accompanying flow between their respective classes
- Various models: Bell-LaPadula, Biba, Gougen-Meseguer, Brewer-Nash, ...
- Formalised by Denning in the late 1970s. More generally known as Lattice-Based Access Control (LBAC)

LBAC

Definition [Information flow policy] – A triple $\langle SC, \rightarrow, \oplus \rangle$ where:

- SC is a set of security classes
- $\rightarrow \subseteq SC \times SC$ is a binary can-flow relation on SC
- $\oplus : SC \times SC \rightarrow SC$ is a binary class-combining (or join) operator on SC.

Examples: (See the Hasse diagrams!)

1. High-Low policy
2. Isolated classes

Note: Can-flow is often called *dominance* relation, but the other way round!

LBAC

Definition [Denning's axioms] –

(i) The set of security clases SC is finite

(ii) The can-flow relation $\rightarrow$ is a partial order on SC

(iii) SC has a lower bound w.r.t. $\rightarrow$

(iv) The join operator $\oplus$ is a totally defined least upper bound operator:

(a) $\forall A, B \in SC$, $A \rightarrow A \oplus B$ and $B \rightarrow A \oplus B$

(b) $\forall A, B, C \in SC$, if $A \rightarrow C$ and $B \rightarrow C$ then $A \oplus B \rightarrow C$

LBAC

The military / government lattice

- Security classes (a.k.a. *classification levels* vary from country to country, but most have 4 o 5 classes corresponding to the risk incurred if information is made publicly available :

Top Secret (TS)	*exceptionally grave damage*
Secret (S)	*grave damage*
Confidential (C)	*damage / prejudicial*
Restricted (R)	*undesirable effects*
Unclassified (U)	*technically no classification*

- See e.g. `en.wikipedia.org/wiki/Classified_information`
- Users are also assigned a security class: clearance.
- Additionally, access is restricted on a *need to know* basis.

LBAC

Examples

- Subset lattices
- Product lattices (totally ordered lattice + subset lattice)

Bell-LaPadula (BLP)

- One of the earliest MAC models (1973)
- Targets confidentiality
- Basic idea: augment DAC with MAC policies
  - Authorisations specified by a DAC matrix D...
  - ... but the operation must be authorised by the MAC policy
- Users might have control over D, but generally not over the MAC policy

Bell-LaPadula (BLP)

Model (simplified):

- $\mathcal{O} = \{o_1, o_2, \ldots, o_n\}$ objects
- $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$ subjects
- $\forall o \in \mathcal{O} \ : \ \lambda(o)$ security classification of o
- $\forall s \in \mathcal{S} \ : \ \mu(s)$ security clearance of s
- $\succeq$ a total ordering over $\mathcal{S}$ and $\mathcal{O}$
  - We write $\mu(s) \succeq \lambda(o)$, meaning that clearance of s is greater than or equal to confidentiality of o.
- **3 basic properties:**

Bell-LaPadula (BLP)

Simple-security property – Subject s can read object o only if $\mu(s) \succeq \lambda(o)$

– No Read Up (NRU)

★-property – Subject s can write object o only if $\lambda(o) \succeq \mu(s)$

– No Write Down (NWD)

Tranquility property – Classifications $\lambda(o)$ and clearances $\mu(s)$ cannot be changed while the system is running.

Notes:

- How does information flow?
- Note the "only ifs"!!! Mandatory controls are necessary but not sufficient

Integrity models

- **Commercial** vs Military applications
- Focus on preserving data integrity
- Plenty of models:
  - Biba
  - Goguen-Meseguer
  - Sutherland
  - Clark-Wilson
  - Brewer-Nash
- Can be integrated with confidentiality models (e.g. BLP) into a single hierarchy

Biba model

- Ken Biba (1977)
- Dual to BLP

Model (simplified):

- $\mathcal{O} = \{o_1, o_2, \ldots, o_n\}$ objects
- $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$ subjects
- $\forall o \in \mathcal{O} \;:\; \sigma(o)$ integrity classification of o
- $\forall s \in \mathcal{S} \;:\; \tau(s)$ integrity clearance of s
- $\succeq$ a total ordering over $\mathcal{S}$ and $\mathcal{O}$
  - We write $\tau(s) \succeq \sigma(o)$, meaning that clearance of s is greater than or equal to integrity level of o.
- **3 basic properties:**

Biba model

Simple-integrity property – Subject s can read object o only if $\sigma(o) \succeq \tau(s)$

- No Read Down (NRD)

★-integrity property – Subject s can write object o only if $\tau(s) \succeq \sigma(s)$

- No Write Up (NWU)

Tranquility property – Integrity classifications $\sigma(o)$ and clearances $\tau(s)$ cannot be changed while the system is running.

Notes:

- How does information flow?
- Note the "only ifs"!!! Mandatory controls are necessary but not sufficient

MLS systems

- Very expensive
- Complex administration
  - Trusted users
  - Problems with declassification
- Avoid both undesirable and desirable accesses
- Inflexibility

Multilateral security

- Goal is not to prevent information flowing down a hierarchy, but *across* between departments.
- Lateral Information Flow Controls
- Three motivating applications
  - Healthcare – Patient's privacy. Data only accessible to some departments.
  - National Intelligence – Spies/collaborators in different countries.
  - Clash of interests within an organisation – E.g., law firms, consultancy, ...
- (*Compartmented security* or *compartmentation* in the US)
- Very general and very complex problem

The Chinese wall model

- Brewer and Nash (1989)
- MAC model focused on confidentiality, particularly in mitigating conflicts of interests
- Goal is to prevent information leakage between competitors
  - One user cannot access data of mutually competing companies
- Data is grouped into three layers:
  - Objects, each concerning a single corporation
  - Datasets of objects which concern the same corporation
  - Conflict of interest classes containing datasets whose corporations are in competition
- Policy implemented through mandatory separation rules known as *Chinese walls*.

The Chinese wall model

Formal model

- $\mathcal{S}$ subjects
- $\mathcal{O}$ objects
- $\mathcal{D}$ datasets
- $\mathcal{C}$ conflict of interest classes
- $\mathbf{y} : \mathcal{O} \to \mathcal{D}$
  - For each $o \in \mathcal{O}$, $\mathbf{y}(o)$ gives the dataset to which o belongs.
- $\mathbf{x} : \mathcal{D} \to \mathcal{C}$
  - For each $d \in \mathcal{D}$, $\mathbf{x}(d)$ gives the conflict of interest class to which d belongs.

The Chinese wall model

Simple-security property [Informal]
People are only allowed read access to information which is not held to conflict with any other information that they already possess.
(It's important to remember who has accessed what!)

Simple-security property – Let $N[s_i, o_j], s_i \in \mathcal{S}, o_j \in \mathcal{O}$ be a Boolean matrix, with $N[s_i, o_j] = \texttt{TRUE}$ iff s_i has accessed o_j. Then s can access o only if $\forall \hat{o} \mid N[s, \hat{o}] = \texttt{TRUE}$:

- $\mathbf{y}(o) = \mathbf{y}(\hat{o})$, or
- $\mathbf{y}(o) \notin \mathbf{x}(\mathbf{y}(\hat{o}))$

The Chinese wall model

★**-property** [Informal]
Write access is only permitted if (1) access is permitted by the simple security rule; and (2) no object can be read which is in a different company dataset to the one for which write access is requested and contains **unsanitised** *information.*

- **Sanitisation** takes the form of disguising a corporation's information, in particular to prevent the discovery of that corporation's identity. It's necessary to allow users to compare data relating to one corporation with that relating to other corporation, e.g. belonging to the same business sector.

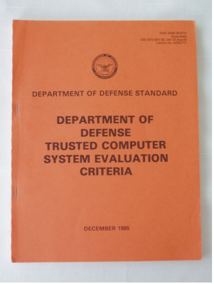
Multilevel and multilateral security

- Each object has both a security level L_i and a security compartment C_i. The pair $\langle L_i, C_i \rangle$ defines the security class of the object.
- Same for users with clearances and compartments.
- Access policy?

Computer security evaluation and certification

- Framework(s) to assess the effectiveness of computer security controls built into a computer system.
- Generally used to classify (and select) systems to be used to process sensitive information.
- Three important standards:
  - **US** – *Trusted Computer System Evaluation Criteria (TCSEC)*, a.k.a. the Orange Book (see DoD Rainbow Series)
  - **EUR** – *Information Technology Security Evaluation Criteria (ITSEC)*
  - **INT** – *The Common Criteria for Information Technology Security Evaluation (ISO/IEC 15408-1, v2.1)*, a.k.a. the Common Criteria, or simply CC.

The orange book (TCSEC)



DOD 5200.28-STD
Supersedes
CSC-STD-001-83, dtd 15 Aug 83
Library No. S225,711

DEPARTMENT OF DEFENSE STANDARD

DEPARTMENT OF
DEFENSE
TRUSTED COMPUTER
SYSTEM EVALUATION
CRITERIA

DECEMBER 1985

- NCSC (NSA), 1983
- Updated 1985
- Obsolete. Replaced by CC
- **Goals:**
  - Give security standards to manufacturers
  - Define evaluation and cert. metrics
  - Establish conditions to select systems
- Four divisions, some of them further divided into classes:

D		
C1	C2	
B1	B2	B3
A1	A1+	

The orange book (TCSEC)

D – Minimal protection

System has been evaluated but fails to meet requirements for higher divisions

C – Discretionary protection

C1 – Discretionary security protection

Identification and authentication; Separation of users and data; DAC policy limiting access on an individual basis; System Documentation and user manuals.

C2 – Controlled Access Protection

Finely grained DAC; User accountability through login procedures; Audit trails; Object reuse; Resource isolation.

The orange book (TCSEC)

B – Mandatory protection

B1 – Labeled Security Protection

Informal security policy model; Data sensitivity labels; MAC policy over selected subjects and objects; Label exportation capabilities; Discovered flaws are removed or mitigated; Design specifications and verification.

B2 – Structured Protection

Clear security policy model; DAC and MAC extended to all subjects and objects; Analysis of covert storage channels; Structured into protection-critical and non-protection-critical elements; [...]

B3 – Security Domains

Security administrator role; Trusted system recovery procedures; Analysis of covert timing channels; [...]

The orange book (TCSEC)

A – Verified protection

A1 – Verified Design

Identical to B3 plus formal design and verification techniques and formal management and distribution procedures.

A1+ / Beyond A1

More security testing, verification, etc. down to the source code level where feasible; [...]

The orange book (TCSEC)

Criticisms

- Focused on confidentiality
- Costly evaluation
- Inflexibility
- Amalgam of functionalities and assurances
- No separation of evaluation and certification

- **Reference:** `http://www.iwar.org.uk/comsec/resources/standards/rainbow/5200.28-STD.html`

European standards (ITSEC / ITSEM)

- Information Technology Security Evaluation Criteria (ITSEC, 1990)
- Information Technology Security Evaluation Methodology (ITSEM, 1993)
- **Definition of security:**
  - Confidentiality, Integrity, Availability
- **Two “objects”:**
  - IT products
  - IT systems
- **Reference:** `http://www.iwar.org.uk/comsec/resources/standards/itsec.htm`

European standards (ITSEC / ITSEM)

- Three basic concepts:
  - Security target (*why?*)
  - Security function (*what?*)
  - Security mechanism (*how?*)
    - → Target of evaluation (TOE)
- **TOE:** System or product subject to security evaluation
- **Security functions:** Identification and authentication; access control; accountability; audit; object reuse; accuracy; service reliability; data exchange.
- **Security target:** Document with the target's security features, which have to be evaluated. Includes implemented functions, security goals, theats and whatever mechanisms employed.

European standards (ITSEC / ITSEM)

- 10 Predefined functionality classes:
  - F-C1, F-C2, F-B1, F-B2, F-B3
  - F-IN, F-AV, F-DI, F-DC, F-DX
- Evaluation assurance level: *Ascending levels of confidence that can be placed in the TOE meeting its security objectives:*
  - E0 – Inadequate assurance
  - E1 – Functionally tested
  - E2 – Structurally tested
  - E3 – Methodically tested and analysed
  - E4 – Semi-formally designed and tested
  - E5 – Semi-formally verified design and tested
  - E6 – Formally verified design and tested

European standards (ITSEC / ITSEM)

- ITSEC vs TCSEC – Approximate equivalences:

E0	D
F-C1,E1	C1
F-C2,E2	C2
F-B1,E3	B1
F-B2,E4	B2
F-B3,E5	B3
F-B3,E6	A

- **ITSEM**

Common Criteria

- `http://www.commoncriteriaportal.org`
- *Common Criteria for Information Technology Security Evaluation (ISO 15408)* – **CC**
- *Common Methodology for Information Technology Security Evaluation (ISO 18045)* – **CEM**
- International standard. Originated out of ITSEC, CTPPEC (Canadian) and TCSEC.

Common Criteria

Key concepts

- **TOE**: as in ITSEC
- **Security Target**: as in ITSEC. Composed of SFRs
- **Sequrity Functional Requirements (SFRs)**: CC offers a standard catalogue, with dependencies among them.
- **Protection Profile (PP)**: Document that identifies security requirements for a class of security devices, e.g. firewalls, smart-card for signing, etc. Serve as templates.
- **Security Assurance Requirements (SARs)**: Measures taken during development and evaluation of the product to assure compliance with the claimed security functionality. E.g., code has been managed through a change management system, functional testing done, etc.

Common Criteria

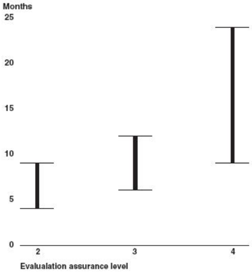
Key concepts

- **Evaluation Assurance Level (EAL)**: Measures depth and rigor of an evaluation. From EAL1 to EAL7.
  - Similar to EALs in TCSEC
  - Higher EALs do not necessarily mean better security, only that the claimed security assurance of the TOE has been more extensively verified.
  - Augmented EALs, e.g. EAL5+: assurance requirements beyond the minimum required for a particular EAL.

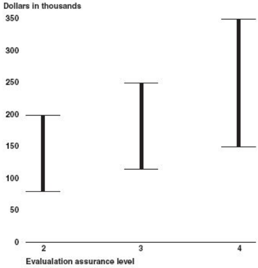
Common Criteria

Some final remarks:

- Evaluation can be lengthy and expensive:



Source: GAO analysis of data provided by laboratories.



Source: GAO analysis of data provided by laboratories.

- **Remember:** This is no Holy Grail: It does NOT ensure "this stuff gives me security," only that the vendor's claims have been more or less thoroughly verified.