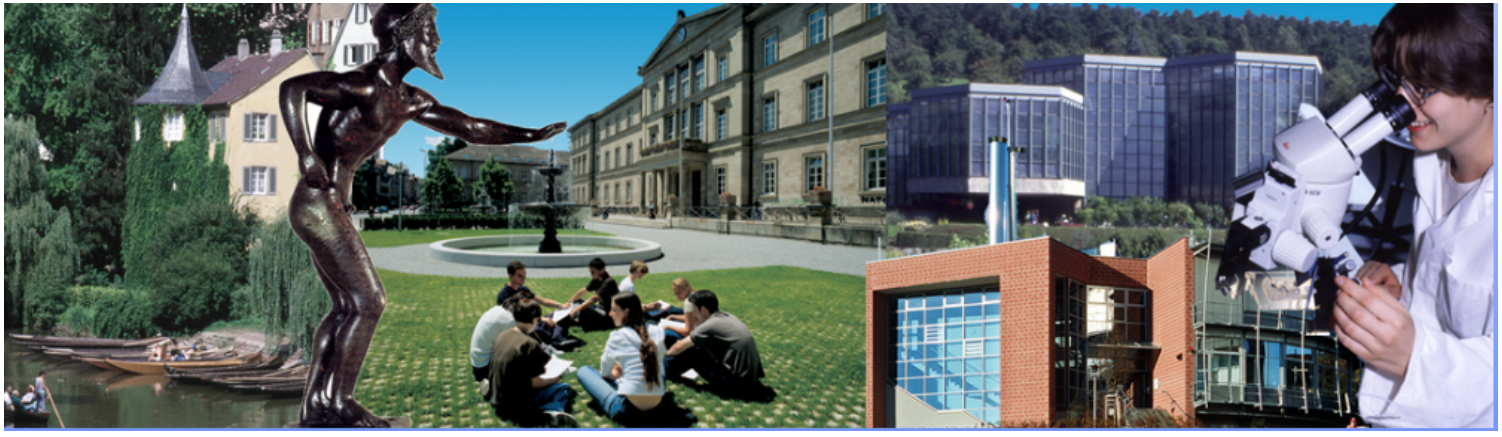
Eberhard Karls
Universität
Tübingen



Introduction to Computer Security

Access Control and Authorization

Pavel Laskov
Wilhelm Schickard Institute for Computer Science

Resource access recapitulated

1. Identification

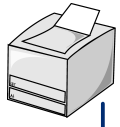
Which object O requests access to resource R?

2. Authentication

Is O indeed what it claims to be?

3. Authorization

Is O authorized to access R?

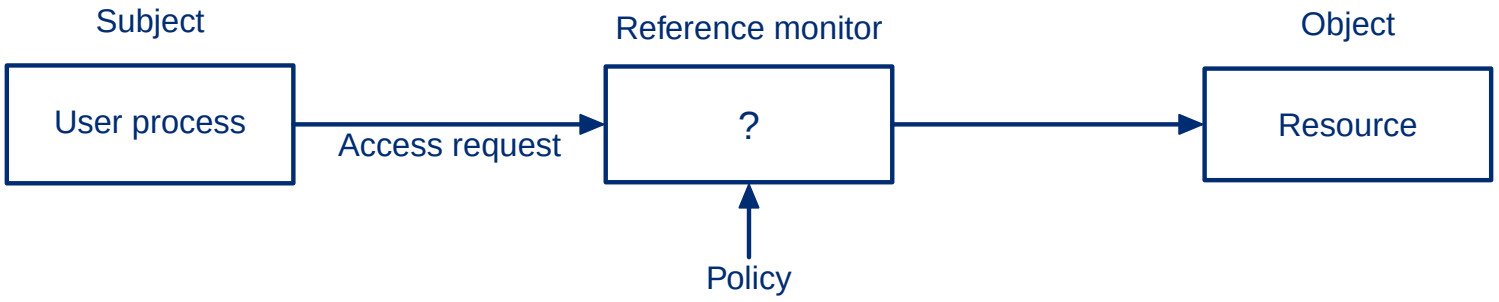


4. Accountability

Log the transaction access(O, R)

Access control overview

- Given a subject, which objects can it access and how?
- Given an object, which subjects can access it and how?



Main concepts of access control

- Subject is an entity that initiates an access request.
- Object is an entity an access to which is requested.
- Rights represent different types of access.
- Reference monitor makes authorization decisions.
- Goals of access control:
  - Granting access
  - Limiting access
  - Preventing access
  - Revoking access

Subjects

- Subjects are any active entities in a system.
- Subjects operate on behalf of principals.
- Each subject must be bound to a unique principal; a principal may be bound to several subjects.
- Examples:
  - Principal: user ID.
  - Subject: process ID.

Objects

- Objects represent passive resources of a system: memory, files, directories, nodes on a network, etc.
- The distinction of objects and subjects is made purely in terms of access requests.
- Depending on circumstances, a resource may be an object or a subject.

Reference monitor and access policies

- Reference monitor is an abstract notion of a mechanism for controlling access requests.
- Access rights represent various access operations supported by a system:
  - read
  - write
  - append
  - execute
  - delete
  - search
  - change owner
  - change permissions
- Access policies map principals, objects and access rights.

Access control structures

- Access control structures are mechanisms for implementing access policies:
  - access control matrix
  - capabilities
  - access control lists
  - intermediate controls (groups, negative permissions, roles, protection rings etc.)
- Requirements for access control structures:
  - an ability to express control policies
  - verifiability of correctness.
  - scalability and manageability

Access control matrix

Access control matrix is a basic control structure.

	bill.doc	edit.exe	fun.com
Alice	–	{execute}	{execute,read}
Bob	{read}	{execute}	{execute,read,write}

Advantages:

- clarity of definition
- easy to verify

Disadvantages:

- poor scalability
- poor handling of changes

Capabilities

Capability is a subject-centered description of access rights:

> Alice: {edit.exe: execute}, {fun.com: execute, read}
> Bob: {bill.doc: read, write}, {edit.exe: execute}, {fun.com: execute, read, write}

Advantages:

- easy ownership transfer
- easy inheritance of access rights

Disadvantages:

- poor overview of access rights per object
- difficulty of revocation
- need for extra integrity protection

Access control lists (ACL)

ACL is an object-centered description of access rights:

bill.doc: {Bob: read, write}
exit.exe: {Alice: execute}, {Bob: execute}
fun.com: {Alice: execute, read}, {Bob: execute, read, write}

Advantages:

- easy access to object access rights
- relative easiness of management using abstractions

Disadvantages:

- poor overview of access rights per subject
- difficulty of revocation
- difficulty of sharing

Access control abstractions

- Group: an collection of related subjects
  - easy sharing
  - easy addition and removal of users
- Negative permission: explicit revocation of access rights
- Privilege: a mapping of users to access rights
  - concise definition of access rights
  - {admin: read, write, execute}, /etc/passwd: {Alice, admin}
- Protection ring: a hierarchy of access right levels

  0 – operating system kernel
  1 – operating system
  2 – services
  3 – user processes

Discretionary access control (DAC)

- Access control is carried out by a resource owner.
- By associating ownership with principals, access rights are easily transferred to other subjects.
- Deployed in a majority of common systems.

Advantages:

- simple and efficient access rights management
- scalability

Disadvantages:

- intentional abuse of access rights
- unintentional abuse of access rights
- no control over information flow

Mandatory access control (MAC)

- Centralized access control by means of system-wide policy.
- Access control rights are fixed by an administrators.
- A limited number of implementations, e.g. SELinux, Systrace.

Advantages:

- strict control over information flow
- strong exploit containment

Disadvantages:

- major usability problems
- cumbersome administration

Role based access control (RBAC)

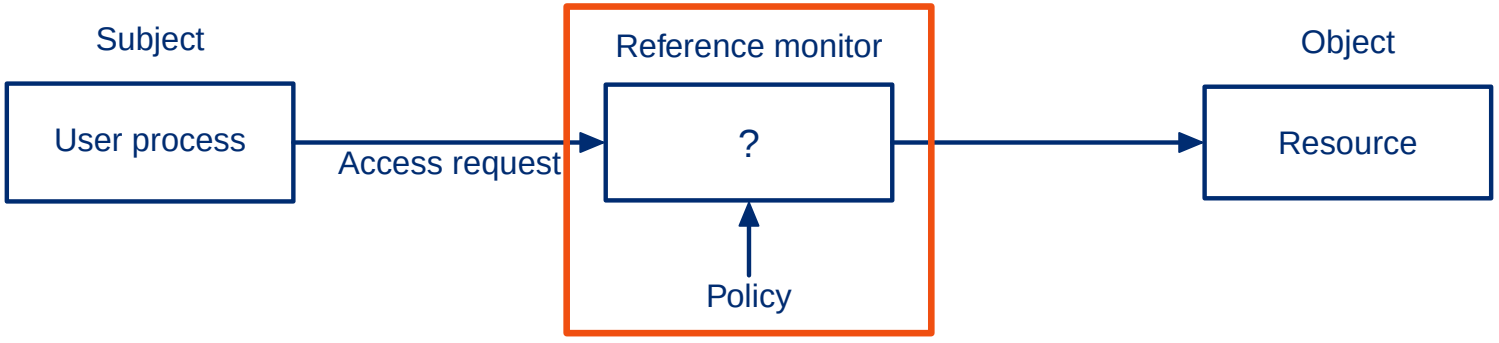
RBAC attempt to handle the complexity of access control by extensive used of abstractions:

- Data types are defined for all objects.
- Procedures are high level access control methods with a more complex semantics than elementary access control rights. Procedures can be only applied to certain data types.
- Procedures are grouped into roles assigned to users. A user can have more than one role and more than one user can have the same role.
- Role hierarchies can be used to match natural relations between roles.

Example: A *Lecturer* can create a role *Student* and give it a privilege “read course material”.

Reference monitors

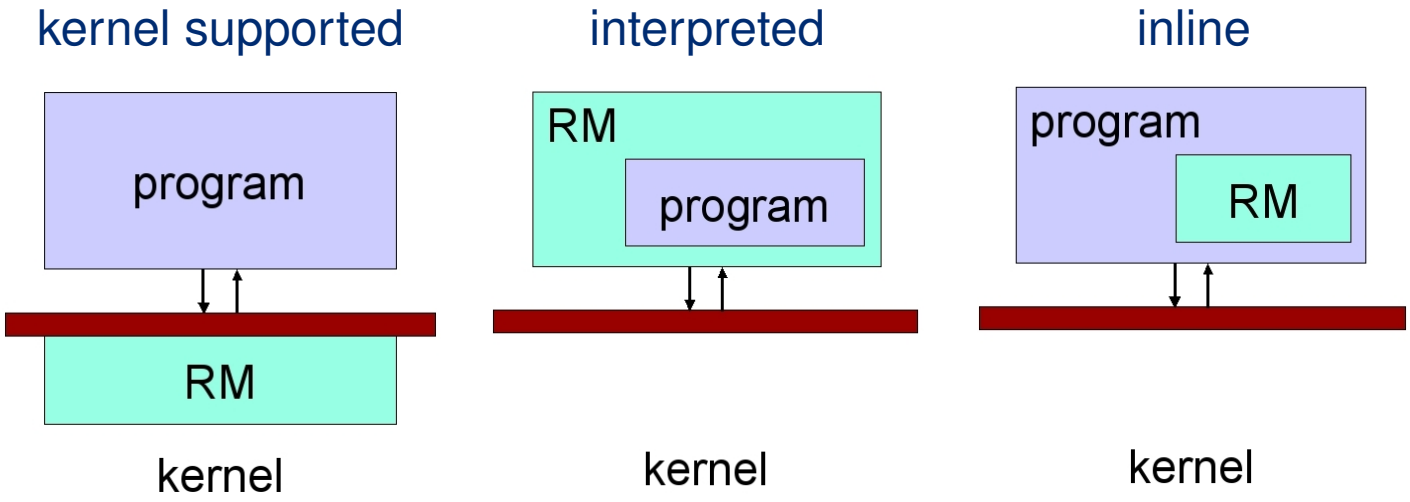
A reference monitor is an abstract device that mediates all accesses of objects to subjects.



Core requirements for a reference monitor implementation:

- Tamper-resistance
- Complete mediation (guaranteed invocation)
- Easiness of verification and testing

Reference monitor design choices



Reference monitor placement

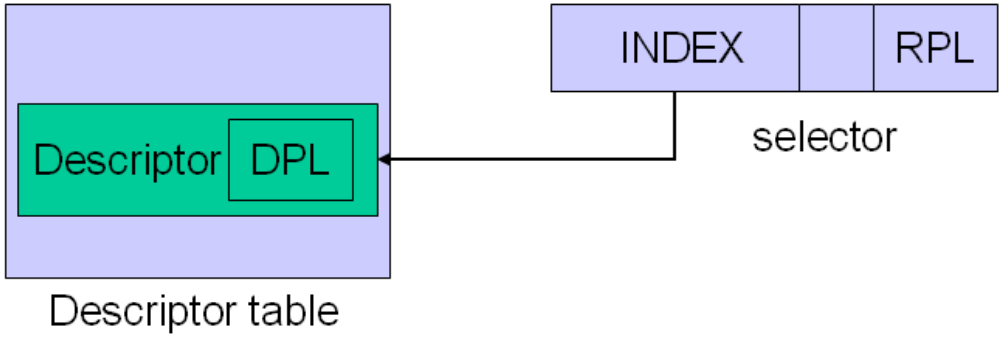
- Hardware:
  - low-level objects, no “layer below”, full system integrity
- Operating system kernel:
  - abstract low-level objects, hard to subvert, encapsulation
- Operating system:
  - conventional objects, not tamper-proof, most common
- Services: databases, JVM, .NET, CORBA
  - high-level abstract objects, very common
- Applications:
  - application-specific objects and access rights

Reference monitors in Intel 80x86

- Status register contains a 2-bit field corresponding to four protection rings (priviledge levels):
  - 0 – operating system kernel
  - 1 – rest of operating system
  - 2 – I/O drivers etc.
  - 3 – application software (user processes)
- Processes can only access resources in their own rings.
- Access to OS objects is controlled by object decriptors stored in descriptor tables.

Access control in Intel 80x86

- Descriptor table is accessed by processes via selectors.
- A selector contains an index of an object's descriptor (in a descriptor table) and a Requested Priviledge Level (RPL) field.
- A selector of a current process is stored in the code segment (CS) register. Its RPL is then compared with the priviledge level in the descriptor (DPL) for access decisions.

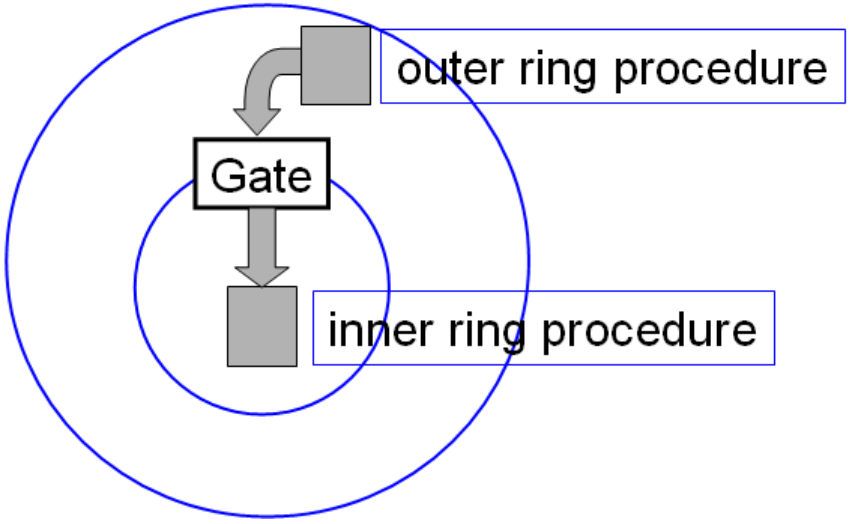


Controlled invocation in Intel 80x86

- How can a program access OS resources?

Controlled invocation in Intel 80x86

- How can a program access OS resources?
- A gate is an object having a ring 3 priviledge level which is able to call objects with higher priviledge levels.
- Gates enable execution but prevent unauthorized manipulation of OS objects.



Summary

- Access control methods implement policies that control which subjects can access which objects in which way.
- Most common practical access control instruments are ACLs, capabilities and their abstractions.
- From the design point of view, access control systems can be classified into discretionary (DAC), mandatory (MAC) and role-based (RBAC).
- Reference monitors are instruments for realization of access control policies. They can be deployed at all levels of system hierarchy.