Access Control

Madiba Hudson-Quansah

Contents

Chapter 1

Introduction

Definition 1.0.1: Access Control

The process of granting or denying specific requests to:
- Obtain and use information and related information processing services
- Enter specific physical facilities

Definition 1.0.2: Subject

An entity capable of accessing objects. There are generally three classes:
Owner - The entity that has control over an object and can determine who has access to it.
Group - A collection of users that share similar access needs.
World - All other users not in the owner or group categories.

Definition 1.0.3: Object

A resource to which access is controlled / An entity used to contain and/or receive information.

Definition 1.0.4: Access Right

Describes the way in which a subject may access and object. e.g. Read, Write, Execute.

Access control can be achieved through a combination of policies, programs and technologies with the focus on the permission or privileges that a subject has on an object.

1.1 Access Control security Requirements (NIST SP 800-171)

Chapter 2

Access Control Structures

2.1 Access Control Matrix (ACM)

An abstract model that defines the rights of all subjects over all objects in a system. It is represented as a matrix where:

- Rows represent subjects.
- Columns represent objects.
- Each cell contains the access rights that the subject has over the object.

It is abstract because it cannot be feasibly implemented because:

- The matrix can be very large and sparse.
- It is difficult to manage and update.

Example:

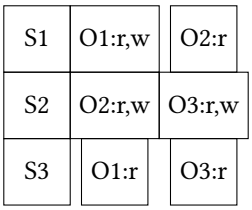
	O1	O2	O3
S1	r,w	r	
S2		r,w	r,w
S3	r		r

2.2 Access Control List (ACL) / Capability List (CL)

A more practical implementation of the ACM, where in the case of ACL, each object has a list of subjects and their corresponding access rights. This is commonly used in file systems. This is preferred when there are more subjects than objects. Example of ACL:

O1	S1:r,w	S3:r
O2	S1:r	S2:r,w
O3	S2:r,w	S3:r

And in the case of CL, each subject has a list of objects and their corresponding access rights. This is commonly used in capability-based systems. This is preferred when there are more objects than subjects. Example of CL:



The main difference between ACL and CL is that ACL maps objects to subject rights, but CL maps subjects to object rights.

2.3 Authorization Table

A table that defines the access rights of subjects to objects. It is commonly used in access control systems such as Role-Based Access Control (RBAC) and can include additional contextual attributes (e.g., time, location) similar to Attribute-Based Access Control (ABAC).

Subject	Object	Access Rights	Conditions
S1	O1	r, w	9am–5pm
S2	O2	r	Location = A
S3	O3	r, w	Location = B

2.4 Extended Access Control Matrix (EACM)

An extension of the ACM that includes additional contextual attributes beyond just subjects, objects, and access rights. These attributes can include conditions based on network addresses, protocols, port, time of day and other criteria. Commonly used in network routers and firewalls to control network traffic. Example:

	O1	**O2**	**O3**
S1	r,w [9am–5pm]	r [Loc=A]	
S2		r,w [Loc=A]	r,w [Loc=B]
S3	r [Loc=B]		r [9am–5pm]

Chapter 3

Access Control Models

3.1 Discretionary Access Control (DAC)

Definition 3.1.1: Discretion

The ability of an entity to determine who has access to its resources.

Provides the ability to share resources or information in a peer-to-peer configuration, i.e. no centralized control. This allows users to control and provide access to their own resources to others at their discretion.

3.2 Non-Discretionary Access Control (NDAC)

Access control decisions are made by a central authority based on predefined policies rather than individual user discretion. This model is often used in environments where security and compliance are critical.

3.2.1 Mandatory Access Control (MAC)

Definition 3.2.1: Label

A tag or marker that specifies the security level of an object or subject. e.g. Unclassified, Confidential, Secret, Top Secret.

- A system-enforced access rights are defined by a central policy not by users.
- Every subject and object has a security label.

3.2.1.1 Rules

Based on the Bell-LaPadula model:

No Read Up - Can't read higher classification, i.e. a user with "Secret" clearance cannot read "Top Secret" documents, but can read "Confidential" documents.

No Write Down - Can't write to lower classification, i.e. a user with "Secret" clearance cannot write to "Confidential" documents, but can write to "Top Secret" documents.

3.2.1.2 Goal

- Prevent information leakage between classification levels.
- Protect confidentiality.

3.2.1.3 Use cases

- Military Systems
- Government Agencies
- Critical Infrastructure

3.2.2 Lattice-Based Access Control (LBAC)

- An extension of MAC that adds categories/compartments, i.e. modifiers to classification levels.
- Categories represent specific areas of information, e.g. Nuclear, Crypto, Finance, etc.

3.2.2.1 Rules

Access is Allowed - If the subject's security label/level is greater than or equal to the object's security label/level and the subject's categories include all of the object's categories, i.e.:

$$\text{Subject Level} \geqslant \text{Object Level} \land \text{Subject Categories} \supseteq \text{Object Categories}$$

3.2.2.2 Goal

- Provides fine-grained control across multiple domains.

3.2.2.3 Use cases

- Multi-Tenant cloud environments.
- Intelligence Agencies
- Research
- Defence

3.2.3 Role-Based Access Control (RBAC)

Definition 3.2.2: Role

A collection of permissions that define a specific job function or responsibility within an organization.

RBAC assigns permission to specific roles within an organization. Users are then assigned roles inheriting the permissions associated with those roles. This model is based on the functions a user performs within an organization simplifying access management.

3.2.3.1 Rules

Role Assignment - A user can only execute a role if they have been assigned to that role.

Role Authorization - A user can only be assigned to roles for which they are authorized.

Permission Authorization - A user can only exercise permissions that are authorized for their active role.

Role Hierarchies - Roles can inherit permissions from other roles, allowing for a hierarchical structure of roles.

3.2.3.2 Constraints

Definition 3.2.3: Constraint

A defined relationship among roles, or a condition related to roles.

Constraints provide a means of adapting RBAC to the specifies of administrative and security policies of an organization.

Types of constraints:

Mutually Exclusive Roles - A user can be assigned to one role in a set of roles but not to others in the same set. Any permission can be granted only one role in the set.

Cardinality - Limits the maximum number of users that can be assigned to a specific role, or the maximum number of roles that can be assigned to a specific user.

Prequisite Roles - Enforces that a user can only be assigned to a particular role if the user is already assigned to another specified role.

3.2.4 Attribute-Based Access Control (ABAC)

Access decisions are based on attributes of the user, rather than roles or identities. A policy defines rules that determine access based on these attributes. This provides:

- Fine-grained, context-aware access control
- More dynamic and scalable access management compared to RBAC

It is commonly used on cloud, government and zero-trust environments.