Access Control and Authorization 9

9.1 Definitions

> Access control is a process to determine "Who does what to what," based on a policy.

One of the system administrator's biggest problems, which can soon turn into a nightmare if it is not well handled, is controlling access of who gets in and out of the system and who uses what resources, when, and in what amounts. Access control is restricting this access to a system or system resources based on something other than the identity of the user. For example, we can allow or deny access to a system's resources based on the name or address of the machine requesting a document.

Access control is one of the major cornerstones of system security. It is essential to determine how access control protection can be provided to each of the system resources. To do this, a good access control and access protection policy is needed. According to Raymond Panko, such a policy has benefits including the following [1]:

- It focuses the organization's attention on security issues, and probably, this attention results in resource allocation toward system security.
- It helps in configuring appropriate security for each system resource based on role and importance in the system.
- It allows system auditing and testing.

As cyberspace expands and the forces of globalization push e-commerce to the forefront of business and commercial transactions, the need for secure transactions has propelled access control to a position among the top security requirements, which also include authorization and authentication. In this chapter, we are going to discuss access control and authorization; authentication will be discussed in the next chapter.

J.M. Kizza, *Guide to Computer Network Security*, Computer Communications
and Networks, DOI 10.1007/978-1-4471-6654-2_9

9.2 Access Rights

To provide authorization, and later as we will see authentication, system administrators must manage a large number of system user accounts and permissions associated with those accounts. The permissions control user access to each system resource. So, user A who wants to access resource R must have permission to access that resource based on any one of the following modes: read, write, modify, update, append, and delete. Access control regimes and programs, through validation of passwords and access mode permissions, let system users get access to the needed system resources in a specified access mode.

Access control consists of four elements: subjects, objects, operations, and a reference monitor. In the normal operation, seen in Fig. 9.1, the subject, for example, a user, initiates an access request for a specified system resource, usually a passive object in the system such as a Web resource. The request goes to the reference monitor. The job of the reference monitor is to check on the hierarchy of rules that specify certain restrictions. A set of such rules is called an *access control list* (ACL). The access control hierarchy is based on the URL path for Web access or the file path for a file access such as in a directory. When a request for access is made, the monitor or server goes in turn through each ACL rule, continuing until it encounters a rule that prevents it from continuing and results in a request rejection or comes to the last rule for that resource, resulting into access right being it granted.

Subjects are system users and groups of users, while objects are files and resources such as memory, printers, and scanners including computers in a network. An access operation comes in many forms including Web access, server access, memory access, and method calls. Whenever a subject requests to access an object, an access mode must be specified. There are two access modes: observe and alter. In the observe mode, the subject may only look at the content of the object; in the alter mode, the subject may change the content of the object. The observe mode is the typical read in which a client process may request a server to read from a file.

Access rights refer to the user's ability to access a system resource. There are four access rights: *execute*, *read*, *append*, and *write*. The user is cautioned not to confuse access rights and access modes. The difference lies in the fact that you can perform any access right within each access mode. Figure 9.2 shows how this can be done. Note that according to the last column in Fig. 9.2, there are X marks in

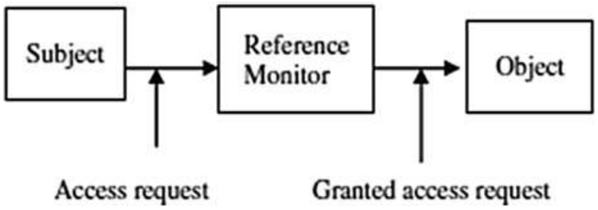
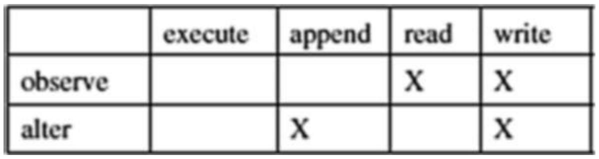


Fig. 9.1 Access control administration



	execute	append	read	write
observe			X	X
alter		X		X

Fig. 9.2 Access modes and access rights (Gollman [2])

both rows because in order to write, one must observe first before altering. This prevents the operating system from opening the file twice, one for the read and another for a write.

Access rights can be set individually on each system resource for each individual user and group. It is possible for a user to belong to a few groups and enjoy those groups' rights. However, user access rights always take precedence over group access rights regardless of where the group rights are applied. If there are inherited group access rights, they take precedence over user default access rights. A user has default rights when the user has no assigned individual or group rights from root down to the folder in question. In the cascading of access rights application, user access rights that are closest to the resource being checked by the monitor take precedence over access rights assignments that are farther away.

We have so far discussed access rights to resources. The question that still remains to be answered is, Who sets these rights? The owner of the resource sets the access rights to the resource. In a global system, the operating systems own all system resources and therefore set the access rights to those resources. However, the operating system allows folders and file owners to set and revoke access rights.

9.2.1 Access Control Techniques and Technologies

Because a system, especially a network system, may have thousands of users and resources, the management of access rights for every user per every object may become complex. Several control techniques and technologies have been developed to deal with this problem; they include access control matrix, capability tables, access control lists, role-based access control, rule-based access control, restricted interfaces, and content-dependent access control.

Many of the techniques and technologies we are going to discuss below are new in response to the growth of cyberspace and the widespread use of networking. These new techniques and technologies have necessitated new approaches to system access control. For a long time, access control was used with user- or group-based access control lists, normally based in operating systems. However, with Web-based network applications, this approach is no longer flexible enough because it does not scale in the new environment. Thus, most Web-based systems employ newer techniques and technologies such as role-based and rule-based access control, where access rights are based on specific user attributes such as their role, rank, or organization unit.

9.2.1.1 Access Control Matrix

All the information needed for access control administration can be put into a matrix with rows representing the subjects or groups of subjects and columns representing the objects. The access that the subject or a group of subjects is permitted to the object is shown in the body of the matrix. For example, in the matrix shown in Fig. 9.2, user A has permission to write in file R4. One feature of the access control matrix is its sparseness. Because the matrix is so sparse, storage consideration becomes an issue, and it is better to store the matrix as a list.

9.2.1.2 Access Control Lists

In the access control lists (ACLs), groups with access rights to an object are stored in association to the object. If you look at the access matrix shown in Fig. 9.2, each object has a list of access rights associated with it. In this case, each object is associated with all the access rights in the column. For example, the ACL for the matrix shown in Fig. 9.3 is shown in Fig. 9.4.

ACLs are very fitting for operating systems as they manage access to objects [2].

9.2.1.3 Access Control Capability

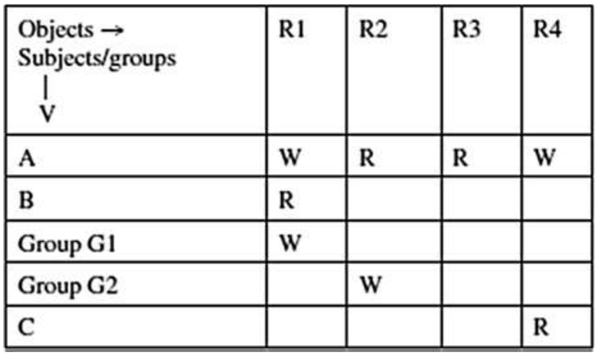
A capability specifies that "the subject may do operation O on object X."

Unlike the ACLs, where the storage of access rights between objects and subjects is based on columns in the access control matrix, capabilities access control storage is based on the rows. This means that every subject is given a capability, a forgery-proof token that specifies the subject's access rights [2].

From the access matrix shown in Fig. 9.3, we can construct a capability as shown in Fig. 9.5.

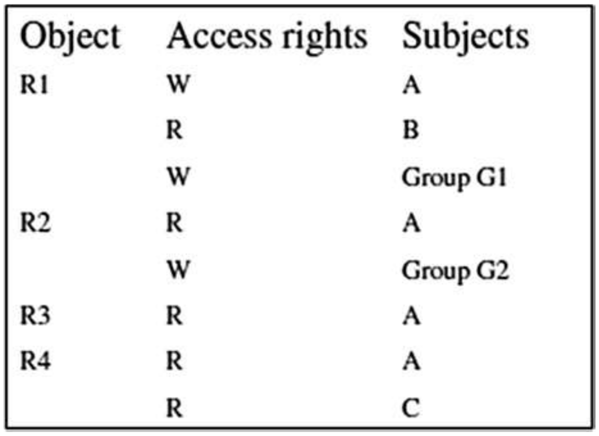
9.2.1.4 Role-Based Access Control

The changing size and technology of computer and communication networks are creating complex and challenging problems in the security management of these large networked systems. Such administration is not only becoming complex as technology changes and more people join the networks, but it is also becoming



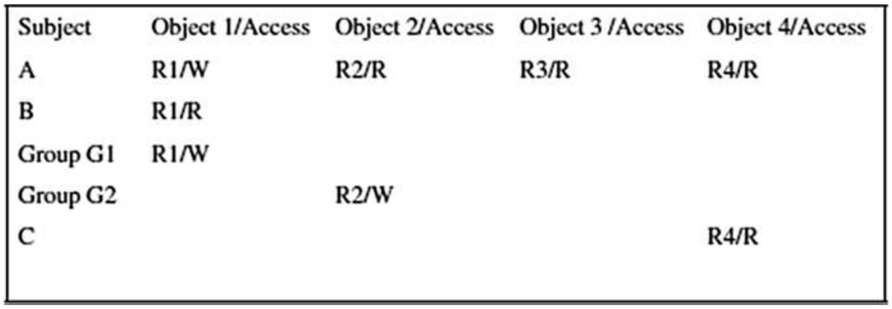
Objects → Subjects/groups | V	R1	R2	R3	R4
A	W	R	R	W
B	R			
Group G1	W			
Group G2		W		
C				R

Fig. 9.3 Access matrix



Object	Access rights	Subjects
R1	W	A
	R	B
	W	Group G1
R2	R	A
	W	Group G2
R3	R	A
R4	R	A
	R	C

Fig. 9.4 Access control list (ACL)



Subject	Object 1/Access	Object 2/Access	Object 3 /Access	Object 4/Access
A	R1/W	R2/R	R3/R	R4/R
B	R1/R			
Group G1	R1/W			
Group G2		R2/W		
C				R4/R

Fig. 9.5 Access control capability lists

extremely costly and prone to error when it is solely based on access control lists for each user on the system individually.

System security in role-based access control (RBAC) is based on roles assigned to each user in an organization. For example, one can take on a role as a chief executive officer, a chief information officer, or a chief security officer. A user may be assigned one or more roles, and each role is assigned one or more privileges that are permitted to users in that role. Access decisions are then based on the roles that individual users have as part of an organization. The process of defining roles should be based on a thorough analysis of how an organization operates and should include input from a wide spectrum of users in an organization.

Access rights are grouped by role name, and the use of resources is restricted to individuals authorized to assume the associated role. A good example to illustrate the role names and system users who may assume more than one role and play those roles while observing an organization's security policy is the following given in the NIST/ITL Bulletin, of December 1995. "Within a hospital system the role of doctor can include operations to perform diagnosis, prescribe medication, and order laboratory tests, and the role of researcher can be limited to gathering anonymous clinical information for studies" [3].

Accordingly, users are granted membership into roles based on their competencies and responsibilities in the organization. The types of operations that a user is permitted to perform in the role he or she assumes are based on that user's role. User roles are constantly changing as the user changes responsibilities and functions in the organizations, and these roles can be revoked. Role associations can be established when new operations are instituted, and old operations can be deleted as organizational functions change and evolve. This simplifies the administration and management of privileges; roles can be updated without updating the privileges for every user on an individual basis.

Like other types of access control, RBAC is also based on the concept of *least privilege* that requires identifying the user's job functions, determining the minimum set of privileges required to perform that function, and restricting the user to a domain with those privileges and nothing more. When a user is assigned a role, that user becomes associated with that role, which means that user can perform a certain and specific number of privileges in that role. Although the role may be associated with many privileges, individual users associated with that role may not be given more privileges than are necessary to perform their jobs.

Although this is a new technology, it is becoming very popular and attracting increasing attention, particularly for commercial applications, because of its potential for reducing the complexity and cost of security administration in large networked applications.

9.2.1.5 Rule-Based Access Control

Like other access control regimes, rule-based access control (RBAC), also known as *policy-based access control* (PBAC), is based on the least privileged concept. It is also based on policies that can be algorithmically expressed. RBAC is a multipart process, where one process assigns roles to users just like in the role-based access control techniques discussed above. The second process assigns privileges to the assigned roles based on a predefined policy. Another process is used to identify and authenticate the users allowed to access the resources.

It is based on a set of rules that determine users' access rights to resources within an organization's system. For example, organizations routinely set policies on the access to the organizations Web sites on the organizations' intranet or Internet. Many organizations, for example, limit the scope and amount, sometimes the times, employees, based on their ranks and roles, can retrieve from the site. Such limits may be specified based on the number of documents that can be downloaded by an employee during a certain time period and on the limit of which part of the Web site such an employee can access.

The role of ACLs has been diminishing because ACLs are ineffective in enforcing policy. When using ACLs to enforce a policy, there is usually no distinction between the policy description and the enforcement mechanism (the policy is essentially defined by the set of ACLs associated with all the resources on the network). Having a policy being implicitly defined by a set of ACLs makes the management of the policy inefficient, error prone, and hardly scalable up to large enterprises with large numbers of employees and resources. In particular, every time an employee leaves a company or even just changes his or her role within the company, an exhaustive search of all ACLs must be performed on all servers, so that user privileges are modified accordingly.

In contrast with ACLs, policy-based access control makes a strict distinction between the formal statement of the policy and its enforcement. It makes rules explicit, and instead of concealing them in ACLs, it makes the policy easier to manage and modify. Its advantage is based on the fact that it administers the concept of least privilege justly because each user can be tied to a role which in turn can be tied to a well-defined list of privileges required for specific tasks in the role. In addition, the roles can be moved around easily and delegated without explicitly de-allocating a user's access privileges [4].

9.2.1.6 Restricted Interfaces

As the commercial Internet grows in popularity, more and more organizations and individuals are putting their data into organization and individual databases and restricting access to it. It is estimated that 88 % of all cyberspace data is restricted data or what is called hidden data [5].

For the user to get access to restricted data, the user has to go via an interface. Any outside party access to restricted data requires a special access request, which many times requires filling in an online form. The interfaces restrict the amount and quality of data that can be retrieved based on filter and retrieval rules. In many cases, the restrictions and filters are instituted by content owners to protect the integrity and proprietary aspects of their data. The Web site itself and the browser must work in cooperation to overcome the over-restriction of some interfaces. Where this is impossible, hidden data is never retrievable.

9.2.1.7 Content-Dependent Access Control

In content-dependent access control, the decision is based on the value of the attribute of the object under consideration. Content-dependent access control is very expensive to administer because it involves a great deal of overhead resulting from the need to scan the resource when access is to be determined. The higher the level of granularity, the more expensive it gets. It is also extremely labor intensive.

9.2.1.8 Other Access Control Techniques and Technologies

Other access control techniques and technologies include those by the US Department of Defense (DoD) that include discretionary access control (DAC), mandatory access control (MAC), context-based access control (CBAC), view-based access control (VBAC), and user-based access control (UBAC).

DAC permits the granting and revoking of access control privileges to be left to the discretion of the individual users. A DAC mechanism departs a little bit from many traditional access control mechanisms where the users do not own the information to which they are allowed access. In DAC, users own the information and are allowed to grant or revoke access to any of the objects under their control.