CHAPTER 2

Access Control Methodologies

After reading this chapter, you will be able to:

- Understand access control basics
- Discuss access control techniques
- Recognize and compare access control models
- Contrast various identification and authentication techniques
- Recognize common attacks and implement controls to prevent them

This chapter presents various methods and techniques for controlling users' access to system resources. You'll learn about different approaches to help ensure that only authorized users can access secured resources. This chapter also covers the basics of access control, general methods and techniques used to manage access to resources, and some common attacks that are launched against access control systems.

2.1 Basics of Access Control

Access control is a collection of methods and components used to protect information assets. Although some information is and should be accessible by everyone, you will most likely need to restrict access to other information. Access control supports both the confidentiality and the integrity properties of a secure system. The confidentiality property protects information from unauthorized disclosure. You use access control to ensure that only authorized users can view information. The integrity property protects information

from unauthorized modification. Access control gives you the ability to dictate what information a user can both view and modify.

Before you can implement a sound access control policy, you must first develop a plan. Here are a few questions you need to answer:

- How do I separate restricted information from unrestricted information?
- What methods should I use to identify users who request access to restricted information?
- What is the best way to permit only users I authorize to access restricted information?
- Where do I start?

2.1.1 Subjects and Objects

Access control is all about, well, controlling access. First, let's define a few terms. The entity that requests access to a resource is called the **subject** of the access. A subject is an active entity because it initiates the access request. The resource a subject attempts to access is called the **object** of the access. The object of an access is the passive part of the access because the subject takes action on the object. So, the goal of a sound access control policy is to allow only authorized subjects to access objects they are permitted to access. It is possible to be an authorized subject but not have access to a specific object.

2.1.2 Least Privilege

Organizations use several general philosophies to design access control rules. The least secure philosophy (read this as "most dangerous") is to give everyone access to all objects by default. Then, you restrict access to only the objects you define as being sensitive. Sounds simple, right? Well it is simple; simple to implement and simple to compromise. The main problem with this philosophy is that you must be absolutely sure you restrict all sensitive objects. This is harder than it sounds. A little sloppy administration can leave large security holes.

Another philosophy, which exists at the opposite end of the spectrum, is much safer and more secure. The philosophy of **least privilege** states that a

subject should be granted only the permissions necessary to accomplish required tasks and nothing more. This approach often requires more administrative maintenance, but it provides more security than more permissive strategies. Least privilege helps to avoid **authorization creep**, which is a condition in which a subject gets access to more objects than was originally intended. The most common causes of authorization creep are ineffective maintenance and poor security philosophy choices.

2.1.3 Controls

Once you decide on the most appropriate access control philosophy for your organization, you can begin to choose the best way to allow subjects to access objects. The mechanisms you put into place to allow or disallow object access are called **controls**. A control is any potential barrier that protects your information from unauthorized access. Controls safeguard your information from threats. There are many types of controls, often organized into several categories. Table 2.1 lists several common control categories.

TABLE 2.1 Common Control Categories

Control Category	Description	Example
Administrative	Policies and procedures designed to enforce security rules	■ Hiring practices ■ Usage monitoring and accounting ■ Security awareness training
Logical (also called technical controls)	Object access restrictions implemented through the use of software or hardware	■ User identification and authentication ■ Encryption ■ Segregated network architecture
Physical	Physical access to hardware limited	■ Fences ■ Walls ■ Locked doors

Sound access control involves choosing the right controls for your organization that will protect and support your security policy.

2.2 Access Control Techniques

You should choose the access control technique that best fits your organization to provide the highest degree of security. Different techniques provide varying levels of security, depending on what the organization needs. In addition to the level of security each technique provides, carefully consider the impact to your users. A grand security scheme will fail if it is so difficult to work with that users commonly try to circumvent it. Consider the techniques covered in the following section, "Access Control Designs," and how each technique could be used in a specific environment. Consider the environmental impact of each technique. Adopt stringent strategies only when absolutely necessary. Remember, a security strategy that is so strict as to encourage users to search for loopholes actually degrades security instead of increasing it.

Each of the following techniques differs in the way objects and subjects are identified, and how decisions are made to approve or deny an access request. First, we look at several models of access control and some of the characteristics of each model. Then we consider and compare several common implementations.

2.2.1 Access Control Designs

An access control design defines rules for users accessing files or devices. We refer to a user, or any entity, that requests access as a subject. Each subject requests access to an entity called an object. An object can be any entity that contains data or resources a subject requests to complete a task. Objects can be files, printers, or other hardware or software entities. The access control type in use for a particular request has the responsibility of evaluating a subject's request to access a particular object and returning a meaningful response. Let's look at three common access control designs.

Mandatory Access Control

Mandatory access control assigns a **security label** to each subject and object. A security label is an assigned level of sensitivity. Some examples of sensitivity levels are public, sensitive, and secret. Tables 2.2 and 2.3 list com-

TABLE 2.2 Military Data Classifications, from Lowest Sensitivity to Highest

Classification	Description
Unclassified	Data that is not sensitive or classified
Sensitive but unclassified (SBU)	Data that could cause harm if disclosed
Confidential	Data for internal use that is exempt from the Freedom of Information Act
Secret	Data that could cause serious damage to national security
Top secret	Data that could cause grave damage to national security

TABLE 2.3 Commercial Data Classifications

Classification	Description
Public	Data not covered elsewhere
Sensitive	Information that could affect business and public confidence if improperly disclosed
Private	Personal information that could negatively affect personnel, if disclosed
Confidential	Corporate information that could negatively affect the organization, if disclosed

mon security labels for military and commercial uses. A subject's security label defines the security clearance, or category of object, that the subject is permitted to access. For example, a subject with a clearance of "secret" can only access objects with a security label of "secret." Some access control methods allow the same subject to access objects of a lower classification, whereas others do not.

One common implementation of mandatory access control is **rule-based access control**. In a rule-based access control system, all access rights are

granted by referencing the security clearance of the subject and the security label of the object. Then, a rule set determines whether an access request should be granted or denied. The rules in place depend on the organization's needs. In addition to matching subject security clearance to an object's security label, many systems require a subject to possess a **need to know**. The need to know property indicates that a subject requires access to an object to complete a task. Thus, access is granted based on both security labels and specific task requirements.

Discretionary Access Control

Discretionary access control uses the identity of the subject to decide whether to grant or reject an access request. The object's owner defines which subjects can access the object, so all access to the object is at the discretion of the object owner. This access control design is generally less secure than mandatory access control, but is the most common design in commercial operating systems. Although it tends to be less secure, it is easier to implement and more flexible for environments that do not require stringent object security. Most objects have permissions, or rights, that specify which users and groups can access the object. This method of granting rights is an example of discretionary access control.

Discretionary access control implementations include identity-based access control and access control lists. **Identity-based access control** makes object access decisions based on a user ID or a user's group membership. An object owner specifies what users or user groups can access each object. When a subject requests access to the object, the subject's credentials are presented and evaluated to grant or deny the request. Most operating systems allow the owners of files and other resources to specify the read, write, and execute permissions based on users and groups. To make the administration a little easier, **access control lists** (**ACLs**) allow groups of objects, or groups of subjects, to be controlled together. An access control list can grant a subject access to a group of objects or grant a group of subjects access to a specific object.

Nondiscretionary Access Control

The third common access control design is **nondiscretionary access control**. This design most commonly uses a subject's role, or a task assigned to the subject, to grant or deny object access. Because nondiscretionary access

control is generally based on roles or tasks, it is also called **role-based access control** or **task-based access control**. This type of access control works well in cases with high turnover or reassignments. When security is associated with a role or task, replacing the person who carries out the task makes security administration easier. At first glance, a role may look like a group, but there are several differences. Although users generally can be associated with multiple groups, users normally are assigned only to a single role. Groups can also represent several types of user associations, but a role represents general tasks a user must perform.

Lattice-based access control is a variation of the nondiscretionary access control design. Instead of associating access rules with specific roles or tasks, each relationship between a subject and an object has a set of access boundaries. These access boundaries define the rules and conditions that allow object access. In most cases, the access boundaries define upper and lower limits that correspond to security classifications and labels.

2.3 Access Control Administration

Once an organization chooses an access control design, the next step is to decide on the method of access control administration. Access control administration can be implemented in both centralized and decentralized modes. It is not uncommon to find hybrid environments where both approaches are used. The best choice of administration style depends on the needs of the organization and the sensitivity of information stored on the affected computer systems.

2.3.1 Centralized Access Control

Centralized access control administration requires that all access requests go through a central authority that grants or denies the request. This approach simplifies administration because objects must be maintained only in a single location. One drawback is that the central access control unit is a single point of failure. If the centralized access control unit fails, no access can be granted to objects, so all objects are effectively unavailable. In addition, the central point of access control can have a negative effect on performance if the system is unable to keep up with all access requests. You can choose from several common packages to implement centralized access control administration.

Remote Authentication Dial-In User Service (RADIUS) provides centralized access control for dial-in users. Users are validated against the user list on the RADIUS server. You can configure the server to hang up and then call the valid user back at a predefined telephone number. Another example of centralized access control for dial-in users is **Challenge Handshake Authentication Protocol (CHAP)**. CHAP presents a challenge when a user requests access. If the user responds to the challenge properly, the access request is granted. CHAP enhances overall security by using encryption during the message exchanges.

Centralized access control for networked applications can use **Terminal Access Controller Access Control System (TACACS)**. TACACS provides general centralized authentication and authorization services. EXtended TACACS (XTACACS) extends TACACS by separating the authentication, authorization, and accounting processes, and TACACS+ adds two-factor authentication.

2.3.2 Decentralized Access Control

Decentralized access control places the responsibility of access control administration closer to the object in question. This approach requires more administration than centralized access control because an object may need to be secured at several locations. It can, however, be more stable because no single point of failure or single point of access exists. Decentralized access control is most often implemented through the use of **security domains**. A security domain is a sphere of trust, or a collection of subjects and objects, with defined access rules or permissions. A subject must be included in the domain to be trusted. This approach makes it fairly easy to exclude an untrusted subject, but makes general administration more difficult due to the fine granularity of security rules.

2.4 Accountability

System auditing assists administrators by keeping logs of activity. These activity logs allow administrators to monitor who is using their systems and how the systems are being used. System logs that are gathered though monitoring can be used to:

- Identify unusual or suspicious activity
- Document usage patterns for possible subsequent action

- Use information to deter future improper actions
- Ensure that users abide by the current security policy

Proper use of the information collected through auditing ensures that each user is accountable for actions performed on or to an information system. Through extensive auditing, all events, whether good or bad, can potentially be traced back to an originating user. The major drawback to complete system auditing is that the process of auditing can have a negative impact on system performance. Administrators must also expend effort to ensure the confidentiality and integrity of sensitive logs.

Many system events can be audited, but prudence often dictates that administrators carefully choose which events to actually audit. One common method to limit the amount of data logged is to use **clipping levels**, which are thresholds for activity that cause no auditing unless exceeded. For example, you may set a clipping level for failed login attempts at 3. If a user fails to log on once or twice, no auditing information is recorded. When the third attempt fails, the third and subsequent failed attempts are logged. This allows administrators to more easily sift through volumes of data and see only the anomalies.

2.5 Access Control Models

Access control models are very useful when deciding what controls are necessary to support your security policy. An access control model provides a conceptual view of your security policy. It allows you to map goals and directives of your security policy to specific system events. This mapping process allows for the formal definition and specification of required security controls. In short, access control models make it possible to decompose complex policies into a series of manageable steps. Many different models have been developed over the years. We look at some of the more important models and discuss some of their unique characteristics in the following sections. Most sound security policy implementations employ a combination of the following access control models.

2.5.1 State Machine Model

A **state machine model** is a collection of defined instances, called states, and specific transitions that permit a modification to occur that changes an object from one state to another. State machines are often used to model

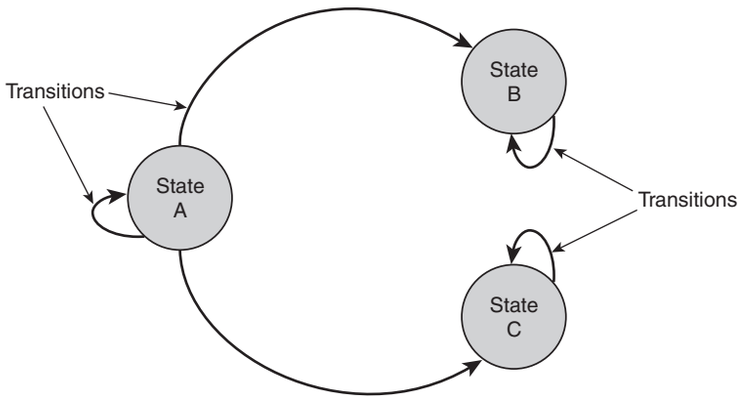

Figure 2.1
Simple state machine

real-life entities when specific states and the transitions from one state to another exist and are understood. Think of a state as being objects at a certain point in time. When a subject requests to read an object, there must be a defined transition that allows an object to change from a closed, unread object to an open object. Figure 2.1 shows a diagram of a simple state machine. States are represented with circles, and transitions are represented with arrows.

The following sections cover four important models: Bell-LaPadula, Biba, Clark-Wilson, and noninterference.

Bell-LaPadula Model

The **Bell-LaPadula model** was developed in the 1970s to help better understand and implement data confidentiality controls. The U.S. military was very interested in protecting classified data while allowing an increasing number of users access to the machines that stored the confidential data. Because the military is most interested in data confidentiality, this model works well in organizations that focus mainly on the confidentiality controls. The Bell-LaPadula model is a state machine model that employs access control lists and security labels to implement object security.

The model uses two basic properties to evaluate access requests. Table 2.4 shows the basic properties and their common names.

The properties may seem confusing at first, but think about what each property states. Remember that confidentiality is the focus. The simple security rule protects information from being disclosed to an unauthorized

TABLE 2.4 Bell-LaPadula Properties

Property	Common Name	Description
Simple security rule	No read up	A subject of a given security clearance cannot read data from a higher security level.
*-property (star property)	No write down	A subject of a given security clearance cannot write to an object at a lower security level.

subject. The *-property protects sensitive or secret data from being inserted into an object of a lower security level. If this were allowed, you could paste a paragraph from a top-secret document into a document that is classified as public. Such a write would disclose the top-secret information to anyone cleared to see public documents. This would clearly violate the confidentiality of the information that was pasted into the public document.

Biba Model

The **Biba model** was developed after the Bell-LaPadula model to address the issue of data integrity. The Biba model is also built on the state machine model and defines states and transitions that focus on the integrity of the data instead of the confidentiality. The Biba model quickly became popular with businesses because its main focus is to ensure that unauthorized subjects cannot change objects.

Similar to the Bell-LaPadula model, the Biba model uses two basic properties to evaluate access requests. Table 2.5 shows the basic Biba properties and their common names.

TABLE 2.5 Biba Properties

Property	Common Name	Description
Simple integrity property	No read down	A subject cannot read an object of a lower integrity level.
*-property (star property)	No write up	A subject cannot write to an object of a higher integrity level.

Clark-Wilson Model

The **Clark-Wilson model** was developed after the Biba model. Unlike the Bell-LaPadula and Biba models, the Clark-Wilson model is not based on the state machine model; it takes a different approach to ensure data integrity. Instead of granting access of a subject to an object, the Clark-Wilson model restricts all accesses to a small number of tightly controlled access programs. The model uses security labels to grant access to objects through the access programs. This approach works well in commercial applications where data integrity is often more important than overall data confidentiality.

The Clark-Wilson model defines several terms that are necessary to understand in order to follow the model's access path:

- **Constrained data item (CDI):** Any data item protected by the model
- **Unconstrained data item (UDI):** Data not protected by the model (for example, data input or output)
- **Integrity verification procedure (IVP):** Procedure that verifies the integrity of a data item
- **Transformation procedure (TP):** Any procedure that makes authorized changes to a data item

The Clark-Wilson model ensures all unconstrained data is validated by the IVP, and then submitted to the system by the TP. All subsequent modifications are first validated by the IVP, and then the modification takes place by the TP. Of course, the IVP and TP are not called until the subject has been properly authenticated and cleared to access the object in question.

Noninterference Model

The last access control model is often an addition to other models. The **noninterference model** ensures that changes at one security level do not "bleed over" into another security level and affect an object in another context. For example, what would happen if you saved a secret document that was embedded in a public document? The dangers in this case are obvious: You risk disclosing secret data when the information is copied to the public document. The basic premise of the noninterference model is that each security level is distinct and changes will not interfere across levels. This assurance reduces the scope of any change and reduces the possibility that a change will

have unintended side effects. By isolating modifications to a specific security level, this model can maintain both data integrity and confidentiality.

2.6 Identification and Authentication Methods

The first user interface element most subjects encounter when accessing an information system is the **identification** and **authentication** challenge. The identification phase allows a subject to claim to be a specific entity by presenting identifying credentials. These credentials could be as simple as a user ID or personal identification number (PIN), or more complex, such as a physical attribute. Once a subject has claimed an identity, the system validates that the user exists in the user database, and then authenticates that the subject really is who she claims to be. The authentication phase asks the subject to present additional information that matches stored information for that subject. These two phases, often called **two-factor authentication**, provide reasonable protection from unauthorized subjects accessing a system. After a subject has been authenticated, the access control system then evaluates the specific rights or permissions for the subject to grant or deny object access requests. This phase is called the authorization phase.

There are three general categories, or types, of authentication information. Best security practices generally dictate that the identification and authentication phases require input from at least two different types. Table 2.6 lists and describes the three common types of authentication data.

The most common and easiest type of authentication to implement is Type 1 authentication. All you have to do is ask the subject to make up a password, passphrase, or PIN. The alternative is to provide one for the user. The

TABLE 2.6 Authentication Types

Authentication Type	Description	Examples
Type 1	What you know	Password, passphrase, PIN, lock combination
Type 2	What you have	Smart card, token device
Type 3	What you are	Biometrics—fingerprint, palm print, retina/iris pattern, voice pattern

difficulty with Type 1 authentication is that you must encourage subjects to create challenge phrases that are very difficult for others to guess, but not so complex that they cannot be easily remembered. If your requirements are so stringent that passwords (or passphrases or PINs) cannot easily be remembered, you will start to see notes stuck to monitors and keyboards with passwords written on them. That negates any value of the password. The same result can occur when administrators require that passwords be changed so often users do not have time to memorize the new ones. Keep passwords safe and secret. The following rules are a good starting point for creating secure passwords:

- Passwords should be at least six characters in length.
- Passwords should contain at least one number or punctuation character.
- Do not use dictionary words or combinations of dictionary words.
- Do not use common personal data, such as birth date, social security number, family member or pet name, or favorite song or hobby.
- Never write down your password.
- Try to make your password easy to remember but hard to guess.

Type 2 authentication data solutions are more complex to administer because subjects are required to carry a device of some sort. The device generally is electronic in nature and either generates a time-sensitive value or generates a value in response to input data. Although Type 2 authentication is more complex, it is almost always more secure than Type 1 authentication.

The most sophisticated authentication type is Type 3, or **biometrics**. Biometrics describes the detection and classification of physical attributes. There are many different biometric techniques, including:

- Fingerprint/palm scan
- Hand geometry
- Retina/iris scan
- Voice print
- Signature/keyboard dynamics

Due to the complexity of biometrics, it is the most expensive authentication type to implement. It is also more difficult to maintain due to the

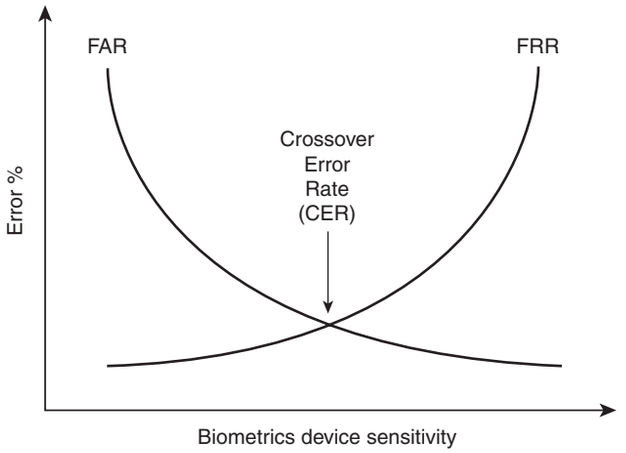

Figure 2.2
Biometrics errors

imperfect nature of biometrics analysis. You should be aware of several important issues regarding biometrics errors. First, a biometrics system could reject an authorized subject. The rate at which this failure occurs is called the **false rejection rate** (**FRR**). On the other hand, the biometrics system could accept an invalid subject. The rate at which this failure occurs is called the **false acceptance rate** (**FAR**). The problem is that when you adjust the sensitivity of the biometrics system to reduce the FRR, the FAR increases. The inverse is also true. So, what is the best setting? The best balance between the FRR and FAR occurs when the rates are equal. This occurs at the **crossover error rate** (**CER**). Figure 2.2 shows the CER in relation to the FRR and FAR of a general biometrics device.

2.6.1 Single Sign-On

The more pieces of information, or factors, you request from a subject, the more assured you can be that the subject is who she claims to be. Thus, two-factor authentication is more secure than single-factor. The problem is that if a subject needs to access several resources on different systems, she may be required to provide identification and authentication information at each different system. This quickly becomes tedious. **Single sign-on** (**SSO**) systems avoid multiple logins by positively identifying a subject and allowing the authentication information to be used within a trusted system or group of systems. Users love SSO, but administrators have a lot of additional work to do. You must take extreme care to ensure the authentication credentials are not compromised or intercepted as they pass across the network.

Several good SSO systems are in use today. It is not important to understand the details of each one. The important concepts and difficulties are fairly common to all SSO products. We look at one product, Kerberos, to examine how these systems work.

2.6.2 Kerberos

The **Kerberos** system came from the Massachusetts Institute of Technology's (MIT's) project Athena. It is named after the three-headed dog from Greek mythology that guards the gates of the underworld. Kerberos provides both authentication and message protection. It uses symmetric key cryptography (both sides have the same key) to encrypt messages. The encryption feature provides end-to-end security, meaning the intermediate machines between the source and target machines cannot see the contents of messages. Kerberos is growing in popularity for use in distributed systems. Although it works well in distributed environments, Kerberos itself uses a centralized server to store the cryptographic keys.

Kerberos includes a data repository and authentication process. The **Key Distribution Center (KDC)** is at the heart of Kerberos. The KDC stores all of the cryptographic keys for subjects and objects. The KDC is responsible for maintaining and distributing these keys, as well as for providing authentication services. When the KDC receives a request for access to an object, it calls the **Authentication Service (AS)** to authenticate the subject and its request. If the subject's request is authenticated, the AS creates an access **ticket** that contains keys for the subject and the object. It then distributes the keys to both the subject and the object. Here are the basic steps in a Kerberos access request cycle:

1. The subject requests access to an object. The subject's Kerberos software prompts for a user ID, and sends the user ID along with the request to the KDC.
2. The KDC calls the AS to authenticate the subject and the object.
3. If authenticated, the KDC sends an encrypted session key to the subject and the object's machine.
4. The subject's Kerberos client software prompts the subject for a password and uses it, along with the subject's secret key, to decrypt the session key.

5. The subject then sends the access request with the session key to the object.
6. The object decrypts the session key it received from the KDC and compares it to the session key it received with the access request.
7. If the two session keys match, access is granted.

The centralized nature of the KDC exposes one of Kerberos' main weaknesses: The KDC is a single point of failure. KDC failure means object access failure. The KDC can also cause a performance bottleneck on heavily utilized machines. Also, there is a small window of time when the session key lives on the client machines. It is possible for an intruder to capture this key and gain unauthorized access to a resource. In spite of several weaknesses, Kerberos is a good example of SSO systems and has enjoyed widespread acceptance.

2.7 File and Data Ownership

Files and data may contain important and valuable information. This important information should be the focus of your security efforts. But who is responsible for ensuring the security of your organization's information? This question is answered by assigning different layers of responsibility to each piece of important information. Each file, or data element, should have at least three different responsible parties assigned. The three layers of responsibility represent different requirements and actions for each group. The most common layers are **data owner**, **data custodian**, and **data user**. Each layer has specific expectations to support the organization's security policy.

2.7.1 Data Owner

The data owner accepts the ultimate responsibility for the protection of the data. The data owner is generally a member of upper management and acts as the representative of the organization in this duty. It is the owner who sets the classification level of the data and delegates the day-to-day responsibility of maintenance to the data custodian. If a security violation occurs, it is the data owner who bears the brunt of any negligence issues.

2.7.2 Data Custodian

The data owner assigns the data custodian to enforce security policies according to the data classification set by the data owner. The custodian is often a member of the IT department and follows specific procedures to secure and protect assigned data. This includes implementing and maintaining appropriate controls, taking backups, and validating the integrity of the data.

2.7.3 Data User

Finally, the users of data are the ones who access the data on a day-to-day basis. They are charged with the responsibility of following the security policy as they access data. You would expect to see more formal procedures that address important data, and users are held accountable for their use of data and adherence to these procedures. In addition to a commitment to follow security procedures, users must be aware of how important security procedures are to the health of their organization. All too often, users use shortcuts to bypass weak security controls because they lack an understanding of the importance of the controls. An organization's security staff must continually keep data users aware of the need for security, as well as the specific security policy and procedures.

2.8 Related Methods of Attacks

The main purpose for implementing access controls is to block unauthorized access to sensitive objects. The primary purpose of attackers is to access these same objects. Several attack types are related to access controls. Most attacks directed toward access controls are designed to thwart, or bypass, the controls and allow access to unauthorized subjects. One of the best ways to decide which controls to put into place is to understand the nature of the attack you are trying to stop. Let's take a look at three of the most common access control attacks.

2.8.1 Brute Force Attack

Brute force attacks are fairly unsophisticated attacks that can be effective. The purpose of such an attack is to attempt every possible combination of characters to satisfy Type 1 authentication. Often called password guessing, a program submits many login attempts, each with a slightly different password. The hope is that the program will hit on the correct

password before anyone notices an attack is underway. One variation of the brute force attack is **war dialing**, in which a program dials a large group of telephone numbers and listens for a modem to answer. When the war dialing program finds a modem, the number is logged for later probing and attacks. These attacks are called brute force attacks because they attempt a very large number of possibilities to find the password or access number.

The best defense is a good offense. A great way to protect your system from a brute force attack is to run one yourself. It is a good idea to run a password cracking or war dialing program against your system periodically. Make sure you have written permission to execute the attack first. You could find that you are violating your security policy as you try to protect it. Once is not enough. Any time a user gets tired of a password or finds that getting access to the Internet is too hard, you will start seeing easily cracked passwords and unauthorized modems showing up. Run your brute force attacks periodically to find users who are taking shortcuts.

In addition to running your own attacks, set your monitoring system clipping levels to warn you when unusual activity occurs. It is also a good idea to set aggressive lockout levels so accounts are locked after a certain number of login failures. As frustrating as this is to honest users who have forgotten passwords, it provides a great defense against brute force attacks.

2.8.2 Dictionary Attack

A **dictionary attack** is actually a subset of a brute force attack. Instead of trying all password combinations, a dictionary attack attempts to satisfy a password prompt by trying commonly used passwords from a list, or dictionary. Many lists of commonly used user IDs and passwords exist and are easy to find. Although they make great input sources for dictionary attacks, they also provide examples of user IDs and passwords to avoid. In fact, one of the best deterrents to a dictionary attack is a strong password policy. A password policy tells users how to construct passwords and what types of passwords to avoid. You can avoid having nearly all of your passwords appear in a password dictionary by creating and enforcing a strong password policy. Once passwords are in place, run dictionary attacks periodically. These attacks are not as intensive as brute force attacks and give you a good idea who is abiding by your password policy. You can also avoid password disclosure by never sending passwords as clear text. Avoid using

HTTP or Telnet for that reason. When you need to send a password to a Web application, use another protocol, such as HTTP-S.

2.8.3 Spoofing Attack

Another interesting type of access control attack is **login spoofing**. An attacker can place a fake login program that prompts a user for a user ID and password. It probably looks just like the normal login screen, so the user likely provides the requested information. Instead of logging the user into the requested system, the bogus program stores or forwards the stolen credentials, then returns a notice that the login has failed. The user is then directed to the real login screen. The beauty of the approach is that few of us would ever think of a spoofing attack if we were presented with a failed login screen. Most of us would chalk it up to a typo.

The best defense against this type of attack is to create trusted paths between users and servers when at all possible. Attempt to minimize the opportunities for attackers to step in between users and servers. In environments where security is extremely important, users should carefully examine all failed login attempts and ensure the failure is properly recorded and reported. If, after being alerted your login has failed, you find that the system thinks the last login failure happened last week, you may have been spoofed. Security awareness goes a long way in preventing and detecting these types of attacks.

2.9 Chapter Summary

- Access control supports data confidentiality and data integrity.
- The least privilege philosophy states that a subject should be granted only the permissions necessary to accomplish required tasks and nothing more.
- A control is any potential barrier you put into place that protects your information.
- Mandatory access control, also called rule-based access control, uses security labels to grant or deny access requests.
- Commercial and military organizations have similar but distinct data classifications.
- Discretionary access control, also called identity-based access control, uses the subject's identity to grant or reject an access request.

- Nondiscretionary access control, also called role-based or task-based access control, uses roles or tasks, as opposed to a subject's identity, to grant or deny access requests.
- Access controls can be centralized, such as RADIUS, CHAP, and TACACS, or decentralized, as with security domains.
- All users of secured information systems are subject to monitoring to ensure they are accountable for all actions.
- Several theoretical access control models help visualize object access issues.
- The Bell-LaPadula model is a state machine model that supports data confidentiality.
- The Biba model is also a state machine model that supports data integrity.
- The Clark-Wilson model supports data integrity by limiting the procedures that can modify data items.
- The noninterference model ensures that changes at one security level have no effect on data at a different security level.
- Identification is a subject claiming to be a specific identity.
- Authentication is the process of validating that a subject is who she claims to be.
- Type 1 authentication is something you know, Type 2 authentication is something you have, and Type 3 authentication is something you are.
- Data owners, custodians, and users each have responsibilities to maintain the security of data.

2.10 Key Terms

access control list (ACL): 1. A list used to grant a subject access to a group of objects or to grant a group of subjects access to a specific object. 2. A list of resources and the users and groups allowed to access them. It is the primary storage mechanism of access permissions in a Windows system.

authentication: A subject provides verification that he is who he claims to be.

Authentication Service (AS): A process in the Kerberos KDC that authenticates a subject and its request.

authorization creep: A condition under which a subject gets access to more objects than was originally intended.

Bell-LaPadula model: An access control model developed in the 1970s to help better understand and implement data confidentiality controls.

Biba model: An access control model developed after the Bell-LaPadula model to address the issue of data integrity.

biometrics: The detection and classification of physical attributes.

brute force attack: An access control attack that attempts all possible password combinations.

centralized access control administration: All access requests go through a central authority that grants or denies the request.

Challenge Handshake Authentication Protocol (CHAP): A centralized access control system that provides centralized access control for dial-in users.

Clark-Wilson model: An access control model that addresses data integrity by restricting all object accesses to a small number of tightly controlled access programs.

clipping levels: Thresholds for activity that trigger auditing activity when exceeded.

constrained data item (CDI): Any data item protected by the Clark-Wilson model.

control: Any potential barrier that protects your information from unauthorized access.

crossover error rate (CER): The point where FRR = FAR.

data custodian: Generally, an IT person who is assigned by the data owner to enforce security policies according to the data classification set by the data owner.

data owner: A member of upper management who accepts the ultimate responsibility for the protection of the data.

data users: System users who access the data on a day-to-day basis.

decentralized access control: Places the responsibility of access control administration close to the object in question.

dictionary attack: An access control attack that attempts passwords from a dictionary of commonly used passwords.

discretionary access control: Object access decisions based on the identity of the subject requesting access.

false acceptance rate (FAR): The rate at which invalid subjects are accepted.

false rejection rate (FRR): The rate at which valid subjects are rejected.

identification: 1. The phase in which a subject claims to be a specific identity. 2. The act of verifying a subject's identity.

identity-based access control: Object access decisions based on a user ID or a user's group membership.

integrity verification procedure (IVP): A procedure that verifies the integrity of a data item.

Kerberos: A popular SSO system that provides both authentication and message protection.

Key Distribution Center (KDC): The network service and data repository that stores all the cryptographic keys for subjects and objects in a Kerberos system.

lattice-based access control: A variation of the nondiscretionary access control model that establishes each relationship between a subject and an object with a set of access boundaries.

least privilege: A philosophy in which a subject should be granted only the permissions needed to accomplish required tasks and nothing more.

login spoofing: An access control attack that replaces a valid login screen with one supplied by an attacker.

mandatory access control: A system-enforced access control mechanism that assigns a security label, which defines the security clearance, to each subject and object.

need to know: A condition when a subject requires access to an object to complete a task.

nondiscretionary access control: Uses a subject's role, or a task assigned to the subject, to grant or deny object access.

noninterference model: An access control model that ensures that changes at one security level do not "bleed over" into another security level and affect an object in another context.

object: The resource a subject attempts to access.

Remote Authentication Dial-In User Service (RADIUS): A centralized access control system that provides centralized access control for dial-in users.

role-based access control: A nondiscretionary access control method that uses a subject's role to grant or deny object access.

rule-based access control: All access rights are decided by referencing the security clearance of the subject and the security label of the object.

security domain: A sphere of trust, or a collection of subjects and objects with defined access rules or permissions.

security label: An assigned level of sensitivity.

single sign-on (SSO): A system that avoids multiple logins by positively identifying a subject and allowing the authentication information to be used within a trusted system or group of systems.

state machine model: A collection of defined instances, called states, and specific transitions that permit a modification to occur that changes an object from one state to another.

subject: The entity that requests access to a resource.

task-based access control: A nondiscretionary access control method that uses the task a subject is working on to grant or deny object access.

Terminal Access Controller Access Control System (TACACS): A centralized access control system that provides centralized access control for networked users.

ticket: A Kerberos authentication message that contains keys for the subject and the object.

transformation procedure (TP): Any procedure that makes authorized changes to a data item.

two-factor authentication: A process of providing two pieces of information to authenticate a claimed identity.

unconstrained data item (UDI): Any data not protected by the Clark-Wilson model.

war dialing: Automated dialing of many telephone numbers searching for a modem.

2.11 Challenge Questions

2.1 What is the access control subject?

a. The passive entity that is the target of an access request

b. The active entity that initiates an access request

c. A specific type of access requested

d. The authentication service that processes the access request

2.2 Which statement best describes the principle of least privilege?

a. Only allow the minimum number of defined users to access a system.

b. An object should allow only data owners to access it.

c. A subject should be granted only the permissions to accomplish a task and nothing more.

d. An object should grant access only to subjects through one model and nothing more.

2.3 What is a control?

a. Any potential barrier that protects your information from unauthorized access

b. Any data source that contains sensitive data

c. A user or program that attempts to access data on a secure system

d. A device for setting the security clearance of data

2.4 Which of the following are logical controls? (Choose all that apply.)

a. Hiring practices

b. Encryption

c. Walls

d. User identification and authentication

2.5 What two terms mean access control defined by the security clearance of the subject and the security label of the object?

a. Discretionary access control

b. Mandatory access control

c. Rule-based access control

d. Role-based access control

2.6 What type of model is identity-based access control?

a. Mandatory access control

b. Discretionary access control

c. Nondiscretionary access control

d. Transitive-discretionary access control

2.7 What are two types of nondiscretionary access control?

a. Role-based access control

b. Identity-based access control

c. Rule-based access control

d. Task-based access control

2.8 Access and activity monitoring supports what security principle?

a. Availability

b. Least privilege

c. Accountability

d. Liability

2.9 Which access control models primarily support data integrity? (Choose all that apply.)

a. Bell-LaPadula

b. Biba

c. Clark-Wilson

d. State machine

2.10 What is the best definition for the term *authentication*?

a. A subject presents credentials to claim an identity.

b. The access control system looks up permissions assigned to a subject.

c. The access control system searches a user database to see if the subject exists.

d. A subject provides additional information that should match information the access control system stores for that subject.

2.11 What types of authentication do you use when you withdraw cash from an automated teller machine (ATM)?

a. Type 1 and Type 2

b. Type 1 and Type 3

c. Type 1

d. Type 2

2.12 What is the rate at which a biometric device rejects valid subjects?

a. FAR

b. FRR

c. CER

d. CDC

2.13 What is an SSO system?

a. Single sign-on

b. Single secure opening

c. Secure signal operation

d. Single secure operation

2.14 Who is ultimately responsible for the protection of data?

a. Data user

b. Data custodian

c. Data owner

d. Data security administrator

2.15 Which type of attack uses a list of common passwords?

a. Brute force attack

b. Spoofing attack

c. Dictionary attack

d. Smurf attack

2.12 Challenge Exercises

Challenge Exercise 2.1

This exercise directs you to a common repository of security-related reports. Because the security profession constantly changes, professionals must continuously strive to stay up to date. Reading rooms and peer reports offer a great way to keep current. In this exercise, you visit a popular online reading room and review a report submitted by a security professional. You need a computer with a Web browser and Internet access. The Web site you visit is the SANS (SysAdmin, Audit, Network, Security) Institute reading room. Security practitioners who pursue certification through SANS must submit at least one current and relevant report for publication. These reports offer excellent information to help you learn more about security.

2.1 In your Web browser, enter the following address: *http://www.sans.org/rr.*

2.2 From the SANS InfoSec Reading Room page, click the "Authentication" link in the Category section.

2.3 Select and read one or more reports and write a brief summary.

Challenge Exercise 2.2

This exercise examines a current security add-on for the Linux environment. It is not necessary for you to have any Linux experience to complete this exercise. The purpose is to look at a viable product and get a better understanding of how real operating systems implement access control. You need a computer with a Web browser and Internet access. The Web site you visit is the Rule Set Based Access Control (RSBAC) for Linux Web site. The RSBAC product implements discretionary access control for Linux systems.

2.1 In your Web browser, enter the following address: *http://www.rsbac.org.*

2.2 Visit both the "What is RSBAC?" and "Why you need RSBAC" pages.

2.3 Write a two- or three-paragraph summary of how RSBAC could increase the security of a commercial Linux system.

Challenge Exercise 2.3

This exercise examines an access control implementation for Microsoft Windows XP Professional. You do not need extensive Windows experience to complete this exercise. The purpose is to look at a viable product and get a better understanding of how real operating systems implement access control. You need a computer with a Web browser and Internet access. The Web site you will visit is the Microsoft Corporation Web site.

2.1 In your Web browser, enter the following address: *http://www.microsoft.com/windowsxp/pro/using/howto/security/accesscontrol.asp.*

2.2 Read the description of Windows XP access control.

2.3 Write a two- or three-paragraph summary of how Windows XP implements access control.

2.4 Create a list of at least five new user groups you would need for a commercial system.

2.5 Assign privileges to your user groups and explain the purpose of each group.

2.13 Challenge Scenarios

Challenge Scenario 2.1

You are the new Chief Information Officer (CIO) for Spatula City, Inc., the leading wholesaler of spatulas of every shape and size. Because Spatula City provides many products to competing retailers, information security is important. (Imagine what could happen if details of a new spatula design leaked out!) Your job is to implement a security strategy that will satisfy Spatula City's security requirements.

The Spatula City system will contain a central database and be connected to the corporate intranet. In-house users will connect and need access to both sensitive and public resources. Additionally, outside sales representatives will need access through dial-up connections, and customers will access some product information through the company's Web site.

Select an access control design to use and explain how you plan to implement access control. Explain your choices and describe the authentication techniques that will provide the best security for your users.

Challenge Scenario 2.2

You are a security manager for Doorknobs-Are-Us, a manufacturer of custom doorknobs and miscellaneous door fixtures and hardware. Your company just landed a new contract to manufacture doorknobs and locks for a retrofit of military ships. The new locks contain sensitive technology, so all data that pertains to this project must be protected. Your engineers need to access pertinent data stored in the corporate database and document management system from various remote locations. You need to select an authentication technique that will ensure only authorized engineers can access this sensitive data.

Select appropriate authentication controls that will uniquely identify an engineer and explain your choices. As you consider alternatives, explain why you would rule out those solutions that you do not choose.