Access Control

THE CISSP EXAM TOPICS COVERED IN THIS CHAPTER INCLUDE:

1. Access Control
   A. Control access by applying the following concepts/methodology/techniques:
      A.1 Policies
      A.2 Types of controls (preventive, detective, corrective, etc.)
      A.3 Techniques (e.g., nondiscretionary, discretionary, and mandatory)
      A.4 Identification and authentication
      A.5 Decentralized/distributed access control techniques
      A.6 Authorization mechanisms
   D. Identity and access provisioning lifecycle (e.g., provisioning, review, revocation)
The Access Control domain in the Common Body of Knowledge (CBK) for the CISSP certification exam deals with topics and issues related to granting and revoking the right to access data or perform an action on a system. Generally, an access control is any hardware, software, or organizational administrative policy or procedure that performs the following tasks:

- Identifies users or other subjects attempting to access resources.
- Determines whether the access is authorized.
- Grants or restricts access.
- Monitors and records access attempts.

In this chapter and in Chapter 2, “Access Control Attacks and Monitoring,” we discuss the Access Control domain. Be sure to read and study the materials from both chapters to ensure complete coverage of the essential material for this domain of the CISSP certification exam objectives.

Access Control Overview

Controlling access to resources is one of the central themes of security. Access control addresses more than just which users can access which files or services. It is about the relationships between entities (that is, subjects and objects). The transfer of information from an object to a subject is called access, which makes it important to understand the definition of both subject and object.

**Subject**  A subject is an active entity that accesses a passive object to receive information from, or data about, an object. Subjects can be users, programs, processes, computers, or anything else that can access a resource. When authorized, subjects can modify objects.

**Object**  An object is a passive entity that provides information to active subjects. Some examples of objects include files, databases, computers, programs, processes, printers, and storage media.

You can often simplify these access control topics by substituting the word user for subject and the word file for object. For example, instead of a subject accesses an object, you can think of it as a user accesses a file. However, it’s also important to remember that subjects comprise more than users and objects comprise more than just files.
You may have noticed that some examples, such as programs and computers, are listed as both subjects and objects. This is because the roles of subject and object can switch back and forth. In many cases, when two entities interact, they perform different functions. Sometimes they may be requesting information and other times providing information. The key difference is that the subject is always the active entity that receives information about, or data from, the passive object. The object is always the passive entity that provides or hosts the information or data.

For example, consider a common web application that provides dynamic web pages to users. Users query the web application to retrieve a web page, so the application starts as an object. The application then switches to a subject role as it queries the user’s computer to retrieve a cookie and then queries a database to retrieve information about the user based on the cookie. Finally, the application switches back to an object as it sends back the dynamic web page.

Access control is not limited to logical and technical applications. It also applies to physical security and can involve controlling access to entire complexes, entire buildings, or even individual rooms.

**Users, Owners, and Custodians**

When discussing access to objects, three subject labels are used: user, owner, and custodian.

**User** A user is any subject who accesses objects on a system to perform some action or accomplish a work task.

**Owner** An owner, or information owner, is the person who has final organizational responsibility for classifying and labeling objects and protecting and storing data. The owner may be liable for negligence if they fail to perform due diligence in establishing and enforcing security policies to protect and sustain sensitive data.

**Custodian** A custodian is a subject who has been assigned or delegated the day-to-day responsibility of properly storing and protecting objects.

A user is any end user on the system. The owner is typically the CEO, president, or department head. The custodian is typically the Information Technology (IT) staff or the system security administrator.

**The CIA Triad**

One of the primary reasons that access control mechanisms are implemented is to prevent losses. There are three categories of IT loss: loss of *confidentiality*, loss of *availability*, and loss of *integrity*. Protecting against these losses is so integral to IT security that they are frequently referred to the *CIA Triad* (or sometimes the AIC Triad or Security Triad).

Confidentiality  Access controls help ensure that only authorized subjects can access objects. When unauthorized entities are able to access systems or data, it results in a loss of confidentiality.

Integrity  Integrity ensures that data or system configurations are not modified without authorization. If unauthorized or unwanted changes to objects occur, or go undetected, this is known as loss of integrity.

Availability  Authorized requests for objects must be granted to subjects within a reasonable amount of time. In other words, systems and data should be available to users and other subjects when they are needed. If the systems are not operational, or the data is not accessible, this is a loss of availability.

Policies

A security policy is a document that defines the security requirements for an organization. It identifies assets that need protection and the extent to which security solutions should go to protect them. Some organizations create a security policy as a single document and other organizations create multiple security policies with each one focused on a separate area. (Security policies are explored in greater depth in Chapter 5.)

Policies are an important element of access control because they help personnel within the organization understand what security requirements are important. The security policy is created or approved by senior leadership, and it provides a broad overview of an organization’s security needs but usually does not go into details about how to fulfill the needs. For example, it may state the need to implement and enforce separation of duties and least privilege principles but not state how to do so. Professionals within the organization use the security policies as a guide to implement security requirements. Standards are also created from security policies.

Compare Permissions, Rights, and Privileges

When studying access control topics, you’ll often come across the terms permissions, rights, and privileges. These are sometimes used interchangeably, but they don’t always mean the same thing.

Permissions  In general, permissions refer to the access granted for an object and determine what you can do with it. If you have read permission for a file, you’ll be able to open it and read it. Users may be granted permissions to create, read, edit, or delete a file on a file server. Similarly, users can be granted access rights to a file, so in this context, access rights and permissions are synonymous. For example, you may be granted read and execute permissions for an application file, which gives you the right to run the application. Additionally, you may be granted data rights within a database, allowing you to retrieve or update information in the database.

Rights  A right also refers to the ability to take an action on an object. For example, a user might have the right to modify the system time on a computer or the right to restore backed-up data. This is a subtle distinction and not always stressed, but the right to take action on a system is rarely referred to as a permission.
Privileges  Combined, rights and permissions are commonly referred to as privileges. For example, an administrator for a computer will have full privileges, granting the administrator full rights and permissions on the computer. The administrator will be able to perform any actions and access any data on the computer.

Types of Access Control

The term access control refers to a broad range of controls that perform such tasks as ensuring that only authorized users can log on and preventing unauthorized users from gaining access to resources. Controls mitigate a wide variety of information security risks.

The three primary access control types are preventive, detective, and corrective.

Whenever possible you want to prevent any type of security problem or incident. Of course, this isn’t always possible and unwanted events occur. When they do, you want to detect the event as soon as possible. And once you detect the event, you want to correct it.

There are also four other access control types, commonly known as deterrent, recovery, directive, and compensation access controls.

As you read through the controls in the following sections, you’ll notice that some are listed as an example in more than one access control type. For example, a fence (or perimeter-defining device) placed around a building can be a preventive control (physically barring someone from gaining access to a building compound) and/or a deterrent control (discouraging someone from trying to gain access).

Preventive access control  A preventive access control (sometimes called a preventative access control in CISSP materials) is deployed to thwart or stop unwanted or unauthorized activity from occurring. Examples of preventive access controls include fences, locks, biometrics, mantraps, lighting, alarm systems, separation of duties, job rotation, data classification, penetration testing, access control methods, encryption, auditing, presence of security cameras or closed circuit television (CCTV), smart cards, callback procedures, security policies, security awareness training, antivirus software, firewalls, and intrusion prevention systems.

Detective access control  A detective access control is deployed to discover or detect unwanted or unauthorized activity. Detective controls operate after the fact and can discover the activity only after it has occurred. Examples of detective access controls include security guards, motion detectors, recording and reviewing of events captured by security cameras or CCTV, job rotation, mandatory vacations, audit trails, honeypots or honeynets, intrusion detection systems, violation reports, supervision and reviews of users, and incident investigations.

Corrective access control  A corrective access control modifies the environment to return systems to normal after an unwanted or unauthorized activity has occurred. They attempt to correct any problems that occurred as a result of a security incident. Corrective controls can be simple, such as terminating malicious activity or rebooting a system.
They also include antivirus solutions that can remove or quarantine a virus, backup and restore plans to ensure that lost data can be restored, and active intrusion detection systems that can modify the environment to stop an attack in progress.

Chapter 14, “Incident Management” covers intrusion detection systems and intrusion prevention systems in more depth.

Deterrent access control A deterrent access control is deployed to discourage violation of security policies. Deterrent and preventive controls are similar, but deterrent controls often depend on individuals deciding not to take an unwanted action. In contrast, a preventive control actually blocks the action. Some examples include policies, security awareness training, locks, fences, security badges, guards, mantraps, and security cameras.

Recovery access control A recovery access control is deployed to repair or restore resources, functions, and capabilities after a violation of security policies. Recovery controls are an extension of corrective controls but have more advanced or complex abilities. Examples of recovery access controls include backups and restores, fault-tolerant drive systems, system imaging, server clustering, antivirus software, and database or virtual machine shadowing.

Directive access control A directive access control is deployed to direct, confine, or control the actions of subjects to force or encourage compliance with security policies. Examples of directive access controls include security policy requirements or criteria, posted notifications, escape route exit signs, monitoring, supervision, and procedures.

Compensation access control A compensation access control is deployed to provide various options to other existing controls to aid in enforcement and support of security policies. They can be any controls used in addition to, or in place of, another control. For example, an organizational policy may dictate that all personally identifiable information (PII) must be encrypted. A review discovers that a preventive control is encrypting all PII data within databases, but PII transferred over the network is sent in cleartext. A compensation control would be added to protect the data in transit.

The terms types and categories are sometimes used interchangeably when grouping controls. For example, the CISSP Candidate Information Bulletin (CIB) lists “types of controls” as “preventive, detective, corrective,” but many other sources identify these as categories of controls instead of types. Similarly, other sources identify administrative, technical, and physical controls as access control types instead of categories. For the exam, it isn’t important to know if a control grouping is a type or category, but you should be able to differentiate between the meanings of the different controls.
Access controls are also categorized by how they are implemented. Controls can be implemented administratively, logically/technically, or physically. Any of the access control types mentioned previously can include any of these types of implementation.

**Administrative controls**  Administrative access controls are the policies and procedures defined by an organization’s security policy and other regulations or requirements. They are sometimes referred to as management controls. These controls focus on personnel and business practices. Examples of administrative access controls include policies, procedures, hiring practices, background checks, data classifications and labeling, security awareness and training efforts, vacation history, reports and reviews, work supervision, personnel controls, and testing.

**Logical/technical controls**  Logical access controls (also known as technical access controls) are the hardware or software mechanisms used to manage access and to provide protection for resources and systems. As the name implies, they use technology. Examples of logical or technical access controls include authentication methods (such as usernames, passwords, smart cards, and biometrics), encryption, constrained interfaces, access control lists, protocols, firewalls, routers, intrusion detection systems, and clipping levels.

**Physical controls**  Physical access controls are items you can physically touch. They include physical mechanisms deployed to prevent, monitor, or detect direct contact with systems or areas within a facility. Examples of physical access controls include guards, fences, motion detectors, locked doors, sealed windows, lights, cable protection, laptop locks, badges, swipe cards, guard dogs, video cameras, mantraps, and alarms.

When preparing for the CISSP exam, you should be able to easily identify the type of any control. For example, you should recognize that a firewall is a preventive control because it can prevent attacks by blocking traffic, while an intrusion detection system (IDS) is a detective control because it can detect attacks in progress or after they’ve occurred. You should also be able to identify both as logical/technical controls.

**Defense in Depth**

Access controls are implemented using a defense-in-depth strategy, in which multiple layers or levels of access controls are deployed to provide layered security. As an example, consider Figure 1.1. It shows two servers and two disks to represent assets owned by an organization that need to be protected. Intruders or attackers need to overcome multiple layers of defense to reach these protected assets.
Controls are implemented using multiple methods. You can’t depend on technology alone to provide security; you must also use physical access controls and administrative access controls. For example, if a server has strong authentication but is stored on an unguarded desk, a thief can easily steal it and take his time hacking into the system. Similarly, users may have strong passwords, but social engineers may trick them into giving up their password if they haven’t been adequately trained.

This concept of defense in depth highlights several important points:

- An organization’s security policy, one of the administrative access controls, provides the first or innermost layer of defense for assets.
- Personnel are a key focus for access controls. Only with proper training and education can they implement, comply with, and support security elements defined in your security policy.
- A combination of administrative, technical, and physical access controls provides a much stronger defense. Using only administrative, only technical, or only physical controls results in weaknesses that attackers can discover and exploit.

**Access Control Elements**

The different security elements that come together to support access control are grouped into four types: identification, authentication, authorization, and accountability. This list provides a short introduction:

**Identification** A subject claims an identity. For example, users claim identities based on usernames.
Authentication  A subject proves a claimed identity. For example, users can prove usernames are theirs by providing a password with the username.

Authorization  Subjects are granted access to objects based on proven identities. For example, a user can be granted access to files based on the user’s proven identity.

Accountability  Users and other subjects can be held accountable for their actions when auditing is implemented. Auditing tracks subjects and records when they access objects, creating an audit trail in one or more audit logs. For example, auditing can record when a user reads, modifies, or deletes a file. Auditing provides accountability.

All four of these elements are needed in an effective access control system. Subjects must be uniquely identified and authenticated before authentication and accountability can occur. When subjects are identified and authenticated, and their actions are recorded in audit logs, they can be held accountable for their actions.

Identification
Identification is the process by which a subject professes an identity and accountability is initiated. For example, a user provides a username, a logon ID, or a smart card to represent an identification process. Similarly, an application can provide a process ID number as identification. Once a subject has identified itself, the claimed identity becomes accountable for any further actions undertaken by that subject. IT systems track activity by identities, not by subjects themselves. A computer doesn’t know one human from another, but it does know that your user account is different from all other user accounts.

Authentication
Authentication is the process of verifying or testing that a claimed identity is valid. Authentication requires that a subject provide additional information that must correspond exactly to the professed identity. An authentication system checks the professed identity and the authentication against a database. If the database includes the identity and the correct authentication is included, the subject is authenticated.

The three basic methods of authentication are also known as types or factors. They are introduced here and expanded in the section “Identification and Authentication Techniques” later in this chapter.

Type 1  A Type 1 authentication factor is something you know. It is any string of characters you have memorized and can reproduce on a keyboard when prompted. Examples include a password, personal identification number (PIN), passphrase, or mother’s maiden name.

Type 2  A Type 2 authentication factor is something you have. It is a physical device that you must have in your possession at the time of authentication. Examples include a token device, smart card, memory card, or USB drive.
Chapter 1 • Access Control

Type 3  A Type 3 authentication factor is *something you are* or *something you do*. It is a physical characteristic of a person identified with different types of biometrics. Examples in the “something you are” category include fingerprints, voice prints, retina patterns, iris patterns, face shapes, palm topology, and hand geometry. Examples in the “something you do” category include signature and keystroke dynamics, also known as behavioral biometrics.

These types are progressively stronger when implemented correctly, with Type 1 being the weakest and Type 3 being the strongest. In other words, passwords (Type 1) are the weakest, and a fingerprint (Type 3) is stronger than a password—but even Type 3 authentication factors can be breached. For example, an attacker may be able to create a duplicate fingerprint on a gummi bear candy and fool a fingerprint reader.

**Somewhere You Are**

These three basic factors (“something you know,” “something you have,” and “something you are”) are the most common elements in authentication systems. However, a factor known as *somewhere you are* is sometimes used. It can identify a subject’s location based on a specific computer, a phone number identified by caller ID, or a country identified by an IP address. Controlling access by physical location forces a subject to be present in a specific location. For example, remote access users may be authorized to dial in from home. Caller ID and callback techniques are used to verify that the user is actually calling from home. “Somewhere you are” is sometimes considered part of Type 2, “something you have.”

This factor isn’t reliable on its own because any type of address information can be spoofed by a dedicated attacker. However, it can be effective when used in combination with other factors.

**Authorization**

*Authorization* indicates who is trusted to perform specific operations. If the action is allowed, the subject is authorized; if disallowed, the subject is not authorized. Here’s a simple example: If a user attempts to open a file, the authorization mechanism checks to ensure that the user has at least read permission on the file.
It’s important to realize that just because users or other entities can authenticate to a system, that doesn’t mean they are given access to anything and everything. Instead, subjects are authorized access to specific objects based on their proven identity. The process of authorization ensures that the requested activity or object access is possible based on the privileges assigned to the subject.

Identification and authentication are “all-or-nothing” aspects of access control. Either a user’s credentials prove a professed identity, or they don’t. In contrast, authorization occupies a wide range of variations. For example, a user may be able to read a file but not delete it or print a document but not alter the print queue.

Accountability
Accountability, which is done via auditing, logging, and monitoring, ensures that subjects can be held accountable for their actions. Auditing is the process of tracking and recording subject activities within logs. Logs typically record who took an action, when and where the action was taken, and what the action was. One or more logs create an audit trail that can be used to reconstruct events and to verify whether a security policy or authorization was violated. When contents of audit trails are reviewed, people associated with the accounts can be held accountable for their actions. (Logging and monitoring is covered in more depth in Chapter 2.)

There’s a subtle but important point to stress about accountability. Accountability relies on effective identification and authentication, but it does not require effective authorization. In other words, if users are adequately identified and authenticated, accountability mechanisms such as audit logs can track their activity, even when they access resources they shouldn’t.

Identification and Authentication Techniques
Identification is a fairly straightforward concept. A subject must provide an identity to a system to start the authentication, authorization, and accountability processes. Providing an identity might entail typing a username, swiping a smart card, waving a token device, speaking a phrase, or positioning your face, hand, or finger for a camera or scanning device. Without an identity, a system has no way to correlate an authentication factor with the subject.

Authentication verifies the identity of the subject by comparing one or more factors against a database of valid identities, such as user accounts. The authentication information used to verify an identity is considered private information. The ability of the subject and system to maintain the secrecy of the authentication information for identities directly reflects the level of security of that system.

Identification and authentication always occur together as a single two-step process. Providing an identity is the first step, and providing the authentication information is the second step. Without both, a subject cannot gain access to a system.
Each authentication technique or factor has unique benefits and drawbacks. Thus, it is important to evaluate each mechanism in light of the environment in which it will be deployed to determine viability.

**Passwords**

The most common authentication technique is the use of a **password** (a string of characters entered by a user) with Type 1 authentication (something you know), but this is also considered the weakest form of protection. Passwords are poor security mechanisms for several reasons:

- Users often choose passwords that are easy to remember and therefore easy to guess or crack.
- Randomly generated passwords are hard to remember; thus, many users write them down.
- Passwords are easily shared, written down, and forgotten.
- Passwords can be stolen through many means, including observation, recording and playback, and security database theft.
- Passwords are sometimes transmitted in cleartext or with easily broken encryption protocols.
- Password databases are sometimes stored in publicly accessible online locations.
- Weak passwords can be discovered quickly in brute-force attacks.

**Password Encryption**

Passwords are rarely stored in plain text. Instead, a system will create a hash of a password using a hashing algorithm such as Message Digest 5 (MD5) or Secure Hash Algorithm 1 (SHA-1). The hash is a number and the algorithm will always create the same number if the password is the same. When a user enters the password for authentication, it is hashed and compared to the stored password’s hash. If they are the same, the user is authenticated.

**Password Selection**

Passwords can be effective if selected intelligently and managed properly. A **password policy** can be part of the organization’s written policy that dictates the requirements for passwords. Many systems also include technical password policies that enforce the **password restriction** requirements. Password policies can, for example, ensure that users change their passwords regularly (a maximum age setting might specify that users must change their password every 45 days). The following list includes some other password policy settings:
Password length  The length is the number of characters in the password. End user passwords should be at least eight characters long, and many organizations require privileged account passwords to be at least 15 characters long. This specifically overcomes a weakness in how passwords are stored in some Windows systems.

Password complexity  The complexity of a password refers to how many character types it includes. An eight-character password using uppercase characters, lowercase characters, symbols, and numbers is much stronger than an eight-character password using only numbers.

Password history  Many users get into the habit of switching between two passwords. A password history remembers a certain number of previous passwords (perhaps six) and prevents users from reusing a password in the history. This is often combined with a minimum password age setting, preventing users from changing a password repeatedly until they can set the password back to the original one. Minimum password age is often set to one day.

However, even with strong software-enforced password restrictions, it remains possible to create passwords that may be easily guessed or cracked. Users don’t always understand the need for strong passwords, or even how to create them. An organization’s security policy will usually stress the need for strong passwords and define the contents of a strong password. If end users create their own passwords, suggestions like the following can help them create strong ones:

- Do not use any part of your name, logon name, email address, employee number, Social Security number, phone number, extension, or other identifying name or code.
- Do not use dictionary words (including words in foreign dictionaries), slang, or industry acronyms.
- Do use nonstandard capitalization and spelling.
- Do switch letters and replace letters with numbers.

In some environments, initial passwords for user accounts are generated automatically. Often the generated password is a form of a composition password, which is constructed from two or more unrelated words joined together with a number or symbol in between. Composition passwords are easy for computers to generate, but they should not be used for extended periods of time because they are vulnerable to password-guessing attacks. If the algorithm for computer-generated passwords is discovered, all passwords created by the system are in jeopardy of being compromised.

Password Phrases

A password mechanism that is more effective than a basic password is a passphrase. A passphrase is a string of characters similar to a password but it has unique meaning to the user. Passphrases are often basic sentences modified to simplify memorization. Here’s an example: “I passed the CISSP exam” can be converted to the following passphrase: “IP@$edTheCISSPEx@m.” Using a passphrase has several benefits. It is difficult to crack a passphrase using a brute-force tool, and it encourages the use of a lengthy string with numerous characters, but it is still easy to remember.
Cognitive Passwords

Another interesting password mechanism is the cognitive password. A cognitive password is usually a series of questions about facts or predefined responses that only the subject should know. For example, three to five questions such as these might be asked of the subject:

- What is your birth date?
- What is your mother’s maiden name?
- What is the name of your division manager?
- What was your score on your last evaluation exam?
- Who was your favorite player in the 1984 World Series?

If all questions are answered correctly, the subject is authenticated. The most effective cognitive password systems ask a different set of questions each time. The primary limitation for cognitive password systems is that each question must be answered at the time of user enrollment (in other words, user account creation) and answered again during the logon process, which increases the time to complete that process.

Cognitive passwords are often employed to assist with password management using self-service password reset systems or assisted password reset systems. For example, if users forget their original password, they can ask for help. The password management system can then challenge the user with one or more of these cognitive password questions presumably known only by the user. If the user answers correctly, the user is either provided with the original password or granted the ability to change the password.

One of the flaws associated with cognitive passwords is that the information is often easily available via the Internet. For example, an attacker broke into Sarah Palin’s personal Yahoo! email account when she was a vice presidential candidate in 2008. He accessed biographical information about her that he found on the Internet and was able to answer questions posed by Yahoo!’s account recovery process.

Smart Cards and Tokens

Smart cards and tokens (or smart tokens) are both examples of a Type 2, or “something you have,” factor of authentication. They are rarely used by themselves but are commonly combined with another factor of authentication, providing multifactor authentication.

Smart Cards

A smart card is a credit-card-sized ID or badge and has an integrated circuit chip embedded in it. Smart cards contain information about the authorized bearer that can be used for identification and/or authentication purposes. Most current smart cards include a microprocessor and one or more certificates. The certificates are used for asymmetric cryptography such
as encryption and digitally signing email. (Asymmetric cryptography topics are covered in more depth in Chapter 10, “PKI and Cryptographic Applications.”) Smart cards are tamper resistant and provide users with an easy way to carry and use complex encryption keys.

Users insert the card into a smart card reader when authenticating. It's common to require users to also enter a PIN or password as a second factor of authentication with the smart card.

Note that smart cards can provide both identification and authentication. Because users could share or swap smart cards, they aren’t effective identification methods by themselves. Another authentication factor must be used. Smart cards are almost always used with PINs as a secondary factor to improve their security value.

Personnel within the US government use either common access cards (CACs) or Personal Identity Verification (PIV) cards. CACs and PIV cards are smart cards that include pictures and other identifying information about the owner. Users wear them as a badge while walking around and insert them into card readers at their computer when logging on. Chapter 19, “Physical Security Requirements,” has more information on smart cards.

Tokens
A token, or token device, is a password-generating device that users can carry with them. A common token used today includes an LCD that displays a number that is used as a password and changes at a fixed time interval, such as every 60 seconds. This number is derived from several elements, including a unique token device identifier, a built-in clock, and a cryptographic key that is different for each specific token. An authentication server stores the details of the token, so at any moment, the server knows what number is displayed in the LCD of the user’s token device.

The token device is rarely used by itself, but it is used with another method of authentication. For example, a user could use the token to log onto a company website. The authentication page might include text boxes for the user to enter a username, password or PIN, and the number displayed in the token. As with any method of multifactor authentication, this is stronger than using a single factor of authentication. If the token device is lost or stolen, it can’t be used by itself.

However, token systems do have failings. If the battery dies or the device breaks, the user won’t be able to gain access. Additionally, users may be tempted to write their access code or PIN on the device, severely compromising its effectiveness. If it is lost, anyone who finds it can try to use it. Users should also understand that the device identifies them, so they should not loan the token and PIN to anyone else, including co-workers.

The two most common types of tokens are synchronous and asynchronous dynamic password tokens, but static tokens are also used. Synchronous and asynchronous tokens work as one-time password generators. They include a unique identifier similar to a serial number, which is mapped to the user’s account.
Synchronous dynamic password tokens  The LCD token described earlier is a synchronous dynamic password token. It generates passwords at fixed time intervals, such as every 60 seconds. Time interval tokens must have their clocks synchronized to an authentication server. To authenticate, the user enters the password shown on the LCD along with a PIN or passphrase as a second factor of authentication. The generated password provides identification, and the PIN/passphrase provides authentication.

Asynchronous dynamic password tokens  An asynchronous dynamic password token does not use a clock; it generates passwords based on an occurrence of some event. These tokens often generate a password after the user enters a PIN into the token device. The authentication process commonly includes a challenge and a response in which a server sends the user a PIN and the user enters the PIN to create the password. These tokens have a unique seed (or random number) embedded along with a unique identifier for the device. The authentication server also knows the seed and identifier that is assigned to any user.

For example, a user would first submit a username and password to a web page. After validating the user’s credentials, the authentication system uses the token’s identifier and seed to create a challenge number and sends it back to the user. The challenge number changes each time a user authenticates, so it is often called a nonce (short for “number used once”). The challenge number will only produce the correct one-time password on the device belonging to that user.

The user enters the challenge number into the token and the token creates a password. The user then enters the password into the website to complete the authentication process.

Static tokens  A static token can be a swipe card, a smart card, a floppy disk, a USB dongle, or even something as simple as a key for a physical lock. Static tokens often require an additional factor to provide authentication, such as a password or biometric factor.

Many static token devices host a cryptographic key such as a private key, digital signature, or encrypted logon credentials. Some disk encryption schemes, such as Microsoft’s BitLocker, support the use of a USB startup key. As long as the USB thumb drive with the startup key is inserted into the system when it starts, BitLocker will read the key and unlock the drive.
Biometrics

Another common authentication and identification technique is the use of biometrics. Biometric factors fall into the Type 3, “something you are,” authentication category.

Biometric factors can be used as an identifying or authentication technique, or both. Using a biometric factor instead of a username or account ID as an identification factor requires a one-to-many search of the offered biometric pattern against a stored database of enrolled and authorized patterns. Capturing a single image of a person and searching a database of many people looking for a match is an example of a one-to-many search. As an identification technique, biometric factors are used in physical access controls.

Using a biometric factor as an authentication technique requires a one-to-one match of the offered biometric pattern against a stored pattern for the offered subject identity. In other words, the user claims an identity, and the biometric factor is checked to see if the person matches the claimed identity. As an authentication technique, biometric factors are used in logical access controls.

Biometric characteristics are often defined as either physiological or behavioral. Physiological biometric methods include fingerprints, face scans, retina scans, iris scans, palm scans (also known as palm topography or palm geography), hand geometry, and voice patterns. Behavioral biometric methods include signature dynamics and keystroke patterns (keystroke dynamics). These are sometimes referred to as “something you do” authentication.

Fingerprints

Fingerprints are the visible patterns on the fingers and thumbs of people. They are unique to an individual and have been used for decades in physical security for identification. Fingerprint readers are now commonly used on laptop computers and USB flash drives as a method of identification and authentication.

Face scans

Face scans utilize the geometric patterns of faces for detection and recognition. If you've ever watched the TV show Las Vegas, you've probably seen how they can take a picture of a person and then match the characteristics of the face against a database. This allows them to quickly identify a person. Similarly, face scans are used to identify and authenticate people before accessing secure spaces such as a secure vault.

Retina scans

Retina scans focus on the pattern of blood vessels at the back of the eye. They are the most accurate form of biometric authentication and are able to differentiate between identical twins. However, they are the least acceptable biometric scanning means because retina scans can reveal medical conditions, such as high blood pressure and pregnancy. Older retinal scans blew a puff of air into the user's eye, but newer ones typically use an infrared light instead.

Iris scans

Focusing on the colored area around the pupil, iris scans are the second most accurate form of biometric authentication. Iris scans are often recognized as having a longer useful authentication life span than other biometric factors because the iris remains relatively unchanged throughout a person's life (barring eye damage or illness). Iris scans are considered more acceptable by general users than retina scans because they don't reveal personal medical information. Some scanners can be fooled with a high-quality image in place of a person's eye. Additionally, accuracy can be affected by changes in lighting.
Palm scans  Palm scans, sometimes called palm topography or palm geography, scan the palm of the hand for identification. They use near-infrared light to measure vein patterns in the palm, which are as unique as fingerprints. Individuals don’t need to touch the scanner but instead place their palm over a scanner. For example, Boca Ciega High School in Gulfport, Florida, replaced fingerprint scanners with palm scanners to identify students in their lunch lines, and some hospitals are also starting to use palm scanners. Some palm scanners include the fingers and measure the layout of ridges, creases, and grooves, as a full hand scan.

Hand geometry  Hand geometry recognizes the physical dimensions of the hand. This includes the width and length of the palm and fingers. It captures a silhouette of the hand, but not the details of fingerprints or vein patterns. Hand geometry is rarely used by itself since it is difficult to uniquely identify an individual using this method.

Heart/pulse patterns  Measuring the user’s pulse or heartbeat ensures that a real person is providing the biometric factor. It is often employed as a secondary biometric to support another type of authentication. Some researchers theorize that heartbeats are unique between individuals and claim it is possible to use electrocardiography for authentication. However, a reliable method has not been created or fully tested.

Voice pattern recognition  This type of biometric authentication relies on the characteristics of a person’s speaking voice, known as a voiceprint. The user speaks a specific phrase, which is recorded by the authentication system. To authenticate, they repeat the same phrase and it is compared to the original. Voice pattern recognition is sometimes used as an additional authentication mechanism but rarely used by itself.

Speech recognition is commonly confused with voice pattern recognition, but they are different. Speech recognition software, such as dictation software, extracts communications from sound. In other words, voice pattern recognition differentiates between one voice and another for identification or authentication, while speech recognition differentiates between words within any person’s voice.

Signature dynamics  This recognizes how a subject writes a string of characters. Signature dynamics examine both how a subject performs the act of writing and features in a written sample. The success of signature dynamics relies upon pen pressure, stroke pattern, stroke length, and the points in time when the pen is lifted from the writing surface. The speed at which the written sample is created is usually not an important factor.

Keystroke patterns  Keystroke patterns (also known as keystroke dynamics) measure how a subject uses a keyboard by analyzing flight time and dwell time. Flight time is how long it takes between key presses, and dwell time is how long a key is pressed. Using keystroke patterns is inexpensive, nonintrusive, and often transparent to the user (for both use and enrollment). Unfortunately, keystroke patterns are subject to wild variances. Simple changes in user behavior greatly affect this biometric factor, such as using only one hand, being cold, standing rather than sitting, changing keyboards, or sustaining an injury to the hand or a finger.
The use of biometrics promises universally unique identification for every person on the planet. Unfortunately, biometric technology has yet to live up to this promise. For biometric factors to be useful, they must be extremely sensitive.

**Biometric Factor Error Ratings**

The most important aspect of a biometric device is its accuracy. To use biometrics for identification, a biometric device must be able to detect minute differences in information, such as variations in the blood vessels in a person's retina or tones and timbres in their voice. Because most people are basically similar, biometric methods often result in false negative and false positive authentications. Biometric devices are rated for performance by examining the different types of errors they produce.

**Type 1 error** A Type 1 error occurs when a valid subject is not authenticated. This is also known as a false negative authentication. For example, Dawn could use her fingerprint to authenticate herself, but the system incorrectly rejects her valid fingerprint. The ratio of Type 1 errors to valid authentications is known as the false rejection rate (FRR).

**Type 2 error** A Type 2 error occurs when an invalid subject is authenticated. This is also known as a false positive authentication. The ratio of Type 2 errors to valid authentications is called the false acceptance rate (FAR). For example, hacker Joe doesn't have an account but he uses his fingerprint to authenticate and the system recognizes him.

Most biometric devices have a sensitivity adjustment. When a biometric device is too sensitive, Type 1 errors (false negatives) are more common. When a biometric device is not sensitive enough, Type 2 errors (false positives) are more common.

You can compare the overall quality of biometric devices with the crossover error rate (CER), also known as the equal error rate (ERR). Figure 1.2 shows the FRR and FAR percentages when a device is set to different sensitivity levels. The point where the FRR and FAR percentages are equal is the CER, and the CER is used as a standard assessment value to compare the accuracy of different biometric devices. Devices with lower CERs are more accurate than devices with higher CERs.

**FIGURE 1.2** Graph of FRR and FAR errors indicating the CER point
It’s not necessary, and often not desirable, to operate a device with the sensitivity set at the CER level. For example, an organization may use a facial recognition system to allow or deny access to a secure area because they want to ensure that unauthorized individuals are never granted access. In this case, the organization would set the sensitivity very high so there is very little chance of a Type 2 error (false acceptance). This may result in more false rejections, but a false rejection is more acceptable than a possible false acceptance.

**Biometric Registration**

Biometric devices can be ineffective or unacceptable due to factors known as enrollment time, throughput rate, and acceptance. For a biometric device to work as an identification or authentication mechanism, a process called enrollment (or registration) must take place, during which a subject’s biometric factor must be sampled and stored in the device’s database. The stored sample of a biometric factor is called a reference profile (also known as a reference template).

The time required to scan and store a biometric factor depends on which physical or performance characteristic is measured. The longer it takes to enroll using a biometric mechanism, the less willingly the user community accepts the inconvenience. In general, enrollment times over 2 minutes are unacceptable. If you use a biometric characteristic that changes over time, such as a person’s voice tones, facial hair, or signature pattern, reenrollment must occur at regular intervals, adding inconvenience.

The throughput rate is the amount of time the system requires to scan a subject and approve or deny access. The more complex or detailed a biometric characteristic, the longer processing takes. Subjects typically accept a throughput rate of about 6 seconds or faster.

**Multifactor Authentication**

Multifactor authentication is any authentication using two or more factors. Two-factor authentication requires two different factors to provide authentication. For example, when using a debit card at the grocery store, you must usually swipe the card ("something you have") and enter a PIN ("something you know") to complete the transaction. Similarly, smart cards almost always require users to insert their card into a reader and also enter a PIN. As a general rule, the more types of factors that are used, the more secure is the resultant authentication.

Multifactor authentication must use multiple types of factor, such as the “something you know” factor and the “something you have” factor. For example, requiring users to enter a password and a PIN is not multifactor authentication because both methods are from a single authentication factor ("something you know").
When two authentication methods of the same factor are used together, the strength of the authentication is no greater than it would be if just one method were used because the same attack that could steal or obtain one could also obtain the other. For example, using two passwords together is no more secure than using a single password because a password-cracking attempt could discover both in a single successful attack or the user might write both passwords on the same piece of paper.

In contrast, when two or more different factors are employed, two or more different methods of attack must succeed to collect all relevant authentication elements. For example, if a token, a password, and a biometric factor are all used for authentication, then a physical theft, a password crack, and a biometric duplication attack must all succeed simultaneously to allow an intruder to gain entry into the system.

Access Control Techniques

Once a subject has been identified and authenticated, it must be authorized to access resources or perform actions based on its proven identity. Systems provide authorization to subjects through the use of access controls that manage the type and extent of access granted to subjects for different objects.

There are several categories for access control techniques and the CISSP CIB specifically mentions three: discretionary, nondiscretionary, and mandatory. The following sections introduce some basic security operations principles used by these techniques and describe common access control techniques.

Security Operations Principles

This section provides a short explanation of some common security operations concepts. These are important to understand when reviewing the different access control techniques.

Need to know  This principle ensures that subjects are granted access only to what they need to know for their work tasks and job functions. Subjects may have clearance to access classified or restricted data but are not granted authorization to the data unless they actually need it to perform a job.

Least privilege  This principle ensures that subjects are granted only the privileges they need to perform their work tasks and job functions. This is sometimes lumped together with need to know. The only difference is that least privilege will also include rights to take action on a system.

Separation of duties and responsibilities  This principle ensures that sensitive functions are split into tasks performed by two or more employees. It helps to prevent fraud and errors by creating a system of checks and balances.

Discretionary Access Controls

A system that employs discretionary access controls (DACs) allows the owner or creator of an object to control and define subject access to that object. All objects have owners, and access control is based on the discretion or decision of the owner. For example, if a user creates a new spreadsheet file, that user is the owner of the file. As the owner, the user can modify the permissions of the file to grant or deny access to other users.

DAC is also referred to as identity-based access control because access is granted to subjects based on their identity. The identity is typically based on a user’s account, but it can also be based on their membership in a group.

A DAC model is implemented using access control lists (ACLs) on objects. Each ACL defines the types of access granted or denied to subjects. It does not offer a centrally controlled management system because owners can alter the ACLs on their objects at will. Access to objects is easy to change, especially when compared to the static nature of mandatory access controls.

Within a DAC environment, users’ privileges can easily be suspended while they are on vacation, resumed when they return, or terminated when they leave an organization.

Here are some distinguishing points about DAC: Every object has an owner, and owners have full control over their objects. Permissions are maintained in an ACL, and owners can easily change permissions. This makes the model very flexible.

Nondiscretionary Access Controls

The major difference between discretionary and nondiscretionary access controls is in how they are controlled and managed. Administrators centrally administer nondiscretionary access controls and can make changes that affect the entire environment. In contrast, with discretionary access controls, owners can make their own changes, and their changes don’t affect other parts of the environment.

In a non-DAC model, access does not focus on user identity. Instead, a static set of rules governing the whole environment is used to manage access. Non-DAC systems are centrally controlled and easier to manage (although less flexible). Rule-based access controls and lattice-based access controls are both considered nondiscretionary.

Rule-based Access Controls

Rule-based access controls are used in a rule-based system. A set of rules, restrictions, or filters determines what can and cannot occur on the system, such as granting a subject access to an object or granting the ability to perform an action.
In general, rule-based access control systems are more appropriate for environments that experience frequent changes to data permissions, such as changes to the security domain or the labels for objects. Rule-based systems can implement sweeping changes just by changing centralized rules without having to manipulate or “touch” every subject and/or object in the environment. However, in most cases, once rules are established, they remain fairly static and unchanged throughout the life of the environment.

A common example of a rule-based access control system is a firewall. A firewall is governed by a set of rules or filters defined by the administrator. The firewall examines all the traffic going through it and allows only traffic that meets a specific rule. Firewalls often include a final rule denying all other traffic. In other words, if traffic didn’t meet the condition of any previous rule, then the final rule ensures that the traffic is blocked. This final rule is sometimes explicitly stated, but an ACL can also include an implicit deny rule.

Lattice-Based Access Controls

Many nondiscretionary access controls can be labeled as lattice-based access controls. Lattice-based access controls were originally developed to address information flow, which primarily concerns itself with confidentiality. Lattice-based access controls define upper and lower bounds of access for every relationship between a subject and an object. These boundaries usually follow military or corporate security label levels (although they can also be arbitrary).

A subject with the lattice permissions shown in Figure 1.3 can access resources up to Private and down to Sensitive but cannot access resources labeled Confidential/Proprietary or Public. Subjects under lattice-based access controls acquire a least upper bound and a greatest lower bound of access to labeled objects based on their assigned lattice positions. A common example of a lattice-based access control is a mandatory access control.

**FIGURE 1.3** A representation of the boundaries provided by lattice-based access controls
Mandatory Access Controls

A mandatory access control (MAC) system relies upon the use of classification labels. Each classification label represents a security domain, or a realm of security. A security domain is a collection of subjects and objects that share a common security policy. For example, a security domain could have the label Secret, and all objects with the Secret label would be protected in the same manner. Similarly, the requirement for subjects to gain the Secret label is the same for all subjects.

Subjects are labeled by their level of clearance, which is a form of privilege. Objects are labeled by their level of classification or sensitivity. For example, the military uses the labels of top secret, secret, confidential, sensitive but unclassified (SBU), and unclassified to classify its data. Businesses in the private sector often use labels such as confidential (or proprietary), private, sensitive, and public. Chapter 5 discusses data classification topics in more depth.

Once the labels are identified and assigned to subjects and objects, the system determines access based on the labels. In a MAC system, subjects are able to access objects that have the same or a lower level of classification. For example, someone with a secret clearance and secret label is approved to access any data marked as secret or lower.

An expansion of this access control method is known as need to know. Subjects with specific clearance levels are granted access to resources only if their work tasks require such access. In other words, someone with a secret clearance and a secret label is not automatically authorized to access all secret data. If they don't have a need to know the information, they are denied access, even if they have sufficient clearance.

Mandatory access control is prohibitive rather than permissive, and it uses an implicit deny philosophy. If access is not specifically granted, it is forbidden. It is generally recognized as being more secure than DAC, but it isn't as flexible or scalable.

A distinguishing factor between MAC and rule-based access controls is that MAC controls have labels while the nondiscretionary rule-based access controls do not use labels.

Using security labels in mandatory access controls presents some interesting problems. First, for a mandatory access control system to function, every subject and object must have a security label. Depending on the environment, security labels can refer to sensitivity, value to the organization, need for confidentiality, classification, department, project,
and so on. A large environment with multiple classifications, subjects, and objects becomes increasingly difficult to manage.

Security classifications indicate a hierarchy of sensitivity. For example, if you consider the military security labels of top secret, secret, confidential, sensitive but unclassified (SBU), and unclassified, the top secret label includes the most sensitive data and unclassified is the least sensitive. Because of this hierarchy, it’s logical that if someone is cleared for top secret data, they would also be cleared for secret and less-sensitive data. However, classifications don’t have to include lower levels. It is possible to use MAC labels so that a clearance for a higher-level label does not include clearance for a lower-level label.

A distinguishing point about the MAC model is that every object and every subject has a label. These labels are predefined and the system makes a determination of access based on them.

Additionally, it’s possible to segment or compartmentalize labels within each level. For example, instead of a single top secret label, objects can be further identified with labels for multiple compartments within a level. The top secret level could have compartments called CISSP Student, CISSP Associate, and CISSP Professional with matching labels. Subjects will be authorized access to only the specific compartment based on their labels.

Classifications within a MAC model use one of the following three types of environment:

Hierarchical environment  A hierarchical environment relates various classification labels in an ordered structure from low security to medium security to high security. Each level or classification label in the structure is related. Clearance in one level grants the subject access to objects in that level as well as to all objects in all lower levels but prohibits access to all objects in higher levels.

Compartmentalized environment  In a compartmentalized environment, there is no relationship between one security domain and another. Each domain represents a separate isolated compartment. To gain access to an object, the subject must have specific clearance for its security domain.

Hybrid environment  A hybrid environment combines both hierarchical and compartmentalized concepts so that each hierarchical level may contain numerous subdivisions that are isolated from the rest of the security domain. A subject must have the correct clearance and also the need to know for a specific compartment to gain access to the compartmentalized object. A hybrid MAC environment provides more granular control over access but becomes increasingly difficult to manage as it grows.

Role-Based Access Control

Systems that employ role-based or task-based access controls define a subject’s ability to access an object based on the subject’s role or assigned tasks. Roles are commonly identified by job descriptions or work functions. If a subject occupies a management position,
it will have greater access to resources than a subject who is in a temporary job. Role-based access controls are useful in dynamic environments with frequent personnel changes because access depends on a role rather than on subject identity.

Role-based access control (RBAC) is often implemented using groups. For example, a bank may have tellers, loan officers, and managers. Administrators can create groups named Tellers, Loan Officers, and Managers and assign privileges to the groups based on their needs. Any user account placed into one of the groups automatically has the privileges granted to that group. Additionally, as soon as a user is removed from a group, they no longer have the privileges assigned to the group. This helps enforce the principle of least privilege.

RBAC is sometimes called a discretionary model, and other times it’s called a nondiscretionary model. While people can get into spirited debates over which it is, the category it falls into isn’t important for the CISSP exam. It is important to know that access is based on the subject’s job, role, or assigned tasks.

It’s easy to confuse DAC and RBAC because they can both use groups, but they differ in their deployment and use.

DAC and RBAC are similar in that groups can be used in both and they serve as containers to organize users into manageable units.

However, in a strict RBAC system, users have only the privileges granted by assignment to a role—privileges are not assigned to users directly. Furthermore, access is not determined by owner discretion, as it is in DAC. Access is derived from the inherent responsibilities of an assigned role based on the job description and not on the user’s identity. Two different users with the same assigned role will have the same access and privileges.

Another method related to RBAC is called task-based access control (TBAC). TBAC is basically the same as RBAC, but instead of being assigned a single role, each user is assigned an array of tasks. These items all relate to assigned work tasks for the person associated with a user account. Under TBAC, the focus is on controlling access by assigned tasks rather than by user identity.

A distinguishing point about the role-based access control model is that access is granted based on membership in a role. Roles can be created based on jobs or tasks, and privileges can be assigned to the role.

Centralized versus Decentralized Access Control

Access control techniques generally fall into one of two categories: centralized and decentralized/distributed.
Centralized access control implies that all authorization verification is performed by a single entity within a system.

Decentralized access control (also known as distributed access control) implies that various entities located throughout a system perform authorization verification.

Centralized and decentralized access control methodologies offer the same benefits and drawbacks found in any centralized or decentralized system. A small team or individual can manage centralized access control. Administrative overhead is lower because all changes are made in a single location and a single change affects the entire system. Decentralized access control often requires several teams or multiple individuals. Administrative overhead is higher because changes must be implemented across numerous locations. Maintaining consistency across a system becomes more difficult as the number of access control points increases. Changes made to any individual access control point need to be repeated at every access point.

Within a single organization, a centralized access control system is often used. For example, a directory service is a centralized database of objects that includes information about resources available to a network along with information about subjects such as users and computers. You can think of it as a telephone directory for network services and assets. Users, clients, and processes consult the directory service to learn where a desired system or resource resides. Subjects must authenticate to the directory service before performing queries and lookup activities. Even after authentication, the directory service will reveal only certain information to a subject, based on that subject's assigned privileges.

Directory services are often based on the Lightweight Directory Access Protocol (LDAP). Microsoft's Active Directory and Novell's NetWare Directory Services (NDS), now known as eDirectory, are well-known directory services.

Multiple domains and trusts are commonly used in decentralized access control systems. As mentioned previously, a security domain is a collection of subjects and objects that share a common security policy, and individual domains can operate separately from other domains. Trusts are established between the domains to create a security bridge and allow users from one domain to access resources in another. Trusts can be one way only, or they can be two way.

Both centralized and decentralized access control systems can be used to support single sign-on capabilities.

**Single Sign-On**

Single sign-on (SSO) is a centralized access control technique that allows a subject to be authenticated only once on a system and to access multiple resources without repeated authentication prompts. For example, users can authenticate once on a network and then access resources throughout the network without being required to authenticate again.
SSO is very convenient for users, but it also increases security. When users have to remember multiple usernames and passwords, they often resort to writing them down, ultimately weakening security. Users are less likely to write down a single password. SSO also eases administration by reducing the number of accounts required for a subject.

The primary disadvantage to SSO is that once an account is compromised, an attacker gains unrestricted access to all of the authorized resources. However, most SSO systems include methods to protect user credentials.

The following sections discuss several common SSO mechanisms.

**Kerberos**

*Ticket authentication* is a mechanism that employs a third-party entity to prove identification and provide authentication. The most common and well-known ticket system is *Kerberos*.

---

**NOTE**

The Kerberos name is borrowed from Greek mythology. A three-headed dog named Kerberos guards the gates to the underworld. The dog faces inward, preventing escape rather than denying entrance.

---

Kerberos provides a single sign-on solution for users and also provides protection for logon credentials. The current version, Kerberos 5, relies upon symmetric-key cryptography (also known as secret-key cryptography) using the Advanced Encryption Standard (AES) symmetric encryption protocol. Kerberos provides confidentiality and integrity for authentication traffic using end-to-end security and helps prevent against eavesdropping and replay attacks. It uses several different elements that are important to understand:

**Key distribution center (KDC)** The KDC is the trusted third party that provides authentication services. Kerberos uses symmetric-key cryptography to authenticate clients to servers. All clients and servers are registered with the KDC, and it maintains the secret keys for all network members.

**Kerberos authentication server (KAS)** The authentication server hosts the functions of the KDC: a ticket-granting service (TGS), and an authentication service (AS). The *authentication service* verifies or rejects the authenticity and timeliness of tickets. This server is often called the KDC.

**Ticket-granting ticket (TGT)** A TGT provides proof that a subject has authenticated through a KDC and is authorized to request tickets to access other objects. A TGT is encrypted and includes a symmetric key, an expiration time, and the user’s IP address. Subjects present the TGT when requesting tickets to access objects.

**Ticket** A ticket is an encrypted message that provides proof that a subject is authorized to access an object. It is sometimes called a service ticket (ST). Subjects request tickets to access objects, and if they have authenticated and are authorized (based on having a TGT),
they are given a ticket. Kerberos tickets have specific lifetimes and usage parameters. Once a ticket expires, a client must request a renewal or a new ticket to continue communications with any server.

Kerberos requires a database of accounts, which is often contained in a directory service. It uses an exchange of tickets between clients, network servers, and the KDC to prove identity and provide authentication. This allows a client to request resources from the server with both the client and server having assurances of the identity of the other. These encrypted tickets also ensure that logon credentials, session keys, and authentication messages are never transmitted in cleartext.

The Kerberos logon process is as follows:
1. The user types a username and password into the client.
2. The client encrypts the credentials with AES for transmission to the KDC.
3. The KDC verifies the user credentials against a database of known credentials.
4. The KDC generates an encrypted time-stamped TGT.
5. The TGT is encrypted with AES before transmission to the client.
6. The client installs the TGT for use until it expires.

When a client wants to access an object, such as a resource, hosted on the network, it must request a ticket through the Kerberos server. The following steps are involved in this process:
1. The client sends its TGT back to the KDC with a request for access to the resource.
2. The KDC verifies that the TGT is valid and checks its access control matrix to verify that the user has sufficient privileges to access the requested resource.
3. The KDC generates a service ticket and sends it to the client.
4. The client sends the ticket to the server or service hosting the resource.
5. The server or service hosting the resource verifies the validity of the ticket with the KDC.
6. Once identity and authorization is verified, Kerberos activity is complete. The server or service host then opens a session with the client and begins communications or data transmission.

Kerberos is a versatile authentication mechanism that works over local LANs, remote access, and client-server resource requests. However, Kerberos presents a single point of failure—the KDC. If the KDC is compromised, the secret key for every system on the network is also compromised. Also, if a KDC goes offline, no subject authentication can occur.

Kerberos also has strict time requirements, and the default configuration requires that all systems be time-synchronized within five minutes of each other. If a system is not synchronized or the time is changed, a previously issued TGT will no longer be valid and the system will not be able to receive any new tickets. In effect, the client will be denied access to any protected network resources.
Federated Identity Management and SSO

SSO has been common on internal networks for quite a while, but not on the Internet. However, with the explosion of cloud-based applications, an SSO solution has been needed for users accessing resources over the Internet. Federated identity management is often used as a form of decentralized or distributed access control to support this need.

Identity management is the management of user identities and their credentials. Federated identity management extends this beyond a single organization. Multiple organizations can join a federation, or group, where they agree on a method to share identities between them. Users in each organization can log on once in their own organization and their credentials are matched with a federated identity. They can then use this federated identity to access resources in any other organization within the group.

A federation can be composed of multiple unrelated networks within a single university campus, multiple college and university campuses, multiple organizations sharing resources, or any other group that can agree on a common federated identity management system. User credentials within an organization are matched with a federated identity.

Many corporate online training websites use federated SSO systems. When the organization coordinates with the online training company for employee access, they also coordinate the details needed for federated access. These are managed behind the scenes and are usually transparent to users. Users simply access the training website using a web browser and they are automatically authenticated without any further action. Based on their identities, they are then given access to appropriate training courses.

A challenge with multiple companies communicating together is finding a common language. They often have different operating systems, but they still need to share a common language. Federated identity systems often use the Service Provisioning Markup Language (SPML). As background, here’s a short description of some markup languages.

Hypertext Markup Language (HTML)  Commonly used to display static web pages. HTML was derived from the Standard Generalized Markup Language (SGML) and the Generalized Markup Language (GML). HTML describes how data is displayed using tags to manipulate the size and color of the text. For example, the following H1 tag causes the heading to be displayed as a level one heading: `<H1>I Passed The CISSP</H1>`.

Extensible Markup Language (XML)  XML goes beyond just describing how the data is displayed and describes the data. XML can include tags to describe data as anything desired. For example, the following tag identifies the data as the results of taking an exam: `<CISSPExamResults>Passed</CISSPExamResults>`.

Databases from multiple vendors can import and export data to and from an XML format, which means that XML is a common language used to exchange information. Many specific schemas have been created so that companies know exactly what tags are being used for specific purposes.

Security Assertion Markup Language (SAML)  Security Assertion Markup Language is an XML-based language that is commonly used to exchange authentication and authorization (AA) information between federated organizations. It is often used to provide SSO capabilities for browser access.
Service Provisioning Markup Language (SPML)  SPML is a newer framework based on XML but specifically designed for exchanging user information for federated identity single sign-on purposes. It is based on the Directory Service Markup Language (DSML), which can display LDAP-based directory service information in an XML format.

Extensible Access Control Markup Language (XACML)  XACML is used to define access control policies within an XML format, and it commonly implements role-based access controls. It helps provide assurances to all members in a federation that they are granting the same level of access to different roles.

Other Examples of Single Sign-On
Although Kerberos may be the most widely recognized and deployed form of single sign-on, it is not the only one of its kind. In this section, we summarize other SSO mechanisms you may encounter.

Scripted access or logon scripts establish communication links by providing an automated process by which logon credentials are transmitted to resource hosts at the start of a logon session. Scripted access can often simulate SSO even though the environment still requires a unique authentication process to connect to each server or resource. Scripts can be used to implement SSO in environments where true SSO technologies are not available. Scripts and batch files should be stored in a protected area because they usually contain access credentials.

The Secure European System for Applications in a Multivendor Environment (SESAME) is a ticket-based authentication system developed to address weaknesses in Kerberos. However, it did not compensate for all the problems with Kerberos. Eventually, later Kerberos versions and various vendor implementations resolved the initial problems with Kerberos, bypassing SESAME. In the professional security world, SESAME is no longer considered a viable product.

KryptoKnight is a ticket-based authentication system developed by IBM. It is similar to Kerberos but uses peer-to-peer authentication instead of a third party. It was incorporated into the NetSP product. Like SESAME, KryptoKnight and NetSP never took off and are no longer widely used.

AAA Protocols
Several protocols are designed to provide authentication, authorization, and accounting and are sometimes referred to as AAA protocols. These are commonly used with remote access systems such as virtual private networks (VPNs) and other types of network access servers to provide centralized access control. They prevent internal LAN authentication systems and other servers from being attacked remotely. When a separate system is used for remote access, only the remote access users are affected if this system is successfully attacked. In other words, the attacker won’t have access to internal accounts. The AAA protocols are also commonly used for mobile IP, which provides access to mobile users with smart phones.

These AAA protocols use the access control elements of identification, authentication, authorization, and accountability as described in “Access Control Elements,” earlier in this
chapter. They ensure that users have valid credentials to authenticate and verify that the user is authorized to connect to the remote access server based on the user’s proven identity. Additionally, the accounting element can track the user’s network resource usage and can be used for billing purposes. Some common AAA protocols are RADIUS, TACACS+, and Diameter.

**RADIUS**

*Remote Authentication Dial-in User Service (RADIUS)* centralizes authentication for remote dial-up connections. It is typically used when an organization has more than one remote access server. A user can connect to any remote access server, which then passes on the user’s credentials to the RADIUS server to verify authentication and authorization and to track accounting. In this context, the remote access server is the RADIUS client and a RADIUS server will provide AAA services for multiple remote access servers.

Many Internet service providers (ISPs) use RADIUS for authentication. Users can access the ISP from anywhere and the ISP server then forwards the user’s connection request to the RADIUS server. Organizations can also use RADIUS; it is often implemented with callback security for an extra layer of protection. Users call in, and after authentication, the RADIUS server terminates the connection and initiates a call back to the user’s predefined phone number. If a user’s authentication credentials are compromised, the callback security prevents an attacker from using them.

RADIUS uses the User Datagram Protocol (UDP) and encrypts only the exchange of the password. It doesn’t encrypt the entire session, but additional protocols can be used to encrypt the data session. The current version is defined in RFC 2865.

**TACACS+**

Terminal Access Controller Access-Control System (TACACS) was introduced as an alternative to RADIUS. Cisco later introduced extended TACACS (XTACACS) as a proprietary protocol. However, TACACS and XTACACS are not commonly used today. TACACS Plus (TACACS+) was later created as an open publicly documented protocol, and it is the most commonly used of the three.

TACACS+ provides several improvements over the earlier versions and over RADIUS. It separates authentication, authorization, and accounting into separate processes, which can actually be hosted on three separate servers if desired. The other versions combine two or three of these processes. Additionally, TACACS+ encrypts all of the authentication information, not just the password as RADIUS does. TACACS and XTACACS used UDP port 49, while TACACS+ uses Transmission Control Protocol (TCP) port 49, providing a higher level of reliability for the packet transmissions.

**Diameter**

Building on the success of RADIUS and TACACS+, an enhanced version of RADIUS named Diameter was developed. It supports a wide range of protocols, including traditional IP, Mobile IP, and Voice over IP (VoIP). Because it supports extra commands, it is
becoming popular in situations where roaming support is desirable, such as with wireless devices and smart phones.

Diameter uses TCP port 3868 or Stream Control Transmission Protocol (SCTP) port 3868, providing better reliability than UDP used by RADIUS. It also supports Internet Protocol Security (IPsec) and Transport Layer Security (TLS) for encryption.

**NOTE**
In geometry, the radius of a circle is the distance from the center to an edge, and the diameter is twice the radius going from edge to edge through the circle. The Diameter name implies that Diameter is twice as good as RADIUS. While that may not be exactly true, it is an improvement over RADIUS and helps to reinforce that Diameter came later and is an improvement.

**Authorization Mechanisms**

There are many different types of authorization mechanisms, or methods used to control who can access specific objects. This section provides a brief introduction to some common mechanisms.

**Implicit Deny**  A basic principle of access control is implicit deny. Most authorization mechanisms use it. The implicit deny principle ensures that access to an object is denied unless access has been explicitly granted to a subject. For example, if Jeff is granted Full Control to a file but no one else is granted access, Jeff is the only user that has access. All other users are denied access.

ACLs on firewalls use this principle. Explicit rules identify traffic that is allowed, and all other traffic is blocked. The last rule in the ACL is a “deny all” rule that specifically blocks all traffic that hasn’t been previously allowed. This last rule can be explicitly stated as “deny all” to deny all traffic in or out for any traffic that hasn’t been allowed. However, on many firewalls the deny rule is implicit and does not need to be explicitly stated. Configuration settings within the firewall can apply the rule even if it isn’t in the ACL.

**Access control matrix**  An access control matrix is a table that includes subjects, objects, and assigned privileges. When a subject attempts an action, the system evaluates the access control matrix to determine if the subject is authorized.

For example, an access control matrix can include a group of files as the objects and a group of users as the subjects. It will show the exact permissions authorized by each user for each file. Note that this covers much more than a single ACL. In this example, each file within the matrix has a separate ACL that lists the authorized users and their assigned permissions.

Capability tables are another way that an access control matrix can be implemented. They are different from ACLs in that a capability table is created for each subject, such as for each role. For example, a capability table created for the accounting role will include a list of all objects that the accounting role can access and will also include the specific
privileges assigned to the accounting role for these objects. In contrast, ACLs are assigned to objects. An ACL for a file would list all the users and/or groups that are authorized access to the file and the specific access granted to each.

The difference between an ACL and a capability table within an access control matrix is based on the focus or perspective. ACLs are assigned to each object and identify access granted to subjects. Capability tables are created for each subject, and they identify the objects that the subject can access.

Access control matrices are described in more detail in Chapter 11, “Principles of Security Models, Design, and Capabilities.”

Constrained interface A constrained or restricted interface is implemented within an application to restrict what users can do or see based on their privileges. Users with full privileges have access to all the capabilities of the application. Users with restricted privileges have limited access.

Applications constrain the interface using different methods. A common method is to hide the capability if the user doesn’t have permissions to use it. Commands might be available to administrators via a menu or by right-clicking an item, but if a regular user doesn’t have permissions, the command does not appear. Other times, the command is shown but is dimmed or disabled. The regular user can see it but will not be able to use it.

Content- or context-dependent control Some authentication mechanisms control access based on the content of an object or the context of the activity taken by a subject.

Content-dependent access controls restrict access to data based on the content within an object. Database views are commonly used as content-dependent controls. Chapter 7, “Software Development Security,” discusses database views in more depth, but in short, a view is a virtual table. It retrieves a limited data set from one or more tables and restricts what users can see. For example, database tables could include customer names, email addresses, phone numbers, and credit card data. A view might show only certain information within these tables, such as only names and email addresses. Users are granted access to content via the view but cannot access the data in the underlying tables.

Context-dependent access controls require specific activity before access is granted. For example, transactions are commonly completed using context-dependent controls. Here’s a simple example: Users can’t view a web page used to provide credit card information until they begin a purchase transaction. The user’s activities are controlled through a specific group of web pages with previous pages setting up the context for future pages.

Date and time controls are also considered context dependent. For example, it’s possible to restrict access to computers and applications based on the current day and/or time. If a user tries to access the resource outside the allowed time, they are denied access.
Identity and Access Provisioning Life Cycle

The *identity and access provisioning life cycle* refers to the creation, management, and deletion of accounts. Although these activities may seem mundane, they are essential to a system’s access control capabilities. Without properly defined and maintained user accounts, a system is unable to establish accurate identity, perform authentication, provide authorization, or track accountability. As mentioned previously, identification occurs when a subject claims an identity. This identity is most commonly a user account, but it also includes computer accounts and service accounts.

Access control administration is the collection of tasks and duties involved in managing accounts, access, and accountability during the life of the account. These tasks are contained within three main responsibilities of the identity and access provisioning life cycle:

- Provisioning
- Review
- Revocation

**Provisioning**

An initial step in identity management is the creation of new accounts and provisioning them with appropriate privileges. Creating new user accounts is usually a simple process, but the process must be protected and secured via organizational security policy procedures. User accounts should not be created at an administrator’s whim or in response to random requests. Rather, proper provisioning ensures that a specific procedure is followed.

The initial creation of a new user account is often called an *enrollment*. The enrollment process creates a new identity and establishes the factors the system needs to perform authentication. It is critical that the enrollment process be completed fully and accurately. It is also critical that the identity of the individual being enrolled be proved through whatever means your organization deems necessary and sufficient. Photo ID, birth certificate, background check, credit check, security clearance verification, FBI database search, and even calling references are all valid forms of verifying a person’s identity before enrolling them in any secured system.

Many organizations have automated provisioning systems. For example, once a person is hired, the HR department completes initial identification and in-processing steps and then forwards a request to the IT department to create an account. Users within the IT department enter information such as the employee’s name and their assigned department via an application. The application then creates the account using predefined...
Automated provisioning systems ensure that accounts are created consistently, such as always creating usernames the same way and treating duplicate usernames consistently. If the policy dictates that usernames are created from first and last names, then the application will create a username as **suziejones** for a user named Suzie Jones. If a second employee is hired with the same name, then the second username might be **suziejones2**.

If the organization is using groups (or roles), the application can automatically add the new user account to the appropriate groups based on the user’s department or job responsibilities. The groups will already have appropriate privileges assigned, so this step provisions the account with appropriate privileges.

As part of the hiring process, new employees should be trained on organization security policies and procedures. Before hiring is complete, employees are typically required to review and sign an agreement committing to uphold the organization’s security standards. This often includes an acceptable usage policy. Chapter 5 covers the importance of developing and managing a security education, training, and awareness program to ensure that employees are aware of the organization’s policies and procedures.

Throughout the life of a user account, ongoing maintenance is required. Organizations with fairly static organizational hierarchies and low employee turnover or promotion will conduct significantly less account administration than an organization with a flexible or dynamic organizational hierarchy and high employee turnover and promotion rates. Most account maintenance deals with altering rights and privileges. Procedures similar to those used when creating new accounts should be established to govern how access is changed throughout the life of a user account. Unauthorized increases or decreases in an account’s access capabilities can cause serious security repercussions.

**Account Review**

Accounts should be reviewed periodically to ensure that security policies are being enforced. This includes ensuring that inactive accounts are disabled and employees do not have excessive privileges.

Many administrators use scripts to periodically check for inactive accounts. For example, if an account has not been logged into in the past 30 days, the script can disable and isolate the account. Similarly, scripts can be used to ensure that membership in privileged administrator groups is limited to specific users. When other users are added to these groups, the script can automatically remove them.

Account review is often formalized in auditing procedures. Chapter 2 covers assessing the effectiveness of access control, including the ability to assess user entitlement and perform access reviews and audits.
Excessive Privilege and Creeping Privileges

It’s important to guard against two problems related to access control: excessive privilege and creeping privileges. **Excessive privilege** occurs when users have more privileges than their assigned work tasks dictate. If a user account is discovered to have excessive privileges, the unnecessary privileges should be immediately revoked. **Creeping privileges** involve a user account accumulating privileges over time as job roles and assigned tasks change. This can occur because new tasks are added to a user’s job and additional privileges are added but no privileges are ever removed, even if the user no longer needs them. Creeping privileges result in excessive privilege.

Both of these situations violate the basic security principle of least privilege, and account reviews are effective at discovering these problems.

Account Revocation

When employees leave an organization for any reason, their user accounts should be disabled as soon as possible. Additionally, if an employee takes a leave of absence, their account should be disabled to prevent access while they are gone. Whenever possible, this task should be automated and tied into the HR department. For example, if an employee is terminated, the account should be disabled during the exit interview.

If a terminated employee retains access to a user account after the exit interview, the risk for sabotage is very high. Even if the employee doesn’t take malicious action, someone else may be able to use the account if they discover the password. The activity will be logged in the name of the terminated employee instead of the person actually taking the action.

When it’s determined that the account is no longer needed, it should be deleted. Deletion of the account is normally done within 30 days after an account is disabled, but it can vary depending on the needs of the organization.

Many systems have the ability to automatically set specific expiration dates for any account. These can be set for temporary or short-term employees when the accounts are created. This maintains a degree of control without requiring ongoing administrative oversight.

Real World Scenario

**Dangers of Failing to Revoke Account Access**

Fannie Mae learned firsthand of the dangers of not immediately revoking account access after firing an employee. At about 2 p.m. on October 24, 2008, a UNIX engineer at Fannie Mae was fired. He turned in his badge at 4:45 but he retained administrative access until about 10:00 p.m. that day.
He used his account after being fired to grant himself remote access to Fannie Mae’s servers and at some point inserted malicious code in a legitimate script that ran daily at 9 a.m. The full content of his malicious code was set to run on January 31, 2009, as a logic bomb and would have destroyed data on 4,000 Fannie Mae servers. Many experts believe it would have taken Fannie Mae as long as a week to restore functionality if the code ran successfully.

Another engineer discovered the malicious code about a week after the fired employee inserted it so it didn’t cause any damage. However, the incident could have been avoided completely by revoking the employee’s access immediately.

Summary

The first domain of the CISSP CBK is Access Control. It covers the management, administration, and implementation aspects of granting or restricting subject access to objects. Subjects are active entities (such as users), and objects are passive entities (such as files). Access controls are central to establishing a secure system. They protect against security incidents, which can result in the loss of confidentiality, integrity, and/or availability of resources.

Users, owners, and custodians are three specific types of subjects. A user accesses objects such as files on a system. The owner is ultimately responsible for classifying, labeling, and protecting objects. A custodian is delegated day-to-day responsibilities for properly storing and protecting objects.

Three primary types of access controls are preventive, detective, and corrective. Preventive access controls attempt to prevent incidents before they can occur. Detective access controls attempt to detect incidents after they’ve occurred, and corrective access controls attempt to correct problems caused by incidents once they’ve been detected.

Controls are implemented as administrative, logical, and physical. Administrative controls are also known as management controls and include policies and procedures. Logical controls are also known as technical controls and are implemented through technology. Physical controls use physical means to protect objects.

Key access control elements include identification, authentication, authorization, and accountability. Subjects claim an identity, which is proved with authentication. The three factors of authentication are “something you know” (such as passwords or PINs), “something you have” (such as smart cards or tokens), and “something you are” (identified with biometrics). Multifactor authentication uses more than one authentication factor, and it is stronger than using any single authentication factor. Once subjects are authenticated, authorization mechanisms control their access and audit trails log their activities so that they can be held accountable for their actions.

There are various models for access control or authorization. These include discretionary, nondiscretionary, mandatory, and role-based access controls. With discretionary
access controls, all objects have an owner, and the owner has full control over the object. Nondiscretionary controls are centrally controlled by an administrator. Mandatory access controls require all objects to have labels, and the access is based on subjects having a matching label. Role-based access controls use roles, and these roles are granted appropriate privileges based on jobs or tasks. Subjects are placed into roles and they inherit the privileges assigned to the roles.

Single sign-on allows users to authenticate once and access any resources in a network without authenticating again. Kerberos is a popular single sign-on authentication protocol using ticket authentication for identification and authentication. Kerberos uses a database of subjects, symmetric cryptography, and time synchronization of systems to issue tickets.

Federated identity management is a single sign-on solution that can extend beyond a single organization. Multiple organizations create or join a federation and agree on a method to share identities between the organizations. Users can authenticate within their organization and access resources in other organizations without authenticating again.

AAA protocols provide authentication, authorization, and accounting. Popular AAA protocols are RADIUS, TACACS+, and Diameter.

The identity and access provisioning life cycle includes the processes to create, manage, and delete accounts used by subjects. Provisioning includes the initial steps of creating the accounts and ensuring that they are granted appropriate access to objects. As users’ jobs change, they often require changes to the initial access, and provisioning includes modifying access while also ensuring that the principle of least privilege is followed. When accounts are no longer needed, they should be disabled as soon as possible and then deleted.

Exam Essentials

Know the difference between subjects and objects and know common subject labels. You’ll find that CISSP questions and security documentation commonly use the terms subject and object, so it’s important to know the difference between them. Subjects are active entities (such as users) that access passive objects (such as files). A user is a subject who accesses objects in the course of performing some action or accomplishing a work task. The owner is the subject responsible for classifying and labeling objects and for protecting and storing data on any system. A custodian has day-to-day responsibilities for protecting and storing objects.

Know the various types of access control. You should be able to identify the type of any given access control. Access controls may be preventive (to stop unwanted or unauthorized activity from occurring), detective (to discover unwanted or unauthorized activity), or corrective (to restore systems to normal after an unwanted or unauthorized activity has occurred). Other access controls are deterrent (to discourage violation of security policy), recovery (to repair or restore resources, functions and capabilities after a violation of security policy has occurred), directive (to direct, confine, or control the action of subjects to
force or encourage compliance with security policy), or compensation (to provide various options to other existing controls to aid in enforcement and support of security policy). Additionally, controls are implemented as administrative (policies or procedures to implement and enforce overall access control), logical/technical (hardware or software mechanisms used to manage access to resources and systems and to provide protection for those resources and systems), and physical (physical barriers deployed to prevent direct contact with systems or areas within a facility).

Understand the difference between identification, authentication, and authorization. Access controls depend on effective identification, authentication, and authorization, so it’s important to understand the differences among them. Subjects claim an identity, and identification can be as simple as a username for a user. Subjects prove their identity by providing authentication credentials such as the matching password for a username. Subjects are then granted authorization to objects based on their proven identity.

Understand the details of the three authentication factors. The three factors of authentication are something you know (such as a password or PIN), something you have (such as a smart card or token), and something you are (based on biometrics). Multifactor authentication includes two or more authentication factors, and using it is more secure than using a single authentication factor. Passwords are the weakest form of authentication, but password policies help increase their security by enforcing complexity and history requirements. Smart cards include microprocessors and cryptographic certificates, and tokens create one-time passwords. Biometric methods identify users based on characteristics such as fingerprints. The accuracy of a biometric method is identified by the crossover error rate, where Type 1 errors (false rejection rate) are equal to Type 2 errors (false acceptance rate).

Know details about each of the access control techniques. There are several categories of access control techniques commonly tested on the CISSP exam, including discretionary, nondiscretionary, mandatory, and role-based access controls. With discretionary controls, all objects have owners and the owners can modify permissions. Nondiscretionary controls are centrally managed, such as rules on a firewall. Mandatory access controls use labels for subjects and objects, and matching labels are required for access. Role-based access controls use task-based roles and users gain privileges when their accounts are placed within a role.

Identify common authorization mechanisms. Authorization ensures that the requested activity or object access is possible given the privileges assigned to the authenticated identity. Common authorization mechanisms include implicit deny, access control matrices, access control lists, constrained interfaces, and content- or context-dependent controls.

Understand single sign-on. Single sign-on (SSO) is a mechanism that allows a subject to be authenticated once on a system and be able to access multiple objects without authenticating again. It is commonly used within and between organizations, so it’s an important mechanism to understand. Kerberos is the most common SSO method used.
within organizations, and it uses symmetric cryptography and tickets to prove identification and provide authentication. When multiple organizations want to use a common SSO system, they often use a federated identity management system, where the federation, or group of organizations, agrees on a common method of authentication. Service Provisioning Markup Language (SPML) is commonly used to share federated identity information. Other SSO methods are scripted access, SESAME and KryptoKnight.

**Understand the purpose of AAA protocols.** Several protocols provide centralized authentication, authorization, and accounting services and are commonly used for remote access. RADIUS uses UDP and encrypts the password only. TACACS+ uses TCP and encrypts the entire session. Diameter is based on RADIUS and improves many of the weaknesses of RADIUS, but Diameter is not compatible with RADIUS. Diameter is becoming more popular with mobile IP systems such as smart phones.

**Understand the identity and access provisioning life cycle.** The identity and access provisioning life cycle refers to the creation, management, and deletion of accounts. Provisioning accounts ensures that they have appropriate privileges based on task requirements. Periodic reviews ensure that accounts don’t have excessive privileges and thus violate the principle of least privilege. Revocation includes disabling accounts as soon as possible when they are not needed, such as when an employee leaves the company, and deleting them when it’s determined they are no longer needed.