© Ruslan Gi/Shutterstock

CHAPTER 2

# Security in the Microsoft Windows Operating System

ONE OF THE MORE DIFFICULT TASKS when securing any computer system or network is identifying where to start. There are many components in any computing environment. Each component is a potential point of attack. Since the operating system provides the ability for software and hardware to interact, it is a good starting point for securing an entire environment. On any computer, the operating system enables software to access physical resources. For example, it is the operating system that governs how any application actually reads from, or writes to, a physical disk. Consequently, the operating system is a prime candidate for attack and a valuable resource to protect.

From an attacker's point of view, a compromised operating system provides easy access to protected information and resources. Compromising operating system controls gives the attacker the ability to remove evidence of attacks and "clean up" any leftover log entries or other traces of the attack. A secure operating system is the basis of a secure environment.

In this chapter, you will learn about the Windows operating system architecture and controls to ensure system security. You will also learn how attackers search for, identify, and exploit operating system vulnerabilities. With the knowledge of how attackers operate, you'll be able to select and implement the right controls to secure your environment.

## Chapter 2 Topics

This chapter covers the following topics and concepts:

- What the organization of the operating system components and architecture is
- What the basic Windows operating system architecture is
- What access controls and authentication are
- What security tokens, rights, and permissions are
- What users, groups, and Active Directory are
- What Windows attack surfaces and mitigation are
- What fundamentals of Microsoft Windows security monitoring and maintenance are

### Chapter 2 Goals

When you complete this chapter, you will be able to:

- Distinguish operating system components from architecture
- Describe the basic Windows OS architecture
- Discuss access controls and authentication
- Explain security tokens, rights, and permissions
- Research the features of several common security tokens
- Identify the purposes of users and groups
- Discuss the features of directory services
- Analyze the business advantages and challenges of Active Directory
- Describe Windows attack surfaces and mitigation
- Summarize the fundamentals of Microsoft Windows security monitoring and maintenance

## Operating System Components and Architecture

In spite of the reference to an "operating system" as a single entity, an operating system is not a single huge program. An operating system is actually a collection of many programs working together, along with data, to provide access to physical resources. The goal of secure information can simply be expressed as simultaneously ensuring both of the following situations:

1. All required information is available to authorized users.
2. No information is available to unauthorized users.

Although these two basic situations appear to be clear and straightforward, simultaneously ensuring both can be challenging. A **control** is any mechanism or action that prevents, detects, or addresses an attack. Once security controls are in place, attackers must identify methods to compromise the controls and gain the privileges of an authorized user. The methods attackers use will rely on one or more vulnerabilities. Your goal is to ultimately harden your operating system from attack; however, if it is compromised, you would need to further identify and then mitigate as many vulnerabilities as possible to deny the opportunity for an attacker to realize a threat vector.

The first step in planning how to secure the operating system is to understand the purpose of various operating system components. In order to do that effectively, understanding the entire attack surface is required. Each operating system component, and the communication between components, can be a potential point of attack. Many different operating systems are in use today, and often multiple versions of each one; however, some components and services are common among them.

## The Kernel

The central component of most operating systems is the **kernel**. The kernel is the part of the operating system that may reside partially in memory and provides the backbone of the operating system's services (**FIGURE 2-1**). The classic definition of the kernel states that the entire kernel resides in memory. Today's more complex operating system kernels are made up of both the main memory-resident components and external loadable modules. The use of loadable modules reduces the kernel's memory footprint. The kernel provides access to physical resources and often runs other operating system programs to complete a task. The memory-resident kernel code will directly handle access to the CPU, where efficiency is crucial. In other cases, where flexibility is more important, the kernel will run device driver programs to handle physical resource access. The second approach is slower but makes it easy for the operating system to support a wide variety of hardware devices from different vendors. All that is needed is a device driver program for the operating system to support a new device.

Many current operating systems actually implement microkernel architecture. A **microkernel** only implements the minimal required functionality in the memory-resident portion of the operating system, such as memory management, interprocess communication, and process scheduling. Other necessary functionality is supported by external programs. The main difference between internal and external programs is the privilege level at which each runs. A pure microkernel allows only memory-resident components to run at kernel, or maximum privilege, mode.



**NOTE**

Of course, each new device driver means a new potential vulnerability and possible point of attack. Ensure your systems do not have device drivers installed for retired devices.

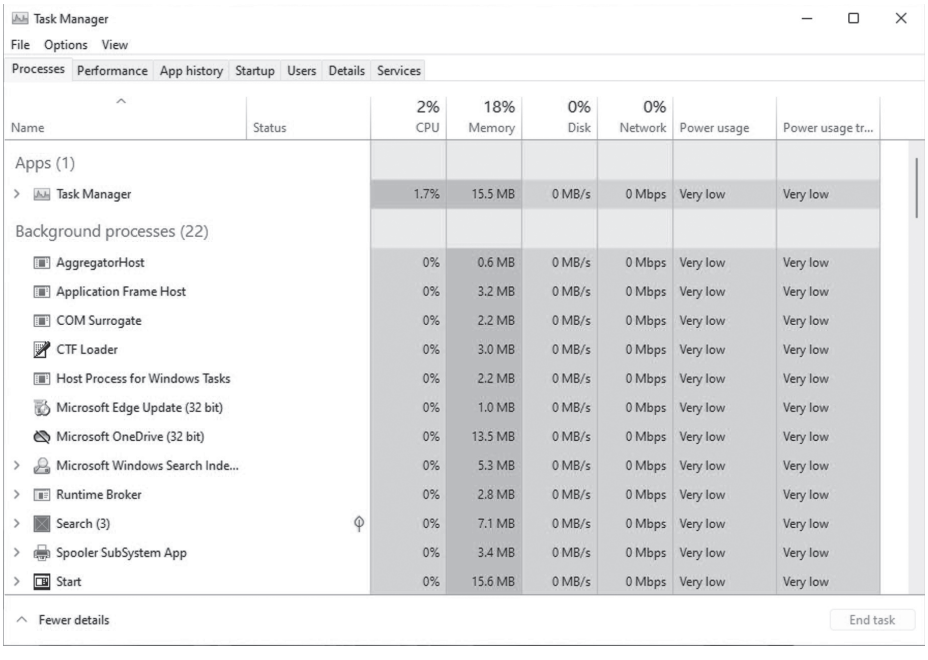



**FIGURE 2-1**
Operating system kernel.
© Jones & Bartlett Learning.

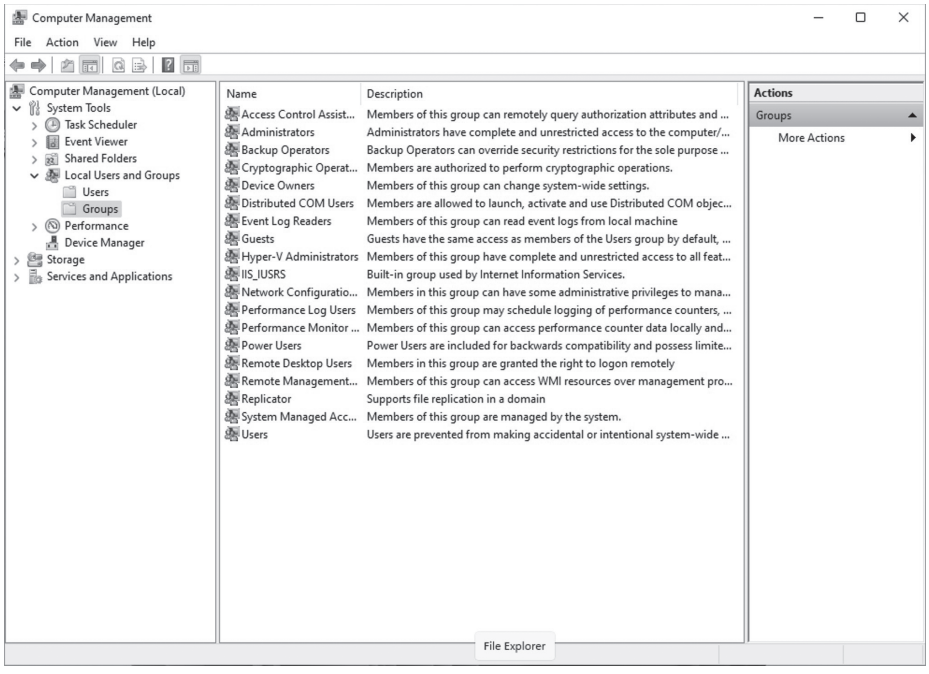



**FIGURE 2-2**
Windows operating system components.
Courtesy of Microsoft Corporation.

The kernel also includes areas of memory reserved for the operating system data structures. One example of an operating system data structure is the process, or task, table. The process table contains one entry for each running process. Each operating system stores different process properties, but the basic information in the process table is consistent. **FIGURE 2-2** shows a few of the types of information the operating system maintains for processes.

One common process property is the mode in which a process is run. Processes generally run in either **user mode** or **supervisor mode**. You may also see supervisor mode referred to as **kernel mode**. Processes that run in supervisor mode can perform more tasks and access more restricted parts of the computer system. One way for an attacker to access a protected resource is to modify the process table entry and change a user mode process to supervisor mode. The process table is only one example of a kernel data structure that must be secured from unauthorized changes. However, processes running in user mode should be able to view the information in the process table to see what is running at any point in time. The kernel keeps track of what processes can and cannot do.

**NOTE**
Some operating system implementations also refer to the maximum privilege mode as supervisor mode or kernel mode.

## Operating System Components

Operating systems contain far more than just the kernel. The kernel provides core services of the operating system and calls external programs to provide many more operating system services. **TABLE 2-1** contains a list of general services most current operating systems provide.

**TABLE 2-1** Operating System General Services

| OPERATING SYSTEM SERVICE | DESCRIPTION |
|---|---|
| Program/process management | The operating system manages locating, loading, and actually executing all programs. It handles memory allocation, CPU scheduling, and providing the necessary environment for each program. |
| Input and output | Nearly every program requires input and produces output. The operating system hides the details of the physical hardware and provides programs with the ability to process input and output. |
| File system | The operating system also provides access to long-term storage, such as disk drives, and helps to organize the information to make it easily and efficiently accessible. |
| Communication | Programs often need to communicate with other programs, both locally and on other computers. The operating system provides the support for exchanging information between programs. |
| Error detection and alerts | The operating system is responsible for monitoring activities that occur within the computer and for responding when errors occur. |

© Jones & Bartlett Learning.



From a security view, the most important concept of this section is that the operating system is the collection of programs that control access to the physical hardware. Since information is stored and transmitted on physical hardware, ensuring the security of protected information starts with ensuring the security of the operating system.

## Basic Windows Operating System Architecture

The Microsoft Windows operating system has changed a lot from the simple personal computer operating system, DOS. It has grown from a single product offering to a full family of products to meet different needs. The current versions of Windows address a wide variety of computing needs, from portable devices and workstations to enterprise-class, high-performance platforms. The Windows operating system is designed to be a modular system to provide the widest variety of services for most platform requirements.

The current versions of client and server Windows operating systems in use today are all based on the legacy Windows NT code base. Windows NT was Microsoft's first operating system designed with security in mind. The first commercial version of Windows NT was version 3.1, released in 1993. At the time, Windows NT was a ground-breaking product from Microsoft. Several versions of Windows share a common ancestry back to Windows NT.

Based on the same kernel, the current Windows operating systems are designed with modified microkernel architectures. The operating system design allows many system functions to be implemented as external programs that run in kernel mode. In a pure

**NOTE**

Windows 7 and Windows Server 2008 R2 are based on the NT 6.1 kernel. Windows 8/8.1 and Windows Server 2012 are based on the NT 6.2 kernel. The most current versions of Windows, Windows 10, Windows 11, Windows Server 2016, Windows Server 2019, and Windows Server 2022, are based on the NT 10.0 kernel.

microkernel architecture, external programs are not allowed to run in kernel mode. Because of the modular nature of Windows, major components can be removed, replaced, or enhanced without having to rewrite the entire operating system. This design allows Microsoft to create different versions of the same base operating system to provide specific services for different client and server environments.

## Windows Run Modes

The architecture of the Windows operating system consists of two main layered components—kernel (or supervisor) mode and user mode programs. Kernel mode programs run in a privileged mode, also called kernel or supervisor mode, and interact closely with the physical hardware. User mode programs interact with both users and kernel mode programs. **FIGURE 2-3** shows the basic architecture of a Windows operating system.

## Kernel Mode

Programs running in kernel mode have complete access to the computer's hardware and system services. This level of access is needed by the operating system and provides an attractive target for attackers.

**TABLE 2-2** shows the main kernel mode program components.

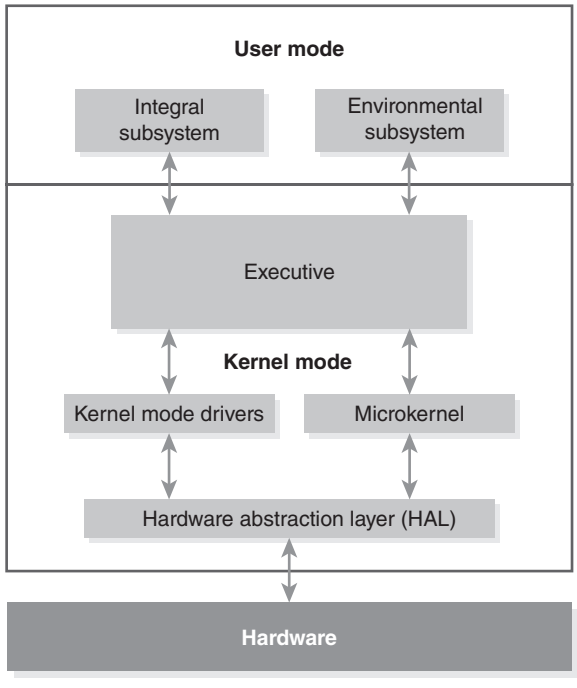



**FIGURE 2-3**
Basic architecture of a Windows operating system.

© Jones & Bartlett Learning.

**TABLE 2-2** Windows Kernel Mode Components

| COMPONENT | DESCRIPTION |
|---|---|
| Hardware Abstraction Layer (HAL) | The HAL provides the actual access to physical hardware. All other kernel mode programs interact with hardware through the HAL. This allows Microsoft to support multiple hardware platforms by just writing different HAL modules, instead of rewriting all operating system programs. |
| Kernel mode drivers | Kernel mode drivers provide user programs and other kernel mode programs access to individual hardware devices, through the HAL. These drivers provide the translation to allow other programs to access devices as file objects. |
| Microkernel | The microkernel is the memory-resident portion of the operating system that provides core functionality, including CPU synchronization, process thread/interrupt scheduling, and exception handling. |
| Executive | The executive is at the "highest level" of the kernel mode programs. It provides services, such as managing objects, I/O, security, and process management. User mode programs interact with the operating system via the executive. |

© Jones & Bartlett Learning.



**Technical TIP**

One common goal of attackers is to run a program of their choice in kernel mode. At that level of privilege, an attacker can pretty much own a computer. Pay special attention to any vulnerability you encounter that could allow an attacker to escalate privilege to kernel mode.

### *Windows Server 2022 Kernel Vulnerabilities*

With Windows Server 2022 and the movement into cloud computing, many of the newer technologies are aimed at developing server operating systems that can reside in a virtualized architecture. In keeping with the adherence to using kernel mode architecture, these newer technologies can be exploited in the same fashion as on-premise technologies. Two newer exploitable vulnerabilities that are able to gain privilege access based on this older architecture is Windows Kernel Information Disclosure Vulnerability CVE-2022-30162 and Windows Clustered Shared Volume Information Disclosure

Vulnerability CVE-2022-29123. Both of these vulnerabilities have allowed for kernel mode exploitation and have since been patched by Microsoft. It is important, however, to discuss these types of attacks so we can better understand how the underlying architecture is examined by threat actors looking for a way to gain privilege.

When considering Windows Kernel Information Disclosure Vulnerability CVE-2022-30162 and reviewing the threat, it was found quickly and patched by Microsoft so it was not a widespread concern like other vulnerabilities, such as PrintNightmare which we learned about in Chapter 1. Impact from unpatched exploitables help to ensure that the topic of patch management is always considered in any system administration security management plan.

In the case of Windows Clustered Shared Volume Information Disclosure Vulnerability CVE-2022-29123, the threat was a little more concerning in that it exploited the use of containerized applications, which is a newer technology utilized with Windows Server 2022 Azure-based cloud systems. The same kernel mode level **exploit** that attempts to gain privilege through the use of exploiting the weakness and then self-granting access based upon least-privilege will allow access to protected resources. In this threat, the AppContainer execution environment, which is designed to isolate applications to make them more secure, was vulnerable and needed to be patched. Microsoft did respond to the CVE and provided a fix for the exploit in the release of a hotfix.

## User Mode

All nonkernel mode programs run under user mode. This includes application programs and the user mode layer of Windows. The Windows user mode layer programs handle all user interaction and processing requests, and pass I/O requests to the necessary kernel mode drivers, using the executive. **TABLE 2-3** lists the two main user mode program components.

**TABLE 2-3** Windows User Mode Components

| COMPONENT | DESCRIPTION |
|---|---|
| Environment subsystem | The environment subsystem provides the ability to run programs written for different operating systems, including previous Windows versions and Linux. |
| Integral subsystem | The integral subsystem handles the user mode functions on behalf of the environment subsystem, including logon and access control, network access, and providing network services. |

© Jones & Bartlett Learning.

# Access Controls and Authentication

All computer users, including attackers, need to establish access to a computer system before accessing its resources. The operating system is responsible for providing access to authorized users while denying access to unauthorized users. This process of providing and denying access is called **access control**. As developers of operating systems have become more concerned with security issues, access control has matured with each new operating system release. Access control is a multistep process, starting with **identification** and **authentication**. Regardless of the methods used, the operating system needs to identify the user asking for access to a resource. Most often, the user provides a username (or user ID). A username alone is not sufficient to use as a basis for granting access to resources, however. Anyone can claim to be a user. There must be a process that validates that a user is authentic.

The authentication process ensures that users are who they claim to be. There are many methods to authenticate users. The most common method in use is the password. Although anyone can claim to be a particular user, no one else but the real user should know the user's password. The ability to provide both the username and password provides the authentication that the user is valid and authentic.



## Authentication Methods

Authentication methods aren't limited to just passwords—they can be one of three types. Each type of authentication is useful in different applications and has strengths and weaknesses. TABLE 2-4 lists the three authentication types: **Type I**, **Type II**, and **Type III authentication**.

Regardless of the type, or types, of authentication used, the authentication system performs the following tasks:

- Collects identification credentials, such as a username.
- Collects authentication credentials, such as a password.
- Finds the stored information that corresponds to the supplied credentials in the user list, often in an authentication database.
- Compares the stored credentials with the supplied credentials. If they match, the user is authenticated.

**⚠ WARNING**

Attackers know that passwords are common and many people have trouble memorizing the complex ones. So, an attacker will likely explore words and numbers that are easily remembered, such as birthdays, anniversaries, and names of pets or children. Another favorite attack is to walk around an office and look for sticky notes on monitors and under keyboards. Many users write down passwords there.

The strongest authentication comes from using more than one type at the same time. Using two types of authentication is called **two-factor authentication**, and using more than two types is called **multifactor authentication (MFA)**. Any operating system or application that requests more than one response during authentication is using two-factor or multifactor authentication. Using more than a single authentication type strengthens the process by increasing the difficulty of impersonating a valid user.

### *Multifactor Authentication*

With the release of Windows Server 2022 and Microsoft's push toward private, public, and hybrid cloud use, the concepts of MFA should be understood as it grows in use to provide

**TABLE 2-4** Authentication Types

| TYPE | DESCRIPTION | STRENGTHS | WEAKNESSES |
|---|---|---|---|
| I—What you know | Some pieces of information only a valid user knows. The most common example of Type I authentication is password. | Simple to implement<br>Simple for users | Password overuse—using one password in multiple applications allows a compromised password to provide access to multiple systems.<br>Writing down complex passwords makes them easy to find. |
| II—What you have | A physical object that contains identity information, such as a token, card, or other device. | More secure than Type I<br>Little or nothing to remember | Can be costly and complex to implement.<br>Risk of lost devices. |
| III—What you are | Physical biometric characteristics, such as a fingerprint, hand print, or retina characteristic. | Very secure<br>Hard to compromise<br>Nothing to carry or remember | More costly than Type I and II.<br>Difficult to implement.<br>Slow process in some applications. |

© Jones & Bartlett Learning.

secure access control. Much like single sign-on (SSO), MFA raises the bar in access control complexity reduction and increased security. As was just discussed in Type I, II and III authentication types, MFA builds on the reduction of exploitation when using, for example, a Type I authentication type such as a password. Although a password can be made and enforced to be complex, it is still something that can be compromised. When you add an additional layer like MFA, you can then create an option to have a code sent to a device to further validate your identity and add an additional (multi) layer of security.

## Access Control Methods

Once a user is identified and properly authenticated, the operating system can grant or deny access based on different rule sets. Access control doesn't only apply to users. Any user or program that requests access to a resource is called the access subject. The resource to which the subject requests access is called the access object. Using these terms, access control is the process of granting or denying subjects access to specific objects.

The most common types of access control rules are **discretionary access control (DAC)**, **mandatory access control (MAC)**, and **role-based access control (RBAC)**. DAC strategies

are defined primarily at the user, or **subject** level. Each **object** has a defined owner, and the owner has complete control over which users can access the object. In an operating system environment, this means access is granted to resources based on the user's settings. Windows extends this concept to allow object properties that can further restrict access. While this approach is simple to define, it can be challenging to maintain in large environments with many users.

The next main type of access control is MAC. A familiar MAC implementation is used in military and government environments. In such an environment, all data objects are labeled with a specific **classification**. Government classifications are Unclassified, Restricted, Confidential, Secret, and Top Secret. Likewise, all subjects are granted a specific **clearance**. A subject must hold a clearance at, or above, the classification level of the desired object to access it. In most environments, subjects must also demonstrate a specific "need to know." This means even if a subject has the correct clearance, they will not be able to access all of the information at that level unless they have a valid reason to do so.

The final type of access control is RBAC. In one point of view, RBAC extends, or generalizes, DAC. Object access is defined by role, as opposed to individual users. Each user is assigned one or more roles. The roles to which a user is assigned define which objects that user (subject) can access. Most operating systems, including Windows, do not directly use RBAC but use a combination of DAC and RBAC. This combination of access control methods is implemented using both user- and user group–based permissions.



## Security Access Tokens, Rights, and Permissions

In a Windows environment, each local system defines local users and groups during the installation process. You can add more local users and groups at any time using the Computer Management tool.

The Local Users and Groups section of the Computer Management tool allows you to add, remove, and manage local users and groups. This tool is most commonly used to create new users and groups, and to associate users with groups. FIGURE 2-4 shows the Local Users and Groups section of the Computer Management tool.

**Technical TIP**

Here's how to open the Computer Management tool:

1. Press the Windows + X keys simultaneously to open the Power User menu.
2. Select Computer Management.

### Security Identifier

Each local user and group in Windows has a unique **security identifier (SID)**. Windows uses the SID to identify users and groups, not the names. Once you create a local user or group,

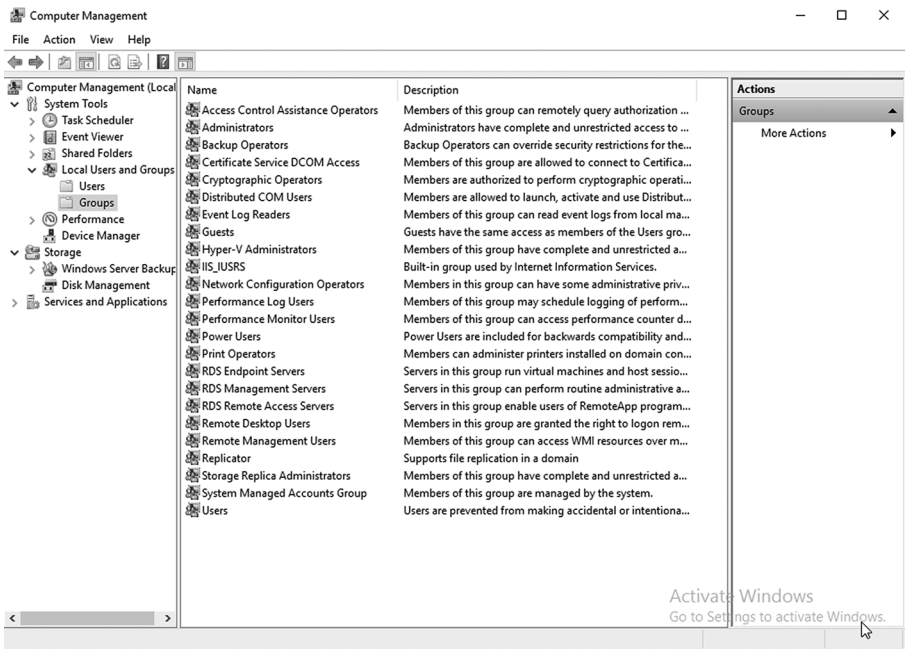



**FIGURE 2-4**

Computer Management tool with open Local Users and Groups.

Courtesy of Microsoft Corporation.

the SID remains the same, even if you change the user or group name. Several well-known users and groups are defined for all Windows machines, but aside from the Microsoft-defined SIDs, all other SIDs are unique to a local machine. **TABLE 2-5** lists a few of the well-known Windows SIDs.

If you create a local user named "Fred" on two different Windows computers, each account will have a different SID. To Windows, the users are completely different, even though they share the same username. This uniqueness between machines makes it difficult to synchronize security settings among multiple standalone computers.

**TABLE 2-5** Well-Known SIDs

| WELL-KNOWN SID | STRING VALUE | IDENTIFIES |
|---|---|---|
| Null SID | S-1-0-0 | Group with no members, often used when an SID is not known |
| World | S-1-1-0 | Group that includes all members |
| Local | S-1-2-0 | Users who log on local terminals |
| Creator Owner ID | S-1-3-0 | SID replaced by the SID of the user who created a new object |
| Creator Group ID | S-1-3-1 | SID replaced by the primary group SID of the user who created a new object |

© Jones & Bartlett Learning.

Every time a Windows user logs on, the operating system fetches the user's SID and the SIDs for all groups to which the user is assigned. The operating system also looks up any local rights for this computer. All of the SIDs and local rights are written to an ID object called your Security Access Token (SAT). Your session's SAT is attached to all your processes. When you run any process, Windows looks at the SAT and any defined access control information for resources to decide whether to grant or deny access to any requested resources.



### Access Rules, Rights, and Permissions

Defining local users and groups is only the first part of the access control implementation process. Windows allows you to associate specific **rights** and permissions to each user that tell Windows what a user can do. User rights define tasks that a user is permitted to carry out, such as take ownership of objects or shut down the computer. **Permissions** define what a user can do to a specific object, such as read or delete the object. Windows stores access rules, or permissions, for resources (objects) in **access control lists (ACLs)**. Each object has an associated ACL and can be used to allow or deny access to the object by user or group. The most common use of ACLs to the general user is to protect files or folders. The Properties dialog box of Files and Folders contains a Security page, which allows you to change the object's access permissions for specific users or groups. The collection of access permissions for each object is called the object's ACL. Windows uses the SAT attached to the currently running process and the ACLs defined for a requested resource to decide whether to grant or deny the requested access. With ACLs, the fundamental building blocks that make one is a list of access control entries (ACE). The list of access control entries is what is used to identify a trustee to derive the rights allowed to grant access, deny access or decide what is auditable. There are two kinds of ACLs, a discretionary access control list (DACL) and a system access control list (SACL). The DACL will be used to identify trustees allowed or that are denied access to an object. The SACL allows an administrator to create log events that can be found in the security event logs within Computer Management, Event Viewer.

## Users, Groups, and Active Directory

Today's computing environments are quickly becoming more diverse and geographically separated, but still integrated using networks. Services that are provided by third parties and delivered via the internet are an integral part of many computing environments. These remote services, generally referred to as cloud services, have transformed the way we look at computing infrastructure. Very few of today's computing environments are composed of unconnected workstations or devices. Windows environments, both home and office, are becoming more dependent on internally and externally shared resources. Even the smallest home networks commonly share printers, storage devices, and network access devices. While defining access permissions for small home networks is fairly easy, the same cannot be said for larger business networks.

### Workgroups

A business network may contain dozens, hundreds, or even thousands of computers. It is desirable to allow users to access network resources from multiple computers attached

to the network. By default, Windows allows computers to share resources by creating a workgroup. A workgroup is a simple peer-to-peer network in which all computers are treated as equals. A workgroup simply allows standalone computers to "see" each other's shared resources. Common shared resources include files, folders, and printers. Each resource can be shared or hidden, and the access to each resource can be controlled by user and group permissions. The main concept behind using a Windows Workgroup is in resource decentralizing. Although this is an easy to set up system that requires little cost, it becomes difficult to manage and secure and is highly discouraged from use.

The main problem with Windows workgroups is that each computer is still defined as a standalone computer. That means users and groups must be defined on multiple computers. In fact, if you want all users to have access on all computers in the workgroup, you have to add the local users on every computer. On small networks, this may seem okay at first, but maintenance can quickly get out of hand. Since all user and group accounts in a workgroup are local accounts, every change to security permissions must be applied to every computer. Administration of workgroups with more than a half dozen computers can quickly become too difficult to remain viable.

## Active Directory

Microsoft offered a solution to local users and groups in the original Windows NT operating system. All Windows operating systems since Windows NT have the ability to share user and group definitions with other computers connected to the same network. Many operating systems support a generic capability to share such information, called directory services, which was a flat file architecture. This functionality has matured into a core Windows feature, called **Active Directory**, which is a true hierarchical directory based on tree model that uses LDAP and X.500 structure. Instead of having to define users and groups locally on each computer, Active Directory allows users and groups to be defined once at a central location (centralized vs. the de-centralized workgroup model) and shared among multiple computers. You get to define the limits of how many computers share users and groups by defining domains. A domain can be thought of as a group of computers that can be grouped together for some purpose. The actual database of shared users and groups is stored on one or more computers designated as domain controllers. This shared database makes it easy to maintain users, groups, and permissions in a central location. Microsoft Active Directory has many more features, but its main feature is the ability to define identity and **authorization** permission that can be shared among multiple computers within one or more domains. This capability greatly simplifies security administration in larger networks.

Implementing Active Directory requires more hardware for domain controller computers and network devices. It also requires additional administration time and resources to ensure shared information is protected and available in a timely fashion. In fact, securing Active Directory information ensures its confidentiality, integrity, and availability. An organization that uses Active Directory depends on the security of its information.

The main reason organizations invest in Active Directory resource requirements is not just to make users happy. While implementing a single sign-on capability is a huge benefit for users in a large organization, the real reason to implement Active Directory is to reduce redundant administrative effort. Securing resources across a network, or multiple networks,

requires substantial administrative effort. The amount of redundant controls administrators must keep current opens opportunities for attacks on stale controls. Active Directory automates and centralizes many controls, making the entire environment more secure.

Unlike workgroup environments, a domain user or group definition is defined on the domain controller, and its SID will be the same for all computers in the domain. In the workgroup environment, a local user named "Fred" may be defined on each computer, but the SID will be different on each one. Auditing Fred's actions across multiple computers can be more difficult when trying to coordinate multiple SIDs. If the user named "Fred" is defined as a domain user instead, Fred's SID will be stored in the domain controller and be shared with computers in the domain when needed. Fred's SID never changes, regardless of what computer in the domain is being accessed. Active Directory removes the problem of different SIDs on each computer.



## Windows Attack Surfaces and Mitigation

Windows, like any operating system, is a collection of services provided to allow users to interact with the physical hardware. Each service provides a specific set of access methods to the hardware and functions that satisfy user requirements. In short, operating system services provide some functionality to users. Each service exposes some part of the computer to external access. Regardless of the care taken to ensure the security of the provided access, there is a risk when exposing any part of a computer to external access. All computing systems can become vulnerable based on lack of hardening, configuration mistakes, bugs, and unknown vulnerabilities—weaknesses that can allow unauthorized access if successfully exploited. The total collection of all possible vulnerabilities that could provide unauthorized access to computer resources is referred to as the **attack surface**. Another way to define the attack surface is the set of all exposed vulnerabilities. The attack surface is further broken down into an **attack tree** and an **attack vector**. An attack tree is what a threat actor will devise, based on vulnerability scanning, testing, and verifying weaknesses, as the blueprint for building the intended attacks. The attack surface is consulted to design the attack tree. The attack vector is considered the road or path the threat actor takes to conduct the attacks. All three are important to know when considering how to create a multilayer defense strategy or apply the concept of defense-in-depth.

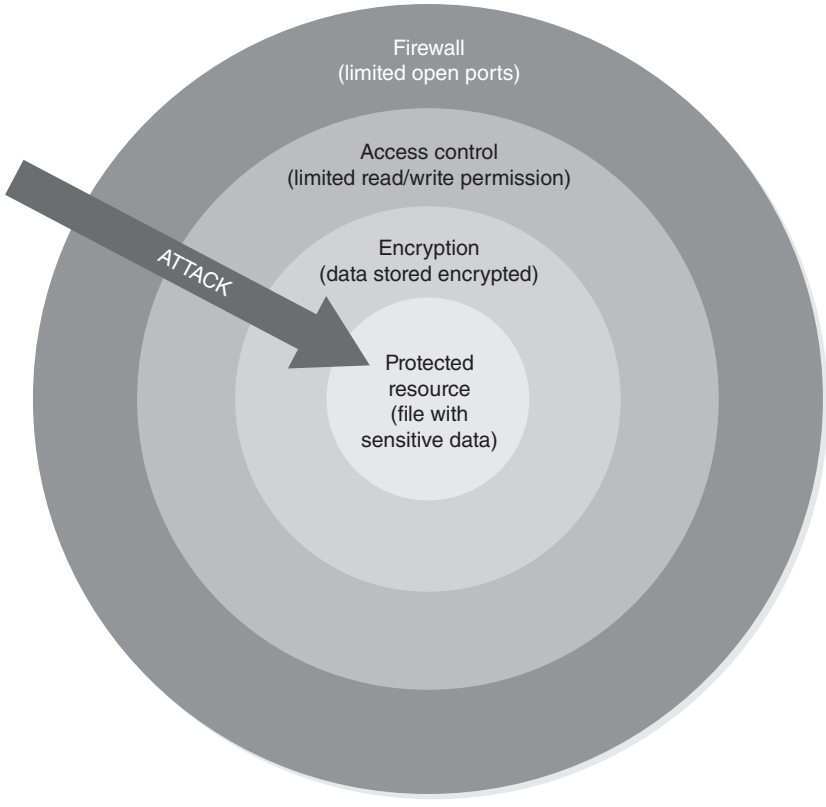
### Multilayered Defense

The discipline of information security is concerned with minimizing the attack surface of any protected resource. This can be achieved by successfully removing, or substantially reducing, the ability of an attacker to conduct an attack against vulnerability. There are several effective methods to minimize the attack surface. The most secure environments employ a combination of strategies. Most important, a solid overall security strategy avoids monolithic solutions. Relying on a single control to protect a resource increases the probability of a successful attack. Always design a defense strategy that is multilayered, which requires multiple controls be compromised to exploit any vulnerability. Such a strategy is often called a defense-in-depth approach to security. **FIGURE 2-5** shows how a multilayered defense strategy protects resources.

**FIGURE 2-5**

Multilayered defense.

© Jones & Bartlett Learning.




At a high level, the easiest way to reduce the attack surface is to remove functionality. Suppose an attacker wants to exploit an Internet Information Services (IIS) web server vulnerability. The quickest way to deny such an attack is to disable or remove the IIS web server. Although disabling IIS may be easy, it may be an unacceptable solution. What if the server computer in question is a web server in a production environment servicing clients or customers? Disabling IIS in that case is not an option! But in many cases, disabling a web server is perfectly okay if it is not in use—this is also considered hardening a system by removing unused services. What steps you take all depends on the purpose of the server computer and the services it must provide to be functional. Defining the computer role and configuring the operating system for that role is one of the best ways to reduce the attack surface of any computer.

The best way to reduce the need to harden systems is to ensure that when installed, all services needed are known up front and nothing else is added. Proper workstation and server role definition and planning make it easier during the installation process to only install and enable services that are necessary for a particular computer. By installing and enabling only the necessary services, you reduce the operating system's complexity and overall attack surface. Windows Server versions since Windows Server 2008 include the Server Manager tool, which makes it easy to define specific roles for a server. There is even a newer installation option available called Server Core that installs only the basic services to support file and print services, Active Directory, and a few other basic server functions.

The resulting installation takes up less disk space, consumes less memory, and has a much smaller attack surface due to fewer installed services. Server Core doesn't include a Graphical User Interface (GUI) and is primarily managed via PowerShell, so it is the common base image for servers that are managed remotely. PowerShell is a task automation and configuration management program from Microsoft, consisting of a command-line shell and cmdlets that allow you the ability to manually (or use automation) to handle any and all administrative tasks in the Windows Operating System(s).

Here's how to open the PowerShell tool:

1. PowerShell can be accessed by going to the Windows 11 or Windows Server 2022 search bar and typing PowerShell.
2. Once open, type help to see a list of online guidance to get you started.

If you are new to PowerShell, start to use the interface to learn, and search Microsoft documentation online to get specific guidance on commands. Most of the security administration done in Windows Server 2022 core is done via the PowerShell interface.



### Mitigation

If you must install and enable a service, such as the IIS web server, you will have to employ measures to protect your system from IIS vulnerabilities. The strategies you will learn to secure any resource, also called mitigating a risk, fall into two main categories:

1. Remove vulnerabilities
2. Stop attacks from exploiting vulnerabilities

Between the two strategies, the former is the better option. Removing vulnerabilities by disabling the service or by updating vulnerable software to a more secure version removes the possibility of a successful attack against a specific vulnerability. However, just applying a security patch doesn't guarantee there is no more vulnerability in the service. That's why a multilayered defense is so important. Never rely on a single control or strategy to protect a resource. Always employ multiple levels of controls. Your goal is to make attackers work very hard to exploit resources on protected computers.

## Fundamentals of Microsoft Windows Security Monitoring and Maintenance

The process of securing a Windows computer system and maintaining a secure system is an iterative process. Three main points in a system's life cycle serve as milestones for security management. It is important to maintain a secure system when you do the following:

1. Install the operating system or application software.
2. Monitor the operation of the computer system.
3. Make any configuration changes to the computer system.

The first and third milestones are easy events to identify, at least for small computing environments, and result from some administrative action. Organizations with more than several dozen servers and workstations find that all of the milestones can be challenging.

As any organization grows, policies that result in configuration standards make the administration process easier. It is important to include security concerns in the installation and modification procedures to avoid introducing unintended vulnerabilities. For example, the procedure to upgrade to a new version of SQL Server should include steps to ensure insecure user accounts or demo procedures are not added to the system environment. It is also desirable to carry out validation tasks after any new installations or configuration changes to ensure system security meets stated goals. The second milestone occurs at intervals specified by security administrators for each system. Monitoring can be as frequent or infrequent as necessary. Typically, the monitoring frequency depends on the volatility of the resource and the risk of attacks against the resource.

## Security Monitoring

In general, the process of security monitoring involves comparing performance or configuration information with a stated baseline. Microsoft offers several tools and resources to help create and maintain secure systems. The basic process of security monitoring is to follow these steps:

1. Define security goals.
2. Describe secure behavior as a baseline.
3. Sample performance information and compare with the baseline.
4. Report anomalies.

The tools and resources available for Windows operating systems include both the suggested baselines and the tools to compare baselines with system configuration and performance information. System configuration information could include user and group definitions, critical resource permissions, and lists of folders. Baseline information could include a list of known vulnerable users, groups, and folders. Any items in the lists of users, groups, or folders that exist in the baseline could indicate vulnerability. Performance information could include web server log files. SIEM and SOAR tools help collect, manage, and automate the process of monitoring such log files and could show attacks that have been carried out against your system. Security, information, and event management (SIEM) is used to collect and manage security log data, while security orchestration, automation, and response (SOAR) is used to help create responses to the data collected. Or even better, you could see the evidence of a preattack reconnaissance effort. In the case of the latter example, you could use such information to take action to protect your system from an expected future attack.

## Identify Vulnerabilities

The idea behind monitoring is to simply consider the current state of a system and help identify any existing security vulnerabilities that may arise from changes in your IT infrastructure. You are proactively taking an approach similar to what an attacker would do in planning an attack. Later in this book, you'll learn about tools that will help you uncover vulnerabilities in your computing environment. After identifying vulnerabilities, you must

decide how to address each one. It sounds simple, and it really is at its core. However, putting that goal into practice can be difficult. There are many options to address each vulnerability, and you have to choose the best option for each situation. The process isn't a single occurrence of each step. Security monitoring and responding to the results is a process you'll repeat over and over to keep your systems secure.



## CHAPTER SUMMARY

Microsoft Windows is a mature family of operating systems that address the needs of many different types of enterprises and users. The wide variety of information managed by Windows systems and the large number of existing systems that run Windows make the platform an attractive target for attacks. Attackers know that each new Windows release gives security administrators new and improved tools to help secure protected resources and information. They know the only way to stay ahead of the game, and outsmart Windows administrators, is to be as knowledgeable about the operating system and its attack surface as possible. And they know it well.

The best way to defend your systems from attack is to know as much, if not more, about your operating system and its vulnerabilities as attackers know. Think like an attacker, a hacker, or a threat actor. Know where and how they'll likely attempt to compromise your systems. Anticipate their moves, and you'll be better prepared to place deterrents in their path. You have now learned the basics of the Windows operating system architecture.

## KEY CONCEPTS AND TERMS

Access control
Access control lists (ACLs)
Active Directory
Attack surface
Attack tree
Attack vector
Authentication
Authorization
Classification
Clearance
Control
Discretionary access control (DAC)
Exploit
Hardware Abstraction Layer (HAL)
Identification
Kernel
Kernel mode
Mandatory access control (MAC)
Microkernel
Multifactor authentication (MFA)
Object
Permissions
Rights
Role-based access control (RBAC)
Security identifier (SID)
Subject
Supervisor mode
Two-factor authentication
Type I authentication
Type II authentication
Type III authentication
User mode

## CHAPTER 2 ASSESSMENT

**1.** Which of the following is *not* a goal of a secure environment?

A. All required information is available to authorized users.
B. No information is available to unauthorized users.
C. All required information is available to unauthorized users.
D. No classified information is available to unauthorized users.

**2.** Which term describes the central component of an operating system?

A. Kernel
B. Shell
C. Hardware Abstraction Layer
D. Executive

**3.** What are the two run modes for Windows programs?

A. Supervisor mode and executive mode
B. Kernel mode and supervisor mode
C. User mode and executive mode
D. Kernel mode and user mode

**4.** Which of the following Windows components resides in memory to provide the core operating system services?

A. Kernel
B. Microkernel
C. Executive
D. Hardware Abstraction Layer

**5.** What is the name of the process that proves you are who you say you are?

A. Identification
B. Authorization
C. Permission
D. Authentication

**6.** Which type of authentication is a smart card?

A. Type I
B. Type II
C. Type III
D. Type IV

**7.** Which access control method relates data classification to user clearance?

A. MAC
B. DAC
C. RBAC
D. LDAC

**8.** What value uniquely identifies a user or group in Windows?

A. UID
B. SAT
C. SID
D. ACE

**9.** If the same user is created on three separate Windows computers, which value is the same on all three computers?

A. SID
B. User code
C. Username
D. SAT

**10.** Which Windows feature allows users and groups to be "shared" among machines?

A. Domain controller
B. Workgroup
C. SID
D. Active Directory

**11.** What defines the limit of how many computers share users and groups?

A. SID
B. GUID
C. Domain
D. Workgroup

**12.** Which of the following best describes the term *attack surface*?

A. All possible vulnerabilities in application software that could be exploited
B. All possible vulnerabilities that could be exploited
C. The most likely avenues of attack
D. Known vulnerabilities that have not been patched

13. When possible, what is the best way to mitigate vulnerability in a specific service?
    A. Remove the service.
    B. Disable the service.
    C. Block access to the service.
    D. Patch the service.

14. When monitoring a Windows system, with what do you compare current system performance to test for security compliance?
    A. The previous monitoring results
    B. A normal performance scan
    C. A stated baseline
    D. A defined security goal

15. Why should you immediately test your system for security compliance after making a configuration change?
    A. Configuration changes generally increase security.
    B. Configuration changes generally decrease security.
    C. Configuration changes may introduce new vulnerabilities.
    D. Configuration changes may remove existing vulnerabilities.