## Chapter 5

# Analyzing Vulnerability Scans

# Outlines

- the Common Vulnerability Scoring System
- Access Vector Metric
- Access Complexity Metric
- Authentication Metric
- Confidentiality Metric
- Integrity Metric
- Availability Metric
- Calculating the Exploitability Score
- Calculating the Impact Score
- Determining the Impact Function Value
- Calculating the Base Score
- Categorizing CVSS Base Scores
- False Positives
- Informational Results
- Reconciling Scan Results with Other Data Sources
- Trend Analysis
- Server and Endpoint Vulnerabilities
- Missing Patches
- Buffer Overflows
- Privilege Escalation
- Arbitrary Code Execution
- Hardware Flaws
- Firmware Vulnerabilities
- Spectre and Meltdown
- Point-of-Sale System Vulnerabilities
- Insecure Protocol Use
- Debug Modes
- Network Vulnerabilities
- Missing Firmware Updates
- SSL and TLS Issues
- Outdated SSL/TLS Versions
- Insecure Cipher Use
- Certificate Problems
- Domain Name System (DNS)
- Internal IP Disclosure
- Virtual Private Network Issues
- Virtualization Vulnerabilities
- VM Escape
- Management Interface Access
- Virtual Host Patching
- Virtual Guest Issues
- Virtual Network Issues
- Internet of Things (IoT)
- IoT Uprising
- Web Application Vulnerabilities
- Injection Attacks
- Cross-Site Scripting

# The Common Vulnerability Scoring System (CVSS)

It provides a technique for scoring each vulnerability on a variety of measures. Cybersecurity analysts often use CVSS ratings to prioritize response actions. Analysts scoring a new vulnerability begin by rating the vulnerability on six different measures:

- Access vector
- Access complexity
- Authentication
- Confidentiality
- Integrity
- Availability

Each measure is given both a descriptive rating and a numeric score. The first three measures evaluate the exploitability of the vulnerability, whereas the last three evaluate the impact of the vulnerability.

## Access Vector Metric

The access vector metric describes how an attacker would exploit the vulnerability.

| Value | Description | score |
|---|---|---|
| Local (L) | The attacker must have physical or logical access to The affected system. | 0.395 |
| Adjacent Network (A) | The attacker must have access to the local network That the affected system is connected to. | 0.646 |
| Network (N) | The attacker can exploit the vulnerability remotely Over a network. | 1.000 |

# Access Complexity Metric

**The access complexity metric describes the difficulty of exploiting the vulnerability.**

| Value | Description | score |
|---|---|---|
| High (H) | Exploiting the vulnerability requires "specialized" Conditions that would be difficult to find. | 0.350 |
| Medium (M) | Exploiting the vulnerability requires "somewhat Specialized" conditions. | 0.610 |
| Low (L) | Exploiting the vulnerability does not require any Specialized conditions. | 0.710 |

# Authentication Metric

**The authentication metric describes the authentication hurdles that an attacker would need to clear to exploit vulnerability.**

| Value | Description | score |
|---|---|---|
| Multiple (M) | Attackers would need to authenticate two or more times to exploit the vulnerability. | 0.450 |
| Single (S) | Attackers would need to authenticate once to exploit The vulnerability. | 0.560 |
| None (N) | Attackers do not need to authenticate to exploit the Vulnerability. | 0.704 |

# Confidentiality Metric

**The confidentiality metric describes the type of information disclosure that might occur if an attacker successfully exploits the vulnerability.**

| Value | Description | score |
|---|---|---|
| None (N) | There is no confidentiality impact. | 0.000 |
| Partial (P) | Access to some information is possible, but the attacker does not have control over what information is compromised. | 0.275 |
| Complete (C) | All information on the system is compromised. | 0.660 |

## Integrity Metric

The integrity metric describes the type of information alteration that might occur if an attacker successfully exploits the vulnerability.

| Value | Description | score |
|---|---|---|
| None (N) | There is no integrity impact. | 0.000 |
| Partial (P) | Modification of some information is possible, but the Attacker does not have control over what information is modified. | 0.275 |
| Complete (C) | The integrity of the system is totally compromised and the attacker may change any information at will. | 0.660 |

## Availability Metric

The availability metric describes the type of disruption that might occur if an attacker successfully exploits the vulnerability.

| Value | Description | score |
|---|---|---|
| None (N) | There is no availability impact. | 0.000 |
| Partial (P) | The performance of the system is degraded. | 0.275 |
| Complete (C) | The system is completely shut down. | 0.660 |

Interpreting the CVSS Vector
The CVSS vector uses a single-line format to convey the ratings of vulnerability on all six of the metrics described in the preceding sections. For example, recall the CVSS vector presented.

**CVSS2#AV: N/AC: M/Au: N/C: P/I: N/A: N**

- vector was composed using CVSS version 2
- Sections correspond to each of the six CVSS metrics.
- Access Vector: Network (score: 1.000)
- Access Complexity: Medium (score: 0.610)
- Authentication: None (score: 0.704)
- Confidentiality: Partial (score: 0.275)
- Integrity: None (score: 0.000)
- Availability: None (score: 0.000)

# Exploitability score, impact score, and impact function

## Calculating the Exploitability Score

### Formula:

**Exploitability = 20 × Access Vector × Access Complexity × Authentication**

**Ex:**

**Exploitability = 20 × 1.000 × 0.610 × 0.704**

**Exploitability = 8.589**

## Calculating the Impact Score

**Formula:**

**Impact = 10.41 × (1 − (1 − Confidentiality) × (1 − Integrity) × (1 − Availability))**

**Ex:**

**Impact = 10.41 × (1 − (1 − 0.275) × (1 − 0) × (1 − 0))**
**Impact = 10.41 × (1 − (0.725) × (1) × (1))**
**Impact = 10.41 × (1 − 0.725)**
**Impact = 10.41 × 0.275**
**Impact = 2.863**

## Determining the Impact Function Value

**The impact function is a simple check. If the impact score is 0, the impact function value is also 0. Otherwise, the impact function value is 1.176. So, in our example case, the result is as follows:**

**Impact Function = 1.176**

## Calculating the Base Score

**Formula:**

**Base Score = ((0.6 × Impact) + (0.4 × Exploitability) − 1.5) × Impact Function**

**Ex:**

BaseScore = ((0.6 × 2.863) + (0.4 × 8.589) − 1.5) × 1.176
BaseScore = (1.718 + 3.436 − 1.5) × 1.176
BaseScore = 3.654 × 1.176
BaseScore = 4.297

## Categorizing CVSS Base Scores

| CVSS score | Risk category |
|---|---|
| Under 4.0 | Low |
| 4.0 or higher, but less than 6.0 | Medium |
| 6.0 or higher, but less than 10.0 | High |
| 10.0 | Critical |

# False Positives

Vulnerability scanners are useful tools, but they aren't foolproof. Scanners do sometimes make mistakes for a variety of reasons. The scanner might not have sufficient access to the target system to confirm vulnerability, or it might simply have an error in a plug-in that generates an erroneous vulnerability report. When a scanner reports a vulnerability that does not exist, this is known as a false positive error.

# Informational Results

Vulnerability scanners often supply very detailed information when run using default configurations. Not everything reported by a vulnerability scanner actually represents a significant security issue. Nevertheless, scanners provide as much information as they are able to determine to show the types of information that an attacker might be able to gather when conducting a reconnaissance scan.

# Reconciling Scan Results with Other Data Sources

- Logs from servers, applications, network devices, and other sources that might contain information about possible attempts to exploit detected vulnerabilities

- Security information and event management (SIEM) systems that correlate log entries from multiple sources and provide actionable intelligence

- Configuration management systems that provide information on the operating system and applications installed on a system

# Trend Analysis

Trend analysis is also an important part of a vulnerability scanning program. Managers should watch for overall trends in vulnerabilities, including the number of new vulnerabilities arising over time.

# Server and Endpoint Vulnerabilities

Computer systems are quite complex. Operating systems run on both servers and endpoints comprising millions of lines of code, and the differing combinations of applications they run makes each system fairly unique. It's no surprise, therefore, that many of the vulnerabilities detected by scans exist on server and endpoint systems, and these vulnerabilities are often among the most complex to remediate.

# Missing Patches

Applying security patches to systems should be one of the core practices of any information security program, but this routine task is often neglected due to a lack of resources for preventive maintenance. One of the most common alerts from a vulnerability scan is that one or more systems on the network are running an outdated version of an operating system or application and require security patch(es).

- AV: N tells us that the vulnerability can be exploited remotely by a hacker over the network.

- AC: L tells us that the access complexity is low, meaning that a relatively unskilled attacker can exploit it.

- Au: N tells us that no authentication is required to exploit the vulnerability.

- C:C, I:C , and A:C tell us that someone exploiting this vulnerability is likely, to completely compromise the confidentiality, integrity, and availability of the system.

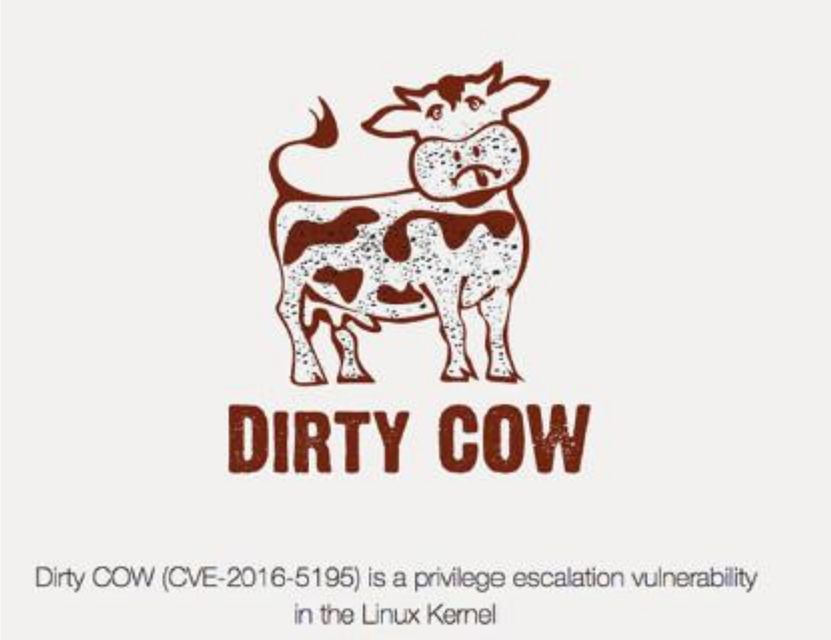
# Buffer Overflows

Buffer overflow attacks occur when an attacker manipulates a program into placing more data into an area of memory than is allocated for that program's use. The goal is to overwrite other information in memory with instructions that may be executed by a different process running on the system.

- CVE 1999-1058: Buffer overflow in Vermillion FTP Daemon
- CVE 2001-0876: Buffer overflow in Universal Plug and Play (UPnP) on Windows 98, 98SE, ME, and XP
- CVE 2002-0126: Buffer overflow in Black Moon FTP Server 1.0 through 1.5
- CVE 2003-0818: Multiple integer overflows in Microsoft ASN.1 library

# Privilege Escalation

Privilege escalation attacks seek to increase the level of access that an attacker has to a target system. They exploit vulnerabilities that allow the transformation of a normal user account into a more privileged account, such as the root superuser account.



Dirty COW (CVE-2016-5195) is a privilege escalation vulnerability in the Linux Kernel

# Arbitrary Code Execution

Arbitrary code execution vulnerabilities allow an attacker to run software of their choice on the targeted system. This can be a catastrophic event, particularly if the vulnerability allows the attacker to run the code with administrative privileges. Remote code execution vulnerabilities are an even more dangerous subset of code execution vulnerabilities because the attacker can exploit the vulnerability over a network connection without having physical or logical access to the target system.

# Hardware Flaws

While most vulnerabilities affect operating systems and applications, occasionally vulnerabilities arise that directly affect the underlying hardware running in an organization. These may arise due to firmware issues or, in rarer cases, may be foundational hardware issues requiring remediation.

# Firmware Vulnerabilities

Many hardware devices contain firmware: computer code stored I nonvolatile memory on the device, where it can survive a reboot of the device. Firmware often contains the device's operating system and/or configuration information. Just like any other code, the code contained in firmware may contain security vulnerabilities.

# Spectre and Meltdown

Hardware may also contain intrinsic vulnerabilities that can be quite difficult to remediate. In 2017, security researchers announced the discovery of two related hardware vulnerabilities in nearly every microprocessor manufactured during the preceding two decades. These vulnerabilities, named Spectre and Meltdown, exploit a feature o the chips known as speculative execution to allow processes to gain access to information reserved for other processes.

## Point-of-Sale System Vulnerabilities

The point-of-sale (POS) systems found in retail stores, restaurants, and hotels are lucrative targets for attackers and penetration testers alike. These systems often store, process, and/or transmit credit card information, making them highly valuable in the eyes of an attacker seeking financial gain.

## Insecure Protocol Use

Many of the older protocols used on networks in the early days of the Internet were designed without security in mind. They often failed to use encryption to protect usernames, passwords, and the content sent over an open network, exposing the users of the protocol to eavesdropping attacks. Telnet is one example of an insecure protocol used to gain command-line access to a remote server. The File Transfer Protocol (FTP) provides the ability to transfer files between systems but does not incorporate security features.

## Debug Modes

Many application development platforms support debug modes that give developers crucial information needed to troubleshoot applications in the development process. Debug modes typically provide detailed information on the inner workings of an application and server as well as supporting databases. Although this information can be useful to developers, it can inadvertently assist an attacker seeking to gain information about the structure of a database, authentication mechanisms used by an application, or other details.

## Network Vulnerabilities

Modern interconnected networks use a complex combination of infrastructure components and network appliances to provide widespread access to secure communications capabilities.
These networks and their component parts are also susceptible to security vulnerabilities that may be detected during a vulnerability scan.

## Missing Firmware Updates

Operating systems and applications aren't the only devices that require regular security updates. Vulnerability scans may also detect security problems in network devices that require firmware updates from the manufacturer to correct.

## SSL and TLS Issues

The Secure Sockets Layer (SSL) protocol and its successor, Transport Layer Security (TLS), offer a secure means to exchange information over the Internet and private networks. Although these protocols can be used to encrypt almost any type of network communication, they are most commonly used to secure connections to web servers and are familiar to end users designated by the S in HTTPS.

## Outdated SSL/TLS Versions

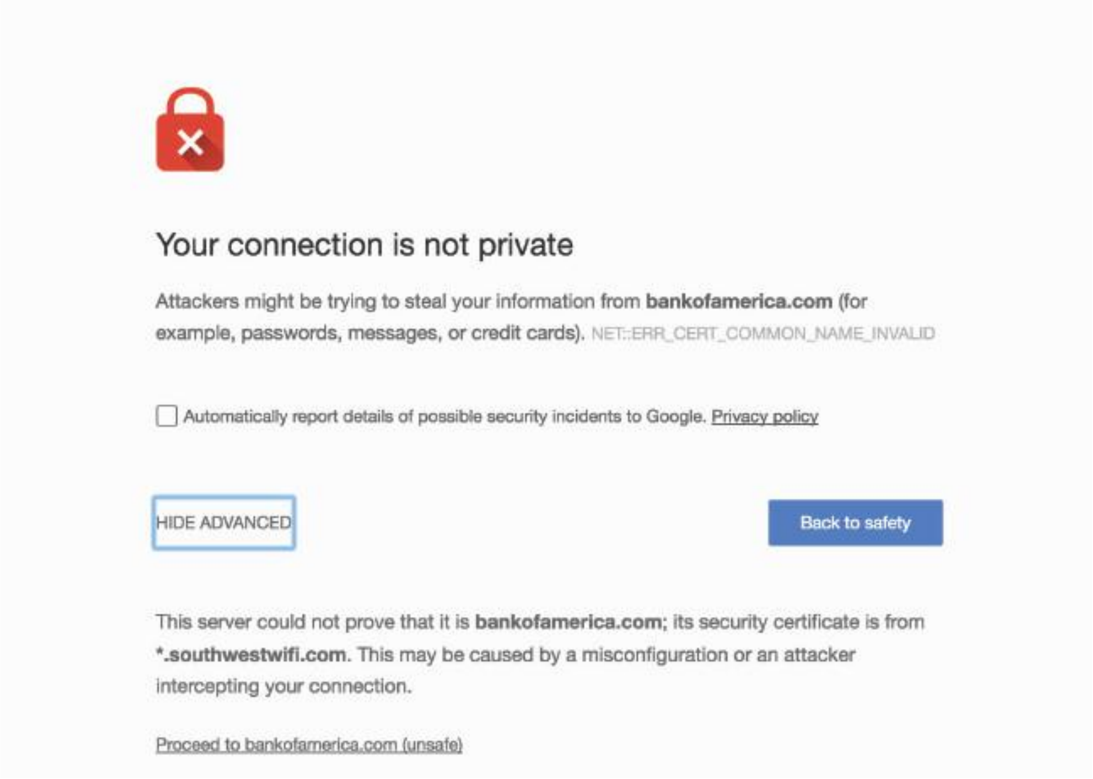
SSL is no longer considered secure and should not be used on production systems. The same is true for early versions of TLS. Vulnerability scanners may report that web servers are using these protocols, and cybersecurity analysts should understand that any connections making use of these outdated versions of SSL and TLS may be subject to eavesdropping attacks.

## Insecure Cipher Use

SSL and TLS are commonly described as cryptographic algorithms, but in fact, this is not the case. The SSL and TLS protocols describe how cryptographic ciphers may be used to secure network communications, but they are not cryptographic ciphers themselves.
Instead, they allow administrators to designate the cryptographic ciphers that can be used with those protocols on a server-by-server basis. When a client and server wish to communicate using SSL/TLS, they exchange a list of ciphers that each system supports and agree on a mutually acceptable cipher. Some ciphers contain vulnerabilities that render them insecure because of their susceptibility to eavesdropping attacks.

# Certificate Problems

**SSL and TLS rely on the use of digital certificates to validate the identity of servers and exchange cryptographic keys. Website users are familiar with the error messages displayed in web browsers.**
**Expiration of the Digital Certificate Digital certificates have validity periods and expiration dates. When you see an expired certificate, it most likely means that the server administrator failed to renew the certificate in a timely manner. Unknown Certificate Authority (CA) anyone can create a digital certificate, but digital certificates are only useful if the recipient of a certificate trusts the entity that issued it. Operating systems and browsers contain instructions to trust well-known CAs but will show an error if they encounter a certificate issued by an unknown or untrusted CA.**



Your connection is not private

Attackers might be trying to steal your information from **bankofamerica.com** (for example, passwords, messages, or credit cards). NET::ERR_CERT_COMMON_NAME_INVALID

□ Automatically report details of possible security incidents to Google. Privacy policy

This server could not prove that it is **bankofamerica.com**; its security certificate is from ***.southwestwifi.com**. This may be caused by a misconfiguration or an attacker intercepting your connection.

Proceed to bankofamerica.com (unsafe)

## Domain Name System (DNS)

The Domain Name System (DNS) provides a translation service between domain names and IP addresses. DNS allows end users to remember user-friendly domain names, such as apple.com, and not worry about the mind-numbing IP addresses actually used by those servers.
DNS servers are a common source of vulnerabilities on enterprise networks. Despite the seemingly simple nature of the service, DNS has a track record of many serious security vulnerabilities and requires careful configuration and patching. Many of the issues with DNS services are those already discussed in this chapter, such as buffer overflows, missing patches, and code execution vulnerabilities, but others are specific to the DNS service. Because DNS vulnerabilities are so prevalent, DNS servers are a common first target for attackers and penetration testers alike

DNS amplification vulnerability on two servers on an organization's network. In this type of attack, the attacker sends spoofed DNS requests to a DNS server that are carefully designed to elicit responses that are much larger in size than the original requests. These large response packets then go to the spoofed address where the DNS server believes the query originated. The IP address used in the spoofed request is actually the target of a denial-of-service attack and is bombarded by very large responses from DNS servers all over the world to queries that it never sent. When conducted in sufficient volume, DNS amplification attacks can completely overwhelm the targeted systems, rendering them inoperable

## Internal IP Disclosure

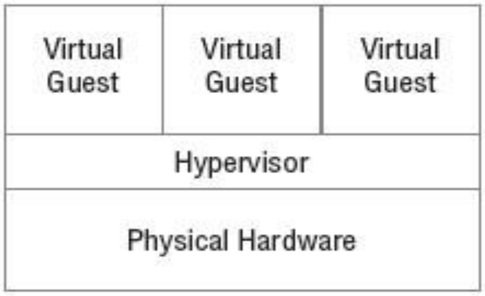
IP addresses come in two different variants: public IP addresses, which can be routed over the Internet, and private IP addresses, which can only be used on local networks. Any server that is accessible over the Internet must have a public IP address to allow that access, but the public IP address is typically managed by a firewall that uses the Network Address Translation (NAT) protocol to map the public address to the server's true, private IP address. Systems on the local network can use the server's private address to access it directly, but remote systems should never be aware of that address. Servers that are not properly configured may leak their private IP addresses to remote systems. This can occur when the system includes its own IP address in the header information returned in the response to an HTTP request. The server is not aware that NAT is in use, so it uses the private address in its response. Attackers and penetration testers can use this information to learn more about the internal configuration of a firewalled network.

## Virtual Private Network Issues

Virtual private networks (VPNs) provide employees with secure remote access to the organization's network. As with any application protocol, administrators must ensure that the VPN services offered by the organization are fully patched to current levels. In addition, VPNs require the use of cryptographic ciphers and suffer from similar issues as SSL and TLS when they support the use of insecure ciphers.

## Virtualization Vulnerabilities

Virtualization technology allows multiple guest systems to share the same underlying hardware. In a virtualized data center, the virtual host hardware runs a special operating system known as a hypervisor that mediates access to the underlying hardware resources. Virtual machines then run on top of this virtual infrastructure provided by the hypervisor, running standard operating systems such as Windows and Linux variants. The virtual machines may not be aware that they are running in a virtualized environment because the hypervisor tricks them into thinking that they have normal access to the underlying hardware when, in reality, that hardware is shared with other systems.



## VM Escape

Virtual machine escape vulnerabilities are the most serious issue that may exist in a virtualized environment, particularly when a virtual host runs systems with differing security levels. In an escape attack, the attacker has access to a single virtual host and then manages to leverage that access to intrude on the resources assigned to a different virtual machine. The hypervisor is supposed to prevent this type of intrusion by restricting a virtual machine's access to only those resources assigned to that machine. Escape attacks allow a process running on the virtual machine to "escape" those hypervisor restrictions.

## Management Interface Access

Virtualization engineers use the management interface for a virtual infrastructure to configure the virtualization environment, set up new guest machines, and regulate access to resources. This management interface is extremely sensitive from a security perspective, and access should be tightly controlled to prevent unauthorized individuals from gaining access. In addition to using strong multifactor authentication on the management interface, cybersecurity professionals should ensure that the interface is never directly accessible from a public network. Vulnerability scans that detect the presence of an accessible management interface will report this as a security concern.

## Virtual Host Patching

This chapter has already discussed the importance of promptly applying security updates to operating systems, applications, and network devices. It is equally important to ensure that virtualization platforms receive security updates that may affect the security of virtual guests or the entire platform. Patches may correct vulnerabilities that allow virtual machine escape attacks or other serious security flaws.

## Virtual Guest Issues

Cybersecurity analysts should think of each guest machine running in a virtualized environment as a separate server that requires the same security attention as any other device on the network. Guest operating systems and applications running on the guest OS must be promptly patched to correct security vulnerabilities and be otherwise well maintained. There's no difference from a security perspective between a physical server and a virtualized server.

# Virtual Network Issues

As data centers become increasingly virtualized, a significant amount of network traffic never actually touches a network! Communications between virtual machines that reside on the same physical hardware can occur in memory without ever touching a physical network. For this reason, virtual networks must be maintained with the same attention to security that administrators would apply to physical networks. This includes the use of virtual firewalls to control the flow of information between systems and the isolation of systems of differing security levels on different virtual network segments.

# Internet of Things (IoT)

In some environments, cybersecurity analysts may encounter the use of supervisory control and data acquisition (SCADA) systems, industrial control systems (ICSs), and other examples of the Internet of Things (IoT). These systems allow the connection of physical devices and processes to networks and provide tremendous sources of data for organizations seeking to make their business processes more efficient and effective. However, they also introduce new security concerns that may arise on vulnerability scans. As with any other device on a network, IoT devices may have security vulnerabilities and are subject to network-based attacks. However, it is often more difficult to patch IoT devices than it is to patch their traditional server counterparts because it is difficult to obtain patches. IoT device manufacturers may not use automatic update mechanisms, and the only way that cybersecurity analysts may become aware of an update is through a vulnerability scan or by proactively subscribing to the security bulletins issued by IoT device manufacturers. IoT devices also often have unique characteristics compared to other devices attached to the networks. They often exist as embedded systems, where there is an operating system and computer running in the device that may not be obvious or accessible to the outside world. For example, large multifunction copier/printer units found in office environments often have an entire Windows or Linux operating system running internally that may act as a file and print server. IoT devices also often run real-time operating systems (RTOS). These are either special purpose operating systems or variants of standard operating systems designed to process data rapidly as it arrives from sensors or other IoT components.

# IoT Uprising

On October 21, 2016, a widespread distributed denial of service (DDoS) attack shut down large portions of the Internet, affecting services run by Amazon, The New York Times, Twitter, Box, and other providers. The attack came in waves over the course of the day and initially mystified technologists seeking to bring systems back online. Investigation later revealed that the outages occurred when Dyn, a global provider of DNS services, suffered a debilitating attack that prevented it from answering DNS queries. Dyn received massive amounts of traffic that overwhelmed its servers. The source of all of that traffic? Attackers used an IoT botnet named Mirai to leverage the bandwidth available to baby monitors, DVRs, security cameras, and other IoT devices in the homes of normal people. Those bonneted devices received instructions from a yet-unknown attacker to simultaneously bombard Dyn with requests, knocking it (and a good part of the Internet!) offline.

# Web Application Vulnerabilities

Web applications are complex environments that often rely not only on web servers but also on backend databases, authentication servers, and other components to provide services to end users. These web applications may also contain security holes that allow attackers to gain a foothold on a network, and modern vulnerability scanners are able to probe web applications for these vulnerabilities.

# Injection Attacks

Injection attacks occur when an attacker is able to send commands through a web server to a backend system, bypassing normal security controls and fooling the backend system into believing that the request came from the web server. The most common form of this attack is the SQL injection attack, which exploits web applications to send unauthorized commands to a backend database server.

# Cross-Site Scripting

**Cross-site scripting (XSS) attack, an attacker embeds scripting commands on a website that will later be executed by an unsuspecting visitor accessing the site. The idea is to trick a user visiting a trusted site into executing malicious code placed there by an untrusted third party.**

# Questions

1. Tom is reviewing a vulnerability scan report and finds that one of the servers on his network suffers from an internal IP address disclosure vulnerability. What protocol is likely in use on this network that resulted in this vulnerability?
A. TLS
B. NAT
C. SSH
D. VPN

2. Which one of the CVSS metrics would contain information about the number of times an attacker must successfully authenticate to execute an attack?
A. AV
B. C
C. Au
D. AC

3. Which one of the following values for the CVSS access complexity metric would indicate that the specified attack is simplest to exploit?
A. High
B. Medium
C. Low
D. Severe

4. Which one of the following values for the confidentiality, integrity, or availability CVSS metric would indicate the potential for total compromise of a system?
A. N
B. A
C. P
D. C

5. What is the most recent version of CVSS that is currently available?
A. 1.0
B. 2.0
C. 2.5
D. 3.0

6. Which one of the following metrics is not included in the calculation of the CVSS exploitability score?
A. Access vector
B. Vulnerability age
C. Access complexity
D. Authentication

7. Kevin recently identified new security vulnerability and computed its CVSSv2 base score as 6.5. Which risk category would this vulnerability fall into?
A. Low
B. Medium
C. High
D. Critical

8. Tara recently analyzed the results of a vulnerability scan report and found that a vulnerability reported by the scanner did not exist because the system was actually patched as specified. What type of error occurred?
A. False positive
B. False negative
C. True positive
D. True negative

9. Which one of the following is not a common source of information that may be correlated with vulnerability scan results?
A. Logs
B. Database tables
C. SIEM
D. Configuration management system

10. Which one of the following operating systems should be avoided on production networks?
A. Windows Server 2003
B. Red Hat Enterprise Linux 7
C. CentOS 7
D. Ubuntu 16

11. In what type of attack does the attacker place more information in a memory location than is allocated for that use?
A. SQL injection
B. LDAP injection
C. Cross-site scripting
D. Buffer overflow

12. The Dirty COW attack is an example of what type of vulnerability?
A. Malicious code
B. Privilege escalation
C. Buffer overflow
D. LDAP injection

13. Which one of the following protocols should never be used on a public network?
A. SSH
B. HTTPS
C. SFTP
D. Telnet

14. Betty is selecting a transport encryption protocol for use in a new public website she is creating. Which protocol would be the best choice?
A. SSL 2.0
B. SSL 3.0
C. TLS 1.0
D. TLS 1.1

15. Which one of the following conditions would not result in a certificate warning during a vulnerability scan of a web server?
A. Use of an untrusted CA
B. Inclusion of a public encryption key
C. Expiration of the certificate
D. Mismatch in certificate name

16. What software component is responsible for enforcing the separation of guest systems in a virtualized infrastructure?
A. Guest operating system
B. Host operating system
C. Memory controller
D. Hypervisor

17. In what type of attack does the attacker seek to gain access to resources assigned to a different virtual machine?
A. VM escape
B. Management interface brute force
C. LDAP injection
D. DNS amplification

18. Which one of the following terms is not typically used to describe the connection of physical devices to a network?
A. IoT
B. IDS
C. ICS
D. SCADA

19. Monica discovers that an attacker posted a message attacking users who visit a web forum that she manages. Which one of the following attack types is most likely to have occurred?
A. SQL injection
B. Malware injection
C. LDAP injection
D. Cross-site scripting

20. Alan is reviewing web server logs after an attack and finds many records that contain semicolons and apostrophes in queries from end users. What type of attack should he suspect?
A. SQL injection
B. LDAP injection
C. Cross-site scripting
D. Buffer overflow