Vulnerabilities

Weak Configurations

Most systems have significant configuration options that administrators can adjust to enable or disable functionality based on usage.

When a system suffers from misconfiguration or weak configuration, it may not achieve all of the desired performance or security objectives.

Misconfiguration can result from omissions, such as when the administrator does not change default credentials.

Open Permissions

Permissions is the term used to describe the range of activities permitted on an object by an actor in a system.

Managing permissions can be tedious, and as the size of the enterprise grows, the scale of permissions requires automation to manage.

When permissions are not properly set, the condition of open permissions exists.

The risk associated with an open permission is context dependent, as for some items, unauthorized access leads to little or no risk, whereas in other systems it can be catastrophic.

Unsecure Root Accounts

Root accounts have access to everything and the ability to do virtually any activity on a network

All root accounts should be monitored, and all accesses should be verified as correct.

Errors

Errors are the condition where something has gone wrong.

Unhandled errors are eventually handled at some level, and the higher up through a system an error goes, the less likely it will be handled correctly.

Trapping and handling errors can reduce the possibility of an error becoming exploitable.

Errors should be trapped by the program and appropriate log files generated.

Examples:

W3C logs are web server logs that focus on recording specific web-related events.

The Windows System log records operating system error messages.

The Windows Application log records events related to local system applications.

Weak Encryption

One typical mistake is choosing to develop your own cryptographic algorithm.

Development of a secure cryptographic algorithm is far from an easy task, and even when it's attempted by experts, weaknesses can be discovered that make the algorithm unusable.

Cryptographic algorithms become trusted only after years of scrutiny and repelling attacks, so any new algorithms would take years to join the trusted set

A similar mistake to attempting to develop your own cryptographic algorithm is attempting to write your own implementation of a known cryptographic algorithm.

Errors in coding implementations are common and lead to weak implementations of secure algorithms that are vulnerable to bypass.

The second major cause of cryptographic weakness, is the employment of deprecated or weak cryptographic algorithms.

Weak cipher suites are those that at one time were considered secure but are no longer considered secure.

Unsecure Protocols

Improperly secured communication protocols and services and unsecure credentials increase the risk of unauthorized access to the enterprise Network infrastructure devices can include routers, switches, access points, gateways, proxies, and firewalls.

It can be as easy as using HTTPS instead of HTTP

Default Settings

Default settings can be a security risk unless they were created with security in mind.

For example:
Older operating systems used to have everything enabled by default.
Old versions of some systems had hidden administrator accounts.

Open Ports and Services

For a service to respond to a request, its port must be open for communication.

Disabling unnecessary services, closing ports, and using firewalls to prevent communications except on approved channels creates a barrier to entry by unauthorized users.

Cloud-based vs On-premises Vulnerabilities

Regardless of whether a system is on premises or cloud based, it will always have potential vulnerabilities

With on-premises vulnerabilities, the enterprise has unfettered access to the infrastructure elements, making the discovery and remediation of vulnerabilities a problem defined by scope and resources.

With the cloud, what is lacking in vulnerability management from the enterprise point of view is visibility into the infrastructure element itself, as this is under the purview of the cloud provider

Zero Day

Most vulnerabilities exist in an unknown state until discovered by a researcher or developer.

Zero day is a term used to define vulnerabilities that are newly discovered and not yet addressed by a patch.

The most frightening thing about a zero-day threat is the unknown factor—its capability and effect on risk are unknown because it is unknown.

Although there are no patches for zero-day vulnerabilities, you can use compensating controls to mitigate the risk.

Exam Tips

Weak configurations greatly increase the likelihood of successful attacks and infiltration. Make every effort to remove unnecessary apps, disable any unnecessary services, change default account usernames and passwords, and close or secure unnecessary ports.

Strong configurations include secure root (Linux) and Administrator (Windows) accounts. Without securing these accounts, anything they are connected to, including processes and services, is exposed to vulnerabilities.

It is important to remember that no matter where data is stored, there will always be potential vulnerabilities that exist.

Zero-day threats have become a common topic in the news and are a likely target for exam questions. Keep in mind that defenses exist, such as compensating controls, which are controls that mitigate the risk indirectly; for example, a compensating control may block the path to the vulnerability rather than directly address the vulnerability.