Title of the project : "Integrated Cybersecurity Analytics ”

Threat Intelligence, Incident Response, SEIM, Nessus, CWE Vulnerabilities, and Web Application Testing

Overview :

This project aims to provide comprehensive analytics on various cybersecurity domains, including Threat Intelligence, Incident Response strategies, Security Information and Event Management (SIEM) principles using Qradar as a focal tool, exploration of Nessus vulnerability scanning, understanding of Common Weakness Enumeration (CWE) vulnerabilities, and practical Web Application Testing techniques. This project offers hands-on experience, allowing participants to delve into real-world scenarios, tools, and methodologies used in securing digital environments and combating cyber threats effectively.

S.No	name	collage	contact
1	Renu Bahuguna	Coer University Roorkee	9917312870

S.no	Vulnerability Name	CWE - No
1	Identification and Authentication Failure	CWE-287: Improper Authentication (4.13)
2	Software and Data integrity failure	CWE-353: Missing Support for Integrity Check
3	Broken Access Control	CWE-284: Improper Access Control
4	Cryptographic failure	CWE-327: Use of a Broken or Risky Cryptographic Algorithm
5	Injection	CWE-94: Improper Control of Generation of Code ('Code Injection')
6	Insecure Design	CWE-657: Violation of Secure Design Principles
7	Security Misconfiguration	CWE-15: External Control of System or Configuration Setting
8	Software login and monitoring failure	CWE-778: Insufficient Logging
9	Server Side Request Forgery	CWE-918: Server-Side Request Forgery (SSRF)
10	Vulnerable And Outdated Components	CWE-1104: Use of Unmaintained Third Party Components

1. Vulnerability Name: Identification and Authentication Failure

   CWE-287: Improper Authentication

   OWASP Category: OWASP Top Ten

   2007

   Description: When an actor claims to have a given identity, the product does not prove or insufficiently proves that the claim is correct.

   Business Impact: CWE-287's Improper Authentication can lead to data breaches, financial losses from fraudulent activities, compliance violations resulting in fines, reputational damage eroding customer trust, and operational disruptions due to security incidents. Businesses risk lawsuits and legal consequences, impacting their finances and reputation. Implementing robust authentication measures and regular security assessments is crucial to mitigate these risks and safeguard against unauthorized access.

2. Vulnerability Name: Software and Data integrity

   failureCWE-353: Missing Support for Integrity

   Check

   OWASP Category: A08:2021

   Description: Product uses a transmission protocol that does not include a mechanism for verifying the integrity of the data during transmission, such as a checksum.

   Business Impact: It lead to compromised data integrity, enabling unauthorized modifications or corruption. Business impacts include loss of trust, potential legal liabilities, operational disruptions, and compromised system reliability, undermining the organization's reputation and leading to financial losses. Resolving this involves implementing integrity checks to safeguard against unauthorized alterations and ensuring data remains intact and trustworthy.

3. Vulnerability Name: Broken Access Control

CWE-284: Improper Access Control OWASP

Category:A01:2021

Description : The product does not restrict or incorrectly restricts access to a resource from an unauthorized actor.

Business Impact: Software and Data Integrity Failure (CWE-353) poses a grave risk to businesses, potentially resulting in corrupted data, unauthorized alterations, compromised system reliability, data breaches, leading to regulatory non-compliance, reputational harm, financial losses, and legal consequences. Implementing robust integrity checks and data validation mechanisms is imperative to mitigate these risks and maintain the trustworthiness and security of business-critical information.

4. Vulnerability Name: Cryptographic failure

   CWE-327: Use of a Broken or Risky Cryptographic Algorithm

   OWASP Category: A02:2021

   Description: product uses a broken or risky cryptographic algorithm or protocol.

   Business Impact: The business impact of Cryptographic Failure (CWE-327) is profound, as it can lead to severe security breaches, compromised data confidentiality, unauthorized access to sensitive information, financial losses due to fraud or theft, regulatory non-compliance, damaged reputation, legal liabilities, and erosion of customer trust. Implementing strong, vetted cryptographic algorithms and regularly updating cryptographic protocols are imperative to mitigate these risks and ensure robust protection of sensitive data and systems.

5. Vulnerability Name: Injection

CWE-94: Improper Control of Generation of Code ('Code Injection')

OWASP Category:A03:2021

Description: Product constructs all or part of a code segment using externally-influenced input from an upstream component, but it does not neutralize or incorrectly neutralizes special elements that could modify the syntax or behavior of the intended code segment.

Business Impact: SQL injection attacks enable malicious insertion of SQL queries via client input, jeopardizing database security. Successful exploits can extract sensitive data, manipulate database content, execute admin tasks (like DBMS shutdown), access DBMS file system files, and, in some instances, issue commands to the underlying operating system. These attacks manipulate data inputs to execute SQL commands, demanding robust security measures like input validation, prepared statements, and regular audits to prevent exploitation and safeguard systems.

6. Vulnerability Name: Insecure Design

   CWE-657: Violation of Secure Design

   PrinciplesOWASP Category: A04:2021

   Description:The product violates well-established principles for secure design.

   Business Impact: The business impact of Insecure Design (CWE-657) is substantial, potentially resulting in compromised systems, increased susceptibility to cyber attacks, data breaches, reputational damage, financial losses due to exploitation, regulatory penalties for non-compliance, and prolonged remediation efforts. Addressing violations of secure design principles demands significant resources, including redesigning systems and applications, to mitigate risks and enhance overall security posture, safeguarding critical assets and customer trust.

7. Vulnerability Name: Security Misconfiguration

CWE-15: External Control of System or Configuration Setting

OWASP Category: A05:2021

Description : One or more system settings or configuration elements can be externally controlled by a user.

Business Impact: Security Misconfigurations (CWE-15) pose significant business risks, potentially leading to unauthorized access, data breaches, service disruptions, compliance violations, and reputational damage. Exploitation of misconfigurations enables attackers to compromise systems, access sensitive information, disrupt operations, and exploit weaknesses in the infrastructure. Mitigating these vulnerabilities demands regular audits, proper configuration management, and adherence to security best practices to fortify defenses, ensuring robust protection of critical assets and maintaining customer trust.

8. Vulnerability Name: software login and monitoring failure

   CWE-778: Insufficient Logging

   OWASP Category:A09:2021

   Description: When a security-critical event occurs, the product either does not record the event or omits important details about the event when logging it.

   Business impact: The business impact of "Software Login and Monitoring Failure" or CWE-778 (Insufficient Logging) is substantial. Insufficient logging and monitoring can lead to delayed detection of security incidents, hindering the ability to identify and respond promptly to threats. This vulnerability can result in prolonged unauthorized access, unnoticed malicious activities, difficulty in forensic investigations, compliance breaches, reputational damage, and increased vulnerability to cyber threats. Mitigating this issue requires robust logging mechanisms, adequate monitoring, and timely analysis of logs to enhance threat detection and response capabilities, minimizing the impact of security incidents on business operations and data integrity.

9. Vulnerability Name: Server Side Request Forgery

CWE-918: Server-Side Request Forgery (SSRF)

OWASP Category A10:2021

Description : The web server receives a URL or similar request from an upstream component and retrieves the contents of this URL, but it does not sufficiently ensure that the request is being sent to the expected destination.

Business Impact: The business impact of Server-Side Request Forgery (SSRF), categorized under CWE-918 and OWASP A10:2021, can be severe. SSRF vulnerabilities enable attackers to manipulate server requests, potentially leading to unauthorized access to internal systems, sensitive data exposure, and exploitation of internal resources. This vulnerability may facilitate attacks like data theft, bypassing security controls, service disruption, and in some cases, accessing sensitive information from within the network. Exploitation of SSRF can result in reputational damage, regulatory penalties, financial losses due to data breaches, and disruption of services. Mitigation involves robust input validation, network segregation, and secure coding practices to prevent SSRF, safeguarding against unauthorized access and data exposure.

10. Vulnerability Name: Vulnerable And Outdated Components

CWE-1104: Use of Unmaintained Third Party Components

OWASP Category: A06:2021

Description: The product relies on third-party components that are not actively supported or maintained by the original developer or a trusted proxy for the original developer.

Business Impact: Relying on outdated or unmaintained third-party components exposes systems to known vulnerabilities, increasing the risk of exploitation by attackers. This vulnerability may lead to severe consequences such as data breaches, loss of sensitive information, regulatory non-compliance penalties, reputational damage, service disruption, and financial losses due to exploitation or system compromise. Mitigating this risk requires proactive monitoring of third-party libraries, regular updates, patch management, and adopting secure development practices to reduce reliance on vulnerable components, ensuring a more resilient and secure application infrastructure.

SANS Top 20 Security Vulnerabilities In Software Applications

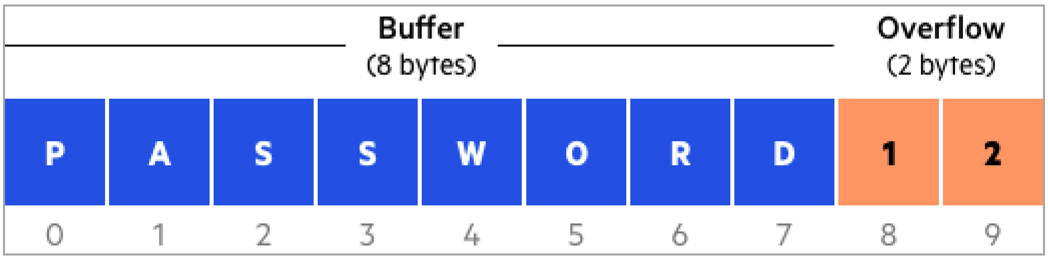
List Of SANS Top 20 Critical Vulnerabilities In Software

S_No	CWE No.	Vulnerabilities Name
1.	**CWE-119**:	Memory Buffer Error
2.	**CWE-79**:	Cross-site Scripting
3.	**CWE-20**	Unvalidated Input Error
4.	**CWE-200**:	Sensitive Information Exposure Error
5.	**CWE-125**:	Out-of-bounds Read Error
6.	**CWE-89**:	SQL Injection
7.	**CWE-416**:	Free Memory Error
8.	**CWE-190**:	Integer Overflow Error
9.	**CWE-352**:	Cross-Site Request Forgery
10.	**CWE-22**:	Traversal
11.	**CWE-78**:	Command Injection
12.	**CWE-787**:	Out-of-bounds Write Error
13.	**CWE-287**:	Improper Authentication Error
14.	**CWE-476**:	Dereferencing NULL Pointer
15.	**CWE-732**:	Incorrect Permission Assignment
16.	**CWE-434**:	Unrestricted File Upload
17.	**CWE-611**:	Information Exposure through XML Entities
18.	**CWE-94**:	Code Injection
19.	**CWE-798**:	Hard-coded Access Key
20.	**CWE-400**:	Uncontrolled Resource Consumption

#1) CWE-119: Memory Buffer Error

This flaw is usually introduced during Architecture and Design, Implementation, Operation stages of the SDLC.This buffer overflow happens when an application process tries to store more data than it can hold in the memory. Since the buffers can only store some level of data and when that level is reached and exceeded, the data flows to another memory location which can corrupt the data already contained in that buffer.

The example below shows a buffer allocated with 8bytes storage. But it overflowed by 2bytes because of more data was sent for execution.



#2) CWE-79: Cross-site Scripting

Cross-site Scripting (XSS) is an injection attack that usually happens when a malicious actor or an attacker injects malicious or harmful script into a web application which can be executed through the web browsers. Once the malicious script finds its way into the compromised system, it can be used to perform different malicious activities.

Cross-site scripting occurrence:

- When un-validated and un-trusted data are inputted into a web application through the web form request.
- When the web application instantly output a web page that contains this malicious data.
- During the process of generating a page, the software fail to validate against the data, which house the content that can be executed by a web browser, like HTML and JavaScript.

- The victim unknowingly visits the page that was generated through a web browser, that house the malicious script that was injected through the use of the untrusted data.
- The malicious script comes from a page that was sent by the attacker's web server, the compromised system web browser then goes ahead to process the malicious script.
- This action violates the web browser's policy about same origin, which stipulates that scripts coming from one domain should not have access to resources or execute code in another different domain except its own domain.

#3) CWE-20: Unvalidated Input Error

The application receives input, but fails to validate the input, whether it has all necessary details needed for it to be accepted into the system for processing.

When there is input sanitization, this can be used to check any potentially dangerous inputs in order to ensure that the inputs are safe to be processed with the source code or when it's an input that is needed to communicate with other components.

The below images show that a good application should not accept script or command as an input. If such inputs are not properly sanitized, the application will process it thinking it's a valid request.

#4) CWE-200: Sensitive Information Exposure Error

This happens when the application knowingly and unknowingly exposes information that is confidential and sensitive to an attacker who does not have the authorization to access these information.

Below are some sensitive information that could be exposed:

- Personal information like personal messages, financial data, health status records, geographic location, or contact details

- System configuration details and environment, **for example,** the operating system and installed packages
- Business Record and intellectual property
- Network configuration details
- Internal application state
- Metadata like the message headers

#5) CWE-125: Out-of-bounds Read Error

This occurs when the application reads data past the normal level, either to the end or before the beginning of the buffer. This gives unprivileged access to an attacker to read sensitive information from other memory locations, which can as well leads to a system or application crash.

If you now check the below example, you will see that the IF statement needs to be modified to include a minimum range validation.

#6) CWE-89: SQL Injection

SQL injection is a form of security vulnerability whereby the attacker injects a Structured Query Language (SQL) code to the Webform input box in order to gain access to resources or change data that is not authorized to access.

This vulnerability can be introduced to the application during the design, implementation, and operation stages.

If the input values are correct, the user is granted access to the application or request, but if the values are incorrect, access will be denied.

#7) CWE-416: Previously Freed Memory

This issue is caused by the referencing of memory after it has been released, which can seriously lead to a program crash. When you use a previously freed memory, this can have adverse consequences, like corrupting of valid data, arbitrary code execution which is dependent on the flaw timing.

Two common causes are:

- Error conditions within the software and in some other exceptional cases.
- No explanation as to which part of the program caused the free memory.

In this instance, the memory is allocated to another pointer immediately after it has been freed. The previous pointer to the freed memory is used again and now points to somewhere around the new allocation. By the time the data is changed, this can corrupt the used memory and could make the application behave in an undefined way.

#8) CWE-190: Integer Overflow Error

When a calculation is processed by an application and there is a logical assumption that the resulting value will be greater than the exact value, integer overflow happens. Here, an integer value increases to a value that cannot be stored in a location.

This issue can trigger buffer overflows, which can be used to execute arbitrary code by an attacker. This integer overflow error is usually introduced into the system during the Design and Implementation stages of the SDLC.

#9) CWE-352: Cross-Site Request Forgery

This is when a web application does not sufficiently verify the HTTP request, whether the request was actually coming from the right user or not. The webservers are designed to accept all requests and to give a response to them.

The below image shows an attacker inducing a user to perform actions that they do not intend to perform.

#10) CWE-22: Directory Traversal

Directory traversal or file path traversal is a web security vulnerability that allows an attacker to read arbitrary files on the server that is currently running an application.

#11) CWE-78: OS Command Injection

It is about the improper sanitization of special elements that may lead to the modification of the intended OS command that is sent to a downstream component. An attacker can execute these malicious commands on a target operating system and can access an environment to which they were not supposed to read or modify.

#12) CWE-787: Out-of-bounds Write Error

This happens when the application writes data past the end, or before the beginning of the designated buffer.When this happens, the end result is usually data corruption, system, or application crash. What the application does is some sort of pointer arithmetic that is used in referencing a memory location outside the buffer boundaries.

#13) CWE-287: Improper Authentication Error

This is when an attacker claims to have a valid identity but the software failed to verify or proves that the claim is correct.

A software validates a user's login information wrongly and as a result, an attacker could gain certain privileges within the application or disclose sensitive information that allows them to access sensitive data and execute arbitrary code.

#14) CWE-476: Dereferencing A NULL Pointer

Dereferencing a null pointer is when the application dereferences a pointer that was supposed to return a valid result instead returns NULL and this leads to a crash. Dereferencing a null pointer can happen through many flaws like race conditions and some programming error.

The processes that are performed with the help of the NULL pointer usually lead to failure, and the possibility of carrying out the process is very slim. This helps attackers to execute malicious code.

#15) CWE-732: Incorrect Permission Assignment

This vulnerability happens when an application assigns permissions to a very important and critical resource in such a manner that exposed the resource to be accessed by a malicious user.

16) CWE-434: Unrestricted File Upload

This vulnerability occurs when the application does not validate the file types before uploading files to the application. This vulnerability is language independent but usually occurs in applications written in ASP and PHP language.

A dangerous type of file is a file that can be automatically processed within the application environment.

#17) CWE-611: Information Exposure Through XML Entities

When an XML document is uploaded into an application for processing and this document contains XML entities with uniform resource identifier that resolves to another document in another location different from the intended location. This anomaly can make the application to attach incorrect documents into its output.

#18) CWE-94: Code Injection

The existence of code syntax in the user's data increases the attacker's possibility to change the planned control behavior and execute arbitrary code. This vulnerability is referred to as "injection weaknesses" and this weakness could make a data control become user-controlled.

#19) CWE-798: Hard-coded Access Key

This is when the password and access key is hard coded into the application directly for inbound authentication purpose and outbound communication to some external components and for encryption of internal data. Hard-coded login details usually cause vulnerability that paves the way for an attacker to bypass the authentication that has been configured by the software administrator.

The system administrator will always find it very hard to detect this vulnerability and fix it.

There are two main streams to this weakness:

- **Inbound**: The application contains an authentication system that validates the input credentials against the hard-coded details.
- **Outbound**: The application connects to another system and details for connecting to the other system are hardcoded into the system.

In the Inbound stream, there is always a default administrator's account that is created, and the credentials to access it will be hard-coded into the application and associated with that default admin account.

#20) CWE-400: Uncontrolled Resource Consumption

This vulnerability happens when the application does not control the allocation properly and maintenance of a limited resource, this allows an attacker to be able to influence the amount of resources consumed, which will eventually lead to the exhaustion of available resources.

Part of the limited resources includes memory, file system storage, database connection pool entries, and CPU.

The three different instances which can lead to resource exhaustion are:

- Shortage of throttling for the number of allocated resources
- Losing out all references to a resource before reaching the shutdown stage
- Failure to close/returning a resource after processing

The issue of resource exhaustion is usually as a result of incorrect implementation of the following scenarios:

- Error conditions and other exceptional circumstances.
- There is mixed reaction over which part of the program releases the resource.

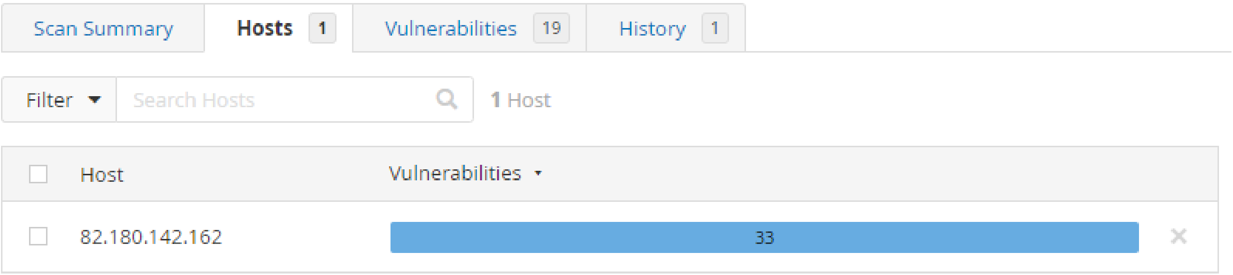
Stage 2 (Nessus)

Overview :

Nessus stands as a powerful vulnerability scanner renowned for its adeptness in detecting security issues across networks, systems, and applications. Leveraging a comprehensive database of known vulnerabilities, it conducts meticulous scans, identifying weaknesses, misconfigurations, and potential entry points for cyber attacks. Its multifaceted approach encompasses various scan types, including credentialed and non-credentialed scans, ensuring a thorough assessment of security postures. Nessus furnishes detailed reports, categorizing vulnerabilities by severity levels and offering actionable insights for remediation. This tool facilitates continuous monitoring by enabling scheduled scans and integrating with other security frameworks. Its user-friendly interface and robust feature set make it adaptable to diverse environments, supporting various platforms and configurations. Nessus aids in compliance adherence by assessing systems against regulatory standards and industry best practices. Incorporating Nessus into cybersecurity strategies enhances proactive threat identification and fortifies defenses against evolving security threats. Its versatility and accuracy empower organizations to proactively manage vulnerabilities, bolstering their overall security posture.

Target website : https://coeruniversity.in/

Target ip address: 82.180.142.162

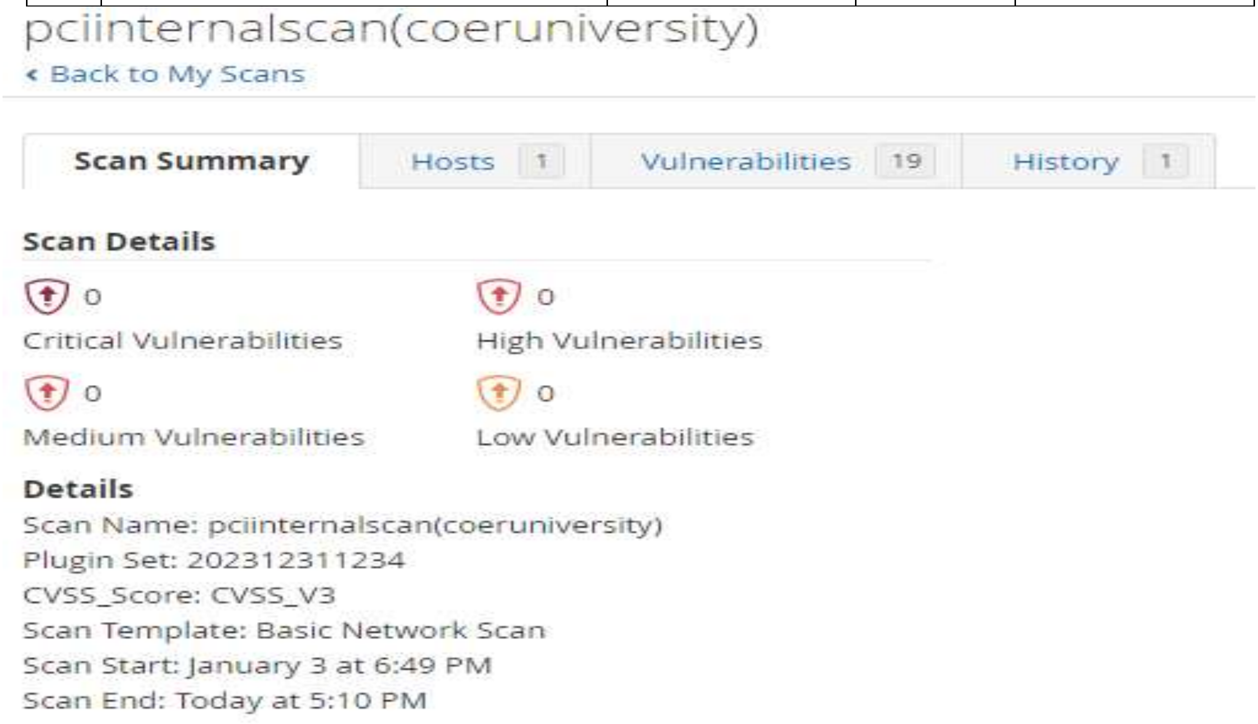


☐	Host	Vulnerabilities
☐	82.180.142.162	33

List of vulnerability

S. No	Vulnerability Name	Family	Severity	Plugins
1	SSL Certificate Information	General	Info	**10863**
2	SSL Cipher Suites Supported	General	Info	**21643**
3	SSL Perfect Forward Secrecy Cipher Suites Supported	General	Info	**57041**
4	SSL / TLS Versions Supported	General	Info	**56984**
5	SSL/TLS Recommended Cipher Suites	General	Info	**156899**
6	TLS Next Protocols Supported	General	Info	**62564**
7	SSL Certificate Signed Using Weak Hashing Algorithm (Known CA)	General	Info	**95631**
8	SSL Root Certification Authority Certificate Information	General	Info	**94761**
9	TLS ALPN Supported Protocol Enumeration	Misc.	Info	**84821**
10	TLS NPN Supported Protocol Enumeration	Misc.	Info	**87242**
11	TLS Version 1.2 Protocol Detection	Service detection	Info	**136318**
12	TLS Version 1.3 Protocol Detection	Service detection	Info	**138330**
13	Nessus SYN scanner	Port scanners	Info	**11219**

14	Service Detection	Service detection	Info	**22964**
15	HTTP Server Type and Version	Web Servers	Info	**10107**
16	Common Platform Enumeration (CPE)	General	Info	**45590**
17	Device Type	General	Info	**54615**
18	FTP Server Detection	Service detection	Info	**10092**
19	FTP Service AUTH TLS Command Support	FTP	Info	**42149**
20	ICMP Timestamp Request Remote Date Disclosure	General	Info	**10114**
21	MySQL Server Detection	Databases	Info	**10719**



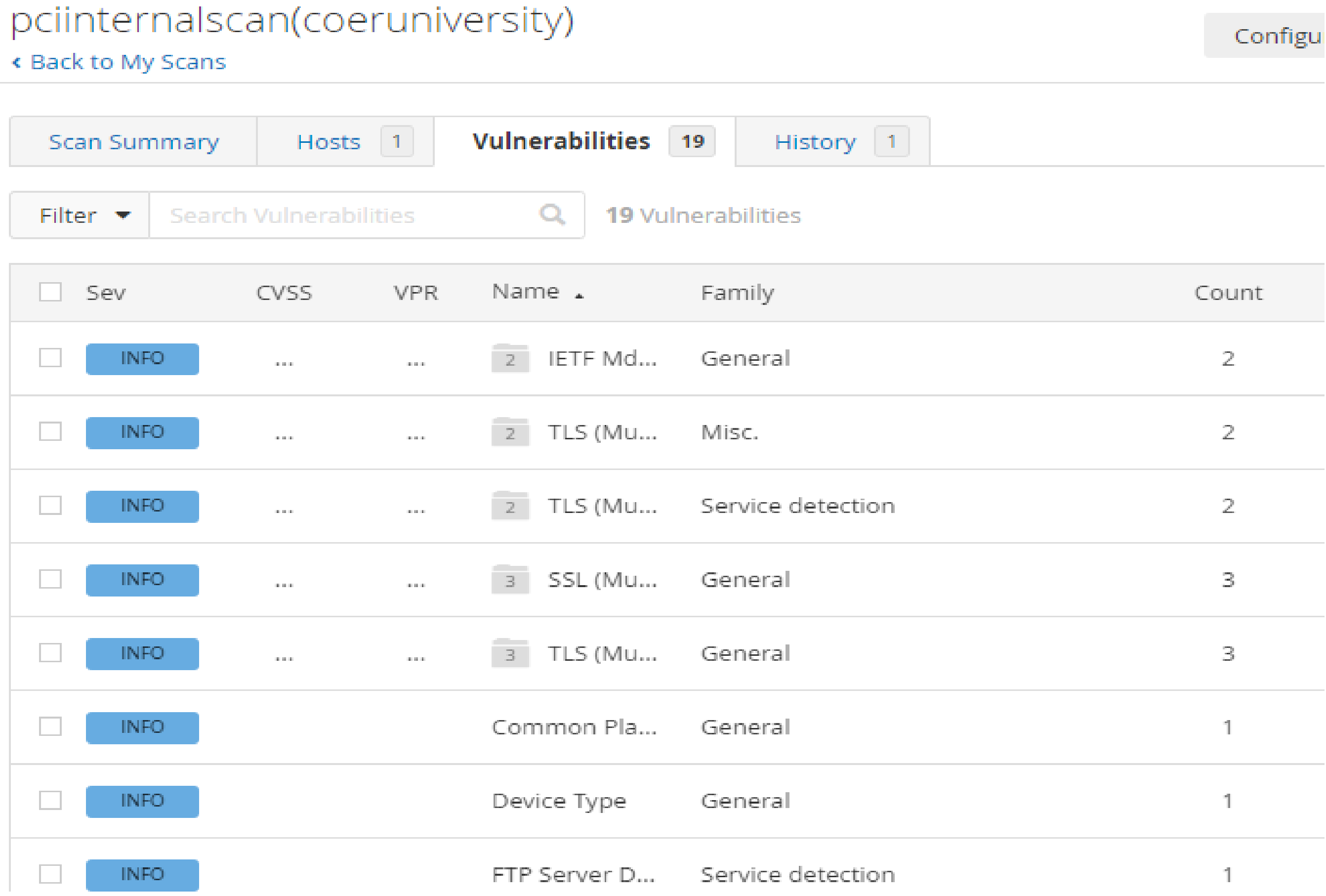
pciinternalscan(coeruniversity)

Scan Details

0	0
Critical Vulnerabilities	High Vulnerabilities
0	0
Medium Vulnerabilities	Low Vulnerabilities

Details

Scan Name: pciinternalscan(coeruniversity)
Plugin Set: 202312311234
CVSS_Score: CVSS_V3
Scan Template: Basic Network Scan
Scan Start: January 3 at 6:49 PM
Scan End: Today at 5:10 PM

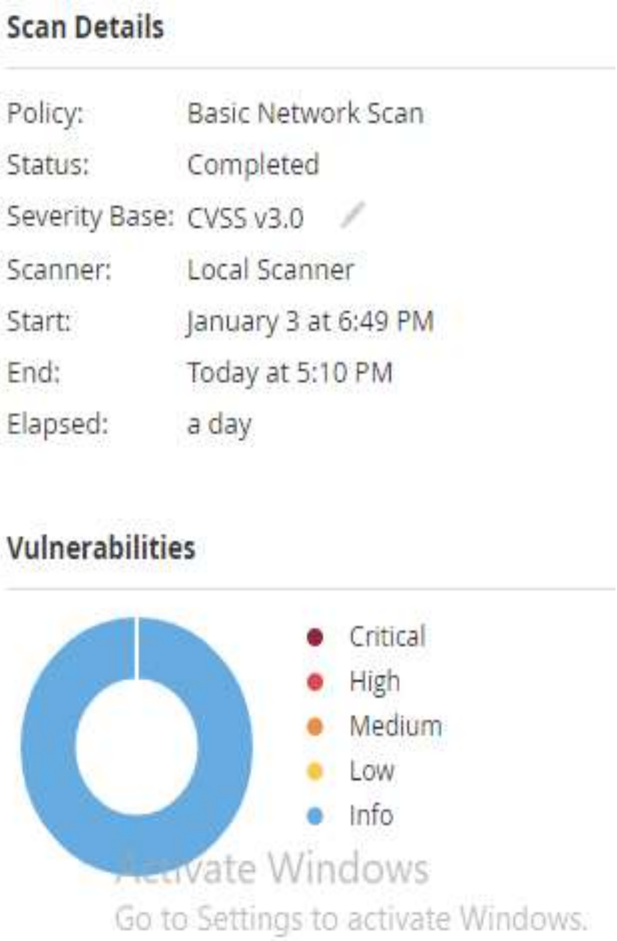


pciinternalscan(coeruniversity)

‹ Back to My Scans

19 Vulnerabilities

□	Sev	CVSS	VPR	Name ▲	Family	Count
□	INFO	...	...	2 IETF Md...	General	2
□	INFO	...	...	2 TLS (Mu...	Misc.	2
□	INFO	...	...	2 TLS (Mu...	Service detection	2
□	INFO	...	...	3 SSL (Mu...	General	3
□	INFO	...	...	3 TLS (Mu...	General	3
□	INFO			Common Pla...	General	1
□	INFO			Device Type	General	1
□	INFO			FTP Server D...	Service detection	1



Scan Details

Policy:	Basic Network Scan
Status:	Completed
Severity Base:	CVSS v3.0
Scanner:	Local Scanner
Start:	January 3 at 6:49 PM
End:	Today at 5:10 PM
Elapsed:	a day

Vulnerabilities

1. **Vulnerability Name : ** SSL Certificate Information

Severity : Info

Plugin:10863

Port :21/tcp/ftp

Description: This plugin connects to every SSL-related port and attempts to extract and dump the X.509 certificate.

Business Impact: Exposes potential weaknesses in SSL certificates, risking data integrity and confidentiality.

Solution: Regularly update and renew SSL certificates, ensuring they meet industry standards. Employ strong encryption algorithms.

2. **Vulnerability Name :** SSL Cipher Suites Supported

   Severity : Info

   Plugin:21643

   Port : 21/tcp/ftp

   Description : This plugin detects which SSL ciphers are supported by the remote service for encrypting communications.

   Business Impact: Weak cipher suites may lead to unauthorized access or data interception.

   Solution: Disable insecure cipher suites, follow best practices for secure configurations, and keep systems updated.

3. **Vulnerability Name : SSL Perfect Forward Secrecy Cipher Suites Supported**

   Severity : Info

   Plugin: 57041

   Port : 21/tcp/ftp

   Description : The remote host supports the use of SSL ciphers that offer Perfect Forward Secrecy (PFS) encryption. These cipher suites ensure that recorded SSL traffic cannot be broken at a future date if the server's private key is compromised.

   Business Impact: Lack of Perfect Forward Secrecy may expose past communication to decryption.

   Solution: Enable and prioritize cipher suites that support Perfect Forward Secrecy to enhance data confidentiality.

4. **Vulnerability Name : SSL / TLS Versions Supported**

   Severity : Info

   Plugin: 56984

   Port : 21/tcp/ftp

Description : This plugin detects which SSL and TLS versions are supported by the remote service for encrypting communications.

Business Impact: Outdated versions may have known vulnerabilities, risking security.

Solution: Use the latest TLS versions and regularly update systems to protect against known exploits.

5. **Vulnerability Name : SSL/TLS Recommended Cipher Suites**

   Severity : Info

   Plugin: 156899

   Port : 21/tcp/ftp

   Description : The remote host has open SSL/TLS ports which advertise discouraged cipher suites.

   Business Impact: Choosing insecure cipher suites can compromise data security.

   Solution: Follow industry best practices for recommended cipher suites, keeping security configurations up to date.

6. **Vulnerability Name : TLS Next Protocols Supported**

   Severity : Info

   Plugin: 62564

   Port : 21/tcp/ftp

   Description :This script detects which protocols are advertised by the remote service to be encapsulated by TLS connections. Nessus did not attempt to negotiate TLS sessions with the protocols shown. The remote service may be falsely advertising these protocols and / or failing to advertise other supported protocols.

   Business Impact: Unsecure protocols may expose systems to attacks or unauthorized access.

Solution: Disable insecure protocols, maintain a list of secure ones, and update configurations.

7. **Vulnerability Name : SSL Certificate Signed Using Weak Hashing Algorithm (Known CA)**

   Severity : Info

   Plugin: 95631

   Port : 21/tcp/ftp

   Description The remote service uses a known CA certificate in the SSL certificate chain that has been signed using a cryptographically weak hashing algorithm (e.g., MD2, MD4, MD5, or SHA1). These signature algorithms are known to be vulnerable to collision attacks (CVE-2004-2761, for example). An attacker can exploit this to generate another certificate with the same digital signature, allowing the attacker to masquerade as the affected service.this plugin reports all SSL certificate chains signed with SHA-1 that expire after January 1, 2017 as vulnerable. This is in accordance with Google's gradual sunsetting of the SHA-1 cryptographic hash algorithm.

   this plugin will only fire on root certificates that are known certificate authorities as listed in Tenable Community Knowledge Article 000001752. That is what differentiates this plugin from plugin 35291, which will fire on any certificate, not just known certificate authority root certificates.

   Known certificate authority root certificates are inherently trusted and so any potential issues with the signature, including it being signed using a weak hashing algorithm, are not considered security issues.

   Business Impact: Weakly signed certificates may be vulnerable to unauthorized issuance or forgery.

   Solution: Regularly update certificates, use strong hashing algorithms, and monitor certificate authorities.

8. **Vulnerability Name : SSL Root Certification Authority Certificate Information**

   Severity : Info

   Plugin: 94761

   Port : 21/tcp/ftp

   Description: The remote service uses an SSL certificate chain that contains a self-signed root Certification Authority certificate at the top of the chain.

   Business Impact: Compromised root certificates may lead to trust issues in secure communications.

   Solution: Securely manage and update root certificates, regularly audit and verify certificate authorities.

9. **Vulnerability Name :TLS ALPN Supported Protocol Enumeration**

   Severity : Info

   Plugin: 84821

   Port : 21/tcp/ftp

   Description:The remote host supports the TLS ALPN extension. This plugin enumerates the protocols the extension supports.

   Business Impact: Enumeration can reveal potentially exploitable information about supported protocols.

   Solution: Limit information disclosure, disable unnecessary protocols, and keep systems patched.

10. **Vulnerability Name :TLS NPN Supported Protocol Enumeration**

    Severity : Info

    Plugin: 87242

    Port : 21/tcp/ftp

Description The remote host supports the TLS NPN (Transport Layer Security Next Protocol Negotiation) extension. This plugin enumerates the protocols the extension supports.

Business Impact: Revealing supported protocols may aid attackers in crafting targeted exploits.

Solution: Minimize protocol information exposure, disable unnecessary protocols, and apply security patches promptly.

11. Vulnerability Name :TLS Version 1.2 Protocol Detection

Severity : Info

Plugin: 136318

Port : 21/tcp/ftp

Description :The remote service accepts connections encrypted using TLS 1.2.

Business Impact: Detection of older TLS versions may indicate vulnerability to known exploits.

Solution: Upgrade to the latest TLS versions, disable obsolete protocols, and apply security updates.

12. Vulnerability Name :TLS Version 1.3 Protocol Detection

Severity : Info

Plugin: 138330

Port : 21/tcp/ftp

Description :The remote service accepts connections encrypted using TLS 1.3.

Business Impact: Detection of outdated TLS versions may expose systems to potential vulnerabilities.

Solution: Ensure systems are updated to the latest TLS version and disable older protocols.

13.Vulnerability Name :Nessus SYN scanner

Severity : Info

Plugin: 11219

Port : 21/tcp/ftp , 80/tcp/www , 443/tcp/www , 3306/tcp/mysql

Description :This plugin is a SYN 'half-open' port scanner. It shall be reasonably quick even against a firewalled target.SYN scans are less intrusive than TCP (full connect) scans against broken services, but they might cause problems for less robust firewalls and also leave unclosed connections on the remote target, if the network is loaded.

Business Impact: SYN scanning may be an attempt to identify vulnerabilities in network services.

Solution: Implement proper network security measures, such as firewalls and intrusion detection systems, to mitigate SYN scan risks.

14.Vulnerability Name :Service Detection

Severity : Info

Plugin:22964

Port : 21 / tcp / ftp, 80 / tcp / www , 443 / tcp / www

Description Nessus was able to identify the remote service by its banner or by looking at the error message it sends when it receives an HTTP request.

Business Impact: Unauthorized service detection may reveal information about the network, aiding potential attackers.

Solution: Employ network segmentation, firewall rules, and disable unnecessary services to limit exposure.

15.Vulnerability Name :HTTP Server Type and Version

Severity : Info

Plugin: 10107

Port : 443 / tcp / www , 80 / tcp / www

Description :This plugin attempts to determine the type and the version of the remote web server.

Business Impact: Disclosing server details may expose vulnerabilities specific to the server type and version.

Solution: Limit server information disclosure, use generic server banners, and keep software up to date.

16.Vulnerability Name :Common Platform Enumeration (CPE)

Severity : Info

Plugin: 45590

Port : NA

Description :By using information obtained from a Nessus scan, this plugin reports CPE (Common Platform Enumeration) matches for various hardware and software products found on a host. if an official CPE is not available for the product, this plugin computes the best possible CPE based on the information available from the scan.

Business Impact: Enumeration may reveal details about system platforms, aiding targeted attacks.

Solution: Minimize information disclosure, regularly update systems, and employ intrusion detection systems.

17.Vulnerability Name :Device Type

Severity : Info

Plugin: 54615

Port :NA

Description:Based on the remote operating system, it is possible to determine what the remote system type is (eg: a printer, router, general-purpose computer, etc).

Business Impact: Identifying device types may assist attackers in crafting targeted exploits.

Solution: Use generic naming conventions, restrict access to device details, and regularly update device firmware.

18.Vulnerability Name :FTP Server Detection

Severity : Info

Plugin: 10092

Port : 21/tcp/ftp

Description:It is possible to obtain the banner of the remote FTP server by connecting to a remote port.

Business Impact: Revealing FTP server details may expose vulnerabilities or aid in unauthorized access attempts.

Solution: Disable unnecessary FTP services, implement secure configurations, and keep software updated.

19.Vulnerability Name :FTP Service AUTH TLS Command Support

Severity : Info

Plugin: 42149

Port : 21/tcp/ftp

Description: The remote FTP service supports the use of the 'AUTH TLS' command to switch from a cleartext to an encrypted communications channel.

Business Impact: Lack of secure FTP authentication support may expose credentials during transmission.

Solution: Enable FTP over TLS (FTPS), use secure authentication methods, and encrypt FTP traffic.

20. Vulnerability Name :ICMP Timestamp Request Remote Date Disclosure

Severity : Info

Plugin: 10114

Port : 0 / icmp

Description: The remote host answers to an ICMP timestamp request. This allows an attacker to know the date that is set on the targeted machine, which may assist an unauthenticated, remote attacker in defeating time-based authentication protocols. Timestamps returned from machines running Windows Vista / 7 / 2008 / 2008 R2 are deliberately incorrect, but usually within 1000 seconds of the actual system time.

Solution : Filter out the ICMP timestamp requests (13), and the outgoing ICMP timestamp replies (14).

Business Impact: Disclosure of system time may assist attackers in synchronization and planning attacks.

Solution: Disable ICMP timestamp responses, employ network security measures, and keep systems synchronized.

21. Vulnerability Name :MySQL Server Detection

Severity : Info

Plugin: 10719

Port : 3306 / tcp / mysql

Description: The remote host is running MySQL, an open source database server.

Business Impact: Revealing MySQL server details may expose vulnerabilities or aid in unauthorized access attempts.

Solution: Limit MySQL information disclosure, apply security best practices, and regularly update MySQL server software.

Stage 3

Tittle : "Evaluation and Optimization of IBM's SOC and SIEM Tools for Enhanced Security Operations"

This project aims to assess the efficacy, capabilities, and operational efficiency of IBM's Security Operations Center (SOC) and Security Information and Event Management (SIEM) tools in mitigating cyber threats and fortifying organizational security. The study will delve into an in-depth analysis of IBM's SOC and SIEM tools, exploring their functionalities, features, and integration capabilities within diverse network environments. Moreover, the project seeks to optimize the utilization of these tools to enhance proactive threat detection, incident response, and overall security posture.

Objectives:

1. Conduct a comprehensive review of IBM's SOC and SIEM tools, including their functionalities, deployment options, and compatibility with various IT infrastructures.
2. Evaluate the effectiveness of these tools in real-time threat detection, log management, correlation of security events, and incident response.
3. Compare IBM's SOC and SIEM tools with industry standards and competitor offerings to identify strengths, weaknesses, opportunities, and potential areas of improvement.
4. Implement a test environment to simulate various cyber threat scenarios and assess the tools' capabilities in detecting, alerting, and mitigating these threats.

5. Develop strategies and recommendations for optimizing the utilization of IBM's SOC and SIEM tools based on the evaluation outcomes.

SOC and SOC Cycle :

SOC:

Security Operations Center (SOC) plays a pivotal role in safeguarding an organization's digital assets and infrastructure from potential cyber threats. The SOC operates as a centralized unit that monitors, detects, analyzes, responds to, and mitigates cybersecurity incidents on an ongoing basis.

The SOC Cycle typically involves the following key stages:

1. Threat Detection and Prevention:

Monitoring: Constant monitoring of networks, systems, applications, and endpoints using various security tools and technologies like SIEM (Security Information and Event Management), IDS/IPS (Intrusion Detection and Prevention Systems), firewalls, etc.

Threat Intelligence: Gathering and analyzing threat intelligence feeds to stay updated about potential cyber threats, vulnerabilities, and attack patterns.

Vulnerability Management: Identifying and assessing vulnerabilities within the organization's IT environment to proactively address weaknesses before they can be exploited.

2. Incident Identification and Analysis:

Alert Triage: Analyzing alerts generated by security tools to determine the severity and validity of potential security incidents.

Incident Investigation: Conducting in-depth analysis and investigation of confirmed security incidents to understand the nature, scope, and impact of the threat.

Forensic Analysis: Gathering and preserving evidence for further analysis, attribution, and potential legal proceedings.

3. Incident Response and Mitigation:

Containment: Taking immediate actions to contain the incident and prevent further spread or damage.

Eradication: Removing the root cause of the incident from the affected systems and networks.

Recovery: Restoring affected systems and data to their normal operational state.

Lessons Learned: Conducting post-incident reviews to identify weaknesses in the security posture and improve response procedures for future incidents.

4. Continuous Improvement:

Metrics and Reporting: Tracking key performance indicators (KPIs) to measure the effectiveness and efficiency of the SOC operations.

Feedback Loop: Incorporating lessons learned from incidents into security policies, procedures, and training programs.

Technology and Process Enhancement: Upgrading security tools, refining processes, and adapting strategies to stay ahead of evolving threats.

The SOC operates in a continuous cycle, as cybersecurity is an ongoing process that requires constant vigilance, adaptation, and improvement. The SOC's effectiveness relies not only on technology but also on skilled personnel, robust processes, and collaboration across various departments within an organization to ensure a proactive and resilient security posture.

SIEM and SIEM Cycle :

SIEM :

SIEM stands for Security Information and Event Management. It refers to a comprehensive approach to security management that combines SIM (Security Information Management) and SEM (Security Event Management) functionalities. SIEM systems provide a centralized platform for collecting, analyzing, and managing security-related data from various sources within an organization's IT infrastructure.

The SIEM Cycle typically involves the following key stages:

1. Data Collection:

Log Collection: Gathering logs and security-related data from diverse sources such as network devices, servers, applications, endpoints, security tools, and more.

Normalization: Standardizing and normalizing collected data into a common format for consistent analysis and correlation.

Aggregation: Aggregating and storing the normalized data in a centralized repository for further analysis and correlation.

2. Event Correlation and Analysis:

Real-Time Monitoring: Analyzing incoming events and logs in real-time to identify potential security incidents or suspicious activities.

Correlation: Correlating disparate security events and logs to detect patterns or relationships that may indicate a security threat.

Alert Generation: Generating alerts or notifications for security analysts or SOC teams based on predefined rules or anomalies detected during correlation.

3. Threat Detection and Incident Response:

Incident Prioritization: Prioritizing alerts based on severity, impact, and relevance to the organization's security posture.

Investigation and Analysis: Investigating identified incidents to understand the scope, impact, and root cause of security threats.

Response Orchestration: Initiating and coordinating appropriate response actions to contain, mitigate, or eradicate the security incident.

4. Forensic Analysis and Reporting:

Forensic Investigation: Conducting detailed forensic analysis to gather evidence, identify the attack vector, and aid in remediation efforts.

Reporting and Compliance: Generating reports for compliance purposes, incident documentation, and analysis of security trends or patterns.

5. Continuous Improvement:

Tuning and Optimization: Fine-tuning SIEM rules, correlations, and configurations based on the analysis of incidents and false positives.

Threat Intelligence Integration: Integrating external threat intelligence feeds to enhance the SIEM's capability to detect emerging threats.

Training and Knowledge Sharing: Providing ongoing training to analysts and SOC teams to stay updated with evolving threats and security technologies.

The SIEM Cycle is an iterative process that requires constant refinement and adaptation to effectively detect, respond to, and mitigate security threats. It serves as a critical component of an organization's cybersecurity strategy, enabling proactive threat management and incident response capabilities.

MISP :

MISP stands for Malware Information Sharing Platform & Threat Sharing. It's an open-source threat intelligence platform designed to enable sharing, storing, and correlating Indicators of Compromise (IOCs) and threat information among cybersecurity professionals and organizations. MISP serves as a centralized repository for exchanging actionable threat intelligence to improve cyber defense strategies and incident response capabilities.

Key features and functionalities of MISP include:

IOC Management: MISP allows users to collect, store, and manage various types of IOCs, such as IP addresses, domain names, hashes, email addresses, malware samples, and more.

Data Sharing and Collaboration: It facilitates the sharing of threat intelligence and IOCs among trusted communities, organizations, or teams, fostering collaboration and enabling quick dissemination of actionable information.

Automated Data Feeds: MISP supports automated data feeds from various sources, including open-source feeds, commercial feeds, and user-generated feeds, ensuring an up-to-date and comprehensive view of emerging threats.

Correlation and Analysis: The platform enables the correlation of different types of IOCs to identify potential relationships or patterns between indicators, aiding in the detection of complex and coordinated attacks.

Taxonomies and Tagging: MISP uses standardized taxonomies and tagging systems to classify and categorize threat information, enhancing consistency and interoperability between different organizations and sectors.

Integration Capabilities: It offers integration with other security tools and platforms, such as SIEMs, threat intelligence platforms, and analysis tools, allowing for seamless information sharing and automation of response actions.

Customization and Flexibility: Users can customize and extend the platform's functionalities to meet specific organizational requirements, adapt to different use cases, and support various data formats.

MISP plays a crucial role in strengthening cybersecurity defenses by facilitating the exchange of timely and relevant threat intelligence, enabling organizations to proactively defend against cyber threats and enhance their incident response capabilities through collective knowledge and shared insights.

Our College (COER UNIVERSITY ROORKEE) Network Information :

Used Network Topology in Our College (Coer University Roorkee)

Network Topology is the schematic description of a network arrangement, connecting various nodes (sender and receiver) through lines of connection.

RING Topology

It is called ring topology because it forms a ring as each computer is connected to another computer, with the last one connected to the first. Exactly two neighbours for each device.

Features of Ring Topology

1. A number of repeaters are used for Ring topology with large number of nodes, because if someone wants to send some data to the last node in the ring topology with 100 nodes, then the data will have to pass through 99 nodes to reach the 100th node. Hence to prevent data loss repeaters are used in the network.
2. The transmission is unidirectional, but it can be made bidirectional by having 2 connections between each Network Node, it is called **Dual Ring Topology**.
3. In Dual Ring Topology, two ring networks are formed, and data flow is in opposite direction in them. Also, if one ring fails, the second ring can act as a backup, to keep the network up.
4. Data is transferred in a sequential manner that is bit by bit. Data transmitted, has to pass through each node of the network, till the destination node.

Advantages of Ring Topology

1. Transmitting network is not affected by high traffic or by adding more nodes, as only the nodes having tokens can transmit data.
2. Cheap to install and expand

STAR Topology

In this type of topology all the computers are connected to a single hub through a cable. This hub is the central node and all others nodes are connected to the central node.

Features of Star Topology

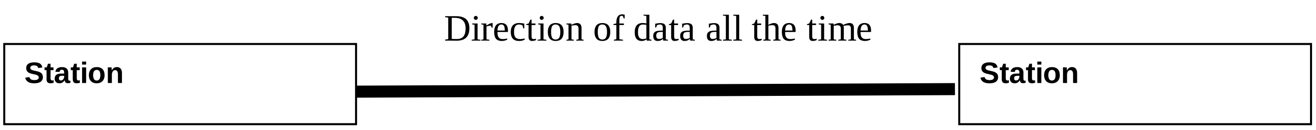
1. Every node has its own dedicated connection to the hub.
2. Hub acts as a repeater for data flow.
3. Can be used with twisted pair, Optical Fibre or coaxial cable.

Advantages of Star Topology

1. Fast performance with few nodes and low network traffic.
2. Hub can be upgraded easily.
3. Easy to troubleshoot.
4. Easy to setup and modify.
5. Only that node is affected which has failed, rest of the nodes can work smoothly.

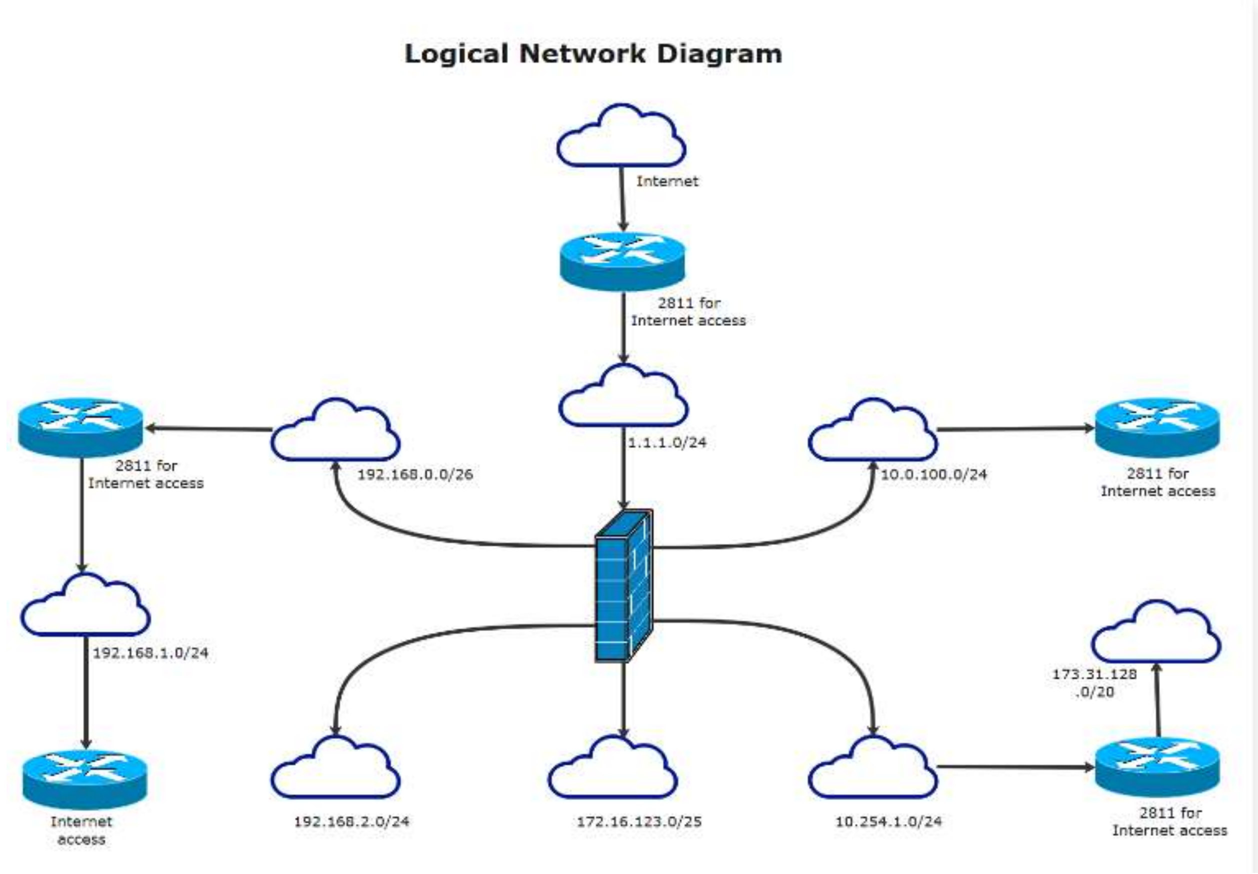
Data Flow:

Communication between two devices half-duplex or ful-duplex we are using ful-duplex communication



NETWORKS

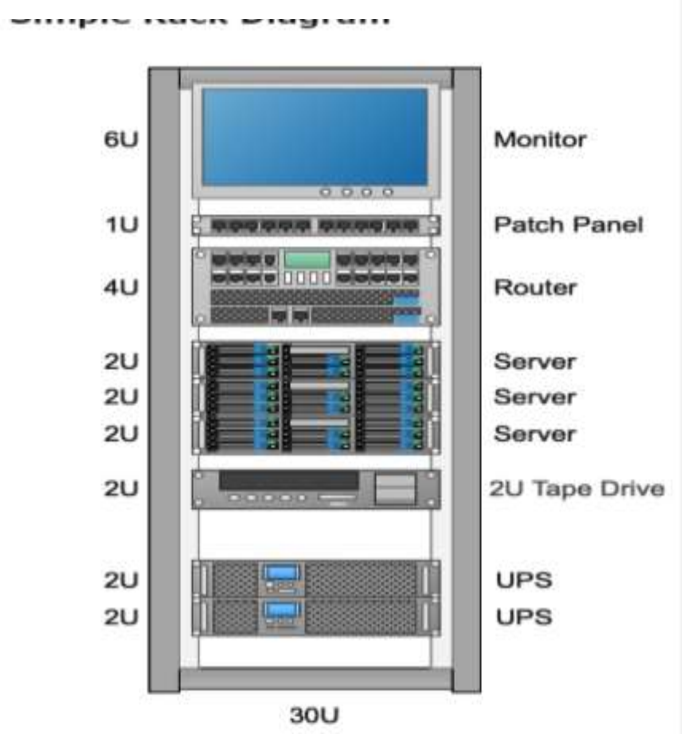
A network is a set of devices (often referred to as nodes) connected by communication links. A node can be a computer, printer, or any other device capable of sending and/or receiving data generated by other nodes on the network.



Network Criteria

A network must be able to meet a certain number of criteria. The most important of these are performance, reliability, and security.

Performance: Performance can be measured in many ways, including transit time and response time. Transit time is the amount of time required for a message to travel from one device to another. Response time is the elapsed time between an inquiry and a response. The performance of a network depends on a number of factors, including the number of users, the type of transmission medium, the capabilities of the connected hardware, and the efficiency of the software. Performance is often evaluated by two networking metrics: throughput and delay. We often need more

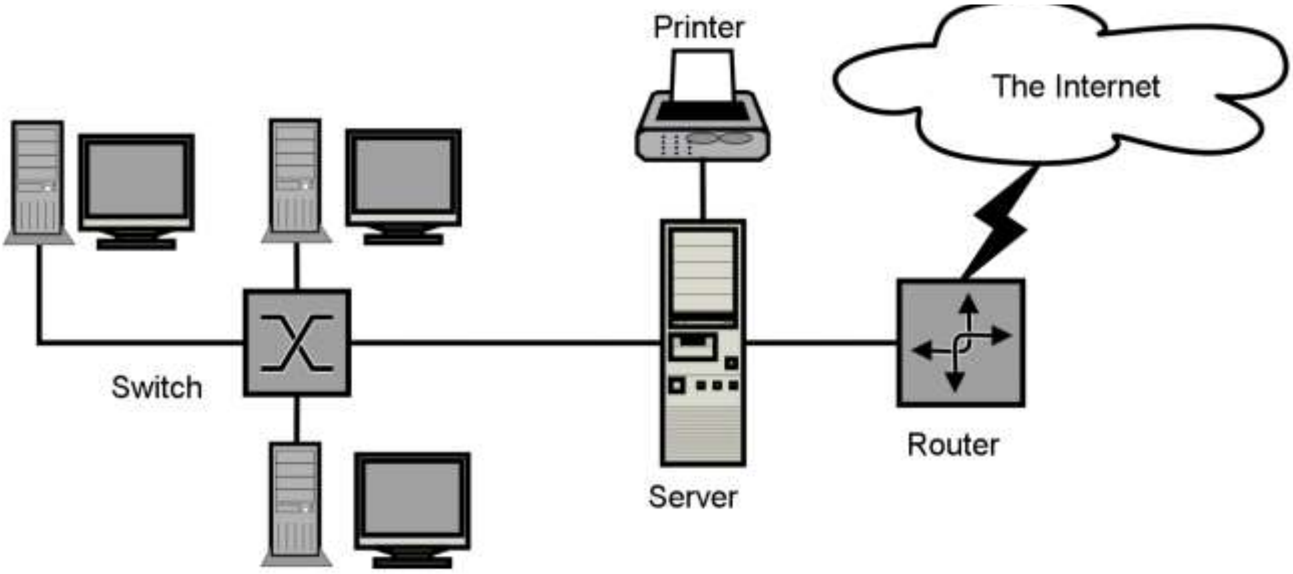


throughput and less delay. However, these two criteria are often contradictory. If we try to send more data to the network, we may increase throughput but we increase the delay because of traffic congestion in the network.

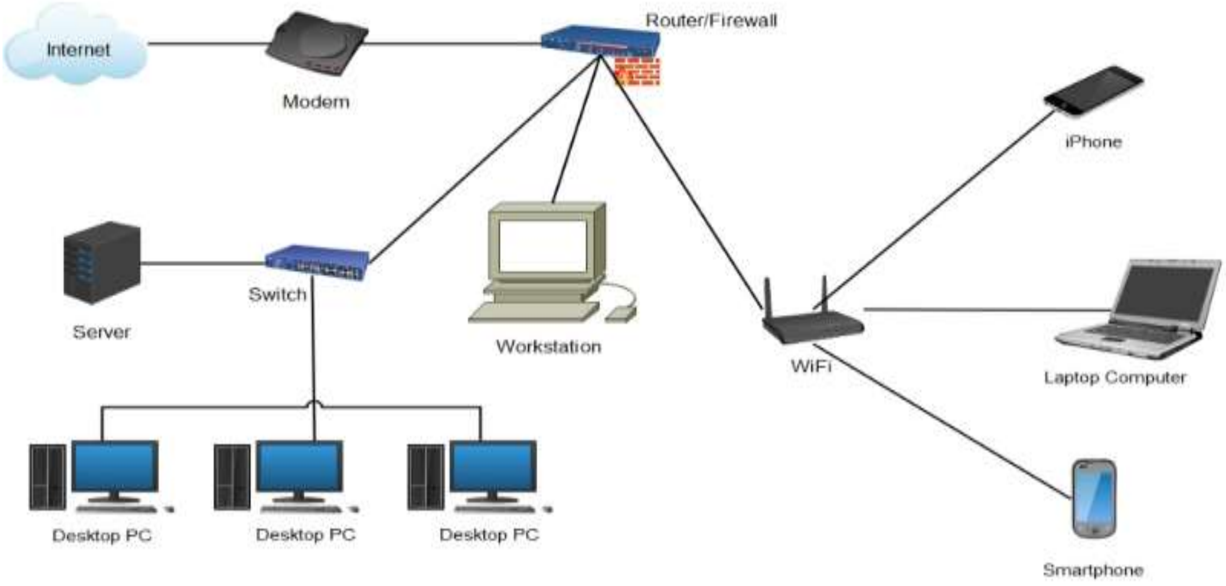
Reliability: In addition to accuracy of delivery, network reliability is measured by the frequency of failure, the time it takes a link to recover from a failure, and the network's robustness in a catastrophe.

Security: Network security issues include protecting data from unauthorized access, protecting data from damage and development, and implementing policies and procedures for recovery from breaches and data losses.

THE INTERNET



The Internet has revolutionized many aspects of our daily lives. It has affected the way we do business as well as the way we spend our leisure time. Count the ways you've used the Internet recently. Perhaps you've sent electronic mail (e-mail) to a business associate, paid a utility bill, read a newspaper from a distant city, or looked up a local movie schedule-all by using the Internet. The Internet is a communication system that has brought a wealth of information to our fingertips and organized it for our use.



1. **Rack Server Dell R710**
2. **Internet Bandwidth** **:** 700 Mbps (1:1) Dedicated Lease line
3. **Internet Connection distributed to the users through Server :** DHCP Server (Automatic & Manually)
4. **Firewall and Bandwidth Management through :** Smart Guard (Third Party Server)
5. **Distribution of LAN** through Giga switch & UTP Cable

Deploying soc in college (Coer University):

Deploying a Security Operations Center (SOC) in a college environment like Coer University involves several critical steps to effectively manage and mitigate cybersecurity risks.

1. Assessment and Planning:

Assess Current Security Posture: Evaluate existing security measures, policies, and infrastructure to identify weaknesses, potential threats, and areas needing improvement.

Define Objectives: Determine the specific security goals and the scope of the

SOC, considering the university's network size, critical assets, and potential risks.

2. Designing the SOC:

Infrastructure Planning: Plan the physical and virtual infrastructure needed for the SOC, including hardware, software, network equipment, and security tools (SIEM, IDS/IPS, firewalls, etc.).

Tool Selection: Choose appropriate security technologies that align with the university's security requirements and budget.

3. Team Establishment:

Staffing: Hire or assign skilled personnel with expertise in cybersecurity, incident response, threat analysis, and SOC operations.

Training: Provide continuous training to the SOC team to keep them updated with the latest threats and technologies.

4. Implementation and Deployment:

Deploy Security Tools: Install, configure, and integrate selected security tools and technologies within the university's network.

Establish Processes: Develop standard operating procedures (SOPs) for incident handling, escalation, and response workflows.

5. Monitoring and Response:

Continuous Monitoring: Monitor network traffic, logs, and security events around the clock using the deployed security tools.

Incident Response: Establish a defined incident response plan to handle security incidents promptly and efficiently.

6. Testing and Optimization:

Simulated Exercises: Conduct simulated security exercises to test the SOC's readiness and effectiveness in responding to various cyber threats.

Refinement: Based on test results and real-world feedback, refine configurations, rules, and procedures to improve SOC performance.

Deploying a SOC in a college environment like Coer University requires a combination of technology, skilled personnel, robust processes, and continuous improvement efforts to effectively protect the network, sensitive data, and ensure a resilient security posture against evolving cyber threats. Adapt these steps according to the specific needs and resources available within the university environment. Consulting with cybersecurity experts or service providers might also be beneficial in this process.

Threat Intelligence :

Threat intelligence refers to information collected, analyzed, and interpreted to understand potential or existing cyber threats, including the tactics, techniques, and procedures (TTPs) employed by threat actors. It serves as a critical component of cybersecurity strategies, providing valuable insights that enable organizations to proactively defend against threats and mitigate risks effectively.

key aspects and components of threat intelligence:

1. Types of Threat Intelligence:

Strategic Intelligence: Focuses on broader trends, motivations, and intentions of threat actors, helping organizations anticipate future threats.

Tactical Intelligence: Provides specific details about threats, such as indicators of compromise (IOCs), attack patterns, vulnerabilities, and methods used by adversaries.

Operational Intelligence: Offers actionable information for immediate use, aiding in real-time threat detection, incident response, and mitigation strategies.

2. Sources of Threat Intelligence:

Open-Source Intelligence (OSINT): Publicly available information from forums, social media, websites, and news sources that can provide valuable insights into threat actor behavior.

Closed Sources: Information obtained from private or subscription-based services, industry-specific groups, sharing communities, and security vendors.

Technical Feeds: Data derived from security tools, malware analysis, intrusion detection systems, and SIEM platforms within an organization's network.

3. Collection and Analysis:

Aggregation: Gathering diverse data sets from multiple sources, including logs, reports, feeds, and threat feeds.

Normalization: Standardizing collected data into a common format for consistency and ease of analysis.

Correlation: Identifying relationships and patterns within the collected data to uncover potential threats and understand the context of attacks.

4. Utilization of Threat Intelligence:

Threat Detection: Using intelligence to identify potential threats by matching indicators against network activity, logs, and security events.

Incident Response: Assisting in rapid response and mitigation efforts during security incidents by providing actionable information to contain and eradicate threats.

Risk Management: Informing risk assessments and aiding in the prioritization of security measures and resource allocation based on identified threats.

5. Challenges and Considerations:

Timeliness and Relevance: Ensuring that threat intelligence remains current, accurate, and relevant to the organization's specific threat landscape.

Data Overload: Managing and prioritizing large volumes of threat data to focus on the most critical threats to the organization.

Sharing and Collaboration: Encouraging information sharing and collaboration among industry peers and trusted communities to enhance collective defense against threats.

6. Continuous Improvement:

Feedback Loop: Incorporating lessons learned from incidents and threat intelligence usage to improve processes, tools, and response capabilities.

Adaptation: Evolving threat intelligence strategies to address emerging threats and evolving tactics used by threat actors.

INCIDENT RESPONSE :

Incident response in cybersecurity refers to the structured approach and actions taken by organizations to manage and mitigate the impact of security incidents. These incidents may include cyber attacks, data breaches, system compromises, malware infections, unauthorized access, or any other security breaches that could potentially harm an organization's IT infrastructure, data, or operations.

Best practices involved in incident response:

1. Preparation:

Develop an Incident Response Plan (IRP): Create a detailed plan that outlines the steps to be taken in the event of a security incident. Define roles, responsibilities, communication protocols, and escalation procedures.

Establish Incident Response Team: Formulate a team of skilled individuals responsible for handling incidents. This team should consist of IT professionals, security experts, legal advisors, public relations, and relevant stakeholders.

Training and Drills: Provide regular training and conduct simulated drills to ensure the incident response team is well-prepared and familiar with their roles and the IRP.

2. Detection and Analysis:

Incident Identification: Use security tools, monitoring systems, and anomaly detection to identify potential security incidents or breaches.

Incident Triage: Assess and prioritize incidents based on severity, impact, and potential risks to the organization's assets and operations.

Forensic Investigation: Conduct in-depth analysis and forensic examination of affected systems and networks to determine the nature and scope of the incident.

3. Containment, Eradication, and Recovery:

Containment: Take immediate actions to limit the scope and spread of the incident. Isolate affected systems or networks to prevent further damage.

Eradication: Remove the root cause of the incident from the affected systems or networks. Patch vulnerabilities, remove malware, or implement necessary fixes.

Recovery: Restore affected systems and data to their normal operational state. Implement backups if necessary.

4. Communication and Reporting:

Internal Communication: Maintain clear and timely communication within the incident response team and relevant stakeholders throughout the incident handling process.

External Communication: Notify appropriate parties, such as law enforcement, regulatory bodies, customers, or partners, as required by regulations or based on the incident severity.

Documentation: Document all actions taken, findings, and lessons learned during the incident response process for future reference and improvement.

5. Post-Incident Analysis and Improvement:

Lessons Learned: Conduct a thorough review and analysis of the incident response process. Identify strengths, weaknesses, and areas for improvement.

Updates to IRP: Update the incident response plan, policies, and procedures based on the findings and lessons learned from the incident.

Continuous Improvement: Implement changes and enhancements to strengthen the organization's incident response capabilities and resilience against future incidents.

Having a well-defined incident response plan, a trained team, and a proactive approach to incident handling is crucial for minimizing the impact of security incidents and maintaining the organization's security posture.

QRADAR :

QRadar is a comprehensive Security Information and Event Management (SIEM) solution designed to provide advanced security intelligence and analytics capabilities for threat detection, incident response, and compliance management. It helps organizations detect, prioritize, and respond to potential cybersecurity threats more effectively by aggregating and correlating data from various sources within the IT environment.

Key features of IBM QRadar include:

1. Log Management and Collection:

QRadar collects and aggregates log data from network devices, servers, applications, endpoints, and security appliances. It normalizes and stores this data for analysis and correlation.

2. Real-Time Event Correlation:

QRadar employs advanced correlation techniques to identify potential security threats by analyzing and correlating security events and logs in real-time. This allows for the detection of anomalous activities and suspicious patterns.

3. Threat Intelligence Integration:

It integrates with external threat intelligence feeds to enrich its capabilities in detecting and responding to known threats, indicators of compromise (IOCs), and emerging attack patterns.

4. Anomaly Detection and Behavioral Analysis:

QRadar utilizes machine learning and behavioral analytics to identify deviations from normal behavior within the network. This helps in detecting insider threats and advanced persistent threats (APTs).

5. Incident Response and Workflow Orchestration:

It provides incident response capabilities by enabling security teams to orchestrate response actions, automate workflows, and initiate response procedures to contain and mitigate security incidents.

6. Visualization and Reporting:

QRadar offers dashboards, visualizations, and customizable reports that provide insights into security events, trends, and compliance posture. This assists in decision-making and regulatory compliance.

7. Integration and Extensibility:

It integrates with various security tools, third-party solutions, and threat intelligence platforms, allowing for a holistic security ecosystem. QRadar also supports customization and extensions through APIs for tailored use cases.

8. Threat Hunting and Investigation:

QRadar facilitates threat hunting activities by enabling security analysts to perform in-depth investigations, conduct searches, and analyze security data to uncover hidden threats or suspicious activities.

IBM QRadar is a powerful SIEM solution that helps organizations improve their cybersecurity posture by providing real-time visibility, advanced threat detection, and effective incident response capabilities across their IT infrastructure. Its comprehensive features make it a valuable tool in managing and mitigating cybersecurity risks.

Conclusion :

Stage 1 :

Stage 1 involves understanding Common Weakness Enumeration (CWE), OWASP Top Ten vulnerabilities, and the SANS Top 20. CWE catalogs software weaknesses, aiding in vulnerability identification, while OWASP Top Ten highlights critical web application security risks. SANS Top 20 focuses on prevalent cybersecurity threats and mitigation strategies. This learning emphasizes the importance of secure coding practices, threat modeling, and vulnerability management. Recognizing these vulnerabilities enables preemptive measures, like input validation, authentication mechanisms, and patch management, enhancing overall software and system security. Understanding these frameworks aids in proactively addressing vulnerabilities, mitigating risks, and fortifying against potential cyber threats.

Stage 2 :

After hands-on experience with Nessus in Stage 2, individuals gain practical insights into conducting web application testing. This experience proves helpful in understanding vulnerabilities specific to web applications, such as SQL injection, cross-site scripting (XSS), and security misconfigurations. It facilitates the identification of weaknesses within web app architectures and helps in comprehending how vulnerabilities manifest are detected by scanning tools like Nessus. Practical application with Nessus enhances skills in mitigating web-related security risks and aids in developing proactive measures to secure web applications effectively.

Stage 3 :

The Security Operations Center (SOC) is the operational hub responsible for managing and responding to cybersecurity threats. The SIEM, such as IBM QRadar, serves as a crucial tool within the SOC, providing comprehensive capabilities for collecting, analyzing, and responding to security events and incidents. The QRadar Dashboard, as a part of the SIEM, offers a user-friendly and customizable interface for security analysts and stakeholders to gain valuable insights into the organization's security landscape, aiding in effective decision-making and incident response.

These components work together to empower organizations in proactively identifying, mitigating, and responding to cyber threats, enhancing their security posture, and ensuring the resilience of their IT infrastructure against evolving security challenges.

Future Scope :

Stage 1 :

The future scope involves integrating these resources into AI-driven vulnerability detection systems, fostering rapid and accurate identification of emerging threats. Aligning with evolving compliance standards, these frameworks will continue to guide secure software development practices, fortifying applications against known vulnerabilities. Collaboration and community-driven knowledge sharing will further enrich these resources, enhancing collective defense against evolving cyber threats.

Stage 2 :

Hands-on experience with Nessus in Stage 2 paves the way for an expanded cybersecurity landscape. The future scope involves leveraging advanced features of Nessus for more intricate vulnerability assessments and compliance checks. Integrating Nessus with AI-driven automation will streamline vulnerability detection processes, enhancing efficiency and accuracy. Utilizing Nessus in cloud environments and IoT ecosystems will broaden its applicability, securing diverse technological landscapes.

Collaborative innovation and community-driven contributions will further enrich Nessus capabilities, addressing evolving cyber threats and bolstering organizations' defenses.

Stage 3 :

A Security Operations Center (SOC) serves as the nerve center for an organization's cybersecurity efforts, housing skilled personnel, processes, and technologies aimed at detecting, analyzing, and responding to potential security threats. Security Information and Event Management (SIEM) solutions, such as IBM's QRadar Dashboard, play a vital role within a SOC by aggregating, correlating, and analyzing vast amounts of security data from various sources within an organization's network. The QRadar Dashboard, as part of the SIEM platform, offers a user-friendly interface that presents actionable insights, visualizations, and reports, aiding security analysts in monitoring security events, identifying anomalies, and making informed decisions to fortify the organization's security posture and swiftly respond to potential cyber threats.

TOPICS EXPLORED :

- Active and Passive Attacks
- Phases of Hacking
- Top Ten Hackers
- Networking
- Different Ports
- Subnetting
- Linux Architecture

- Characteristics and Types of Linux OS
- Linux File System and Directories
- Top 20 Sans
- CWE Vulnerabilities
- Web Application Testing
- Nessus Essential
- Burp Suit
- SOC
- SIEM
- IBM QRadar

TOOLS AND HANDS-ON EXPLORED:

- Nessus Essential
- VMware
- Kali Linux Installation
- Metasploitable
- SQL Injection
- SOC Tool
- SIEM Tool
- QRadar