TEAM -9

Cyber Phishers

Part I-Executive summary

Overview

Developing a comprehensive cybersecurity plan for an organization necessitates a methodical approach to protect against different types of attacks. An outline that can be used to create robust cybersecurity security measures is as follows:

1. **Conduct risk assessments**
Inventory Assets: Keep track of all data, software, hardware, and network sources.
Assess the Risks: Determine potential threats, weak points, and the effects of security lapses.
Set Risks in Order of Priority: Rank risks according to their likelihood and possible impact.

2. **Establish procedures and policies for security**
Create Policy: Establish policies and procedures for incident response, data protection, and permitted use.
Access Control: Enforce minimum required access and put role-based access controls in place.
Establishing protocols for data handling, encryption, and classification is known as data management.

3. **Put Technical Protections in Place**
Network protection can be achieved by utilizing secure network architecture, firewalls, and intrusion detection/prevention systems (IDS/IPS).
Endpoint Protection: Make sure to have up-to-date antivirus software and endpoint detection and response (EDR) technologies installed.
Application Security: Review code, carry out vulnerability analyses, and implement

secure software development techniques.
Sensitive data should be encrypted both during transmission and at rest.

4. **Educate and Train Users**
Security Training: Consistently teach employees on safe online conduct, social engineering, and phishing techniques.
Communicate Policies: Make sure staff members are aware of and obedient to security guidelines and protocols.
Attack Simulation: To gauge user awareness, run security drills and phishing simulations.

5. **Create protocols for responding to incidents**
Response Strategy: Create and record an incident response strategy with responsibilities and processes clearly outlined.
Monitoring: Make use of Security Information and Event Management (SIEM) systems to conduct ongoing monitoring.
Response Team: Put together an incident response team (IRT) that is cross-functional and has distinct duties.

6. **Perform periodic assessments and audits**
Vulnerability Assessments: Conduct penetration testing and vulnerability assessments on a regular basis.
Conduct compliance audits to verify adherence to industry and regulatory requirements, such as ISO 27001, HIPAA, and GDPR.
External Assessments: Hire outside security professionals to conduct unbiased assessments.

7. **Planning for Backup and Recovery**
Data backup: Make sure your regular backups of important data are safe and well-tested.
Prepare a catastrophe recovery plan that outlines the steps involved and the goals for the various recovery locations and times.

8. **Boost Safety Measures**
Access Control: To regulate physical access to critical places, use security staff,

biometric technology, and badges.
Environmental Controls: In data centers, regulate humidity and temperature, and put in place fire suppression.

9. **Utilize Security Tools and Technologies**
Use identity and access management (IAM) systems to control user identities and permissions.
Enforce multi-factor authentication (MFA) to gain access to vital systems.
Cloud Security: Use cloud access security brokers (CASBs) for monitoring and implement best practices for cloud security.

10. **Continuous Improvement**
Threat Intelligence: Use threat intelligence feeds to stay current on new threats and weaknesses.
Feedback Loop: To improve security measures, consider the input received from audits, assessments, and incidents.
Regular Updates: To meet evolving issues, evaluate and upgrade security policies, practices, and technologies.

IP address of the website considered: 45.124.184.10

2. Team Members Involved in vulnerability Assessment

S.No	Name	Designation	Mobile Number
1	Dr. Pimal Khanpara	Assistant Professor	9825757647 pimal.khanpara@nirmauni.ac.in
2	Dr. Rebakah Geddam	Assistant Professor	9925010747 rebakah.geddam@nirmauni.ac.in
3	Ms. Esakkiammal S	Information Officer	8469212089 esakkiammal.s@nirmauni.ac.in

4	Ms.Jalpa Patel	Assistant Professor	9726691064 patel.jal005@gmail.com

3. List of Vulnerable Parameter, location discovered

S.No	Name of the Vulnerability	Reference CWE
1	Broken Access Control	CWE 200- Exposure of Sensitive Information to an unauthorized user
2	Cryptographic Failures	CWE-327: Broken or Risky Crypto Algorithm
3	Injection	CWE 89: SQL Injection
4	Insecure Design	CWE 522: Insufficiently Protected Credentials
5	Security Misconfiguration	CWE 16 - Configuration
6	Vulnerable and OutdatedComponents	CWE 1395: Dependency on Vulnerable Third-Party Components
7	Identification and Authentication Failures	CWE 287 – Improper Authentication
8	Software and Data IntegrityFailures	CWE-494 Download of Code Without Integrity Check
9	Security Logging and Monitoring Failures	CWE-778 Insufficient Logging
10	Server Side Request Forgery	CWE-918:Server Side Request Forgery

1. CWE: CWE 200- Exposure of Sensitive Information to an unauthorized user
OWASP CATEGORY : A01 2021 Broken Access Control
DESCRIPTION: Unauthorized access to confidential information or functionalities may result from inadequate enforcement of access controls.
BUSINESS IMPACT:
Data breaches: When private, financial, or proprietary information is accessed without authorization, it can result in data leaks.
Regulatory Fines: Serious fines and legal repercussions may follow noncompliance with data protection laws like the GDPR, CCPA, or HIPAA.
Loss of Customer Trust: When customer trust is undermined by a breach, it may result in a decline in business and a tarnished reputation.
Financial Losses: There may be substantial expenses related to breach mitigation, litigation, and reimbursing impacted customers.
Operational Disruptions: Unauthorized access may result in vital system functions being manipulated or indefinite system outages, which could compromise business continuity.

2. CWE: CWE-327: Broken or Risky Crypto Algorithm
OWASP CATEGORY : A02 2021 Cryptographic Failures
DESCRIPTION: Issues pertaining to cryptography, like poor encryption, inappropriate key management, or noncompliance with recommended protocols.
BUSINESS IMPACT:
Data Theft: If encryption is inadequate or poorly maintained, sensitive information—such as client information or intellectual property—may be revealed.
Penalties for Non-Compliance: Serious fines and legal action may follow noncompliance with data protection regulations.
Damage to Reputation: The loss of confidential information can undermine consumer confidence in a company and harm its standing.
Intellectual Property Loss: A competitive advantage may be weakened by unauthorized access to confidential information.

3. CWE: CWE 89: SQL Injection
OWASP CATEGORY : A03 2021 Injection
DESCRIPTION: When untrusted data is supplied to an interpreter as an element of a

command or query, injection vulnerabilities arise. SQL Injection occurs when an attacker is able to execute arbitrary SQL code on a database by manipulating user inputs. This type of injection flaw can lead to unauthorized access, data theft, and data manipulation.

BUSINESS IMPACT:

Data Breaches: Attackers can gain unauthorized access to sensitive data stored in the database, such as customer information, financial records, and proprietary data. This exposure can lead to significant data breaches.

Data Corruption: SQL Injection can be used to alter, delete, or corrupt data within the database, leading to loss of data integrity. This can disrupt business operations and lead to incorrect business decisions.

Operational Disruption: Successful SQL Injection attacks can disrupt the normal functioning of applications, leading to downtime and a halt in business processes. This can impact productivity and lead to financial losses.

Financial Losses: Costs associated with responding to SQL Injection attacks can be substantial. This includes incident response efforts, legal fees, regulatory fines, and compensation for affected customers.

Regulatory Fines: Non-compliance with data protection regulations such as GDPR, CCPA, or HIPAA due to data breaches caused by SQL Injection can result in hefty fines and legal penalties.

Reputation Damage: SQL Injection attacks can severely damage an organization's reputation. Loss of customer trust and confidence can lead to a decrease in business and long-term reputational harm.

Fraud and Abuse: Attackers can exploit SQL Injection to perform fraudulent activities, such as unauthorized financial transactions or manipulating application logic to gain unfair advantages.

4. CWE: CWE 522: Insufficiently Protected Credentials

OWASP CATEGORY : A04 2021 Insecure Design

DESCRIPTION: Insecure design flaws can lead to vulnerabilities. This category emphasizes the need for secure design patterns.

BUSINESS IMPACT:

System Breaches: Exploitation of design flaws can lead to unauthorized access and data breaches.

Service Disruptions: Vulnerabilities can be exploited to disrupt business operations, leading to downtime and financial losses.
Redesign Costs: Addressing insecure designs often requires significant resources for redesign and redevelopment.
Reputation Damage: Continuous security issues can damage the organization's reputation and customer trust.

5. CWE: CWE 16 - Configuration
OWASP CATEGORY : A05 2021 Security Misconfiguration
DESCRIPTION: Misconfigurations can occur at any level of an application stack, including network services, platforms, and custom code.
BUSINESS IMPACT:
Unauthorized Access: Misconfigurations can be exploited to gain unauthorized access to systems and data.
Service Disruptions: Exploited vulnerabilities can lead to system downtime and operational disruptions.
Remediation Costs: Fixing misconfigurations and mitigating their impact can be costly and resource-intensive.
Regulatory Non-Compliance: Misconfigurations leading to data breaches can result in fines and legal penalties.

6. CWE: CWE 1395: Dependency on Vulnerable Third-PartyComponents
OWASP CATEGORY : A06 2021 Vulnerable and Outdated Components
DESCRIPTION: Use of outdated or vulnerable software components can be exploited by attackers.
BUSINESS IMPACT:
System Breaches: Known vulnerabilities in outdated components can be exploited to gain unauthorized access.
Data Compromises: Exploited vulnerabilities can lead to data breaches and exposure of sensitive information.
Regulatory Fines: Non-compliance with security standards can result in fines and legal actions.
Operational Downtime: Vulnerabilities can be exploited to disrupt business operations, leading to downtime and loss of productivity.

7. CWE: CWE 287 – Improper Authentication

OWASP CATEGORY : A07 2021 Identification and Authentication Failures

DESCRIPTION: Issues in the implementation of authentication mechanisms can lead to unauthorized access.

BUSINESS IMPACT:

Unauthorized Access: Weak authentication mechanisms can be bypassed, leading to data breaches and system compromises.

Fraud: Unauthorized access can result in fraudulent activities, causing financial losses.

Reputation Damage: Breaches due to authentication failures can harm the organization's reputation and erode customer trust.

Financial Losses: Costs related to breach mitigation, legal actions, and compensating affected customers.

8. CWE: CWE-494 Download of Code Without Integrity Check

OWASP CATEGORY : A08 2021 Software and Data Integrity Failures

DESCRIPTION: Integrity failures occur when software updates, critical data, or CI/CD pipelines are compromised.

BUSINESS IMPACT:

Deployment of Malicious Code: Compromised software can introduce malicious code, leading to data breaches and system disruptions.

Data Corruption: Unauthorized changes to data can compromise its integrity, leading to incorrect business decisions and financial losses.

Operational Disruption: Integrity failures can disrupt business operations, leading to downtime and productivity loss.

Legal Consequences: Failing to protect data integrity can result in regulatory fines and legal actions.

9. CWE: CWE-778 Insufficient Logging

OWASP CATEGORY: A09 2021 Security Logging and Monitoring Failures

DESCRIPTION: Insufficient logging and monitoring can delay the detection of security breaches.

BUSINESS IMPACT:
Delayed Incident Detection: Lack of adequate monitoring can lead to prolonged undetected breaches, increasing damage.
Extended Downtime: Delayed response to incidents can prolong system downtime and operational disruptions.
Higher Remediation Costs: The longer a breach goes undetected, the more costly it becomes to mitigate.

10. CWE: CWE-918 Server Side Request Forgery
OWASP CATEGORY : A10 2021 - Server Side Request Forgery
DESCRIPTION: SSRF vulnerabilities occur when a web application is tricked into making requests to unintended locations.
BUSINESS IMPACT:
Internal System Access: Attackers can leverage SSRF to access internal systems, leading to data breaches.
Infrastructure Compromise: SSRF can be used to bypass firewalls and other security controls, compromising critical infrastructure.
Operational Disruptions: Exploited SSRF vulnerabilities can lead to service disruptions and loss of business continuity.
Financial Losses: Costs related to breach mitigation, system repairs, and potential legal actions.

STAGE 2

Overview :-
An unsupported version of Apache Tomcat is installed on the remote host.

Target website – https://inflibnet.ac.in/
Target ip address:- 45.124.184.10

List of vulnerability –

s.no	Vulnerability name	Severity	plugins
1.	111066 (1) - Apache	High	Published: 2018/07/24, Modified:

	Tomcat 7.0.0 < 7.0.89		2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
2.	121120 (1) - Apache Tomcat 7.0.0 < 7.0.76	Medium	Published: 2019/01/11, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
3.	171351 (1) - Apache Tomcat SEoL (7.0.x	Critical	Published: 2023/02/10, Modified: 2024/05/06 Plugin Output 45.124.184.32 (tcp/8080/www)
4.	197818 (1) - Apache Tomcat 7.0.0 < 7.0.72 multiple vulnerabilities0	Medium	Published: 2024/05/23, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
5.	197843 (1) - Apache Tomcat 7.0.0 < 7.0.100 multiple vulnerabilities	High	Published: 2024/05/23, Modified: 2024/05/24 Plugin Output 45.124.184.32 (tcp/8080/www)
6.	197848 (1) - Apache Tomcat 7.0.0 < 7.0.73 multiple vulnerabilities	High	Published: 2024/05/23, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
7.	88936 (1) - Apache Tomcat 7.0.0 < 7.0.68 multiple vulnerabilities	Medium	Published: 2016/02/24, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
8.	103329 (1) - Apache Tomcat 7.0.0 < 7.0.81 multiple vulnerabilities	Medium	Published: 2017/09/19, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
9.	103782 (1) - Apache Tomcat 7.0.0 < 7.0.82	Medium	Published: 2017/10/11, Modified: 2024/05/23 Plugin Output 45.124.184.32

			(tcp/8080/www)
10.	121118 (1) - Apache Tomcat 7.0.5 < 7.0.67	Medium	Published: 2019/01/11, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
11.	121119 (1) - Apache Tomcat 7.0.0 < 7.0.70	High	Published: 2019/01/11, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
12.	121121 (1) - Apache Tomcat 7.0.28 < 7.0.88	Medium	Published: 2019/01/11, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
13.	124064 (1) - Apache Tomcat 7.0.0 < 7.0.94 multiple vulnerabilities	High	Published: 2019/04/16, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
14.	136770 (1) - Apache Tomcat 7.0.0 < 7.0.104	Medium	Published: 2020/05/22, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
15.	138851 (1) - Apache Tomcat 7.0.27 < 7.0.105	Medium	Published: 2020/07/23, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
16.	147163 (1) - Apache Tomcat 7.0.0 < 7.0.108 multiple vulnerabilities	Medium	Published: 2021/03/05, Modified: 2024/05/24 Plugin Output 45.124.184.32 (tcp/8080/www)
17.	197820 (1) - Apache Tomcat 7.0.0 < 7.0.77	Medium	Published: 2024/05/23, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
18.	197823 (1) - Apache Tomcat 7.0.0 < 7.0.75	Medium	Published: 2024/05/23, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)

19.	197826 (1) - Apache Tomcat 7.0.25 < 7.0.90	Medium	Published: 2024/05/23, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
20.	197831 (1) - Apache Tomcat 7.0.0 < 7.0.78	Medium	Published: 2024/05/23, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
21.	197838 (1) - Apache Tomcat 7.0.0 < 7.0.99 multiple vulnerabilities	Medium	Published: 2024/05/23, Modified: 2024/05/24 Plugin Output 45.124.184.32 (tcp/8080/www)
22.	12085 (1) - Apache Tomcat Default Files	Medium	Published: 2004/03/02, Modified: 2019/08/12 Plugin Output 45.124.184.32 (tcp/8080/www)
23.	83764 (1) - Apache Tomcat 7.0.0 < 7.0.59	Medium	Published: 2015/05/21, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
24.	102587 (1) - Apache Tomcat 7.0.41 < 7.0.79	Medium	Published: 2017/08/18, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
25.	106975 (1) - Apache Tomcat 7.0.0 < 7.0.85 multiple vulnerabilities	Medium	Published: 2018/02/23, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
26.	118035 (1) - Apache Tomcat 7.0.23 < 7.0.91	Medium	Published: 2018/10/10, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
27.	121117 (1) - Apache Tomcat 7.0.0 < 7.0.65	Medium	Published: 2019/01/11, Modified: 2024/05/23 Plugin Output 45.124.184.32 (tcp/8080/www)
28.	148405 (1) - Apache Tomcat 7.0.0 < 7.0.107	Medium	Published: 2021/04/09, Modified: 2024/05/23 Plugin Output 45.124.184.32

			(tcp/8080/www)
29.	11219 (3) - Nessus SYN scanner	None	Published: 2009/02/04, Modified: 2024/05/20 Plugin Output 45.124.184.32 (tcp/80/www)
30.	11422 (1) - Web Server Unconfigured - Default Install Page Present	None	Published: 2003/03/20, Modified: 2018/08/15 Plugin Output 45.124.184.32 (tcp/8080/www)
31.	20108 (1) - Web Server / Application favicon.ico Vendor Fingerprinting	None	Published: 2005/10/28, Modified: 2020/06/12 Plugin Output 45.124.184.32 (tcp/8080/www)
32.	42981 (1) - SSL Certificate Expiry - Future Expiry	None	Published: 2009/12/02, Modified: 2020/09/04 Plugin Output 45.124.184.32 (tcp/443/www)
33.	46180 (1) - Additional DNS Hostnames	None	Published: 2010/04/29, Modified: 2022/08/15 Plugin Output 45.124.184.32 (tcp/0)
34.	83298 (1) - SSL Certificate Chain Contains Certificates Expiring Soon	None	Published: 2015/05/08, Modified: 2015/05/08 Plugin Output 45.124.184.32 (tcp/443/www)
35.	94761 (1) - SSL Root Certification Authority Certificate Information	None	Published: 2016/11/14, Modified: 2018/11/15 Plugin Output 45.124.184.32 (tcp/443/www)
36	95631 (1) - SSL Certificate Signed Using Weak Hashing Algorithm (Known CA)	None	Published: 2016/12/08, Modified: 2022/10/12 95631 (1) - SSL Certificate Signed Using Weak Hashing Algorithm (Known CA) 93 Plugin Output 45.124.184.32 (tcp/443/www)

37.	156899 (1) - SSL/TLS Recommended Cipher Suites	None	Published: 2022/01/20, Modified: 2024/02/12 Plugin Output 156899 (1) - SSL/TLS Recommended Cipher Suites 96 45.124.184.32 (tcp/443/www)

REPORT:-

Vulnerability Name:-

111066 (1) - Apache Tomcat 7.0.0 < 7.0.89
121120 (1) - Apache Tomcat 7.0.0 < 7.0.76
171351 (1) - Apache Tomcat SEoL (7.0.x
197818 (1) - Apache Tomcat 7.0.0 < 7.0.72 multiple vulnerabilities0
197843 (1) - Apache Tomcat 7.0.0 < 7.0.100 multiple vulnerabilities
197848 (1) - Apache Tomcat 7.0.0 < 7.0.73 multiple vulnerabilities
88936 (1) - Apache Tomcat 7.0.0 < 7.0.68 multiple vulnerabilities
103329 (1) - Apache Tomcat 7.0.0 < 7.0.81 multiple vulnerabilities
103782 (1) - Apache Tomcat 7.0.0 < 7.0.82
121118 (1) - Apache Tomcat 7.0.5 < 7.0.67
121119 (1) - Apache Tomcat 7.0.0 < 7.0.70
121121 (1) - Apache Tomcat 7.0.28 < 7.0.88
124064 (1) - Apache Tomcat 7.0.0 < 7.0.94 multiple vulnerabilities
136770 (1) - Apache Tomcat 7.0.0 < 7.0.104
138851 (1) - Apache Tomcat 7.0.27 < 7.0.105
147163 (1) - Apache Tomcat 7.0.0 < 7.0.108 multiple vulnerabilities
197820 (1) - Apache Tomcat 7.0.0 < 7.0.77
197823 (1) - Apache Tomcat 7.0.0 < 7.0.75
197826 (1) - Apache Tomcat 7.0.25 < 7.0.90
197831 (1) - Apache Tomcat 7.0.0 < 7.0.78
197838 (1) - Apache Tomcat 7.0.0 < 7.0.99 multiple vulnerabilities
12085 (1) - Apache Tomcat Default Files
83764 (1) - Apache Tomcat 7.0.0 < 7.0.59
102587 (1) - Apache Tomcat 7.0.41 < 7.0.79

106975 (1) - Apache Tomcat 7.0.0 < 7.0.85 multiple vulnerabilities
118035 (1) - Apache Tomcat 7.0.23 < 7.0.91
121117 (1) - Apache Tomcat 7.0.0 < 7.0.65
148405 (1) - Apache Tomcat 7.0.0 < 7.0.107

severity : - High, Medium, Critical

Plugin:-

Port :- 8080

Description:-

The remote Apache Tomcat server is affected by a vulnerability.

An unsupported version of Apache Tomcat is installed on the remote host.

The version of Tomcat installed on the remote host is prior to 7.0.89. It is, therefore, affected by a vulnerability as referenced in the fixed_in_apache_tomcat_7.0.89_security-7 advisory. - The defaults settings for the CORS filter provided in Apache Tomcat 9.0.0.M1 to 9.0.8, 8.5.0 to 8.5.31, 8.0.0.RC1 to 8.0.52, 7.0.41 to 7.0.88 are insecure and enable 'supportsCredentials' for all origins. It is expected that users of the CORS filter will have configured it appropriately for their environment rather than using it in the default configuration. Therefore, it is expected that most users will not be impacted by this issue.

The ResourceLinkFactory implementation in Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 did not limit web application access to global JNDI resources to those resources explicitly linked to the web application. Therefore, it was possible for a web application to access any global JNDI resource whether an explicit ResourceLink had been configured or not. (CVE-2016-6797)

- A malicious web application running on Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 was able to bypass a configured SecurityManager via manipulation of the configuration parameters for the JSP Servlet. (CVE-2016-6796) –

When a SecurityManager is configured, a web application's ability to read system properties should be controlled by the SecurityManager. In Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70, 6.0.0 to 6.0.45 the system property replacement feature for configuration files could be used by a malicious web application to bypass the SecurityManager and read system properties that should not be visible. (CVE-2016-6794)

- In Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 a malicious web application was able to bypass a configured SecurityManager via a Tomcat utility method that was accessible to web applications. (CVE-2016-5018)

- The Realm implementations in Apache Tomcat versions 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 did not process the supplied password if the supplied user name did not exist. This made a timing attack possible to determine valid user names. Note that the default configuration includes the LockOutRealm which makes exploitation of this vulnerability harder.

- Remote code execution is possible with Apache Tomcat before 6.0.48, 7.x before 7.0.73, 8.x before 8.0.39, 8.5.x before 8.5.7, and 9.x before 9.0.0.M12 if JmxRemoteLifecycleListener is used and an attacker can reach JMX ports. The issue exists because this listener wasn't updated for consistency with the CVE-2016-3427 Oracle patch that affected credential types. (CVE-2016-8735)

- The code in Apache Tomcat 9.0.0.M1 to 9.0.0.M11, 8.5.0 to 8.5.6, 8.0.0.RC1 to 8.0.38, 7.0.0 to 7.0.72, and 6.0.0 to 6.0.47 that parsed the HTTP request line permitted invalid characters. This could be exploited, in conjunction with a proxy that also permitted the invalid characters but with a different interpretation, to inject data into the HTTP response. By manipulating the HTTP response the attacker could poison a webcache, perform an XSS attack and/or obtain sensitive information from requests other then their own.

According to its version, Apache Tomcat is 7.0.x. It is, therefore, no longer maintained by its vendor or provider. Lack of support implies that no new security patches for the product will be released by the vendor. As a result, it may contain security vulnerabilities.

The Mapper component in Apache Tomcat 6.x before 6.0.45, 7.x before 7.0.68, 8.x before 8.0.30, and 9.x before 9.0.0.M2 processes redirects before considering security constraints and Filters, which allows remote attackers to determine the existence of a directory via a URL that lacks a trailing / (slash) character. (CVE-2015-5345) - The (1) Manager and (2) Host Manager applications in Apache Tomcat 7.x before 7.0.68, 8.x before 8.0.31, and 9.x before 9.0.0.M2 establish sessions and send CSRF tokens for arbitrary new requests, which allows remote attackers to bypass a CSRF protection mechanism by using a token. (CVE-2015-5351)

Apache Tomcat 6.x before 6.0.45, 7.x before 7.0.68, 8.x before 8.0.31, and 9.x before 9.0.0.M2 does not place org.apache.catalina.manager.StatusManagerServlet on the org/apache/catalina/ core/RestrictedServlets.properties list, which allows remote authenticated users to bypass intended SecurityManager restrictions and read arbitrary HTTP requests, and consequently discover session ID values, via a crafted web application. (CVE-2016-0706)

The session-persistence implementation in Apache Tomcat 6.x before 6.0.45, 7.x before 7.0.68, 8.x before 8.0.31, and 9.x before 9.0.0.M2 mishandles session attributes, which allows remote authenticated users to bypass intended SecurityManager restrictions and execute arbitrary code in a privileged context via a web application that places a crafted object in a session. (CVE-2016-0714)

The setGlobalContext method in org/apache/naming/factory/ResourceLinkFactory.java in Apache Tomcat 7.x before 7.0.68, 8.x before 8.0.31, and 9.x before 9.0.0.M3 does not consider whether ResourceLinkFactory.setGlobalContext callers are authorized, which allows remote authenticated users to bypass intended SecurityManager restrictions and read or write to arbitrary application data, or cause a denial of service (application disruption), via a web application that sets a crafted global context.

When using FORM authentication with Apache Tomcat 9.0.0.M1 to 9.0.29, 8.5.0 to 8.5.49 and 7.0.0 to 7.0.98 there was a narrow window where an attacker could perform a session fixation attack. The window was considered too narrow for an exploit to be practical but, erring on the side of caution, this issue has been treated as a security vulnerability. (CVE-2019-17563)

When Apache Tomcat 9.0.0.M1 to 9.0.28, 8.5.0 to 8.5.47, 7.0.0 and 7.0.97 is configured

with the JMX Remote Lifecycle Listener, a local attacker without access to the Tomcat process or configuration files is able to manipulate the RMI registry to perform a man-in-the-middle attack to capture user names and passwords used to access the JMX interface. The attacker can then use these credentials to access the JMX interface and gain complete control over the Tomcat instance.

When serving resources from a network location using the NTFS file system, Apache Tomcat versions 10.0.0-M1 to 10.0.0-M9, 9.0.0.M1 to 9.0.39, 8.5.0 to 8.5.59 and 7.0.0 to 7.0.106 were susceptible to JSP source code disclosure in some configurations. The root cause was the unexpected behaviour of the JRE API File.getCanonicalPath() which in turn was caused by the inconsistent behaviour of the Windows API (FindFirstFileW) in some circumstances

solution:-

Upgrade to Apache Tomcat version. Upgrade to a version of Apache Tomcat that is currently supported.

Business Impact::- Apache Tomcat latest version is 9. But this server version is 7. Need to upgrade to version 9.

Vulnerability Name:- 11219 (3) - Nessus SYN scanner

severity : - None

Plugin:- Published: 2009/02/04, Modified: 2024/05/20 Plugin Output 45.124.184.32 (tcp/80/www)

Port :- 443, 8080

Description:-

This plugin is a SYN 'half-open' port scanner. It shall be reasonably quick even against a firewalled target. Note that 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.

solution:-

Protect your target with an IP filter.

Vulnerability Name:- 11422 (1) - Web Server Unconfigured - Default Install Page Present

severity : - None

Plugin:- Published: 2003/03/20, Modified: 2018/08/15 Plugin Output 45.124.184.32 (tcp/8080/www)

Port :- 8080

Description:- The remote web server uses its default welcome page. Therefore, it's probable that this server is not used at all or is serving content that is meant to be hidden.

solution:- Disable this service if you do not use it.

Vulnerability Name:- 20108 (1) - Web Server / Application favicon.ico Vendor Fingerprinting

severity : - None

Plugin:- Published: 2005/10/28, Modified: 2020/06/12 Plugin Output 45.124.184.32 (tcp/8080/www)

Port :- 8080

Description:- The 'favicon.ico' file found on the remote web server belongs to a popular web server. This may be used to fingerprint the web server.

solution:- Remove the 'favicon.ico' file or create a custom one for your site.

Vulnerability Name:- 42981 (1) - SSL Certificate Expiry - Future Expiry

severity : - None

Plugin:- Published: 2009/12/02, Modified: 2020/09/04 Plugin Output 45.124.184.32 (tcp/443/www)

Port :- 443

Description:- The SSL certificate associated with the remote service will expire soon.

solution:- Purchase or generate a new SSL certificate in the near future to replace the existing one.

Vulnerability Name:- 46180 (1) - Additional DNS Hostnames

severity : - None

Plugin:- Published: 2010/04/29, Modified: 2022/08/15 Plugin Output 45.124.184.32 (tcp/0)

Port :- 0

Description:- Hostnames different from the current hostname have been collected by miscellaneous plugins. Nessus has generated a list of hostnames that point to the remote host. Note that these are only the alternate hostnames for vhosts discovered on a web server. Different web servers may be hosted on name-based virtual hosts.

solution:- If you want to test them, re-scan using the special vhost syntax.

Vulnerability Name:- 66334 (1) - Patch Report

severity : - None

Plugin:- Published: 2013/07/08, Modified: 2024/06/27 Plugin Output 45.124.184.32 (tcp/0)

Port :- 0

Description:- The remote host is missing one or more security patches. This plugin lists the newest version of each patch to install to make sure the remote host is up-to-date. Note: Because the 'Show missing patches that have been superseded' setting in your scan policy depends on this plugin, it will always run and cannot be disabled.

solution:- Install the patches listed below.

Vulnerability Name:- 83298 (1) - SSL Certificate Chain Contains Certificates Expiring

Soon

severity : - None

Plugin:- Published: 2015/05/08, Modified: 2015/05/08 Plugin Output 45.124.184.32 (tcp/443/www)

Port :- 443

Description:- The remote host has an SSL certificate chain with one or more SSL certificates that are going to expire soon. Failure to renew these certificates before the expiration date may result in denial of service for users.

solution:- Renew any soon to expire SSL certificates.

Vulnerability Name:- 94761 (1) - SSL Root Certification Authority Certificate Information

severity : - None

Plugin:- Published: 2016/11/14, Modified: 2018/11/15 Plugin Output 45.124.184.32 (tcp/443/www)

Port :- 443

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

solution:- Ensure that use of this root Certification Authority certificate complies with your organization's acceptable use and security policies.

Vulnerability Name:- 95631 (1) - SSL Certificate Signed Using Weak Hashing Algorithm (Known CA)

severity : - None

Plugin:- Published: 2016/11/14, Modified: 2018/11/15 Plugin Output 45.124.184.32 (tcp/443/www)

Port :- 443

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. Note that 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. Note that 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.

solution:- Contact the Certificate Authority to have the certificate reissued.

Vulnerability Name:- 156899 (1) - SSL/TLS Recommended Cipher Suites

severity : - None

Plugin:- Published: 2016/11/14, Modified: 2018/11/15 Plugin Output 45.124.184.32 (tcp/443/www)

Port :- 443

Description:- The remote host has open SSL/TLS ports which advertise discouraged cipher suites. It is recommended to only enable support for the following cipher suites: TLSv1.3: - 0x13,0x01 TLS13_AES_128_GCM_SHA256 - 0x13,0x02 TLS13_AES_256_GCM_SHA384 - 0x13,0x03 TLS13_CHACHA20_POLY1305_SHA256 TLSv1.2: - 0xC0,0x2B ECDHE-ECDSA-AES128-GCM-SHA256 - 0xC0,0x2F ECDHE-RSA-AES128-GCM-SHA256 - 0xC0,0x2C ECDHE-ECDSA-AES256-GCM-SHA384 - 0xC0,0x30 ECDHE-RSA-AES256-GCM-SHA384 - 0xCC,0xA9 ECDHE-ECDSA-CHACHA20-POLY1305 -

0xCC,0xA8 ECDHE-RSA-CHACHA20-POLY1305 This is the recommended configuration for the vast majority of services, as it is highly secure and compatible with nearly every client released in the last five (or more) years.

solution:- Only enable support for recommended cipher suites.

STAGE 3:ABILITY OF SOC / SEIM

SOC

A Security Operations Center (SOC) is a centralized facility within an organization tasked with monitoring, detecting, analyzing, and responding to cybersecurity incidents in real-time. It is staffed by a team of skilled security professionals, including analysts, engineers, and managers, who work around the clock to ensure the security of the organization's information systems, networks, devices, and data. The SOC uses advanced technologies and processes, such as Security Information and Event Management (SIEM) systems, intrusion detection/prevention systems (IDS/IPS), and threat intelligence platforms, to identify and mitigate potential threats. By continuously monitoring network traffic, system activities, and data flows, the SOC can quickly detect anomalies and respond to incidents to prevent or minimize damage.

In addition to real-time threat monitoring and response, the SOC is responsible for conducting detailed incident investigations, providing comprehensive reporting, and ensuring regulatory compliance. The SOC's activities include incident triage, deep dive analysis, threat hunting, and post-incident reviews to understand the attack vectors and techniques used by adversaries. This continuous cycle of monitoring, detection, analysis, and response helps organizations improve their security posture over time. The SOC also plays a vital role in the development and refinement of security policies, procedures, and controls, ensuring that the organization remainresilient against evolving cyber threats.

SOC – cycle

The SOC cycle, also known as the Security Operations Center lifecycle, encompasses a series of ongoing and iterative processes that ensure effective cybersecurity operations. The cycle typically includes the following stages:

1. **Preparation**:

- **Asset Inventory**: Identifying and cataloging all assets within the organization, including hardware, software, data, and network components.
- **Baseline Development**: Establishing normal behavior patterns for systems and users to identify deviations indicative of potential threats.
- **Policy and Procedure Development**: Creating and updating security policies, incident response plans, and standard operating procedures.

2. **Detection and Monitoring**:
   - **Real-time Monitoring**: Continuously observing network traffic, system activities, and logs using various tools such as Security Information and Event Management (SIEM) systems, Intrusion Detection Systems (IDS), and Intrusion Prevention Systems (IPS).
   - **Threat Intelligence**: Integrating threat intelligence feeds to stay updated on the latest threats, vulnerabilities, and attack techniques.
   - **Alert Generation**: Automatically generating alerts when suspicious activities or anomalies are detected.
3. **Analysis and Investigation**:
   - **Triage**: Prioritizing alerts based on their severity, potential impact, and likelihood of being a true positive.
   - **Deep Dive Analysis**: Conducting detailed investigations of high-priority alerts to determine the nature, scope, and impact of potential incidents.
   - **Threat Hunting**: Proactively searching for signs of hidden threats or compromised systems that may not have triggered automated alerts.
4. **Response**:
   - **Containment**: Isolating affected systems or network segments to prevent the spread of the threat.
   - **Eradication**: Removing malicious software, closing vulnerabilities, and restoring affected systems to their normal state.
   - **Recovery**: Bringing systems back online, verifying their integrity, and ensuring business continuity.
5. **Post-Incident Activities**:
   - **Lessons Learned**: Conducting post-incident reviews to identify what worked well and areas for improvement.
   - **Reporting**: Documenting the incident, actions taken, and outcomes for internal stakeholders and regulatory compliance.

- **Feedback Loop**: Updating policies, procedures, and security controls based on insights gained from the incident.

6. **Continuous Improvement**:
   - **Training and Awareness**: Providing ongoing training for SOC staff and raising awareness among employees about cybersecurity best practices.
   - **Technology Upgrades**: Implementing new tools and technologies to enhance detection, analysis, and response capabilities.
   - **Process Refinement**: Regularly reviewing and refining SOC processes to improve efficiency and effectiveness.

> This cycle ensures that the SOC remains adaptive and responsive to the dynamic nature of cybersecurity threats, continuously improving its ability to protect the organization's assets.

SIEM

Security Information and Event Management (SIEM) is a comprehensive cybersecurity approach that combines two key functions: security information management (SIM) and security event management (SEM). SIEM systems collect, normalize, and aggregate log and event data from a wide array of sources, including network devices, servers, applications, and security systems such as firewalls and intrusion detection/prevention systems (IDS/IPS). By centralizing this data, SIEM provides a unified view of an organization's security posture, enabling more effective monitoring and analysis. Advanced SIEM solutions employ sophisticated correlation rules, machine learning algorithms, and behavioral analytics to detect anomalies, identify potential threats, and generate real-time alerts for security teams.

In addition to threat detection and alerting, SIEM systems play a crucial role in incident response and compliance management. They facilitate the investigation of security incidents by providing detailed logs and event data that help trace the actions of malicious actors and understand the scope of an attack. SIEM also aids in regulatory compliance by offering comprehensive reporting and auditing capabilities, ensuring that organizations meet the requirements of various industry standards and regulations. By integrating data from diverse sources and providing actionable insights, SIEM enhances the ability of security operations centers (SOCs) to respond swiftly to incidents, mitigate risks, and improve overall cybersecurity resilience.

SIEM Cycle

The SIEM cycle is a continuous process that ensures effective management of security information and events.

1. **Data Collection**: Gather log and event data from various sources such as firewalls, IDS/IPS, servers, applications, and network devices.
2. **Normalization**: Convert collected data into a standardized format for consistency and easier analysis.
3. **Aggregation**: Combine data from multiple sources to provide a unified view of security events.
4. **Parsing**: Break down collected data into individual fields for detailed analysis.
5. **Correlation**: Identify relationships between different events to detect patterns indicative of potential security incidents.
6. **Detection**: Use predefined rules, machine learning, and anomaly detection to identify threats and suspicious activities.
7. **Alerting**: Generate alerts based on detected threats and anomalies for immediate attention.
8. **Investigation**: Conduct detailed analysis of alerts to determine the nature, scope, and impact of potential incidents.
9. **Prioritization**: Rank alerts based on severity, impact, and urgency to focus on the most critical issues.
10. **Response**: Take appropriate actions to mitigate and contain identified threats, including isolation, eradication, and recovery.
11. **Reporting**: Generate reports for internal stakeholders and regulatory compliance, detailing incidents, actions taken, and outcomes.
12. **Forensics**: Perform in-depth forensic analysis to understand the root cause and attack vectors of incidents.
13. **Feedback Loop**: Incorporate insights and lessons learned from incidents into the SIEM system to improve detection and response capabilities.
14. **Compliance Management**: Ensure ongoing adherence to industry regulations and standards through continuous monitoring and reporting.
15. **Continuous Improvement**: Regularly update and refine SIEM processes, rules, and technologies to adapt to evolving threats and enhance overall security posture.

MISP

MISP (Malware Information Sharing Platform) is an open-source threat intelligence platform designed to improve the sharing, storing, and analysis of structured threat information among organizations. It enables the collection and centralization of threat data, such as indicators of compromise (IOCs), from various sources, facilitating collaborative defense efforts. By providing tools for analyzing and correlating threat data, MISP helps identify trends and patterns, enhancing the ability to detect and respond to security incidents. Additionally, MISP supports automation through APIs, allowing for seamless integration with other security tools, and promotes community-driven information sharing to strengthen overall cybersecurity posture.

Your college network information

Network Information Description for 200 Computers in an ITNU, Nirma University

The engineering college's network infrastructure is designed to support 200 computers, ensuring robust connectivity, high performance, and secure access for students, faculty, and administrative staff. The network is segmented into various functional areas, each tailored to meet specific needs and ensure optimal performance.

Network Topology

The network employs a hybrid topology combining star and bus configurations. This design optimizes the balance between performance and scalability.

Backbone Infrastructure

The backbone network is built on high-speed Gigabit Ethernet, ensuring rapid data transfer and low latency across the campus. Fiber optic cables are used for inter-building connections, providing a high-speed backbone that connects various departments, labs, and administrative offices.

Network Segmentation

To enhance security and manageability, the network is segmented into several VLANs (Virtual Local Area Networks), including:

- **Academic VLAN**: Dedicated to computer labs and classrooms, providing high-speed access to academic resources, software applications, and the internet.

- **Administrative VLAN**: Segregated for administrative and faculty use, ensuring secure access to sensitive data and internal services.
- **Student VLAN**: Separate segment for student personal devices, offering internet access while maintaining isolation from the academic and administrative networks.

Access Points and Wireless Coverage
The campus is equipped with multiple high-performance wireless access points (APs) to provide seamless Wi-Fi coverage in classrooms, lecture halls, libraries, and common areas. The Wi-Fi network supports the latest standards (Wi-Fi 6) to ensure fast and reliable wireless connectivity.

Network Security
Security measures include:

- **Firewalls**: Protecting the network perimeter with advanced firewall solutions to block unauthorized access and prevent cyber threats.
- **Intrusion Detection Systems (IDS)**: Monitoring network traffic for suspicious activities and potential security breaches.
- **Encryption**: Implementing WPA3 encryption for wireless networks and SSL/TLS encryption for secure data transmission.

Network Management
A centralized network management system is in place to monitor and control the entire network infrastructure. This includes:

- **Network Monitoring Tools**: Providing real-time insights into network performance, traffic patterns, and potential issues.
- **Automated Alerts**: Notifying IT staff of any anomalies or failures, ensuring prompt response and resolution.
- **Regular Maintenance**: Scheduled maintenance and updates to keep the network infrastructure up-to-date and secure.

Support and Maintenance

A dedicated IT support team is responsible for maintaining the network, troubleshooting issues, and ensuring minimal downtime. Regular training sessions are conducted to keep the staff updated with the latest network management practices and technologies.

How you think you deploy soc in your college

Deploying a Security Operations Center (SOC) within the existing network infrastructure of an ITNU involves several key steps. The SOC will enhance the overall security posture by monitoring, detecting, and responding to cybersecurity threats in real-time. Here's a comprehensive guide to deploying a SOC in this context:

1. Assessment and Planning

- **Conduct a Risk Assessment**: Identify potential threats and vulnerabilities in the current network setup.
- **Define SOC Objectives**: Clearly outline the goals and scope of the SOC, including the types of incidents to monitor and response strategies.
- **Budget and Resources**: Determine the budget for SOC implementation and allocate necessary resources, including personnel, hardware, and software.

2. Infrastructure Setup

- **Physical Location**: Set up a dedicated, secure room for the SOC with controlled access.
- **Hardware Requirements**: Procure and install the necessary hardware, including servers, workstations, large displays for monitoring, and secure storage for logs and data.

3. Network Integration

- **Deploy Security Tools**: Install essential security tools such as firewalls, intrusion detection/prevention systems (IDS/IPS), endpoint protection, and data loss prevention (DLP) systems.
- **SIEM System**: Implement a Security Information and Event Management (SIEM) system to aggregate and analyze logs from various sources (firewalls, IDS/IPS, servers, endpoints).

- **Network Segmentation**: Ensure that the SOC network is isolated from the academic and administrative networks to prevent any potential breaches from spreading.

4. Data Collection and Monitoring

- **Log Collection**: Configure all network devices, servers, and endpoints to forward logs to the SIEM system.
- **Real-Time Monitoring**: Set up dashboards in the SIEM for real-time monitoring of network traffic, system logs, and security events.
- **Automated Alerts**: Configure automated alerts for suspicious activities, unusual traffic patterns, and potential security incidents.

5. Incident Response Plan

- **Develop Response Procedures**: Create detailed incident response procedures for different types of security incidents (malware infections, DDoS attacks, data breaches).
- **Team Training**: Train the SOC team on incident response protocols and conduct regular drills to ensure readiness.
- **Communication Plan**: Establish clear communication channels for reporting incidents to relevant stakeholders (IT team, management, affected users).

6. Staffing and Roles

- **SOC Team**: Hire or assign dedicated personnel for the SOC, including SOC analysts, incident responders, and a SOC manager.
- **Roles and Responsibilities**: Define clear roles and responsibilities for each team member, ensuring 24/7 coverage if necessary.

7. Continuous Improvement

- **Regular Audits**: Conduct regular security audits and vulnerability assessments to identify and remediate weaknesses.
- **Update Security Policies**: Keep security policies and procedures up-to-date with evolving threats and technologies.
- **Training and Awareness**: Provide ongoing training for SOC staff and conduct security awareness programs for the entire college community.

8. Reporting and Compliance

- **Compliance Monitoring**: Ensure that the SOC operations comply with relevant regulations and standards (e.g., GDPR, FERPA).
- **Reporting**: Generate regular security reports for management, highlighting incidents, response actions, and security improvements.

9. Third-Party Integration

- **Threat Intelligence**: Integrate threat intelligence feeds to stay updated on emerging threats and vulnerabilities.
- **Collaboration**: Collaborate with external security experts, local authorities, and other educational institutions for knowledge sharing and joint response to widespread threats.

Threat intelligence

Threat intelligence refers to the collection, analysis, and dissemination of information about potential and current threats to an organization's security. It plays a critical role in enhancing an organization's ability to defend against cyberattacks by providing actionable insights into the tactics, techniques, and procedures (TTPs) used by cyber adversaries. Here's a detailed explanation of threat intelligence:

1. Definition and Purpose

- **Definition**: Threat intelligence is evidence-based knowledge, including context, mechanisms, indicators, implications, and actionable advice about an existing or emerging menace or hazard to assets.
- **Purpose**: The primary goal of threat intelligence is to help organizations understand the risks they face from cyber threats, enabling them to make informed decisions and take proactive measures to protect their networks, systems, and data.

2. Types of Threat Intelligence

Threat intelligence can be categorized into several types based on the nature of the information and its usage:

- **Strategic Threat Intelligence**: Provides high-level insights into threat landscapes, trends, and potential impacts on business operations. It is often used

by senior management to inform decision-making and risk management strategies.

- **Tactical Threat Intelligence**: Focuses on the tactics, techniques, and procedures (TTPs) of threat actors. It is used by security teams to understand the methods adversaries use and to develop countermeasures.
- **Operational Threat Intelligence**: Relates to specific, time-sensitive events and incidents. It includes information about active campaigns, threat actor activities, and indicators of compromise (IOCs). This type is used for immediate threat detection and response.
- **Technical Threat Intelligence**: Includes detailed technical data such as IP addresses, domain names, file hashes, and malware signatures. It is used to configure security tools like firewalls, intrusion detection systems (IDS), and antivirus software.

3. Sources of Threat Intelligence

Threat intelligence is gathered from various sources, including:

- **Internal Sources**: Logs from network devices, servers, endpoints, and security tools within the organization.
- **External Sources**: Open-source intelligence (OSINT), dark web monitoring, threat intelligence feeds from commercial providers, government and industry reports, and information sharing with other organizations and cybersecurity communities.
- **Human Intelligence (HUMINT)**: Information gathered from human sources, including security researchers, analysts, and informants within cyber threat groups.

4. Lifecycle of Threat Intelligence

The process of threat intelligence involves several stages, commonly referred to as the threat intelligence lifecycle:

1. **Planning and Direction**: Define the objectives and requirements for threat intelligence based on the organization's needs.
2. **Collection**: Gather data from various internal and external sources.
3. **Processing**: Organize and structure the collected data for analysis.

4. **Analysis**: Examine the processed data to extract actionable insights and produce intelligence reports.
5. **Dissemination**: Distribute the intelligence to relevant stakeholders within the organization.
6. **Feedback**: Gather feedback to refine and improve the threat intelligence process.

5. Benefits of Threat Intelligence

- **Proactive Defense**: Enables organizations to anticipate and mitigate threats before they materialize.
- **Improved Incident Response**: Provides context and details that enhance the effectiveness of incident response efforts.
- **Enhanced Security Posture**: Helps in identifying and addressing vulnerabilities, improving overall security measures.
- **Informed Decision-Making**: Empowers management with insights to make informed decisions about security investments and risk management.

6. Challenges in Threat Intelligence

- **Data Overload**: Managing and processing large volumes of data from diverse sources can be overwhelming.
- **Timeliness and Relevance**: Ensuring that the intelligence is timely and relevant to the organization's specific context.
- **Quality and Accuracy**: Validating the accuracy and reliability of the collected intelligence.
- **Integration**: Seamlessly integrating threat intelligence into existing security operations and tools.

Incident response

Incident response (IR) is a structured approach to managing and addressing security incidents, breaches, or cyber threats. The objective is to handle the situation in a way that limits damage and reduces recovery time and costs. A well-defined incident response plan helps organizations deal with potential cybersecurity incidents efficiently and effectively.

1. Definition and Importance

- **Definition**: Incident response is the process of detecting, investigating, and responding to security incidents to mitigate their impact.
- **Importance**: Effective incident response minimizes the impact of security breaches, reduces downtime, protects sensitive data, and helps maintain the organization's reputation.

2. Incident Response Lifecycle

The incident response process typically follows a lifecycle consisting of several phases. These phases ensure a comprehensive and coordinated approach to managing incidents:

1. **Preparation**
    - **Objective**: Establish and train an incident response team, create an incident response plan, and ensure the necessary tools and resources are available.
    - **Activities**: Develop policies, conduct training sessions, and set up communication protocols.
2. **Identification**
    - **Objective**: Detect and identify potential security incidents.
    - **Activities**: Monitor systems and networks for signs of unusual activity, utilize intrusion detection systems (IDS), and analyze logs and alerts.
3. **Containment**
    - **Objective**: Contain the incident to prevent further damage and limit its spread.
    - **Activities**: Implement short-term containment measures (e.g., isolating affected systems) and develop long-term containment strategies (e.g., applying patches, removing malware).
4. **Eradication**
    - **Objective**: Eliminate the root cause of the incident and remove malicious elements from the environment.
    - **Activities**: Identify and remove malware, close vulnerabilities, and clean affected systems.
5. **Recovery**
    - **Objective**: Restore affected systems and services to normal operation.

    - **Activities**: Rebuild and restore systems from clean backups, monitor systems for any signs of lingering issues, and validate that systems are functioning correctly.
6. **Lessons Learned**
    - **Objective**: Review and analyze the incident to improve future response efforts.
    - **Activities**: Conduct a post-incident review, document findings, update the incident response plan, and implement improvements based on lessons learned.

3. Key Components of an Incident Response Plan
An effective incident response plan should include the following components:

- **Incident Response Team (IRT)**: A designated group of individuals with defined roles and responsibilities for managing incidents. This may include IT staff, security analysts, legal advisors, and communication personnel.
- **Communication Plan**: Procedures for internal and external communication during an incident, including notification protocols for stakeholders, customers, and regulatory bodies.
- **Incident Classification**: Criteria for categorizing incidents based on their severity and impact, helping prioritize response efforts.
- **Investigation and Analysis Procedures**: Guidelines for collecting and analyzing evidence, identifying the cause and scope of the incident, and documenting findings.
- **Containment, Eradication, and Recovery Procedures**: Step-by-step instructions for containing the incident, removing the threat, and restoring normal operations.
- **Documentation and Reporting**: Processes for documenting incident details, actions taken, and outcomes, as well as reporting requirements for compliance purposes.

4. Tools and Technologies for Incident Response
Several tools and technologies can aid in the incident response process:

- **Security Information and Event Management (SIEM)**: Aggregates and analyzes log data to detect and respond to security incidents.

- **Intrusion Detection/Prevention Systems (IDS/IPS)**: Monitors network traffic for suspicious activity and takes action to prevent or mitigate threats.
- **Endpoint Detection and Response (EDR)**: Provides visibility into endpoint activity and enables rapid detection and response to threats.
- **Forensic Tools**: Assists in collecting and analyzing digital evidence to understand the scope and impact of an incident.
- **Automated Response Tools**: Automates routine response tasks, enabling faster and more efficient incident management.

5. Challenges in Incident Response

- **Rapid Detection**: Quickly identifying incidents amid vast amounts of data and potential false positives.
- **Coordination**: Ensuring effective communication and coordination among various teams and stakeholders.
- **Resource Constraints**: Managing incidents with limited personnel and technical resources.
- **Evolving Threats**: Keeping up with constantly changing threat landscapes and attack techniques.
- **Legal and Regulatory Compliance**: Navigating legal and regulatory requirements related to incident reporting and data protection.

Qradar & understanding about tool

IBM QRadar is a leading Security Information and Event Management (SIEM) solution designed to provide comprehensive visibility and actionable insights into an organization's IT infrastructure. QRadar helps security teams detect, investigate, and respond to threats more effectively by aggregating and analyzing log data, network traffic, and other security-related information.

1. Key Features of QRadar

- **Log Management**: Collects, normalizes, and stores log data from various sources such as firewalls, servers, applications, and endpoints.
- **Real-Time Monitoring**: Provides real-time visibility into network activity and system events, enabling quick detection of suspicious behavior.
- **Correlation Engine**: Correlates events from multiple sources to identify complex attack patterns and prioritize security alerts.

- **Advanced Analytics**: Uses machine learning and behavioral analysis to detect anomalies and advanced threats that traditional methods might miss.
- **Incident Response**: Facilitates the investigation and response to security incidents through integrated workflows and automated actions.
- **Threat Intelligence**: Incorporates threat intelligence feeds to stay updated on the latest threats and vulnerabilities, enhancing detection and response capabilities.
- **User Behavior Analytics (UBA)**: Monitors user activities and behaviors to identify potential insider threats and compromised accounts.
- **Dashboards and Reporting**: Provides customizable dashboards and detailed reports for security monitoring, compliance, and management purposes.

2. Architecture of QRadar

QRadar's architecture is modular and scalable, allowing it to be tailored to the needs of various organizations, from small businesses to large enterprises. The primary components include:

- **Event Collectors**: Gather log data from different sources and forward it to the Event Processor.
- **Event Processor**: Processes and normalizes the collected log data, applying correlation rules to detect potential security incidents.
- **Data Node**: Stores processed data and provides long-term storage for log and event data.
- **Flow Processor**: Analyzes network traffic data (flows) to provide visibility into network activity and detect anomalies.
- **QRadar Console**: The user interface for managing and monitoring QRadar, including dashboards, reports, and incident management tools.

3. Deployment Options

QRadar can be deployed in various ways to suit different organizational needs:

- **On-Premises**: Installed on physical or virtual servers within the organization’s data center.
- **Cloud-Based**: Delivered as a cloud service, reducing the need for on-site hardware and infrastructure management.
- **Hybrid**: Combines on-premises and cloud-based components to provide flexibility and scalability.

4. Use Cases

QRadar supports a wide range of use cases, including:

- **Security Monitoring**: Continuously monitors network traffic and system logs for signs of suspicious activity.
- **Threat Detection**: Identifies and prioritizes potential threats using advanced analytics and correlation rules.
- **Incident Response**: Facilitates rapid investigation and response to security incidents, helping to minimize damage and downtime.
- **Compliance**: Helps organizations meet regulatory requirements by providing detailed logs, reports, and audit trails.
- **Fraud Detection**: Monitors for unusual patterns that may indicate fraudulent activities.

5. Benefits of QRadar

- **Comprehensive Visibility**: Provides a unified view of security across the entire IT environment, making it easier to detect and respond to threats.
- **Improved Efficiency**: Automates many aspects of security monitoring and incident response, reducing the workload on security teams.
- **Enhanced Detection**: Uses advanced analytics and threat intelligence to detect sophisticated threats that might evade traditional security measures.
- **Scalability**: Can scale to meet the needs of growing organizations, with flexible deployment options.
- **Integration**: Integrates with a wide range of third-party security tools and data sources, enhancing its capabilities and providing a more comprehensive security posture.

6. Challenges and Considerations

- **Complexity**: Setting up and managing QRadar can be complex, requiring skilled personnel to configure and maintain the system.
- **Cost**: The total cost of ownership can be high, particularly for large deployments, including licensing, hardware, and maintenance costs.
- **Tuning**: Requires ongoing tuning and customization to ensure that correlation rules and analytics are effective and that false positives are minimized.

Conclusion :-

Stage 1 :- what you understand from Web application testing .

Web application testing is the process of evaluating and verifying the functionality, security, usability, performance, and compatibility of a web application. The goal is to ensure that the application works as expected and provides a good user experience while being secure from potential threats. Here are the key aspects of web application testing:

1. **Functional Testing**:
    - Verifies that the application functions according to the specified requirements.
    - Includes testing user interfaces, APIs, databases, security, client/server applications, and functionality.
2. **Usability Testing**:
    - Ensures the application is easy to use and provides a good user experience.
    - Involves checking navigation, layout, content, and other user interface elements.
3. **Performance Testing**:
    - Evaluates the application's performance under different conditions, such as varying loads and network speeds.
    - Includes load testing, stress testing, and scalability testing.
4. **Security Testing**:
    - Identifies vulnerabilities and ensures that the application is protected against threats like SQL injection, cross-site scripting (XSS), and other cyber attacks.
    - Involves authentication, authorization, data protection, and secure communication checks.
5. **Compatibility Testing**:
    - Ensures the application works across different browsers, devices, operating systems, and network environments.
    - Includes cross-browser testing, mobile testing, and responsive design testing.
6. **Interface Testing**:
    - Verifies that interactions between different systems or components of the application work correctly.
    - Includes testing APIs, web services, and data exchange processes.
7. **Database Testing**:
    - Ensures that the database functions correctly, and data integrity is maintained.
    - Includes checking data retrieval, data updates, data deletion, and database connections.
8. **Regression Testing**:
    - Ensures that new code changes do not adversely affect existing functionality.
    - Involves re-running previous tests to check for defects in the software.
9. **Automation Testing**:
    - Uses automated tools to execute test cases, which can save time and increase coverage.
    - Popular tools include Selenium, QTP, and TestComplete.

Effective web application testing involves planning, creating test cases, executing tests, and reporting results to ensure the application is of high quality before it is deployed.

Stage 2 :- what you understand from the nessus report .

The remote Apache Tomcat server is affected by a vulnerability. An unsupported version of Apache Tomcat is installed on the remote host. The version of Tomcat installed on the remote host is prior to 7.0.89. It is, therefore, affected by a vulnerability as referenced in the fixed_in_apache_tomcat_7.0.89_security-7 advisory. - The defaults settings for the CORS filter provided in Apache Tomcat 9.0.0.M1 to 9.0.8, 8.5.0 to 8.5.31, 8.0.0.RC1 to 8.0.52, 7.0.41 to 7.0.88 are insecure and enable 'supportsCredentials' for all origins. It is expected that users of the CORS filter will have configured it appropriately for their environment rather than using it in the default configuration. Therefore, it is expected that most users will not be impacted by this issue.

The ResourceLinkFactory implementation in Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 did not limit web application access to global JNDI resources to those resources explicitly linked to the web application. Therefore, it was possible for a web application to access any global JNDI resource whether an explicit ResourceLink had been configured or not. (CVE-2016-6797)

A malicious web application running on Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 was able to bypass a configured SecurityManager via manipulation of the configuration parameters for the JSP Servlet. (CVE-2016-6796) –

When a SecurityManager is configured, a web application's ability to read system properties should be controlled by the SecurityManager. In Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70, 6.0.0 to 6.0.45 the system property replacement feature for configuration files could be used by a malicious web application to bypass the SecurityManager and read system properties that should not be visible. (CVE-2016-6794)

In Apache Tomcat 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 a malicious web application was able to bypass a configured Security Manager via a Tomcat utility method that was accessible to web applications. (CVE-2016-5018)

The Realm implementations in Apache Tomcat versions 9.0.0.M1 to 9.0.0.M9, 8.5.0 to 8.5.4, 8.0.0.RC1 to 8.0.36, 7.0.0 to 7.0.70 and 6.0.0 to 6.0.45 did not process the supplied password if the supplied user name did not exist. This made a timing attack possible to determine valid user names. Note that the default configuration includes the LockOutRealm which makes exploitation of this vulnerability harder.

Remote code execution is possible with Apache Tomcat before 6.0.48, 7.x before 7.0.73, 8.x before 8.0.39, 8.5.x before 8.5.7, and 9.x before 9.0.0.M12 if JmxRemoteLifecycleListener is used and an attacker can reach JMX ports. The issue exists because this listener wasn't updated for consistency with the CVE-2016-3427 Oracle patch that affected credential types. (CVE-2016-8735)

The code in Apache Tomcat 9.0.0.M1 to 9.0.0.M11, 8.5.0 to 8.5.6, 8.0.0.RC1 to 8.0.38, 7.0.0 to 7.0.72, and 6.0.0 to 6.0.47 that parsed the HTTP request line permitted invalid characters. This could be exploited, in conjunction with a proxy that also permitted the invalid characters but with a different interpretation, to inject data into the HTTP response. By manipulating the HTTP response the attacker could poison a webcache, perform an XSS attack and/or obtain sensitive information from requests other then their own.

According to its version, Apache Tomcat is 7.0.x. It is, therefore, no longer maintained by its vendor or provider. Lack of support implies that no new security patches for the product will be released by the vendor. As a result, it may contain security vulnerabilities.

The Mapper component in Apache Tomcat 6.x before 6.0.45, 7.x before 7.0.68, 8.x before 8.0.30, and 9.x before 9.0.0.M2 processes redirects before considering security constraints and Filters, which allows remote attackers to determine the existence of a directory via a URL that lacks a trailing / (slash) character. (CVE-2015-5345) - The (1) Manager and (2) Host Manager applications in Apache Tomcat 7.x before 7.0.68, 8.x before 8.0.31, and 9.x before 9.0.0.M2 establish sessions and send CSRF tokens for arbitrary new requests, which allows remote attackers to bypass a CSRF protection mechanism by using a token. (CVE-2015-5351)

Apache Tomcat 6.x before 6.0.45, 7.x before 7.0.68, 8.x before 8.0.31, and 9.x before 9.0.0.M2 does not place org.apache.catalina.manager.StatusManagerServlet on the

org/apache/catalina/ core/RestrictedServlets.properties list, which allows remote authenticated users to bypass intended SecurityManager restrictions and read arbitrary HTTP requests, and consequently discover session ID values, via a crafted web application. (CVE-2016-0706)

The session-persistence implementation in Apache Tomcat 6.x before 6.0.45, 7.x before 7.0.68, 8.x before 8.0.31, and 9.x before 9.0.0.M2 mishandles session attributes, which allows remote authenticated users to bypass intended SecurityManager restrictions and execute arbitrary code in a privileged context via a web application that places a crafted object in a session. (CVE-2016-0714)

The setGlobalContext method in org/apache/naming/factory/ResourceLinkFactory.java in Apache Tomcat 7.x before 7.0.68, 8.x before 8.0.31, and 9.x before 9.0.0.M3 does not consider whether ResourceLinkFactory.setGlobalContext callers are authorized, which allows remote authenticated users to bypass intended SecurityManager restrictions and read or write to arbitrary application data, or cause a denial of service (application disruption), via a web application that sets a crafted global context.

When using FORM authentication with Apache Tomcat 9.0.0.M1 to 9.0.29, 8.5.0 to 8.5.49 and 7.0.0 to 7.0.98 there was a narrow window where an attacker could perform a session fixation attack. The window was considered too narrow for an exploit to be practical but, erring on the side of caution, this issue has been treated as a security vulnerability. (CVE-2019-17563)

When Apache Tomcat 9.0.0.M1 to 9.0.28, 8.5.0 to 8.5.47, 7.0.0 and 7.0.97 is configured with the JMX Remote Lifecycle Listener, a local attacker without access to the Tomcat process or configuration files is able to manipulate the RMI registry to perform a man-in-the-middle attack to capture user names and passwords used to access the JMX interface. The attacker can then use these credentials to access the JMX interface and gain complete control over the Tomcat instance.When serving resources from a network location using the NTFS file system, Apache Tomcat versions 10.0.0-M1 to 10.0.0-M9, 9.0.0.M1 to 9.0.39, 8.5.0 to 8.5.59 and 7.0.0 to 7.0.106 were susceptible to JSP source code disclosure in some configurations. The root cause was the unexpected behaviour of the JRE API File.getCanonicalPath() which in turn was caused by the inconsistent behaviour of the Windows API (FindFirstFileW) in some circumstances

Stage 3 :- what you understand from SOC / SEIM / Qradar Dashboard .

- **Comprehensive Visibility**: Provides a unified view of security across the entire IT environment, making it easier to detect and respond to threats.
- **Improved Efficiency**: Automates many aspects of security monitoring and incident response, reducing the workload on security teams.
- **Enhanced Detection**: Uses advanced analytics and threat intelligence to detect sophisticated threats that might evade traditional security measures.
- **Scalability**: Can scale to meet the needs of growing organizations, with flexible deployment options.
- **Integration**: Integrates with a wide range of third-party security tools and data sources, enhancing its capabilities and providing a more comprehensive security posture.

Future Scope :-

Stage 1 :- future scope of web application testing

The future of web application testing will be characterized by greater automation, advanced technologies like AI and machine learning, and an increased focus on performance, security, and user experience. These advancements will enable more efficient, effective, and comprehensive testing of web applications.

Stage 2 :- future scope of testing process you understood .

The future of testing processes will be characterized by increased automation, integration of advanced technologies like AI and machine learning, continuous testing within CI/CD pipelines, and a strong focus on security, performance, and user experience. These advancements will enable more efficient, effective, and comprehensive testing, ensuring higher quality applications and faster delivery times.

Stage 3 :- future scope of SOC / SEIM

Future Scope of Security Operations Centers (SOC)

The future scope of Security Operations Centers (SOC) in cybersecurity is poised for transformative advancements driven by the integration of cutting-edge technologies such as artificial intelligence (AI) and machine learning (ML). These technologies will significantly enhance the capabilities of SOCs, enabling more accurate and real-time

threat detection and response. Automated systems will analyze vast amounts of data to identify anomalies and potential threats more efficiently than ever before. Predictive analytics, powered by ML, will allow SOCs to anticipate and mitigate threats before they materialize, shifting the focus from reactive to proactive defense measures. This integration will reduce the workload on human analysts, allowing them to focus on more complex and strategic tasks.

The evolution of cloud and hybrid environments will require SOCs to develop advanced solutions that ensure comprehensive visibility and protection across diverse infrastructures. Cloud-native security solutions will become essential as more organizations migrate their operations to the cloud. These solutions will offer seamless integration with various cloud services, maintaining consistent monitoring and threat detection regardless of where the data resides. Additionally, SOCs will need to manage multi-cloud environments, ensuring that security policies and threat detection mechanisms are uniformly applied across different cloud platforms. This capability will be crucial in providing a unified and robust security posture.

Furthermore, the implementation of Zero Trust architecture will redefine the security landscape within SOCs. The Zero Trust model operates on the principle of continuous verification and validation of users and devices, regardless of their location within or outside the network perimeter. This approach will significantly reduce the risk of lateral movement by attackers, limiting their ability to exploit vulnerabilities within the network. Micro-segmentation, a key aspect of Zero Trust, will further enhance security by dividing the network into smaller, isolated segments. This granular approach will ensure that even if one segment is compromised, the threat is contained and does not spread across the entire network.

The human element within SOCs will also undergo substantial development. Continuous training and skill enhancement for SOC personnel will be crucial to keep up with the rapidly evolving cybersecurity landscape. Advanced training programs focusing on the latest trends, tools, and techniques will be essential to ensure that analysts are well-prepared to handle sophisticated threats. Additionally, reducing alert fatigue through smarter alert prioritization and automation will improve the efficiency and effectiveness of SOC analysts. Enhanced collaboration and information sharing between organizations will foster a more resilient collective defense, enabling SOCs to

stay ahead of emerging threats. Expanding SOC capabilities to include the monitoring and protection of Internet of Things (IoT) and Operational Technology (OT) environments will further solidify their role as critical defenders in the increasingly interconnected and complex cybersecurity ecosystem.

Future Scop of Security Information and Event Management (SIEM

The future scope of Security Information and Event Management (SIEM) in cybersecurity is set to be profoundly influenced by the integration of artificial intelligence (AI) and machine learning (ML). These technologies will transform SIEM systems, enabling them to analyze vast amounts of data in real-time and identify patterns and anomalies that may signify security threats. AI-driven automation will enhance threat detection capabilities and reduce the time needed for manual analysis, allowing for quicker response to incidents. Predictive analytics, powered by ML, will enable SIEM to foresee potential threats by analyzing historical data and trends, thereby facilitating a proactive approach to cybersecurity.

Another significant development in the future of SIEM is its enhanced support for cloud and hybrid environments. As organizations increasingly move their operations to the cloud, SIEM solutions will need to evolve to provide comprehensive visibility and protection across these diverse infrastructures. Cloud-native SIEM solutions will be designed to integrate seamlessly with various cloud services, ensuring consistent monitoring and threat detection regardless of where data resides. Additionally, SIEM systems will need to manage security across multi-cloud environments, ensuring that security policies and threat detection mechanisms are uniformly applied across different cloud platforms, thereby maintaining a robust security posture.

Advanced analytics will play a crucial role in the future of SIEM, offering deeper insights into security events. Behavioral analytics will enable SIEM to understand and establish baselines for normal user behavior, making it easier to detect suspicious activities that deviate from these norms. User and Entity Behavior Analytics (UEBA) will be instrumental in identifying insider threats and compromised accounts by analyzing the behavior of users and entities within the network. Furthermore, real-time integration of threat intelligence feeds will enhance SIEM's ability to correlate internal security events with external threat data, providing a more comprehensive view of the threat landscape and enabling more informed decision-making.

The integration of SIEM with other security tools and platforms will be another key aspect of its future development. SIEM systems will increasingly integrate with Endpoint Detection and Response (EDR), Network Detection and Response (NDR), and Security Orchestration, Automation, and Response (SOAR) platforms. This integration will create a unified security ecosystem, allowing for better correlation of events, streamlined workflows, and enhanced incident response capabilities. Automation within SIEM will further improve incident response by enabling automated playbooks and response actions, ensuring quick containment and mitigation of threats.

Compliance and reporting will also be crucial areas of focus for the future of SIEM. As regulatory requirements continue to evolve, SIEM systems will need to provide robust compliance reporting and auditing capabilities. Advanced reporting features will help organizations meet regulatory requirements and demonstrate compliance with industry standards. Additionally, SIEM solutions will enhance data privacy and protection measures, ensuring that sensitive information is safeguarded against unauthorized access. By addressing these evolving needs, SIEM systems will play a critical role in helping organizations maintain a strong and compliant security posture in an increasingly complex and dynamic cyber threat environment.

Topics explored :-

CEH,100 Soc Tools, Incident Investigation, Mobile hacking, Web Application, Hacking Security Analyst.

Tools explored :-

WPSCAN, Zap Proxy, Wireshark, Nikto , NMAP, Qradar, Metaspoilt, Amass