Mokgari kekana

Task 1.

Threat Identification

- **Sample network**.
- We are going to consider a small e-commerce company as our sample organization, which goes by the name “TechGadgets Online”, TechGadgets sells electronic devices and gadgets through an online platform and here I have explained the sample of the network system setup in detail.

**Sample Network/System Setup for TechGadgets Online:**

**Network Infrastructure:**

- Local Area Network (LAN) connecting all internal devices, including workstations and servers.
- Wi-Fi network for employees and guests accessing the internet within the office premises.
- Dedicated servers for hosting the company website, managing inventory, and processing customer orders.
- Firewalls and routers to control incoming and outgoing network traffic.

**Software and Applications:**

- E-commerce platform for managing online product listings, customer orders, and payments.
- Customer Relationship Management (CRM) software for tracking customer interactions and managing customer data.
- Accounting software for financial transactions and bookkeeping.
- Secure Socket Layer (SSL) certificates to ensure secure communication on the website.

**Endpoints:**

- Desktop computers for employees handling customer service, inventory management, and administrative tasks.
- Laptops for sales representatives who may need to access the system remotely.
- Mobile devices for management and employees requiring on-the-go access to company resources.

**E-commerce Website Components:**

- Front-end web servers handling customer requests and displaying the online store interface.
- Back-end servers managing inventory, processing transactions, and storing customer data securely.
- Payment gateway integration for processing online transactions securely.
- Content Delivery Network (CDN) for efficient content delivery and website performance.

**Data Storage:**

- Database servers storing customer information, product details, and transaction records.
- Regular backups of critical data to prevent data loss in case of system failures or cyber incidents.

Here I have listed the potential threats and vulnerabilities within the TechGadgets system.

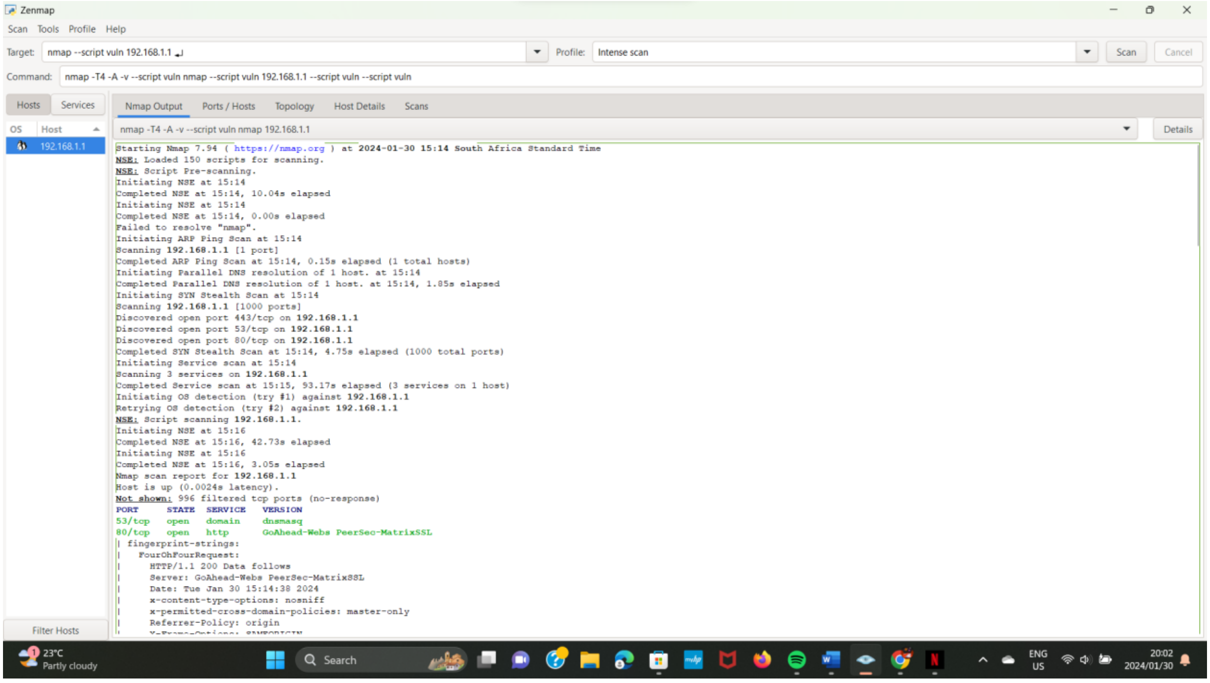
Mokgari kekana

## Threat and Vulnerability Identification:

1. **Denial of Service (DoS) Attacks:**
   - Threat: Malicious attempts to overwhelm the website servers, making it unavailable to legitimate customers.
   - Vulnerability: Lack of effective DoS protection measures and traffic monitoring.
2. **Payment Fraud:**
   - Threat: Unauthorized access to customer payment information or fraudulent transactions.
   - Vulnerability: Weaknesses in payment processing security, inadequate encryption, or compromised payment gateways.
3. **Data Breach:**
   - Threat: Unauthorized access leading to the exposure of customer and financial data.
   - Vulnerability: Inadequate encryption, weak access controls, or unsecured database configurations.
4. **Website Defacement:**
   - Threat: Hackers altering the appearance of the company website to damage its reputation.
   - Vulnerability: Weak website security, outdated content management system (CMS), or poor password practices.
5. **Supply Chain Risks:**
   - Threat: Compromised software or hardware components from suppliers affecting the integrity of the company's systems.
   - Vulnerability: Lack of vendor security assessments, insufficiently secured supply chain channels, or reliance on outdated technologies.
6. **Employee Negligence:**
   - Threat: Unintentional actions by employees leading to security incidents.
   - Vulnerability: Lack of employee training on security best practices, weak password policies, or insufficient user awareness.
7. **Insecure APIs:**
   - Threat: Exploitation of vulnerabilities in APIs used for integration with third-party services.
   - Vulnerability: Poorly secured APIs, lack of regular security audits, or insufficient authentication mechanisms.

## 2. Vulnerability scanning

Mokgari kekana



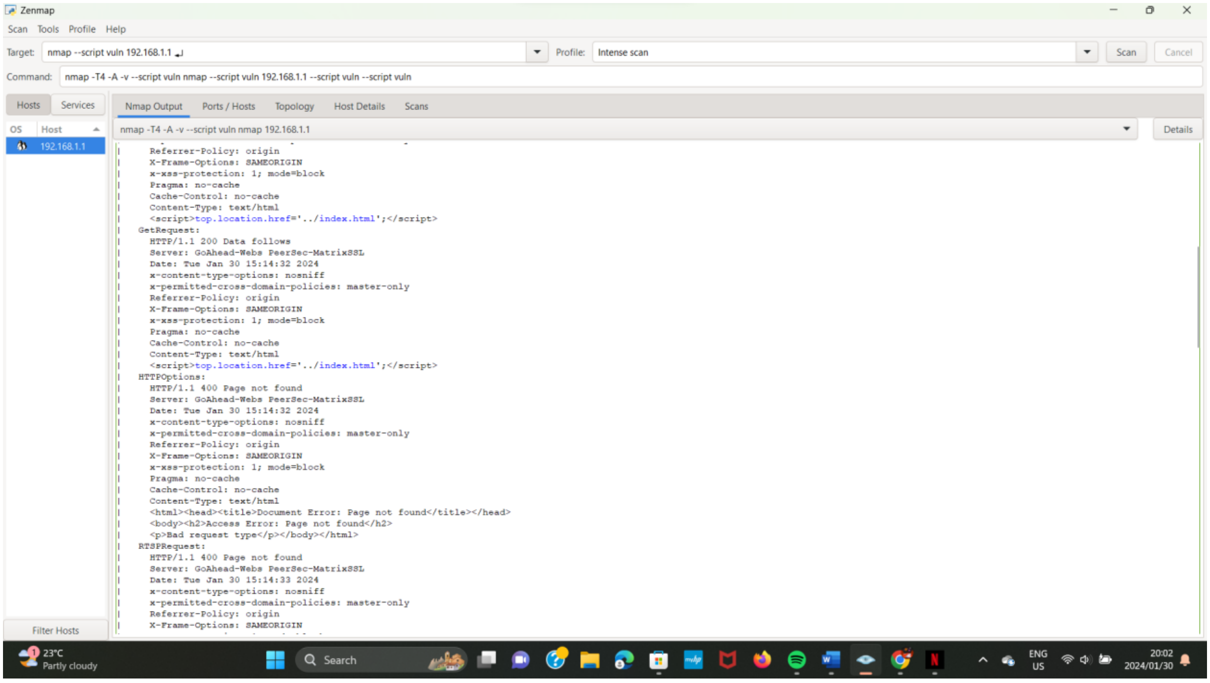
Target: nmap --script vuln 192.168.1.1

Profile: Intense scan

Command: nmap -T4 -A -v --script vuln nmap --script vuln 192.168.1.1 --script vuln --script vuln

nmap -T4 -A -v --script vuln nmap 192.168.1.1

```
Starting Nmap 7.94 ( https://nmap.org ) at 2024-01-30 15:14 South Africa Standard Time
NSE: Loaded 150 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 15:14
Completed NSE at 15:14, 10.04s elapsed
Initiating NSE at 15:14
Completed NSE at 15:14, 0.00s elapsed
Failed to resolve "nmap".
Initiating ARP Ping Scan at 15:14
Scanning 192.168.1.1 [1 port]
Completed ARP Ping Scan at 15:14, 0.15s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 15:14
Completed Parallel DNS resolution of 1 host. at 15:14, 1.85s elapsed
Initiating SYN Stealth Scan at 15:14
Scanning 192.168.1.1 [1000 ports]
Discovered open port 443/tcp on 192.168.1.1
Discovered open port 53/tcp on 192.168.1.1
Discovered open port 80/tcp on 192.168.1.1
Completed SYN Stealth Scan at 15:14, 4.75s elapsed (1000 total ports)
Initiating Service scan at 15:14
Scanning 3 services on 192.168.1.1
Completed Service scan at 15:15, 93.17s elapsed (3 services on 1 host)
Initiating OS detection (try #1) against 192.168.1.1
Retrying OS detection (try #2) against 192.168.1.1
NSE: Script scanning 192.168.1.1.
Initiating NSE at 15:16
Completed NSE at 15:16, 42.73s elapsed
Initiating NSE at 15:16
Completed NSE at 15:16, 3.05s elapsed
Nmap scan report for 192.168.1.1
Host is up (0.0024s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE  SERVICE  VERSION
53/tcp   open   domain   dnsmasq
80/tcp   open   http     GoAhead-Webs PeerSec-MatrixSSL
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.1 200 Data follows
|     Server: GoAhead-Webs PeerSec-MatrixSSL
|     Date: Tue Jan 30 15:14:38 2024
|     x-content-type-options: nosniff
|     x-permitted-cross-domain-policies: master-only
|     Referrer-Policy: origin
```



Target: nmap --script vuln 192.168.1.1

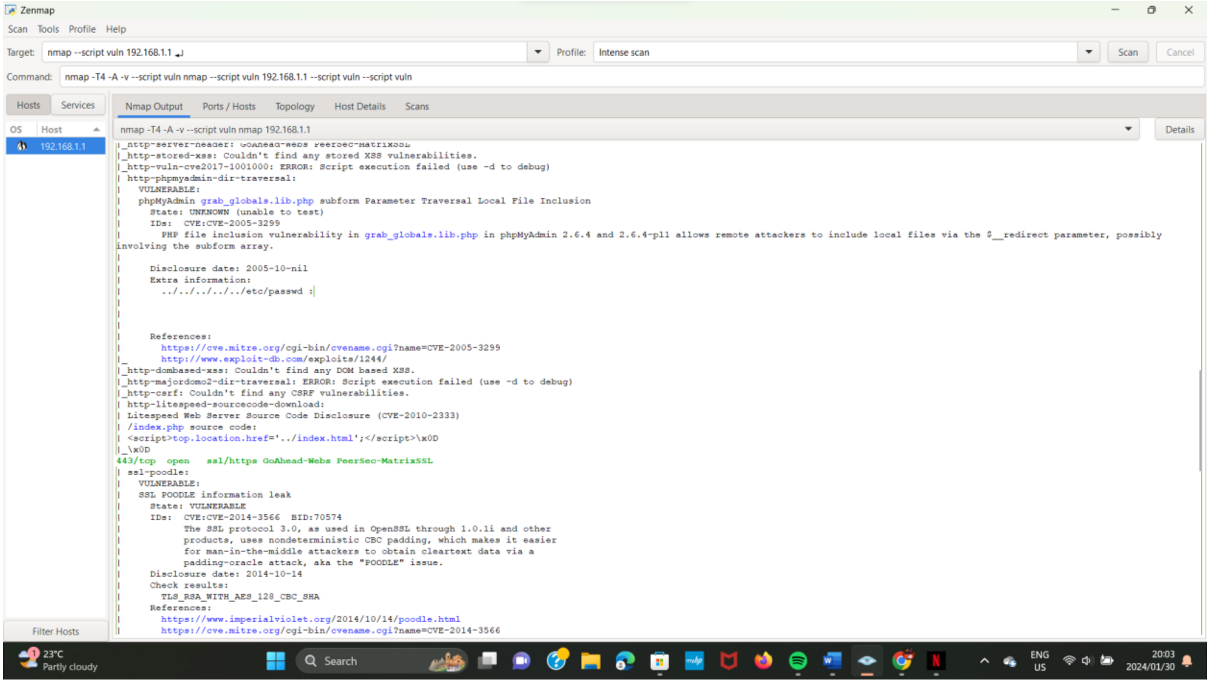
Profile: Intense scan

Command: nmap -T4 -A -v --script vuln nmap --script vuln 192.168.1.1 --script vuln --script vuln

nmap -T4 -A -v --script vuln nmap 192.168.1.1

```
|     Referrer-Policy: origin
|     X-Frame-Options: SAMEORIGIN
|     x-xss-protection: 1; mode=block
|     Pragma: no-cache
|     Cache-Control: no-cache
|     Content-Type: text/html
|     <script>top.location.href='../index.html';</script>
|   GetRequest:
|     HTTP/1.1 200 Data follows
|     Server: GoAhead-Webs PeerSec-MatrixSSL
|     Date: Tue Jan 30 15:14:32 2024
|     x-content-type-options: nosniff
|     x-permitted-cross-domain-policies: master-only
|     Referrer-Policy: origin
|     X-Frame-Options: SAMEORIGIN
|     x-xss-protection: 1; mode=block
|     Pragma: no-cache
|     Cache-Control: no-cache
|     Content-Type: text/html
|     <script>top.location.href='../index.html';</script>
|   HTTPOptions:
|     HTTP/1.1 400 Page not found
|     Server: GoAhead-Webs PeerSec-MatrixSSL
|     Date: Tue Jan 30 15:14:32 2024
|     x-content-type-options: nosniff
|     x-permitted-cross-domain-policies: master-only
|     Referrer-Policy: origin
|     X-Frame-Options: SAMEORIGIN
|     x-xss-protection: 1; mode=block
|     Pragma: no-cache
|     Cache-Control: no-cache
|     Content-Type: text/html
|     <html><head><title>Document Error: Page not found</title></head>
|     <body><h2>Access Error: Page not found</h2>
|     <p>Bad request type</p></body></html>
|   RTSPRequest:
|     HTTP/1.1 400 Page not found
|     Server: GoAhead-Webs PeerSec-MatrixSSL
|     Date: Tue Jan 30 15:14:33 2024
|     x-content-type-options: nosniff
|     x-permitted-cross-domain-policies: master-only
|     Referrer-Policy: origin
|     X-Frame-Options: SAMEORIGIN
```

Mokgari kekana



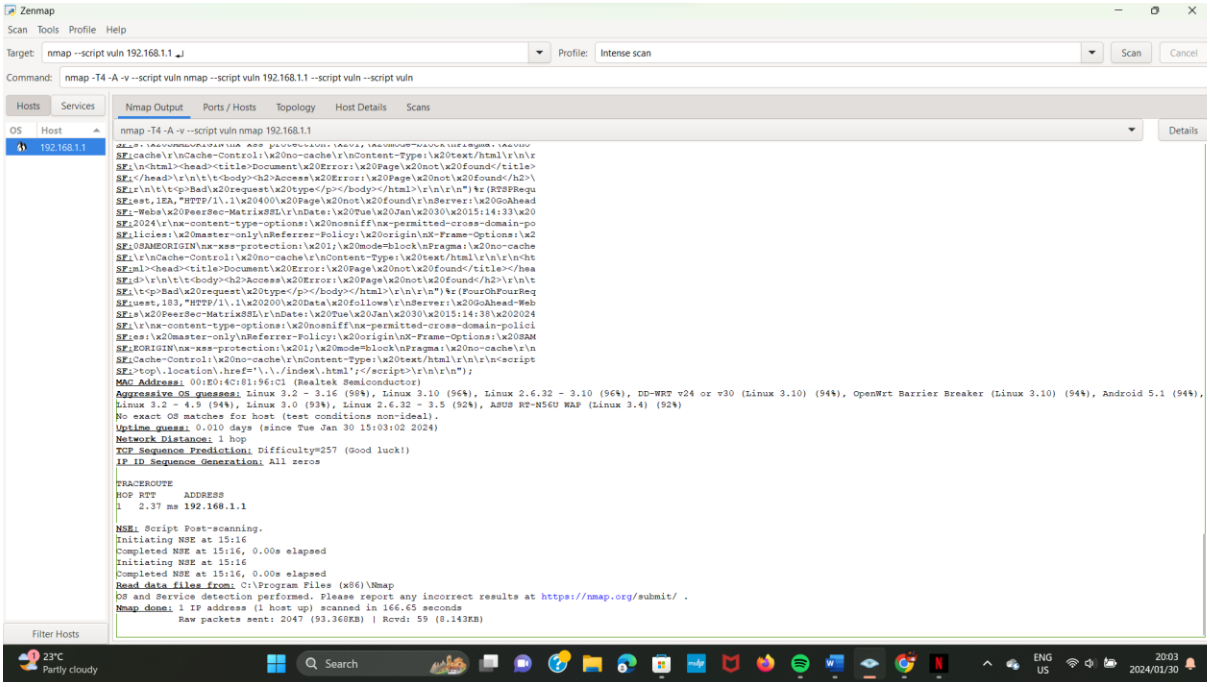
Target: nmap --script vuln 192.168.1.1

Profile: Intense scan

Command: nmap -T4 -A -v --script vuln nmap --script vuln 192.168.1.1 --script vuln --script vuln

nmap -T4 -A -v --script vuln nmap 192.168.1.1

```
|_http-server-header: GoAhead-Webs PeerSec-MatrixSSL
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-vuln-cve2017-1001000: ERROR: Script execution failed (use -d to debug)
| http-phpmyadmin-dir-traversal:
|   VULNERABLE:
|   phpMyAdmin grab_globals.lib.php subform Parameter Traversal Local File Inclusion
|     State: UNKNOWN (unable to test)
|     IDs:  CVE:CVE-2005-3299
|       PHP file inclusion vulnerability in grab_globals.lib.php in phpMyAdmin 2.6.4 and 2.6.4-pl1 allows remote attackers to include local files via the $__redirect parameter, possibly
involving the subform array.
|
|     Disclosure date: 2005-10-nil
|     Extra information:
|       ../../../../../etc/passwd :
|
|
|
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2005-3299
|_      http://www.exploit-db.com/exploits/1244/
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-majordomo2-dir-traversal: ERROR: Script execution failed (use -d to debug)
|_http-csrf: Couldn't find any CSRF vulnerabilities.
| http-litespeed-sourcecode-download:
| Litespeed Web Server Source Code Disclosure (CVE-2010-2333)
| /index.php source code:
| <script>top.location.href='../index.html';</script>\x0D
|_\x0D
443/tcp  open   ssl/https GoAhead-Webs PeerSec-MatrixSSL
| ssl-poodle:
|   VULNERABLE:
|   SSL POODLE information leak
|     State: VULNERABLE
|     IDs:  CVE:CVE-2014-3566  BID:70574
|           The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and other
|           products, uses nondeterministic CBC padding, which makes it easier
|           for man-in-the-middle attackers to obtain cleartext data via a
|           padding-oracle attack, aka the "POODLE" issue.
|     Disclosure date: 2014-10-14
|     Check results:
|       TLS_RSA_WITH_AES_128_CBC_SHA
|     References:
|       https://www.imperialviolet.org/2014/10/14/poodle.html
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566
```



Target: nmap --script vuln 192.168.1.1

Profile: Intense scan

Command: nmap -T4 -A -v --script vuln nmap --script vuln 192.168.1.1 --script vuln --script vuln

nmap -T4 -A -v --script vuln nmap 192.168.1.1

```
SF:s:\x20SAMEORIGIN\nx-xss-protection:\x201;\x20mode=block\nPragma:\x20no
SF:cache\r\nCache-Control:\x20no-cache\r\nContent-Type:\x20text/html\r\n\r
SF:\n<html><head><title>Document\x20Error:\x20Page\x20not\x20found</title>
SF:</head>\r\n\t\t<body><h2>Access\x20Error:\x20Page\x20not\x20found</h2>\
SF:r\n\t\t<p>Bad\x20request\x20type</p></body></html>\r\n\r\n")%r(RTSPRequ
SF:est,1EA,"HTTP/1\.1\x20400\x20Page\x20not\x20found\r\nServer:\x20GoAhead
SF:-Webs\x20PeerSec-MatrixSSL\r\nDate:\x20Tue\x20Jan\x2030\x2015:14:33\x20
SF:2024\r\nx-content-type-options:\x20nosniff\nx-permitted-cross-domain-po
SF:licies:\x20master-only\nReferrer-Policy:\x20origin\nX-Frame-Options:\x2
SF:0SAMEORIGIN\nx-xss-protection:\x201;\x20mode=block\nPragma:\x20no-cache
SF:\r\nCache-Control:\x20no-cache\r\nContent-Type:\x20text/html\r\n\r\n<ht
SF:ml><head><title>Document\x20Error:\x20Page\x20not\x20found</title></hea
SF:d>\r\n\t\t<body><h2>Access\x20Error:\x20Page\x20not\x20found</h2>\r\n\t
SF:\t<p>Bad\x20request\x20type</p></body></html>\r\n\r\n")%r(FourOhFourReq
SF:uest,183,"HTTP/1\.1\x20200\x20Data\x20follows\r\nServer:\x20GoAhead-Web
SF:s\x20PeerSec-MatrixSSL\r\nDate:\x20Tue\x20Jan\x2030\x2015:14:38\x202024
SF:\r\nx-content-type-options:\x20nosniff\nx-permitted-cross-domain-polici
SF:es:\x20master-only\nReferrer-Policy:\x20origin\nX-Frame-Options:\x20SAM
SF:EORIGIN\nx-xss-protection:\x201;\x20mode=block\nPragma:\x20no-cache\r\n
SF:Cache-Control:\x20no-cache\r\nContent-Type:\x20text/html\r\n\r\n<script
SF:>top\.location\.href='\.\./index\.html';</script>\r\n\r\n");
MAC Address: 00:E0:4C:81:96:C1 (Realtek Semiconductor)
Aggressive OS guesses: Linux 3.2 - 3.16 (98%), Linux 3.10 (96%), Linux 2.6.32 - 3.10 (96%), DD-WRT v24 or v30 (Linux 3.10) (94%), OpenWrt Barrier Breaker (Linux 3.10) (94%), Android 5.1 (94%),
Linux 3.2 - 4.9 (94%), Linux 3.0 (93%), Linux 2.6.32 - 3.5 (92%), ASUS RT-N56U WAP (Linux 3.4) (92%)
No exact OS matches for host (test conditions non-ideal).
Uptime guess: 0.010 days (since Tue Jan 30 15:03:02 2024)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=257 (Good luck!)
IP ID Sequence Generation: All zeros

TRACEROUTE
HOP RTT     ADDRESS
1   2.37 ms 192.168.1.1

NSE: Script Post-scanning.
Initiating NSE at 15:16
Completed NSE at 15:16, 0.00s elapsed
Initiating NSE at 15:16
Completed NSE at 15:16, 0.00s elapsed
Read data files from: C:\Program Files (x86)\Nmap
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 166.65 seconds
           Raw packets sent: 2047 (93.368KB) | Rcvd: 59 (8.143KB)
```

## These are the identified vulnerabilities, severity and potential impact.

Based on the intense scanning that was conducted for the vulnerabilities using Nmap tool this is what I found and detected.

1. **SSL POODLE Vulnerability (CVE-2014-3566):**

    - **Severity:** Medium

    - **Potential Impact:** SSL POODLE is a vulnerability that affects the SSL 3.0 protocol. It allows attackers to exploit the way encryption is handled in SSL 3.0, potentially leading to the disclosure of sensitive information. While SSL 3.0 is considered obsolete, the impact depends on the usage of this protocol on the target system.

2. **phpMyAdmin Local File Inclusion (CVE-2005-3299):**
    - **Severity:** Unknown (untested)
    - **Potential Impact:** This vulnerability involves a PHP file inclusion issue in phpMyAdmin 2.6.4 and 2.6.4-pl1. If exploitable, it could allow remote attackers to include local files, potentially leading to unauthorized access and disclosure of sensitive information. However, the scanner couldn't confirm the existence of this vulnerability, and further testing is needed to determine its severity and impact.

## 3 Risk Analysis

1. **SL POODLE Vulnerability (CVE-2014-3566):**
    - **Potential Risk:**
        - SSL POODLE targets the SSL 3.0 protocol, which is considered obsolete and insecure. If the system is still using SSL 3.0, an attacker could potentially exploit this vulnerability to perform a padding oracle attack, leading to the disclosure of sensitive information.
    - **Mitigation:**
        - Disable SSL 3.0 and use more secure protocols like TLS.
        - Implement proper TLS configurations to prevent downgrade attacks.
2. **phpMyAdmin Local File Inclusion (CVE-2005-3299):**
    - **Potential Risk:**
        - If the phpMyAdmin version on the system is affected, and the local file inclusion vulnerability is exploitable, an attacker might include local files, leading to unauthorized access and potential disclosure of sensitive information.
    - **Mitigation:**
        - Ensure that phpMyAdmin is updated to a secure version.
        - Regularly monitor for security updates and apply patches promptly.
        - Consider restricting access to phpMyAdmin based on IP or using additional authentication mechanisms.

Likelihood of exploitation and Prioritization

- **SSL POODLE Vulnerability (CVE-2014-3566):**
  **Severity: High**
  **Likelihood of Exploitation: Moderate**
- **Reasoning:**
  High severity due to the potential for information disclosure through a well-known and exploitable SSL vulnerability.
  Moderate likelihood as it depends on the specific configuration and usage of SSL 3.0 on the system.

Mokgari kekana

- **phpMyAdmin Local File Inclusion (CVE-2005-3299):**
  **Severity: Medium**
  **Likelihood of Exploitation: Low to Medium**
- **Reasoning:**
  Medium severity due to the potential for unauthorized access and disclosure of sensitive information.
  Likelihood depends on the specific version of phpMyAdmin and whether the local file inclusion vulnerability is exploitable.

Mitigation strategies

- **SSL POODLE Vulnerability (CVE-2014-3566):**
  **Mitigation:**
  **Disable SSL 3.0:** Since SSL 3.0 is vulnerable to POODLE, disable it on your web server. This can be done by modifying server configurations.
  **Upgrade to TLS:** Transition to a more secure protocol like TLS (Transport Layer Security). Ensure that only secure TLS versions are enabled.
  **Web Server Configuration:** Adjust web server settings to prioritize secure ciphers and disable vulnerable ones.

- **phpMyAdmin Local File Inclusion (CVE-2005-3299):**
  **Mitigation:**
  **Update phpMyAdmin:** If possible, upgrade to the latest version of phpMyAdmin. This may address known vulnerabilities and improve overall security.
  **Access Controls:** Implement proper access controls to restrict access to phpMyAdmin. Limit the number of users who have access to the application.
  **Regular Security Audits:** Conduct regular security audits to identify and address any new vulnerabilities that may arise.

  **General Recommendations:**
- **Regular Patching:** Keep all software, including the operating system, web server, and applications, up to date with the latest security patches.
- **Network Segmentation:** Implement network segmentation to limit the impact of a potential breach. Isolate critical systems from less secure parts of the network.
  **Monitoring and Incident Response:**
- **Implement Monitoring:** Set up continuous monitoring for unusual or suspicious activities on the network and systems.
- **Incident Response Plan:** Develop and regularly update an incident response plan. This includes procedures for quickly responding to and containing security incidents.
  **User Education:**
- **Phishing Awareness:** Educate users about phishing attacks and social engineering tactics. Many vulnerabilities are exploited through user actions, such as clicking on malicious links.

Recommendations to address identified risks effectively.

1. **regular Security Audits and Vulnerability Assessments:**

   - Conduct regular security audits and vulnerability assessments to proactively identify and address potential risks. This includes both automated scans and manual checks.

2. **Patch Management:**

    - Establish a robust patch management process to ensure that all software, including the operating system, web server, and applications, is kept up to date with the latest security patches.

3. **Network Segmentation:**

    - Implement network segmentation to isolate critical systems from less secure parts of the network. This helps contain the impact of a potential security breach.

4. **Access Controls:**

    - Enforce strong access controls and least privilege principles. Limit user access to systems and applications to only what is necessary for their roles.

5. **Encryption Best Practices:**

    - Follow encryption best practices, ensuring that secure protocols and ciphers are used. Disable outdated and vulnerable protocols (e.g., SSL 3.0) and prioritize the use of secure encryption standards.

6. **User Education and Awareness:**

    - Provide ongoing security awareness training for users to recognize and report phishing attempts, social engineering, and other security threats. Educated users are a critical line of defense.

7. **Incident Response Plan:**

    - Develop, test, and regularly update an incident response plan. Clearly define roles and responsibilities, and establish procedures for detecting, responding to, and recovering from security incidents.

8. **Continuous Monitoring:**

    - Implement continuous monitoring of network and system activities. Use intrusion detection systems and security information and event management (SIEM) tools to detect and respond to suspicious behavior.

9. **Web Application Security:**

    - Regularly assess and secure web applications. Use web application firewalls (WAFs) to filter and monitor HTTP traffic between a web application and the internet, protecting against various attacks.

10. **Regular Backups:**

    - Implement regular data backups and ensure that backup and recovery procedures are tested. This helps mitigate the impact of data loss due to security incidents.

11. **Regular Security Training for IT Staff:**

    - Ensure that IT staff responsible for system administration and security are well-trained and updated on the latest security practices and technologies.

12. **Engage Security Professionals:**

- Consider engaging external security professionals or penetration testing services to conduct thorough security assessments and provide recommendations for improving the overall security posture.