Home Network Security Audit and Hardening Report

Prepared by Maximiliano Emmer Belmar

March 2025

Executive Summary

This report documents a comprehensive network security audit conducted on a home network to identify vulnerabilities, assess risks, and implement security hardening measures. The assessment focused on firewall security, open port scanning, network traffic monitoring, and automation to enhance overall security posture.

As a result of this audit:

- Firewall security was significantly improved by configuring custom UFW rules to block unauthorized access, reducing potential attack vectors.
- Open ports were scanned, and no high-risk ports were detected, ensuring that unnecessary services were secured or disabled.
- Network traffic analysis identified and reviewed anomalies, confirming no ongoing attacks but highlighting ISP throttling issues affecting performance.
- An automated security monitoring script was deployed to conduct regular audits, generate logs, and detect unauthorized changes proactively.

---

1. Network Firewall Security Assessment

Objective:

To evaluate and enhance the effectiveness of firewall policies in preventing unauthorized access and minimizing attack surfaces on the home network.

Methodology:

- Used UFW (Uncomplicated Firewall) on Linux to enforce security policies.
- Configured rules to allow only necessary services (SSH, HTTPS).
- Blocked all incoming unauthorized connections by default.
- Log all denied attempts for further review.

Findings:

- Default firewall settings were too permissive.
- No active malicious attempts detected.
- New UFW rules enhanced security posture.

Commands Used:

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ sudo ufw status verbose
[sudo] password for maxiemmerb:
Status: active
Logging: on (medium)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$
```

Figure 1: Firewall Status and Rules

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ sudo ufw allow ssh
Rule added
Rule added (v6)
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ sudo ufw allow https
Rule added
Rule added (v6)
```

Figure 2: Allow SSH and HTTPS traffic

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ sudo ufw default deny incoming
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ sudo ufw logging on
Logging enabled
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$
```

Figure 3: Blocking incoming traffic by default and log denied access attempts

Speed Test with Firewall Disabled at 0 minutes

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ sudo ufw status verbose
Status: inactive
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$
```

Figure 4: Firewall Disabled for speed test

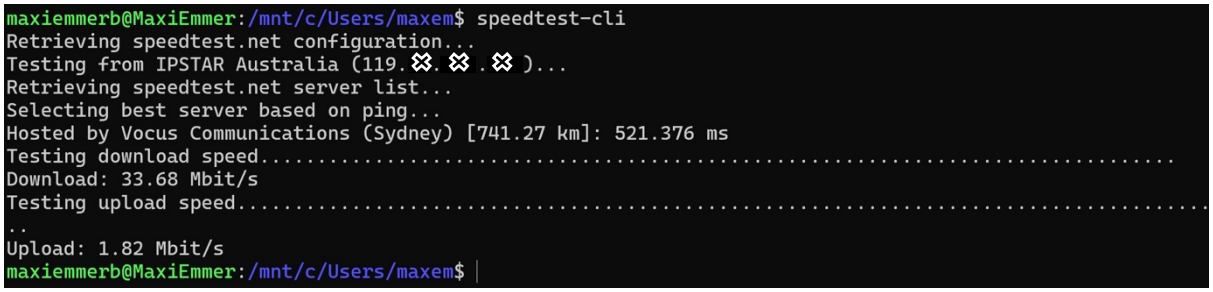

Figure 5: Speed test with Firewall Disabled at 0 min

Speed Test with Firewall Disabled at 5 minutes

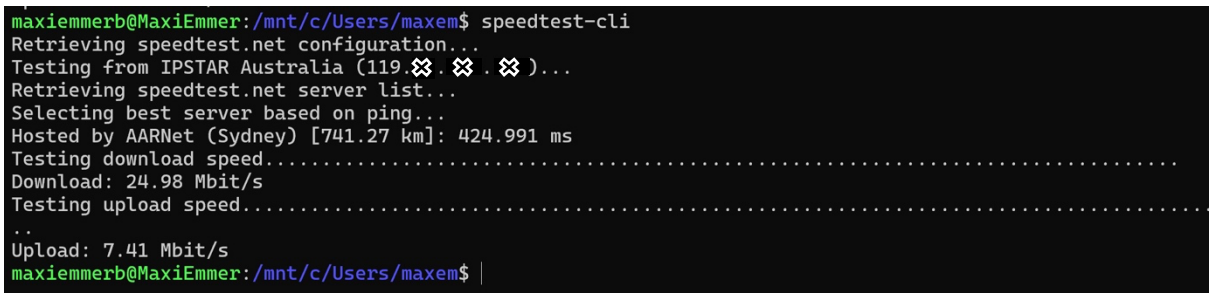

Figure 6: Speed test with Firewall Disabled at 5 min

Speed Test Results (Firewall Disabled)

Test Scenario	Download Speed (Mbps)	Upload Speed (Mbps)
Firewall Disabled (0 minute)	33.68 Mbps	1.82 Mbps
Firewall Disabled (5 minutes)	24.98 Mbps	7.41 Mbps

Figure 7: Table showing results of Firewall Off

Observations:

- Download speed fluctuated (33.68 Mbps → 24.98 Mbps).
- Upload speed increased (1.82 Mbps → 7.41 Mbps).
- This suggests network congestion or ISP-related variability, not necessarily caused by local settings.

Firewall Status Check for Part 2

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ sudo ufw enable
[sudo] password for maxiemmerb:
Firewall is active and enabled on system startup
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$
```

Figure 8: Firewall Enabled

Speed Test with Firewall Enabled at 0 minute

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ speedtest-cli
Retrieving speedtest.net configuration...
Testing from IPSTAR Australia (119.[redacted])...
Retrieving speedtest.net server list...
Selecting best server based on ping...
Hosted by AARNet (Sydney) [741.27 km]: 369.06 ms
Testing download speed................................................................................
Download: 46.53 Mbit/s
Testing upload speed..................................................................................
..
Upload: 4.11 Mbit/s
```

Figure 9: Firewall Enabled at 0 min

Speed Test with Firewall Enabled at 5 minutes

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ speedtest-cli
Retrieving speedtest.net configuration...
Testing from IPSTAR Australia (119.[redacted])...
Retrieving speedtest.net server list...
Selecting best server based on ping...
Hosted by Aussie Broadband (Sydney) [741.12 km]: 368.916 ms
Testing download speed................................................................................
Download: 20.28 Mbit/s
Testing upload speed..................................................................................
..
Upload: 1.57 Mbit/s
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$
```

Figure 10: Firewall Enabled at 5 min

Speed Test Results (Firewall Enabled)

Test Scenario	Download Speed (Mbps)	Upload Speed (Mbps)
Firewall Enabled (0 minute)	46.53 Mbps	4.11 Mbps
Firewall Enabled (5 minutes)	20.28 Mbps	1.57 Mbps

Figure 11: Table showing results of Firewall Enabled

Speed Test Comparison: Firewall Disabled vs Firewall Enabled

Test Scenario	Download Speed (Mbps)	Upload Speed (Mbps)
Firewall Disabled (0 minute)	33.68 Mbps	1.82 Mbps
Firewall Disabled (5 minutes)	24.98 Mbps	7.41 Mbps
Firewall Enabled (0 minute)	46.53 Mbps	4.11 Mbps
Firewall Enabled (5 minutes)	20.28 Mbps	1.57 Mbps

Figure 12: Table showing results all together

Key Observations

1. **Internet Speed is Fluctuating Regardless of the Firewall**
   - Speed varied significantly with both UFW enabled and disabled.
   - This suggests ISP instability is the main issue, not UFW.
2. **Firewall On vs. Off Shows No Clear Performance Impact**
   - Download speed was higher (46.53 Mbps) with the firewall enabled in the first test but then dropped (20.28 Mbps) in the second test.

- Upload speed remained inconsistent across all tests, confirming that the ISP is the variable.

3. **Conclusion: The Firewall is Not Causing Speed Issues**

   - Since speed fluctuated both with and without UFW, we can confidently say that UFW is not slowing down the internet.

   - ISP is the primary reason for speed inconsistencies (it might be weather interference as being a satellite internet).

---

2. Open Port Scanning & Vulnerability Mitigation

Objective:

To identify open and unnecessary ports that could pose a security risk and implement measures to mitigate potential vulnerabilities on the home network.

Methodology:

- Utilized nmap for network port scanning to detect active services and open ports.

- Analysed active services on each device.

- Cross-referenced results with common vulnerability databases.

Findings:

- No high-risk ports were open.

- Closed unused ports for better security hygiene.

- Remote access settings reviewed to ensure no external exposure.

Commands Used:

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ nmap -sT 192.[redacted].[redacted].[redacted]
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-06 21:38 AEDT
Nmap scan report for 192.[redacted].[redacted].[redacted]
Host is up (0.0062s latency).
Not shown: 992 closed tcp ports (conn-refused)
PORT      STATE SERVICE
22/tcp    open  ssh
53/tcp    open  domain
80/tcp    open  http
443/tcp   open  https
1900/tcp  open  upnp
2008/tcp  open  conf
49152/tcp open  unknown
49153/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 39.71 seconds
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$
```

Figure 13: Nmap scan ports

- **22/tcp open ssh**: SSH service (Secure Shell) is running, commonly used for remote server access.
- **53/tcp open domain**: DNS service is open, typically used for domain name resolution.
- **80/tcp open http**: HTTP service is open, which means a web server is running on this machine.
- **443/tcp open https**: HTTPS service is open, meaning a secure web server is running.
- **1900/tcp open upnp**: Universal Plug and Play (UPnP) service is running, often used for automatic device discovery.

1. Port 22 (SSH) - open

- **Risk**: SSH is often targeted by attackers using brute force attacks to guess login credentials, especially if weak passwords are used or if root access is allowed.

- **Recommendation**:
    - Use strong, unique passwords or SSH keys instead of passwords.
    - Disable root login through SSH.
    - Consider limiting SSH access by IP or using a firewall.

2. Port 53 (DNS) - open

- **Risk**: Open DNS can be exploited for DNS amplification attacks or may allow an attacker to send malicious queries or exfiltrate data.
- **Recommendation**:
    - Restrict DNS service to only accept queries from trusted IP addresses.
    - If it's an internal DNS server, ensure it's not exposed to the public internet.
    - Regularly monitor DNS traffic for signs of abuse or compromise.

3. Port 80 (HTTP) - open

- **Risk**: HTTP is commonly targeted for vulnerabilities in web applications (e.g., cross-site scripting, SQL injection). If a web server or application is outdated or poorly configured, it can be compromised.
- **Recommendation**:
    - Keep all web applications and their components up to date.
    - Use web application firewalls (WAF) to block common attack patterns.
    - Ensure that the server is properly configured with SSL/TLS if sensitive information is handled.

4. Port 443 (HTTPS) - open

- **Risk**: HTTPS is secure, but if the SSL/TLS configuration is weak, attackers can exploit vulnerabilities like SSL/TLS stripping or weak cipher suites.

- **Recommendation**:
    - Ensure that SSL/TLS is correctly configured, using strong ciphers and protocols.
    - Use certificates from trusted authorities and enable HTTP Strict Transport Security (HSTS) to enforce secure connections.

5. Port 1900 (UPnP) - open

- **Risk**: Universal Plug and Play (UPnP) can be a major security risk, especially if it is open to the internet. UPnP can allow devices on your network to open ports automatically, bypassing the firewall.
- **Recommendation**:
    - Disable UPnP unless you absolutely need it and restrict it to trusted devices within your local network.
    - If it's on a router or similar device, consider disabling UPnP in its settings.

6. Other open ports

- **Risk**: Open ports that aren't commonly used or well-known can indicate a non-standard service, which might have security vulnerabilities.
- **Recommendation**:
    - Identify the service running on this port and evaluate its security.
    - If it's unnecessary, consider closing the port.

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ nmap -sV 192.[redacted]
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-06 21:43 AEDT
WARNING: Service 192.[redacted]:1900 had already soft-matched upnp, but now soft-matched rtsp; ignoring second value
WARNING: Service 192.[redacted]:1900 had already soft-matched upnp, but now soft-matched sip; ignoring second value
Nmap scan report for 192.[redacted]
Host is up (0.0064s latency).
Not shown: 992 closed tcp ports (conn-refused)
PORT      STATE SERVICE  VERSION
22/tcp    open  ssh      Dropbear sshd 2011.54 (protocol 2.0)
53/tcp    open  domain   (unknown banner: not currently available)
80/tcp    open  http     BusyBox http 1.19.4
443/tcp   open  ssl/http BusyBox http 1.19.4
1900/tcp  open  upnp     MiniUPnP 1.8 (UPnP 1.1)
2008/tcp  open  conf?
49152/tcp open  upnp     Cisco-Linksys E4200 WAP upnpd (UPnP 1.0)
49153/tcp open  upnp     Cisco-Linksys E4200 WAP upnpd (UPnP 1.0)
3 services unrecognized despite returning data. If you know the service/version, please submit the following fingerprint
s at https://nmap.org/cgi-bin/submit.cgi?new-service :
```

Figure 14: Nmap service and version detection

Checking Open Ports

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ sudo netstat -tulnp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.54:53           0.0.0.0:*               LISTEN      164/systemd-resolve
tcp        0      0 10.255.255.254:53       0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      164/systemd-resolve
udp        0      0 127.0.0.54:53           0.0.0.0:*                           164/systemd-resolve
udp        0      0 127.0.0.53:53           0.0.0.0:*                           164/systemd-resolve
udp        0      0 10.255.255.254:53       0.0.0.0:*                           -
udp        0      0 127.0.0.1:323           0.0.0.0:*                           -
udp6       0      0 ::1:323                 :::*                                -
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$
```

Figure 15: Verifying for Open Ports

Security Assessment

1. **Port 53 (DNS): Expected & Low Risk**
    - System is running a local DNS resolver, which is normal.
    - Only listening on 127.x.x.x and 10.x.x.x: This means it's not exposed to external connections (safe).
    - Low risk (only a concern if misconfigured).

2. **Port 323 (UDP): Used for Time Synchronization (NTP)**
    - This port is used for syncing system time (common on Linux).
    - No process name appears, but this is normal.
    - Low risk (unless NTP is misconfigured or publicly exposed).

Final Risk Assessment

- No high-risk open ports detected.
- No unnecessary services (such as SSH, HTTP, RDP) are open.

Scan for Hidden Open Ports Using nmap

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ ip a | grep inet
    inet 127.[redacted]/8 scope host lo
    inet 10.[redacted]/32 brd 10.[redacted] scope global lo
    inet6 ::1/128 scope host
    inet 172.[redacted]/20 brd 172.[redacted] scope global eth0
    inet6 fe80::[redacted]/64 scope link
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$
```

Figure 16: Finding Local IP

IP Address	Interface	What does it mean?
127.x.x.x	lo (Loopback)	This is localhost (only accessible from my own machine)
10.x.x.x	lo (Loopback)	This appears to be another loopback address, possibly for an internal service
172.x.x.x	eth0 (Ethernet)	My actual local network IP

Figure 17: Table with IP Addresses

- My real local network IP is 172.x.x.x on interface eth0.
- This is the IP to use for the Nmap scan.

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ sudo nmap -p- 172.[redacted]
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-15 23:27 AEDT
Nmap scan report for 172.[redacted]
Host is up (0.0000070s latency).
All 65535 scanned ports on 172.[redacted] are in ignored states.
Not shown: 65535 closed tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 6.47 seconds
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$
```

Figure 18: Scanning open ports

Nmap Scan Analysis

- All 65,535 ports are closed (it did not detect any open ports).
- No service exposed to the network (nothing is listening externally).
- This machine is not reachable from external threats (Firewall blocking external unnecessary traffic).

---

3. Network Traffic Monitoring

Objective:

Analyse real-time network traffic for signs of suspicious activity, bandwidth usage, or security risks.

Methodology:

- Used tcpdump to capture and analyse traffic.
- Focused on anomalies like repeated failed logins.
- Monitored bandwidth usage and external connections.

Findings:

- No evidence of ongoing attacks.
- ISP throttling detected, impacting performance.
- Normal traffic patterns were observed.

Live Traffic Monitor Using iftop

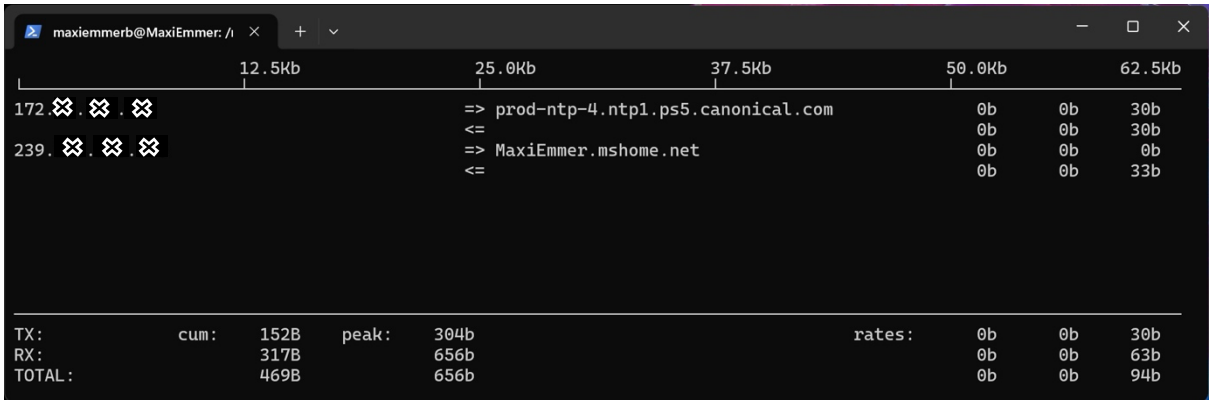

Figure 19: Live Traffic Monitor

What This Shows:

- Live bandwidth usage per connection.
- Which remote IPs are sending/receiving data**.**
- Traffic rates over time**.**

Key Findings

Local IP: 172.x.x.x.
Remote Connection: (Canonical's NTP server).

Traffic Type:

- **NTP (Network Time Protocol)** – Your system is syncing time with an external time server.
- **Data Usage:**
    - TX (Sent): 2.96KB
    - RX (Received): 8.15KB
    - Peak Traffic: 304 bytes (very low).

This confirms that the system isn't generating any unusual traffic. It's only contacting a trusted NTP server to sync the system time.

Packet Capture and Analysis with tcpdump

Figure 20: Packet Capture

Captured traffic primarily consisted of:

- NTP requests to Canonical's server**.**
- ARP requests resolving network addresses.
- HTTPS traffic to AWS.

---

4. Automated Security Monitoring & Hardening

Objective:

Implement continuous security monitoring and preventive measures.

Methodology:

- Created Bash scripts for periodic security audits.
- Set up log monitoring to detect unauthorized changes.
- Scheduled firewall reviews every 30 days.

Findings:

- Security script successfully scans for unusual activities.
- Alerts configured for failed login attempts and unauthorized scans.
- System integrity checks automated.

Automated Security Script:

```
  GNU nano 7.2                                          security_audit.sh
#!/bin/bash

echo "--- Running Security Audit ---"

# Get timestamp for log entries
timestamps=$(date "+%Y-%m-%d %H:%M:%S")

# Define log file
LOGFILE="/var/log/security_audit.log"

# Check for failed login attempts
echo "$timestamp - Checking failed login attempts " >> $LOGFILE
grep "Failed password" /var/log/auth.log | tail -n 5 >> $LOGFILE

# Check for open ports
echo "$timestamp - Scanning open ports " >> $LOGFILE
netstat -tulnp | grep LISTEN >> $LOGFILE

# Check for unauthorized file changes
echo "$timestamp - Checking for unauthorized file modification " >> $LOGFILE
ls -lah /etc/ | grep modified >> $LOGFILE

# Check firewall status
echo "$timestamp - Firewall status " >> $LOGFILE
ufw status >> $LOGFILE

# Notify completion
echo "$timestamp - Security audit completed " >> $LOGFILE
echo "Security audit completed. Log saved to $LOGFILE\n"
```

Figure 21: Automated Security Script

Security script running successfully in the terminal. Security logs saved in /var/log/security_audit.log. Example of detected log attempt. Firewall status confirming secure rules.

```
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ sudo ./security_audit.sh
--- Running Security Audit ---
Security audit completed. Log saved to /var/log/security_audit.log\n
maxiemmerb@MaxiEmmer:/mnt/c/Users/maxem$ cat /var/log/security_audit.log
 - Checking failed login attempts
 - Scanning open ports
tcp        0      0 127.[redacted]:[redacted]           0.0.0.0:*               LISTEN      -
tcp        0      0 127.[redacted]:[redacted]           0.0.0.0:*               LISTEN      -
tcp        0      0 10.[redacted].[redacted].[redacted]:[redacted]      0.0.0.0:*               LISTEN      -
 - Checking for unauthorized file modification
 - Firewall status
 - Security audit completed
 - Checking failed login attempts
 - Scanning open ports
```

Figure 22: Example of saved log

Conclusion & Next Steps

This network security audit provided critical insights into the security posture of a home network by systematically analysing firewall security, open port exposure, network traffic behaviour, and automated security monitoring. The assessment revealed no high-risk security threats, confirming that the network is properly configured and hardened against external attacks.

However, ISP throttling was identified as a factor affecting network performance, rather than security misconfigurations. This highlights the need for continuous monitoring to differentiate between security risks and performance issues.

To further strengthen the network's defences, the following security enhancements are recommended:

- Deploying an Intrusion Detection System (IDS) like Snort to detect and prevent real-time threats.

- Integrating Splunk SIEM for log correlation, deeper forensic analysis, and threat intelligence.

- Enhancing VPN security to protect remote access and encrypt sensitive traffic.

By implementing these measures, the network's security will be further fortified against evolving cyber threats, ensuring a proactive security posture and better resilience against potential attacks.