NETWORK PLAYBOOK

CCDC Network Defense Guide

FIRST 15 MINUTES

Minute 0-5: Map the Network

```
# Quick host discovery
nmap -sn 192.168.1.0/24 --min-rate 1000

# Or without nmap
for i in $(seq 1 254); do
    ping -c 1 -W 1 192.168.1.$i &
done 2>/dev/null | grep "bytes from"

# ARP table (local segment)
arp -a

# Windows
arp -a
Get-NetNeighbor
```

Minute 5-10: Document Topology

Create a quick reference:

```
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■  NETWORK TOPOLOGY                                        ■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■  Subnet: 192.168.1.0/24     Gateway: 192.168.1.1        ■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
■  IP            ■ Hostname     ■ Role         ■ Services    ■
■  192.168.1.10  ■ DC01         ■ Domain Ctrl  ■ DNS,AD,53   ■
■  192.168.1.20  ■ WEB01        ■ Web Server   ■ HTTP,443    ■
■  192.168.1.30  ■ MAIL01       ■ Mail Server  ■ SMTP,25     ■
■  192.168.1.40  ■ DB01         ■ Database     ■ MySQL,3306  ■
■  192.168.1.50  ■ FTP01        ■ File Server  ■ FTP,21      ■
■  192.168.1.1   ■ FW01         ■ Firewall     ■ N/A         ■
■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
```

Minute 10-15: Verify Critical Ports

```
# Quick port scan on critical hosts
nmap -p 22,80,443,3389,445,53,25,110,3306 192.168.1.0/24 -T4

# Verify specific service
nc -zv 192.168.1.20 80
nc -zv 192.168.1.10 53
```

FIREWALL RULE TEMPLATES

Default Deny Policy

```
# Linux (iptables)
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -i lo -j ACCEPT

# Windows
Set-NetFirewallProfile -DefaultInboundAction Block -DefaultOutboundAction Allow
```

Allow by Service

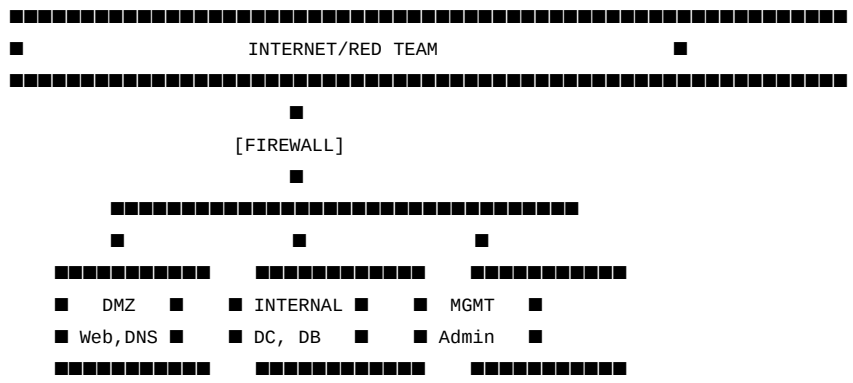
```
# Web Server
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT

# DNS
iptables -A INPUT -p udp --dport 53 -j ACCEPT
iptables -A INPUT -p tcp --dport 53 -j ACCEPT

# Email
iptables -A INPUT -p tcp --dport 25 -j ACCEPT
iptables -A INPUT -p tcp --dport 110 -j ACCEPT
iptables -A INPUT -p tcp --dport 143 -j ACCEPT

# SSH (restrict to admin subnet)
iptables -A INPUT -p tcp --dport 22 -s 192.168.1.0/24 -j ACCEPT

# Database (internal only)
iptables -A INPUT -p tcp --dport 3306 -s 192.168.1.0/24 -j ACCEPT
```

Block Attacker (Speed Drill - 45 seconds)

```
# Linux - Block IP
iptables -I INPUT -s 10.10.10.50 -j DROP

# Linux - Block subnet
iptables -I INPUT -s 10.10.10.0/24 -j DROP

# Windows - Block IP
New-NetFirewallRule -DisplayName "Block_Attacker" -Direction Inbound -RemoteAddress 10.10.10.50 -Action Block

# Windows - Block subnet
New-NetFirewallRule -DisplayName "Block_Range" -Direction Inbound -RemoteAddress 10.10.10.0/24 -Action Block

# pf (BSD/macOS)
echo "block in quick from 10.10.10.50" >> /etc/pf.conf && pfctl -f /etc/pf.conf
```

NETWORK SEGMENTATION

Ideal CCDC Segmentation



Quick Segmentation Rules

```
# Block DMZ to Internal (except specific)
iptables -A FORWARD -s 192.168.1.0/24 -d 192.168.2.0/24 -j DROP

# Allow only specific traffic
iptables -A FORWARD -s 192.168.1.20 -d 192.168.2.40 -p tcp --dport 3306 -j ACCEPT
```

TRAFFIC ANALYSIS

Capture Traffic

```
# Capture all traffic on interface
tcpdump -i eth0 -w capture.pcap

# Capture specific host
tcpdump -i eth0 host 10.10.10.50 -w attacker.pcap

# Capture specific port
tcpdump -i eth0 port 4444 -w suspicious.pcap

# Live view (no file)
tcpdump -i any -nn port not 22
```

Identify Suspicious Patterns

Beaconing (C2)

```
# Look for regular interval connections
tcpdump -i eth0 -nn | grep -E "10\.10\.10\.[0-9]+"

# Check connection frequency
netstat -an | grep ESTABLISHED | awk '{print $5}' | cut -d: -f1 | sort | uniq -c | sort -rn
```

Port Scanning

```
# Many SYN packets from single source
tcpdump -i eth0 'tcp[tcpflags] == tcp-syn' -nn | head -100
```

Data Exfiltration

```
# Large outbound transfers
iftop -i eth0  # Real-time bandwidth

# DNS tunneling indicators
tcpdump -i eth0 port 53 -nn | grep -E "[a-z0-9]{30,}"
```

Quick Connection Review

```
# All established connections
netstat -antup | grep ESTABLISHED

# Windows
Get-NetTCPConnection | Where-Object {$_.State -eq "Established"} |
    Select LocalAddress,LocalPort,RemoteAddress,RemotePort,OwningProcess

# Connections to suspicious ports
netstat -an | grep -E ":4444|:5555|:1337|:31337"
```

DNS SECURITY

Verify DNS Service

```
# Test resolution
nslookup google.com your-dns-server
dig @your-dns-server example.com

# Check DNS is only answering appropriate queries
dig @your-dns-server axfr example.com  # Should fail (zone transfer)
```

DNS Hardening

```
# Disable zone transfers (BIND)
# /etc/bind/named.conf.options
options {
    allow-transfer { none; };
    recursion no;  # Unless needed
};
```

COMMON ATTACK PATTERNS

ARP Spoofing Detection

```
# Check for duplicate MACs
arp -a | awk '{print $4}' | sort | uniq -d

# Static ARP entries (prevents spoofing)
arp -s 192.168.1.1 aa:bb:cc:dd:ee:ff
```

Port Scanning Detection

```
# High rate SYN from single source = scan
# Log with iptables
```

```
iptables -A INPUT -p tcp --syn -m limit --limit 1/s --limit-burst 3 -j LOG --log-prefix "SYN_SCAN: "
iptables -A INPUT -p tcp --syn -m limit --limit 1/s --limit-burst 3 -j DROP
```

Lateral Movement Detection

```
# Unusual RDP/SSH from internal hosts
tcpdump -i eth0 port 3389 -nn
tcpdump -i eth0 port 22 -nn

# Check for PsExec (SMB on 445 + specific patterns)
tcpdump -i eth0 port 445 -nn
```

EMERGENCY PROCEDURES

Isolate Compromised Host

```
# Block all traffic to/from host
iptables -I INPUT -s 192.168.1.50 -j DROP
iptables -I OUTPUT -d 192.168.1.50 -j DROP
iptables -I FORWARD -s 192.168.1.50 -j DROP
iptables -I FORWARD -d 192.168.1.50 -j DROP

# Or at switch level (if you have access)
# Disable port / move to quarantine VLAN
```

Rate Limit Suspicious Traffic

```
# Limit connections per IP
iptables -A INPUT -p tcp --dport 80 -m connlimit --connlimit-above 50 -j DROP

# Limit new connections
iptables -A INPUT -p tcp --syn -m limit --limit 10/s --limit-burst 20 -j ACCEPT
```

MONITORING COMMANDS

Real-time Traffic

```
# Bandwidth per connection
iftop -i eth0

# Connections per second
watch -n 1 'netstat -an | grep ESTABLISHED | wc -l'

# Top talkers
tcpdump -i eth0 -nn -q | awk '{print $3}' | cut -d. -f1-4 | sort | uniq -c | sort -rn | head
```

Connection States

```
# Current connection states
netstat -an | awk '/^tcp/ {print $6}' | sort | uniq -c

# TIME_WAIT flood check
```

```
netstat -an | grep TIME_WAIT | wc -l
```

NETWORK DEFENDER MANTRA

> *""Map first."*
>
> *"Default deny."*
>
> *"Allow explicit."*
>
> *"Block fast.""*