Implementation of IPSec Tunnel Between VMs Using LibreSwan

Network and Systems Security - Winter 2025 Exercise 3, Part 2 Report

Table of Contents

Introduction

This report details the implementation of an IPSec/IKE tunnel between two gateway virtual machines (VMs) using LibreSwan, as specified in Part 2 of the Winter 2025 Networks and Systems Security II Exercise 3. The setup comprises four VMs: two gateway VMs (VM2 and VM3) running Ubuntu, and two endpoint VMs (VM1 and VM4) running Alpine Linux. The primary objective is to establish a secure, encrypted tunnel between VM2 and VM3, enabling VM1 to communicate with VM4 without modifying their routing tables. Authentication is achieved using X.509 certificates, and the traffic egressing VM3 to VM4 bears VM3's source address. This report outlines the configuration process, validation steps, and security analysis, supported by illustrative screenshots.

Network Architecture

The network topology consists of four VMs configured in VMware Workstation with the following specifications:

- **VM1 (Alpine Linux - Client)**
  - Interface 1: NAT (Internet access)
  - Interface 2: Host-only (10.0.0.0/24 network)
  - IP: 10.0.0.10/24
- **VM2 (Ubuntu - Gateway 1)**
  - Interface 1: Host-only (10.0.0.0/24 network)
  - Interface 2: Host-only (20.0.0.0/24 network)
  - IPs: 10.0.0.1/24, 20.0.0.1/24
- **VM3 (Ubuntu - Gateway 2)**

    - Interface 1: Host-only (20.0.0.0/24 network)
    - Interface 2: Host-only (30.0.0.0/24 network)
    - IPs: 20.0.0.2/24, 30.0.0.1/24

- **VM4 (Alpine Linux - Server)**

    - Interface 1: NAT (Internet access)
    - Interface 2: Host-only (30.0.0.0/24 network)
    - IP: 30.0.0.10/24

Figure 1: Virtual network topology showing gateway VMs and protected subnets

---

Implementation Steps

1. VMware Network Setup

1. Configured three host-only networks in VMware Virtual Network Editor:
    - vmnet1: 10.0.0.0/24
    - vmnet2: 20.0.0.0/24
    - vmnet3: 30.0.0.0/24
2. Assigned network adapters to each VM as per the topology.

Figure 2: VMware network adapter configuration

2. VM1 Configuration (Alpine Linux)

- Installed Alpine Linux and configured networking:

```
cat > /etc/network/interfaces << EOF
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet static
    address 10.0.0.10
    netmask 255.255.255.0
    gateway 10.0.0.1
EOF
/etc/init.d/networking restart
ip route add 30.0.0.0/24 via 10.0.0.1
```

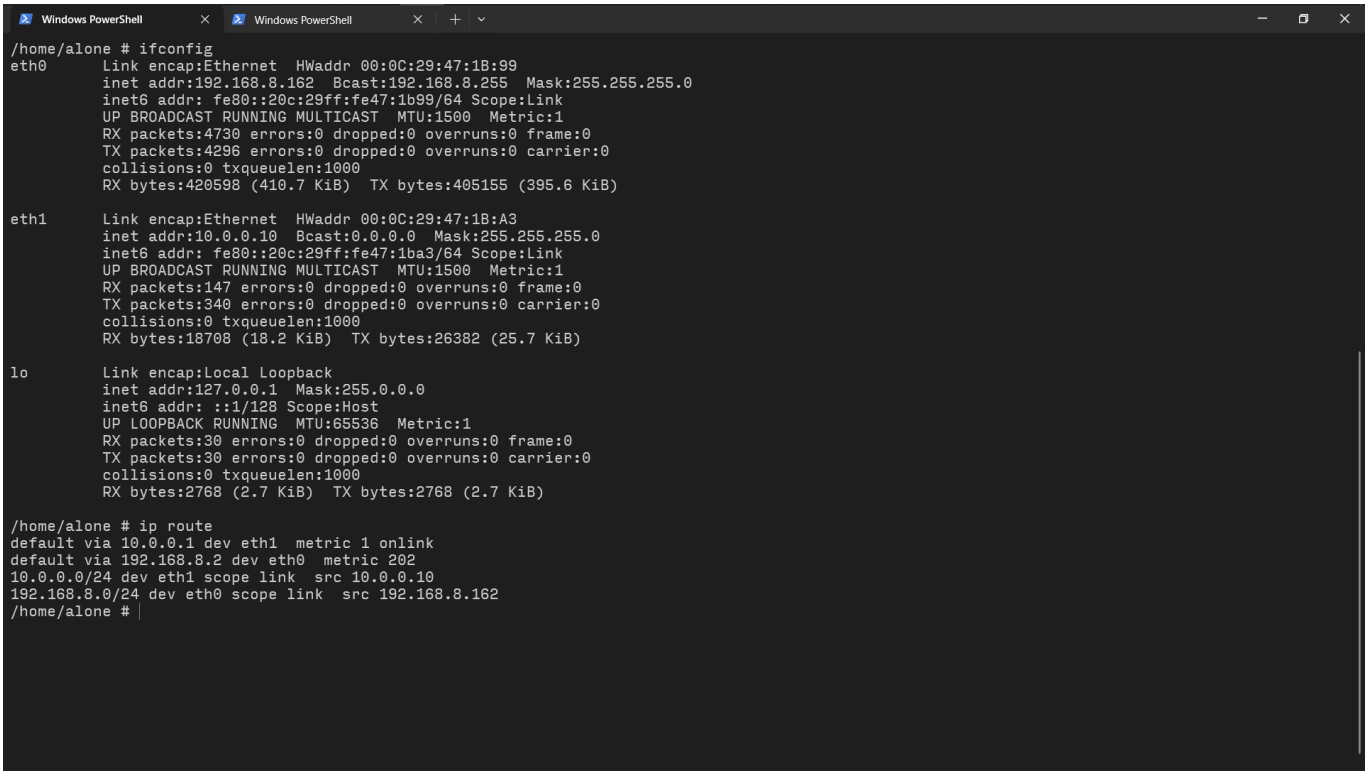
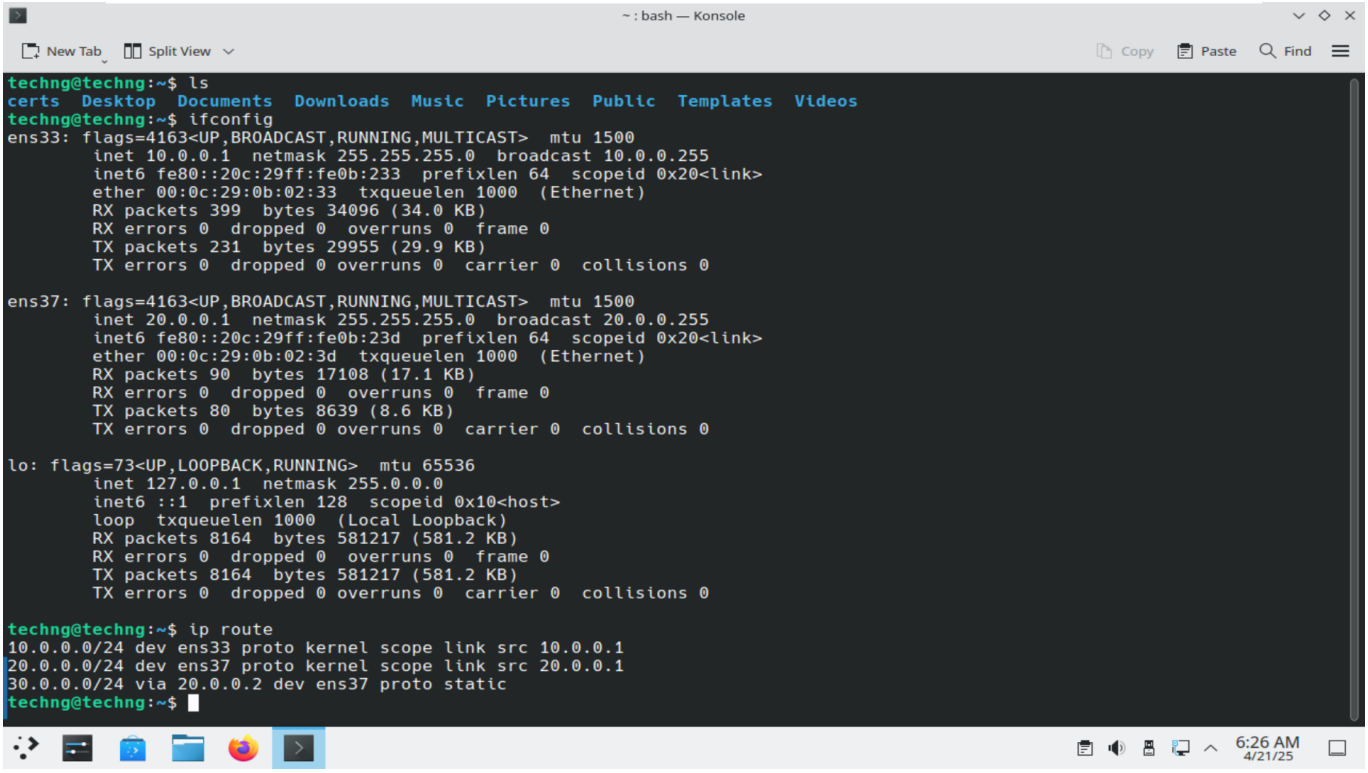


Figure 3: VM1 network configuration

3. VM2 Configuration (Ubuntu - Gateway 1)

- Configured network interfaces:

```
sudo cat > /etc/netplan/01-netcfg.yaml << EOF
network:
  version: 2
  renderer: networkd
  ethernets:
    ens33:
      addresses:
        - 10.0.0.1/24
      dhcp4: no
    ens37:
      addresses:
        - 20.0.0.1/24
      dhcp4: no
      routes:
        - to: 30.0.0.0/24
          via: 20.0.0.2
EOF
sudo netplan apply
```



- Enabled IP forwarding:

```
echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward
echo "net.ipv4.ip_forward = 1" | sudo tee -a /etc/sysctl.conf
sudo sysctl -p
```

- Installed LibreSwan and OpenSSL:

```
sudo apt update
sudo apt install -y libreswan openssl
```

- Generated certificates:

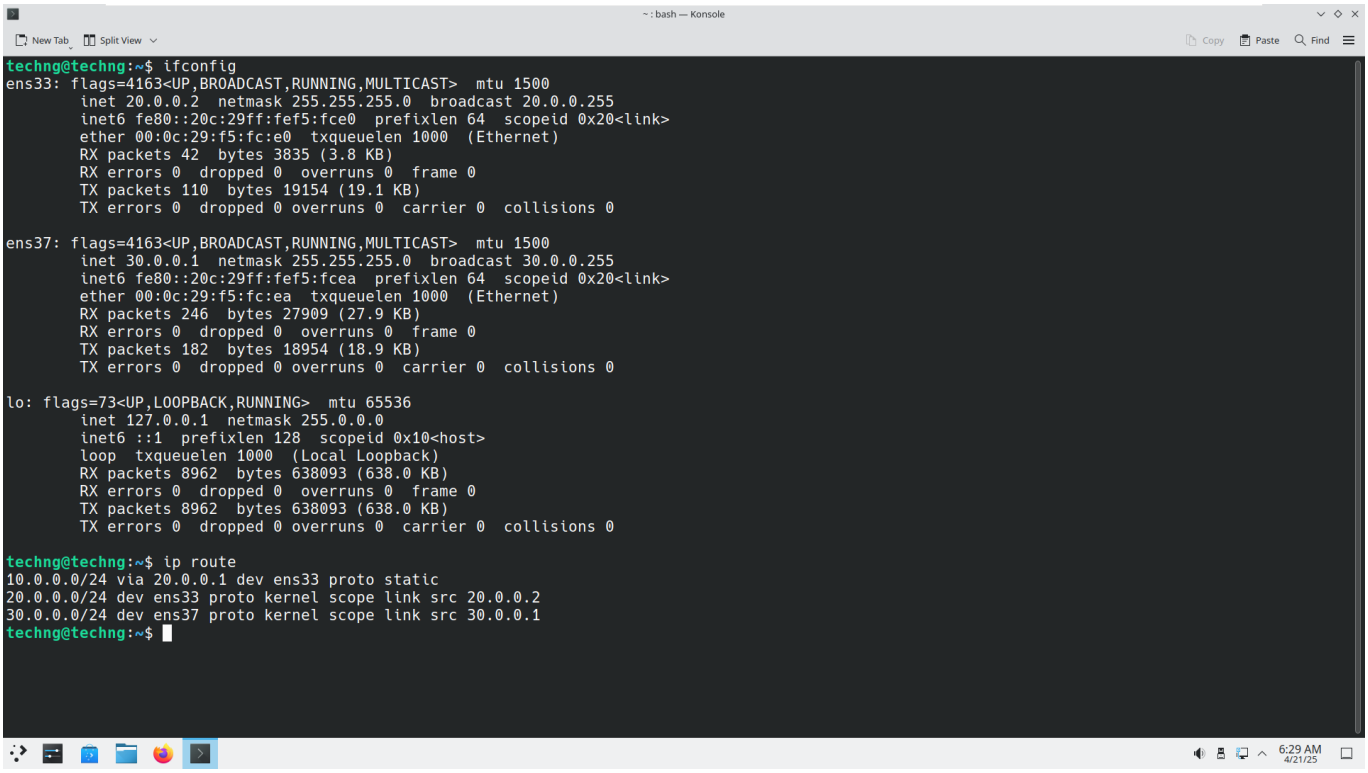
```
sudo mkdir -p /etc/ipsec.d/certs /etc/ipsec.d/private /etc/ipsec.d/cacerts
sudo openssl genrsa -out /etc/ipsec.d/private/ca-key.pem 4096
sudo openssl req -new -x509 -key /etc/ipsec.d/private/ca-key.pem -out
/etc/ipsec.d/cacerts/ca-cert.pem -days 3650 -subj "/CN=VPN CA"
sudo chmod 600 /etc/ipsec.d/private/ca-key.pem
sudo openssl genrsa -out /etc/ipsec.d/private/vm2-key.pem 4096
sudo openssl req -new -key /etc/ipsec.d/private/vm2-key.pem -out
/tmp/vm2.csr -subj "/CN=VM2"
sudo openssl x509 -req -in /tmp/vm2.csr -CA /etc/ipsec.d/cacerts/ca-cert.pem
-CAkey /etc/ipsec.d/private/ca-key.pem -CAcreateserial -out
/etc/ipsec.d/certs/vm2-cert.pem -days 1825
sudo rm /tmp/vm2.csr
```

Figure 4: Certificate generation process on VM2

4. VM3 Configuration (Ubuntu - Gateway 2)

- Configured network interfaces and NAT:

```
sudo cat > /etc/netplan/01-netcfg.yaml << EOF
network:
  version: 2
  renderer: networkd
  ethernets:
    ens33:
      addresses:
        - 20.0.0.2/24
      dhcp4: no
      routes:
        - to: 10.0.0.0/24
          via: 20.0.0.1
    ens37:
      addresses:
        - 30.0.0.1/24
      dhcp4: no
EOF
sudo netplan apply
sudo iptables -t nat -A POSTROUTING -s 10.0.0.0/24 -d 30.0.0.0/24 -j
MASQUERADE
sudo apt install -y iptables-persistent
```



- Installed LibreSwan and configured certificates (transferred CA cert from VM2):

```
sudo apt update
sudo apt install -y libreswan openssl
sudo mkdir -p /etc/ipsec.d/certs /etc/ipsec.d/private /etc/ipsec.d/cacerts
sudo openssl genrsa -out /etc/ipsec.d/private/vm3-key.pem 4096
sudo openssl req -new -key /etc/ipsec.d/private/vm3-key.pem -out
/tmp/vm3.csr -subj "/CN=VM3"
# On VM2: Sign VM3's CSR
sudo openssl x509 -req -in /tmp/vm3.csr -CA /etc/ipsec.d/cacerts/ca-cert.pem
-CAkey /etc/ipsec.d/private/ca-key.pem -CAcreateserial -out /tmp/vm3-
cert.pem -days 1825
```

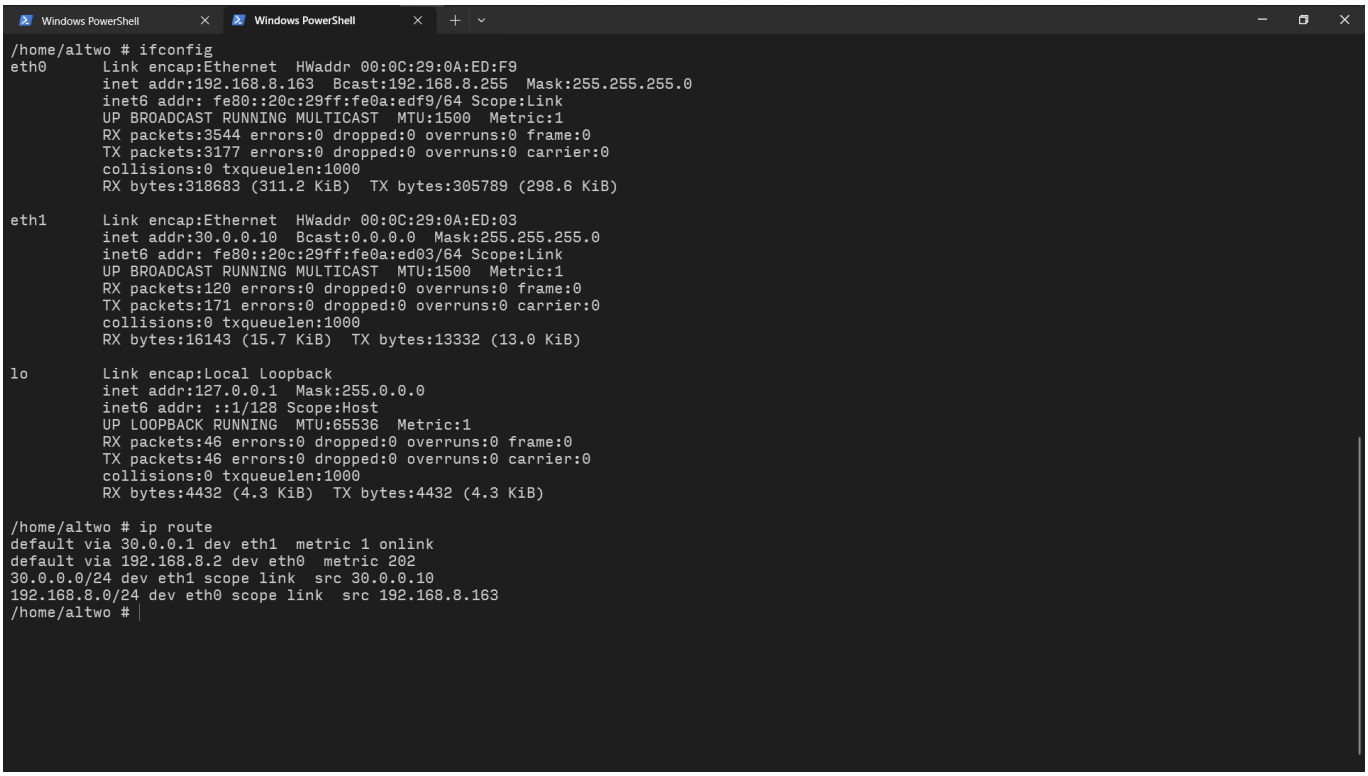
5. VM4 Configuration (Alpine Linux)

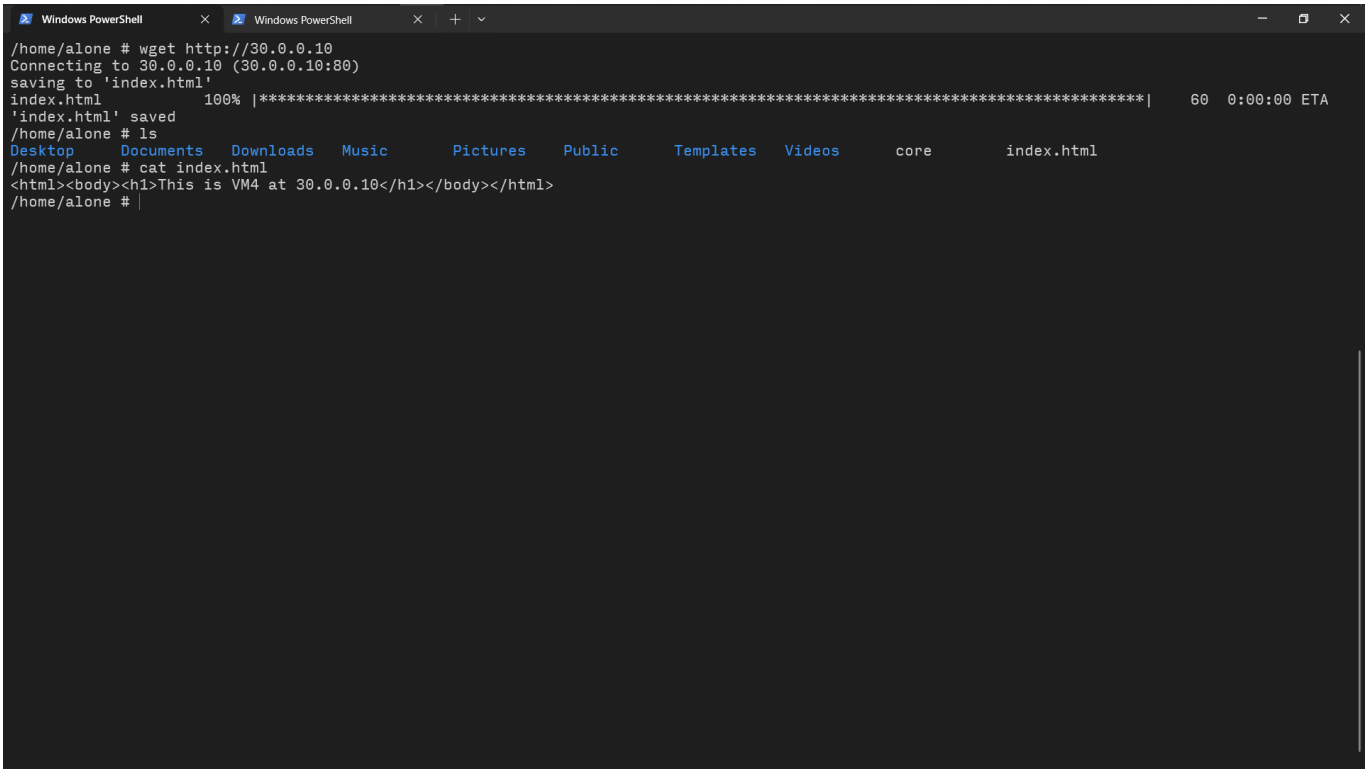
- Configured networking and web server:

```
cat > /etc/network/interfaces << EOF
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet static
    address 30.0.0.10
    netmask 255.255.255.0
    gateway 30.0.0.1
EOF
/etc/init.d/networking restart
apk add lighttpd
rc-service lighttpd start
echo "<html><body><h1>This is VM4 at 30.0.0.10</h1></body></html>" >
/var/www/localhost/htdocs/index.html
```





Configuration Details

VM2 IPSec Configuration

```
sudo cat > /etc/ipsec.conf << EOF
config setup
    logfile=/var/log/pluto.log
    logtime=yes
    logappend=yes
    plutodebug=all
```

```
    dumpdir=/var/run/pluto/
    virtual-private=%v4:10.0.0.0/24,%v4:30.0.0.0/24

conn gateway-tunnel
    authby=rsasig
    left=20.0.0.1
    leftsubnet=10.0.0.0/24
    leftcert=vm2-cert.pem
    leftsendcert=always
    leftid="CN=VM2"
    right=20.0.0.2
    rightsubnet=30.0.0.0/24
    rightid="CN=VM3"
    ike=aes256-sha2_256-modp2048!
    esp=aes256-sha2_256!
    auto=start
    type=tunnel
EOF
sudo cat > /etc/ipsec.secrets << EOF
: RSA vm2-key.pem
EOF
sudo systemctl restart ipsec
```

Figure 5: IPSec configuration on VM2

VM3 IPSec Configuration

```
sudo cat > /etc/ipsec.conf << EOF
config setup
    logfile=/var/log/pluto.log
    logtime=yes
    logappend=yes
    plutodebug=all
    dumpdir=/var/run/pluto/
    virtual-private=%v4:10.0.0.0/24,%v4:30.0.0.0/24

conn gateway-tunnel
    authby=rsasig
    left=20.0.0.2
    leftsubnet=30.0.0.0/24
    leftcert=vm3-cert.pem
    leftsendcert=always
    leftid="CN=VM3"
    right=20.0.0.1
    rightsubnet=10.0.0.0/24
    rightid="CN=VM2"
    ike=aes256-sha2_256-modp2048!
    esp=aes256-sha2_256!
    auto=start
    type=tunnel
```

```
EOF
sudo cat > /etc/ipsec.secrets << EOF
: RSA vm3-key.pem
EOF
sudo systemctl restart ipsec
```

Testing and Validation

1. IPSec Tunnel Status

- Verified tunnel establishment:

```
sudo ipsec status
sudo ipsec verify
```

Figure 6: IPSec tunnel status verification

2. Traffic Capture

- Captured IKE negotiation and ESP packets using Wireshark on the 20.0.0.0/24 network:
  - Filter: `isakmp` for IKE
  - Filter: `esp` for encrypted packets

Figure 7: Successful IKEv2 key exchange process

Figure 8: Wireshark capture showing encrypted ESP packets

3. End-to-End Connectivity

- Tested connectivity from VM1 to VM4:

```
ping 30.0.0.10
busybox wget -O - http://30.0.0.10
```

- Confirmed VM3's IP as the source address on VM4.

Figure 9: End-to-end connectivity test results

Troubleshooting

Common Issues and Resolutions

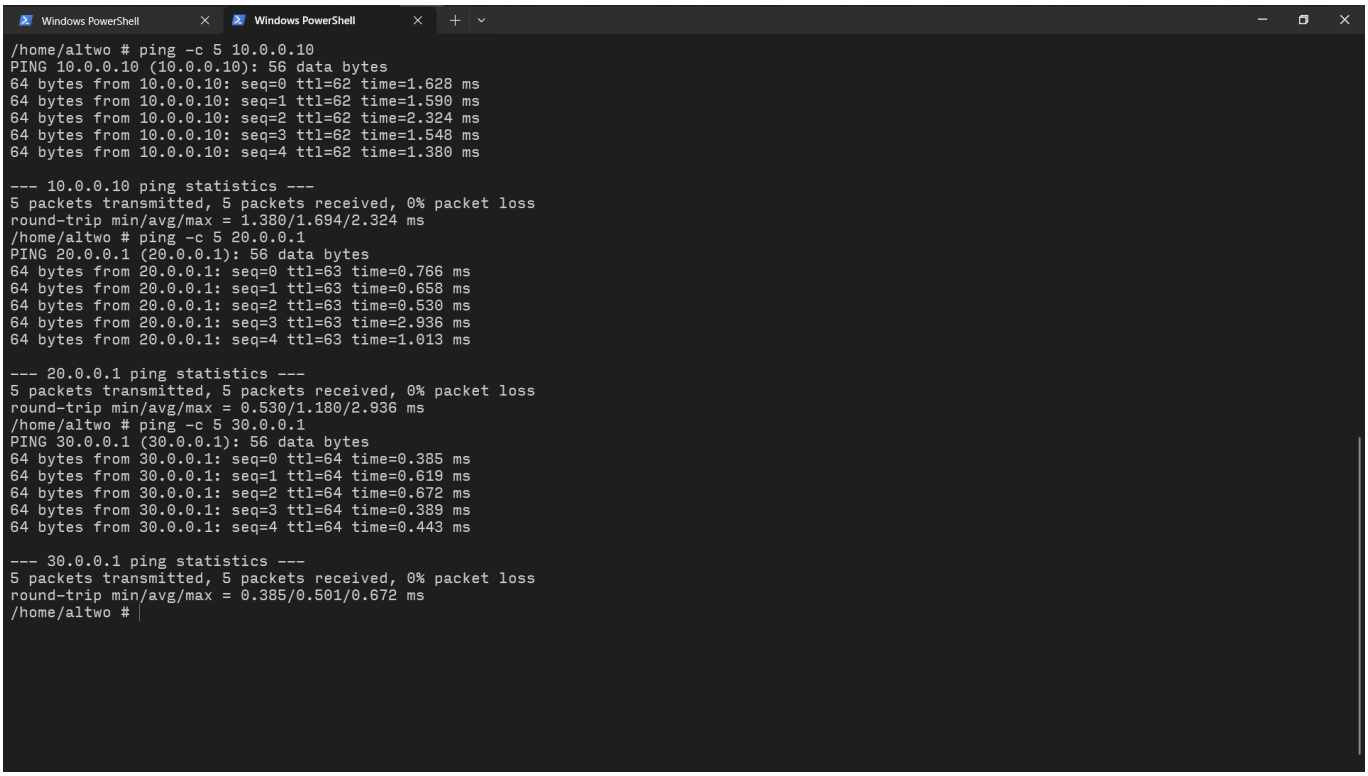
1. **ARP Resolution Failures**
    - Symptom: `<incomplete>` ARP entries
    - Resolution: Added static ARP entries:

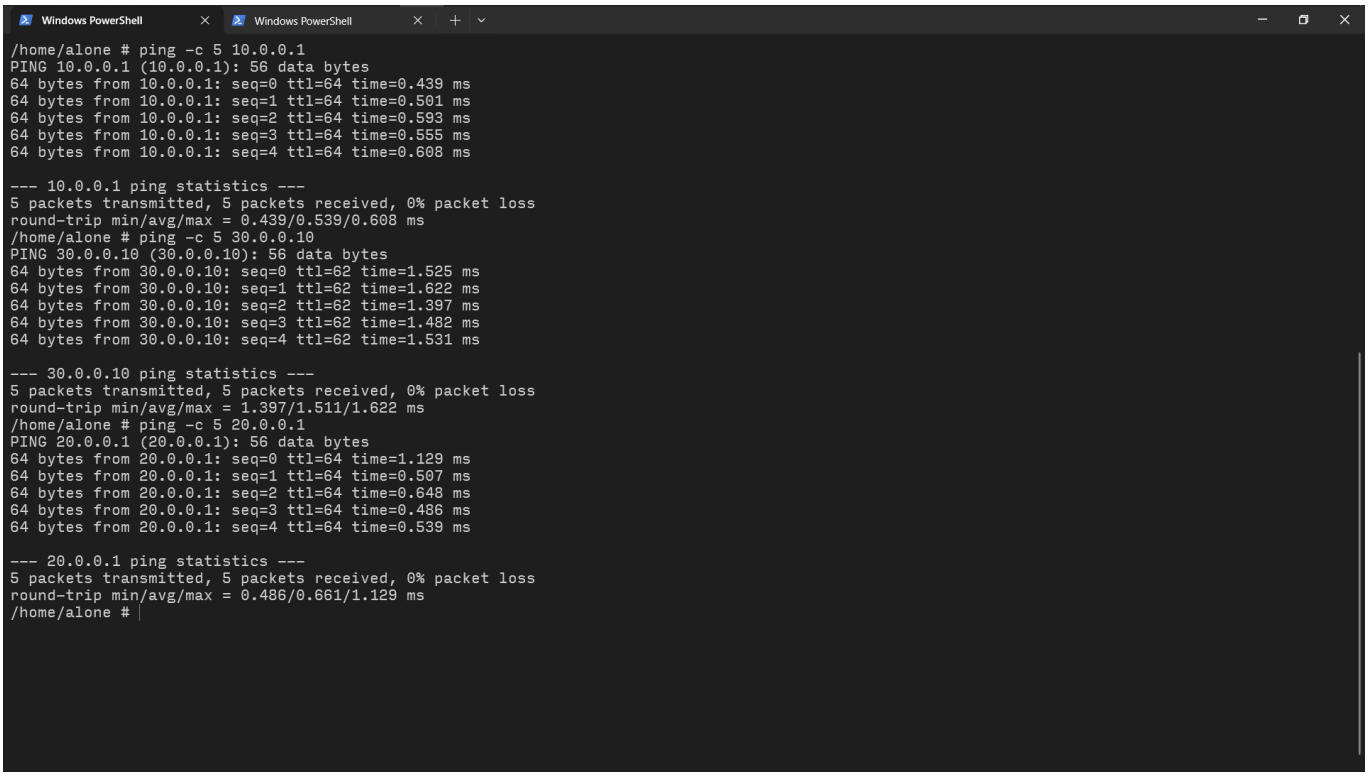
```
# On VM1
arp -s 10.0.0.1 <VM2_MAC>
# On VM2
sudo arp -s 10.0.0.10 <VM1_MAC>
```

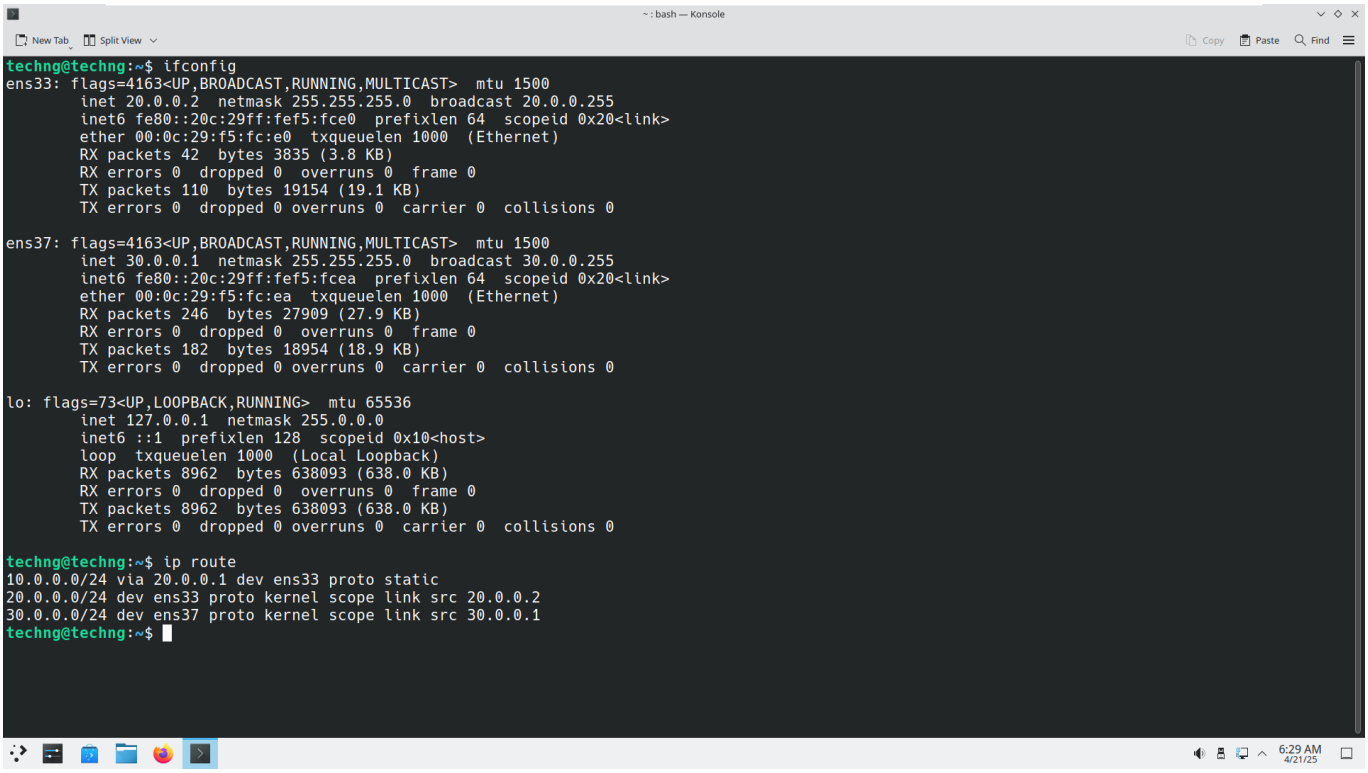
2. **Package Installation Errors**
    - Symptom: IO ERROR on `apk add`
    - Resolution: Updated Alpine repositories:

```
echo "http://dl-cdn.alpinelinux.org/alpine/v3.19/main" >
/etc/apk/repositories
apk update
```

\




Windows PowerShell
Windows PowerShell
/home/alone # ping -c 5 10.0.0.1
PING 10.0.0.1 (10.0.0.1): 56 data bytes
64 bytes from 10.0.0.1: seq=0 ttl=64 time=0.439 ms
64 bytes from 10.0.0.1: seq=1 ttl=64 time=0.501 ms
64 bytes from 10.0.0.1: seq=2 ttl=64 time=0.593 ms
64 bytes from 10.0.0.1: seq=3 ttl=64 time=0.555 ms
64 bytes from 10.0.0.1: seq=4 ttl=64 time=0.608 ms
--- 10.0.0.1 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.439/0.539/0.608 ms
/home/alone # ping -c 5 30.0.0.10
PING 30.0.0.10 (30.0.0.10): 56 data bytes
64 bytes from 30.0.0.10: seq=0 ttl=62 time=1.525 ms
64 bytes from 30.0.0.10: seq=1 ttl=62 time=1.622 ms
64 bytes from 30.0.0.10: seq=2 ttl=62 time=1.397 ms
64 bytes from 30.0.0.10: seq=3 ttl=62 time=1.482 ms
64 bytes from 30.0.0.10: seq=4 ttl=62 time=1.531 ms
--- 30.0.0.10 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.397/1.511/1.622 ms
/home/alone # ping -c 5 20.0.0.1
PING 20.0.0.1 (20.0.0.1): 56 data bytes
64 bytes from 20.0.0.1: seq=0 ttl=64 time=1.129 ms
64 bytes from 20.0.0.1: seq=1 ttl=64 time=0.507 ms
64 bytes from 20.0.0.1: seq=2 ttl=64 time=0.648 ms
64 bytes from 20.0.0.1: seq=3 ttl=64 time=0.486 ms
64 bytes from 20.0.0.1: seq=4 ttl=64 time=0.539 ms
--- 20.0.0.1 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.486/0.661/1.129 ms
/home/alone #


~ : bash — Konsole
New Tab
Split View
Copy
Paste
Find
techng@techng:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 20.0.0.2 netmask 255.255.255.0 broadcast 20.0.0.255
inet6 fe80::20c:29ff:fef5:fce0 prefixlen 64 scopeid 0x20<link>
ether 00:0c:29:f5:fc:e0 txqueuelen 1000 (Ethernet)
RX packets 42 bytes 3835 (3.8 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 110 bytes 19154 (19.1 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
ens37: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 30.0.0.1 netmask 255.255.255.0 broadcast 30.0.0.255
inet6 fe80::20c:29ff:fef5:fcea prefixlen 64 scopeid 0x20<link>
ether 00:0c:29:f5:fc:ea txqueuelen 1000 (Ethernet)
RX packets 246 bytes 27909 (27.9 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 182 bytes 18954 (18.9 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 8962 bytes 638093 (638.0 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 8962 bytes 638093 (638.0 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
techng@techng:~$ ip route
10.0.0.0/24 via 20.0.0.1 dev ens33 proto static
20.0.0.0/24 dev ens33 proto kernel scope link src 20.0.0.2
30.0.0.0/24 dev ens37 proto kernel scope link src 30.0.0.1
techng@techng:~$
6:29 AM
4/21/25

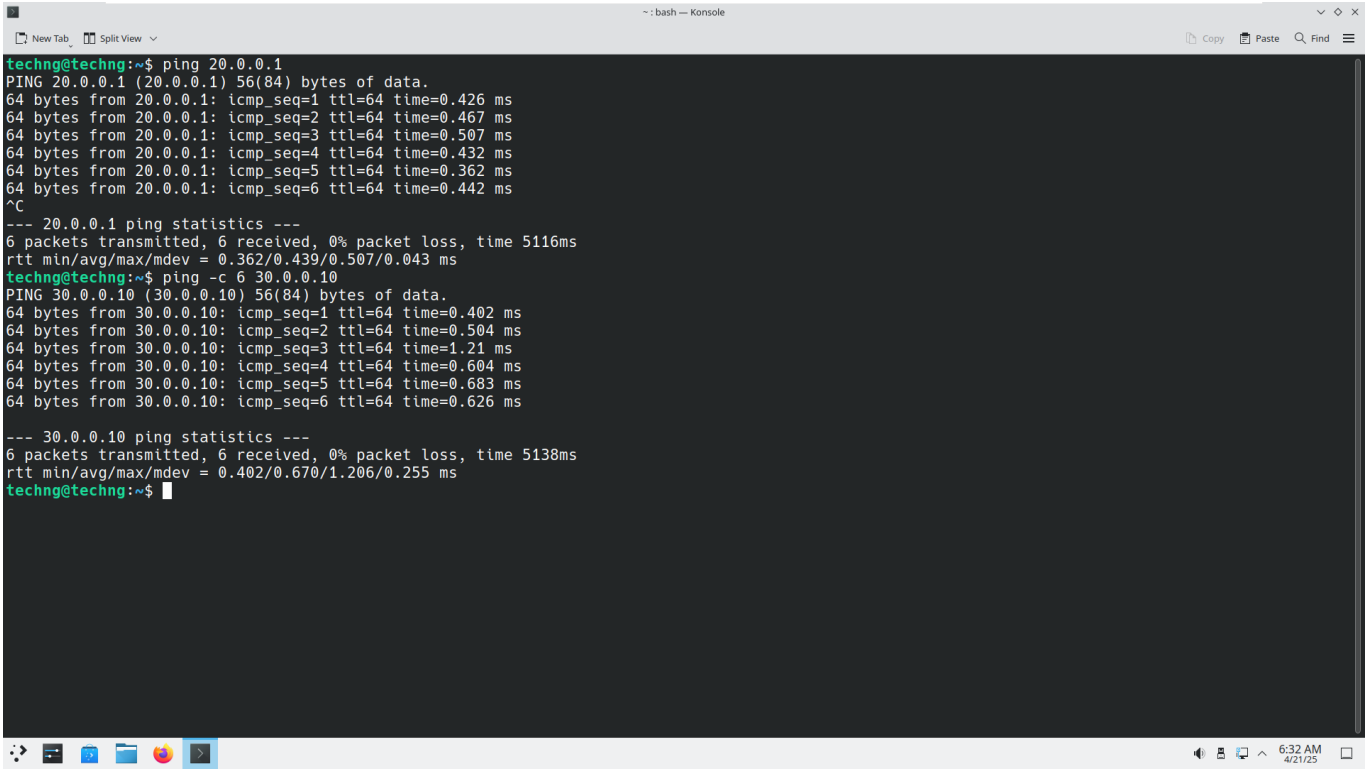


Figure 10: Troubleshooting network connectivity