Lab 05: Virtual Private networks

Network Infrastructure Security (CSP)  
 2024-2025

© Henk Brouckxon, Nico Declerck, Chris Roets

# Introduction

## Lab concept

Up to now, you have been configuring the Firewall over the Internet. This is not a good idea, as the management interface exposed for hacker. We will now change the configuration so that the management can only be done from a specific laptop through a VPN.

A computer with a black screen

Description automatically generatedA diagram of a computer network

Description automatically generated

## Learning goals

* Setup a VPN connection to the management station
* Enable routing so that the OPNSense os reachable through this VPN
* Change the management interface from the WAN interface to the interface of the management network

## Practicalities and prerequisites

You will need the following:

* You laptop/desktop 😊
* Your Proxmox environment with previous exercise successfully done

# Install the Wireguard VPN server on adminstation

1. Log in to the **adminstation** machine and install the Wireguard VPN client [1] [2] [3]. You’ll need the following packages: wireguard, wireguard-tools, iptables

apt-get install wireguard

apt-get install wireguard-tools

apt-get install iptables

1. Create the public/private keypair for the Wireguard VPN client, by running the following commands while you are in /etc/wireguard/:

wg genkey > server.privatekey

cat server.privatekey | wg pubkey > server.publickey

This should create two files, server\_privatekey and server\_publickey, that contain the keypair. Note down both keys here:

**server\_privatekey:**

**server\_publickey:**

**A screenshot of a computer program

Description automatically generated**

# Install the Wireguard VPN client (on your laptop or in a local Windows VM)

1. Download the Wireguard VPN Windows client and install it on your Windows laptop, or in a local Windows VM. The installer can be found at [1].
2. Create a new public/private keypair for the Windows Wireguard VPN client, by creating an empty tunnel “adminstation\_<firstname>” in the Wireguard GUI interface (bottom left, replace <firstname> by your first name, e.g. adminstation\_henk). We will call this keypair the client\_privatekey and client\_publickey. Note both keys here:

**Client\_privatekey: mLX44Dq2XMEzZJb3Q/CIjY8aubK3Zg1XdXGORS6z3Hg=**

**Client\_publickey: 9GWjtTScQ+ZpakTvFCCl49OXBWnmiEszOJrMrUlaxxk=**

**A screenshot of a computer

Description automatically generated**

# A new IP subnet for the VPN

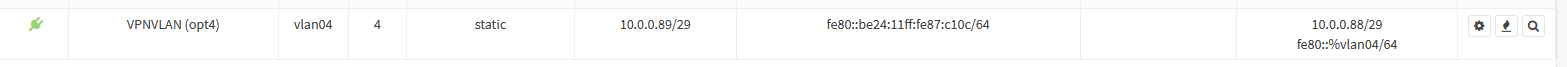
Because the Wireguard VPN works on OSI layer 3[[1]](#footnote-2), all VPN clients will receive an IP address in a specifically reserved IP subnet, which is different from the workstation/DMZ/aministration subnets that we defined in the previous labs. The VPN subnet should accomodate at least 5 client IP addresses, and should be part of the 10.0.0.0/24 network, just like the previously defined subnets. Because the VPN subnet will be connected to the adminstation host, this subnet will become part of that host’s VLAN (administration). Make a new VLAN/subnet table, based on this information:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **VLAN id** | **VLAN/Subnet name** | **Network address** | **prefix** | **netmask** |
| 1 | workstations | 10.0.0.0 | /26 | 255.255.255.192 |
| 2 | administration | 10.0.0.64 | /28 | 255.255.255.240 |
| 3 | DMZ | 10.0.0.80 | /29 | 255.255.255.248 |
| 4 | VPN (in workstations VLAN) | 10.0.0.88 | /29 | 255.255.255.248 |

In the new VPN subnet, assign two new IP addresses: one for the VPN server (adminstation), and one for the remote VPN client (your laptop).

**VPN server: 10.0.0.90/29**

**VPN client: 10.0.0.91/29**



A screenshot of a computer

Description automatically generated

I do not understand what I really did and the instructions are not really clear, like at all. Why new VLAN, how do I make it, where do I make it, and etc..

I created the VLAN, which can handle 5 IPs, assigned the IP, enabled it, put the parent to be the same as the other VLANs

# Configure the Wireguard VPN server on adminstation

1. Now, on adminstation, create a configuration file /etc/wireguard/wg0.conf that will define how the wireguard VPN tunnel will be built. This file should contain the following (replace the highlighted text by the relevant values for your environment):

[Interface]

Address = << SERVER IP address in the VPN subnet, written as x.y.z.u/v >>

PrivateKey = << SERVER PRIVATE KEY >>

PostUp = iptables -A FORWARD -i %i -j ACCEPT; iptables -t nat -A POSTROUTING -o

eth0 -j MASQUERADE;iptables -A FORWARD -o %i -j ACCEPT

PostDown = iptables -D FORWARD -i %i -j ACCEPT; iptables -t nat -D POSTROUTING -

o eth0 -j MASQUERADE;iptables -D FORWARD -o %i -j ACCEPT

ListenPort = 51820

[Peer]

PublicKey = << CLIENT PUBLIC KEY >>

AllowedIPs = << VPN SUBNET including prefix, written as x.y.z.u/v >>

A screenshot of a computer

Description automatically generated

1. Enable the wg0 configuration by running the following command:

wg-quick up wg0

1. Add the WireGuard service to systemd:

systemctl enable wg-quick@wg0.service

systemctl daemon-reload

FIRST DO ALL THE SYSTEMCTL, THEN DO THE UP

1. Start the new service immediately:

sudo systemctl start wg-quick@wg0

1. Allow routing of packets from the remote client through the adminstation machine

sysctl net.ipv4.ip\_forward=1

( alternative: Edit /etc/sysctl.conf to make this setting permanent)

1. To verify that the Wireguard VPN is up and running on the right port, you can run “netstat -tulp” and “watch wg” on adminstation. Also perform an NMAP scan for the relevant port on adminstation. Insert a screenshot of this NMAP scan.  
   important: Wireguard is a stateless VPN, which uses UDP ports!!!

A computer screen shot of a computer code

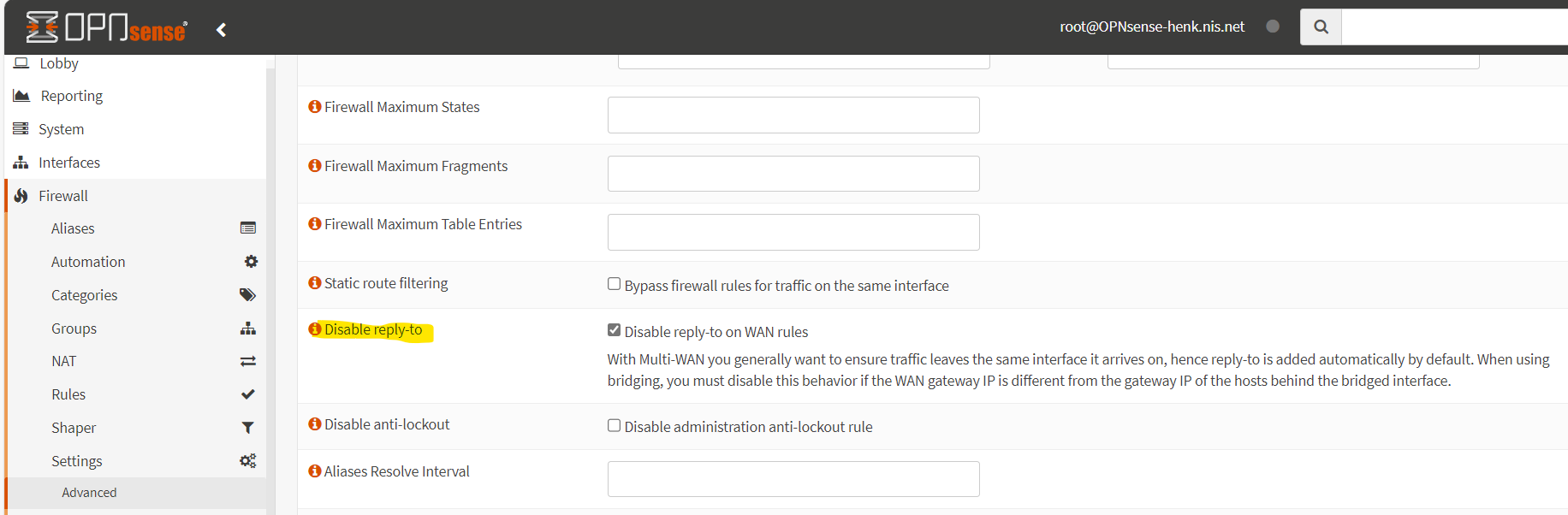
Description automatically generated

A screen shot of a computer

Description automatically generated

# Configuring a portforward through the OPNSense Firewall

1. Configure the Firewall to forward Wireguard packets from a self-chosen port on the WAN interface to the relevant port on adminstation. Please again note that Wireguard is a stateless VPN!  
     
     
   !!! Important !!![[2]](#footnote-3)

Due to the way the CITHub network is configured, an additional configuration step is required in OPNsense to make portforwards available to the outside world. The following checkbox should be selected: Firewall --> advanced --> Disable reply-to --> Disable Reply-to on WAN rules  
  


!!! Important !!!

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. You should now be able to verify that the Wireguard service is running on the WAN port that you selected, by performing an nmap scan of the relevant WAN port from your laptop. Insert a screenshot of this nmap scan here.=

A screenshot of a computer error

Description automatically generated

# Configure the Wireguard VPN client on the Windows machine

1. In the Windows Wireguard client, edit the “adminstation\_<firstname>” tunnel, so that its configuration file contains the following text:

[Interface]

PrivateKey = <<CLIENT PRIVATE KEY>>

Address = << CLIENT IP address in the VPN subnet, written as x.y.z.u/v >>

[Peer]

PublicKey = <<SERVER PUBLIC KEY>>

AllowedIPs = 10.0.0.0/24

Endpoint = <<WAN IP of your OPNsense firewall in the CIThub network>>:<<port>>

And activate the connection. Look at the ‘log’ tab of the application and verify that the connection is effectively built. An active connection should show “keepalive” messages from time to time.

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

1. Verify that you can ping adminstation through the tunnel

A screenshot of a computer

Description automatically generated

1. Now you can add a route from the Windows PC to the administration subnet, using the “adminstation” VM as a router (and in this way going through the VPN tunnel):

route add <<admin subnet IP>> mask <<subnet mask>> <<adminstation IP in VPN subnet>>

route add 10.0.0.64 mask 255.255.255.240 10.0.0.90 (to execute on Windows, first IP is the management subnet and not the first, and the IP that I made above for adminstation. )

1. Verify that you can ping the IP address of the OPNsense firewall in the administration subnet from your Windows machine (through the VPN tunnel). Now you should also be able to access the web interface of the firewall from your Windows machine through the VPN tunnel.

I CAN DO IT BEFORE EVEN SETTING EVERYTHING UP FROM ABOVE (probably because I have floating rule turned on)

A screenshot of a computer

Description automatically generated

1. Make a screenshot of the following on your Windows machine:
   * The Wireguard client configuration file, including the name of the profile (that contains your first name)
   * A terminal window, showing a working ping command from the Windows machine to the interface of the firewall in the administration subnet.

Submit this screenshot in the assignment on Leho.

A screenshot of a computer screen

Description automatically generated

Not sure if this is the things you wanted, so the ping to firewall (10.0.0.5) from LAN works, as well as the default gateway for the management VLAN also works (10.0.0.65)

A screenshot of a computer

Description automatically generated

# References

|  |  |
| --- | --- |
| [1] | "Wireguard - Installation," [Online]. Available: https://www.wireguard.com/install/. |
| [2] | "Setting up Wireguard on Amazon Linux 2," [Online]. Available: https://www.cyberciti.biz/faq/install-set-up-wireguard-on-amazon-linux-2/#Find\_system\_info. |
| [3] | "Installing Wireguard VPN on Alpine Linux," [Online]. Available: https://wiki.alpinelinux.org/wiki/Configure\_a\_Wireguard\_interface\_(wg). |

1. some other VPNs also allow layer-2 connectivity on the ethernet level where the clients become part of the existing subnets, but this is typically more difficult to implement [↑](#footnote-ref-2)
2. thanks to our @Home students for their help with resolving this issue [↑](#footnote-ref-3)