Govind Sharma | 162391189

Peter Callaghan

Enterprise Infrastructure and Security | OPS 300 NAA

February 7, 2024

VPN

Contents

[Introduction 1](#_Toc132295576)

[Setup 2](#_Toc132295577)

[Configuration 3](#_Toc132295578)

[Management 5](#_Toc132295579)

[Testing 6](#_Toc132295580)

[Security Best Practices 6](#_Toc132295581)

[Works Cited 7](#_Toc132295582)

# Introduction

Meet Rahul, he is an IT administrator in charge of an electric car company. The engineering team hired a remote engineer that works in another country and a field technician to repair the charging stations. How can Rahul allow both employees to work while having access to the servers at the headquarters in a secure way if they are both not working in office, VPN! VPN are virtual private network which allows both remote employees the ability to create an encrypted tunnel for information to flow securely from one network to another. Once that encrypted tunnel is created a computer connected to the VPN can access that office network and use all of the network resources like file servers. Without a VPN on the office network the remote engineer would have to use their home router and that information would be sent and pulled through non secure routes, similarly the field technician would be using a sim card in their laptop which would connect to a mobile phone data plan and the information would be going through the satellite routers. In both scenarios a malicious attacker could have access to proprietary information of the car company because the information traffic is not secure. A VPN would create a encrypted tunnel from the office network to their computers and allow for secure transfer of data so no one else can get the proprietary plans for the electric car! In this report however to demonstrate the setup, configuration, and management of a VPN a Raspberry Pi will be used to demonstrative how a simple VPN can be created.

# Setup

To run a simple VPN demonstration, I will be using a Raspberry Pi B+ which is a low cost, low power consumption, and passively cooled computer. However, with those positives it only has a single core processor which makes the connection through the VPN slow as the encryption is done by the processor, Rahul the IT admin would buy a more powerful hosting node or server to host the VPN so that it can encrypt and decrypt traffic faster especially for a enterprise environment.

To start I installed Raspberry Pi Imager which is an imaging software that flashes micro-SD cards to use in a Raspberry Pi. I configured the installer to flash a port of Debian Bullseye called Raspberry Pi Lite that has no graphical interface (to save potentially used hardware power on a GUI), enabled SSH connection using a password authentication, and a username and password so I can use it headless without the default credentials (using default credentials poses a security threat as they are known credentials). After the micro-SD card was flashed, I inserted it into the Raspberry Pi, plugged in an ethernet cable connected to my home router on the other end, and plugged in a micro-B cable for power. Then I went to my home routers settings to set a static IP address of 192.168.2.103 so I can SSH into it. I was then able to SSH into my computer by using into my VPN by using the username I set in the imager and the static IP I set.

To install the VPN I am using PiVPN which is an automated installer that helps to install a VPN and configure it in an easy to understand step by step manner. To install the PiVPN I used the command $curl -L <http://install.pivpn.io> | bash. This starts a simple text-based installer where every choice can be examined and understood fully.

# Configuration

The following numbered list is the list of the configuration questions, all other informational text pages are omitted from this configuration list.

1. IPv6 force routing to block IPv6 leaking
   1. I set this option to no, however I should have selected yes in retrospect. I explain more about this in the “Security Best Practices” section.
2. DHCP Reservation
   1. As I set the static IP address for this device on my router settings I simply added that address into the IP address field “192.168.2.103”, and added the address of my gateway which is my home router “192.168.2.1”.
3. Choose local user that holds VPN configuration
   1. I chose the username I set in the Raspberry Pi imager “gsharma73”
4. Choose a VPN
   1. The installer now asks which VPN and prompts WireGuard and OpenVPN. A VPN itself is just the name of the concept but I actually have to install a VPN tunneling service, OpenVPN is one of them. OpenVPN is a trusted, reliable and flexible VPN software but it has a large codebase and it is difficult to parse through by cybersecurity professionals in a security audit. OpenVPN also uses a block cipher to encrypt and decrypt information in chunks which is slow.
   2. WireGuard is a newer, faster, and more secure VPN as it uses less than 4000 lines of code which makes it easy to audit by just a single cybersecurity professional. WireGuard uses a key pair system to create a public and private key to keep data encrypted and secure through transit. Those keys then undergo a handshake and a together they create a shared secret key that helps to encrypt and decrypt traffic. The encryption also uses XChaCha20 encryption, which is a 256-bit stream cipher encryption that encrypts and decrypts each bit which makes data transfer faster (as opposed to slower block cipher encryption)
5. Port selection
   1. Wireguard has a default port of 51820 and communicates through the UDP protocol so I left the port number as the default and went into my home routers port forwarding rules, I enabled a rule for the IP address of 192.168.2.103 and allowed UDP traffic through port 51820.
6. DNS Provider
   1. I have my own DNS server on another Raspberry Pi that runs PiHole, a DNS client for Raspberry Pi’s. When asked I set the option for a custom DNS provider and entered the IP address of my Raspberry Pi running my personal DNS server which is 192.168.2.207
7. Public IP or DNS
   1. This configuration is asking if the user wants to use a public IP address or a DNS entry. This option is irrelevant as WireGuard uses a conf file that has all the nessisary configuration in it. In other VPN software like OpenConnect that Seneca uses, you must connect using a DNS or public IP and, in that scenario, having a DNS entry is easier for end users to configure on their laptops than having a public IP address. Regardless due to how WireGuard works I used my public IP.
8. Unattended Upgrades
   1. Wire guard allows unattended upgrades however it is an update. So you have to manually reboot every so often. Unattended upgrades help to patch security concerns. And it's important that these concerns are met with a very serious and important intention.

That’s the entire installation and configuration of the PiVPN installer, now we have to move on to management of the WireGuard VPN.

# Management

To start using the VPN I can type “$pivpn help” for all the commands of pivpn. To add a client type “$pivpn add” and then it will prompt the user for a name of the client I typed in “lychee” as my laptop is named “lychee” after the fruit. (Note the name has no bearing or dependency on the hostname of the computer the configuration file will be used on) The software will then create a key pair and create a configuration file with all the pertinent details that the WireGuard client needs to connect to the VPN host including the public key information, DNS server IP address, the VPN host IP and port. It created a file called lychee.conf and saved it in the /home/gsharma73/configs folder on my VPN host.

I now need to get the config file onto my laptop and from the terminal on my laptop I used secure copy to copy the file over to my laptop “% scp [gsharma73@192.168.2.103:/configs/lychee.conf](mailto:gsharma73@192.168.2.103:/configs/lychee.conf) /Users/lychee/desktop”. Now the configuration file is on my laptop. I downloaded the WireGuard client and added the configuration file into this client.

# Testing

Now that I have the config file in the WireGuard client I went over to Seneca on the weekend and while connected to Senecas network I activated the VPN session tunnel to my VPN host at home, it activated the tunnel and I got the same IP address that is my public home IP address! It was a complete success, to further test if I could access network assets I SSH connected to my DNS server at home. I would not have been able to do that because I otherwise would not be able to connect to an IP address that does not exist on Senecas network.

# Security Best Practices

* IPv6 force routing
  + When doing this configuration option, I mistakenly chose no, however upon further research I realized that he should have said yes. IPv6 leaking is a very real threat and occurs when your VPN is communicating to a server via IPv4 but that server is communicating back to you in IPv6. This means the web server you are communicating to can see what your actual laptops IP address is at Seneca. This eliminates one of the purposes of a VPN which is to secure it and make your traffic private. But I have limited my VPN on its functionality.
* Regular automatic updating
  + I would highly advise that all users attempting this VPN implementation enable automatic updating to allow for all patches to be installed as they come. The managing user still has to manually apply those updates by rebooting but a reboot could be scheduled regularly throughout the week so the update and the application of the update are both automated.
* Keep the config file secret
  + The configuration file has all the information the WireGuard client needs to connect to the host VPN thus its best that the file is kept secret in an encrypted or password protected folder.
  + It could also be kept on a separate user account, keeping it more secure behind another layer of security
  + The config file can be kept on a cold storage USB flash drive.

# Works Cited

WireGuard. (n.d.). WireGuard: Fast, Modern, Secure VPN Tunnel. Retrieved April 13, 2023, from <https://www.wireguard.com/>

CISA. (2020, March 16). AA20-073A: Enterprise VPN Security. Cybersecurity & Infrastructure Security Agency. Retrieved April 13, 2023, from <https://www.cisa.gov/news-events/cybersecurity-advisories/aa20-073a>

NordPass. (2021, March 17). XChaCha20 Encryption vs. AES-256: Which One to Choose? Retrieved April 13, 2023, from <https://nordpass.com/blog/xchacha20-encryption-vs-aes-256/>

WireGuard. (n.d.). WireGuard: Fast, Modern, Secure VPN Tunnel. Retrieved April 13, 2023, from <https://www.wireguard.com/>

WireGuard. (n.d.). WireGuard Protocol. Retrieved April 13, 2023, from <https://www.wireguard.com/protocol/>

ProPrivacy. (n.d.). A Complete Guide to IP Leaks (2022). Retrieved April 13, 2023, from <https://proprivacy.com/vpn/guides/a-complete-guide-to-ip-leaks>

PiVPN. (n.d.). PiVPN - Simplest OpenVPN Setup and Configuration, Designed for Raspberry Pi. Retrieved April 13, 2023, from <https://pivpn.io/>