

**AKADEMIA HANDLOWA  
NAUK STOSOWANYCH W RADOMIU**



**RADOM  
ACADEMY OF ECONOMICS**

## **Wydział Studiów Strategicznych i Technicznych**

Kierunek: Informatyka, rok II, semestr III (2021/2022)

# **LABORATORIUM NR 2 PODSTAWY KRYPTOGRAFII**

Prowadzący: dr Piotr Dobosz

**Zespół laboratoryjny:**

Magdalena Szafrńska, nr albumu: 18345

# Spis treści

<b>Cel ćwiczenia</b>	<b>2</b>
<b>Informacje wstępne</b>	<b>2</b>
VPN	2
Tunel	3
OpenVPN	4
PPTP	5
<b>Użyte narzędzia i aplikacje</b>	<b>5</b>
<b>Przebieg ćwiczenia</b>	<b>6</b>
CZEŚĆ I: Badanie ruchu w niezabezpieczonym tunelu	6
Konfiguracja VirtualBox (z Ubuntu)	6
Instalacja serwera PPTP (Ubuntu)	6
Konfiguracja klienta PPTP (Windows 10)	10
Analiza Wiresharka	12
CZEŚĆ II: Badanie ruchu w zabezpieczonym tunelu	13
Instalacja serwera OpenVPN (Ubuntu)	13
Konfiguracja klienta OpenVPN (Windows 10)	17
<b>Wnioski</b>	<b>19</b>

# Cel ćwiczenia

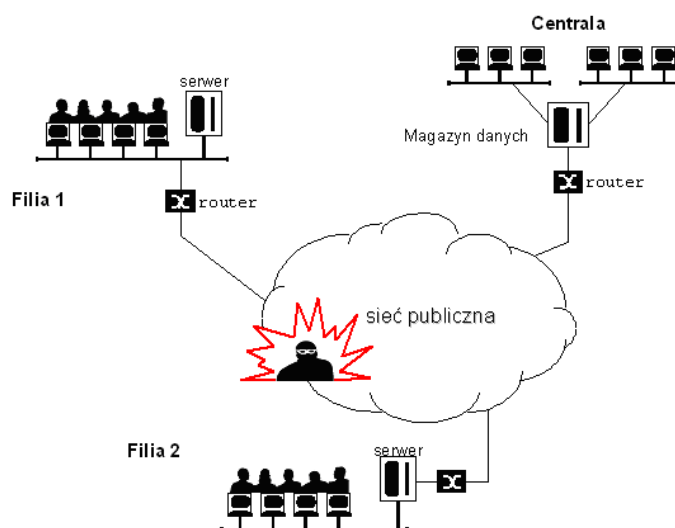
Celem laboratorium jest przetestowanie działania tuneli sieciowych z wykorzystaniem technologii VPN. Sprawdzenie które z połączeń zdalnych do wirtualnej sieci prywatnej jest najbezpieczniejsze i która technologia daje najlepszą ochronę przy jednoczesnej prostocie konfiguracji.

## Informacje wstępne

W wielu sytuacjach, w których mamy do czynienia z rozbudowaną architekturą sieciową zależy nam aby całe środowisko rozproszone (np. oddziały firmy w różnych lokalizacjach) było jak najbardziej spójne. Dotyczy to również adresacji IP. Technologia wirtualnych sieci prywatnych pozwala na rozwiązanie takich problemów. Dzięki utworzeniu sieci VPN można zbudować logiczną sieć komputerową, która będzie łączyć całe rozproszone środowisko, ukrywając sieci łączące odległe od siebie lokalizacje i tym samym uprości wymianę danych między nimi. Budując sieć VPN tworzymy logiczne tunele między wyznaczonymi lokalizacjami. W ten sposób technologia VPN tworzy iluzję, w której odległe od siebie lokalizacje są fizycznie bezpośrednio połączone. Ta cecha sieci VPN wpływa na uproszczenie sposobu wymiany ruchu między tymi lokalizacjami.

### VPN

Virtual Private Network, w skrócie VPN to tunel wirtualny. Jest to kanał komunikacyjny chroniony przed niepożądanym dostępem (odczytem i modyfikacją) poprzez zastosowanie kryptografii. Tunel wirtualny VPN umożliwia chronioną transmisję w obszarze publicznej sieci rozległej, np. w celu realizacji bezpiecznego połączenia pomiędzy różnymi jednostkami, najczęściej geograficznie odległymi.



Rys. Schemat sieci publicznej analizowany jako scenariusz zagrożeń

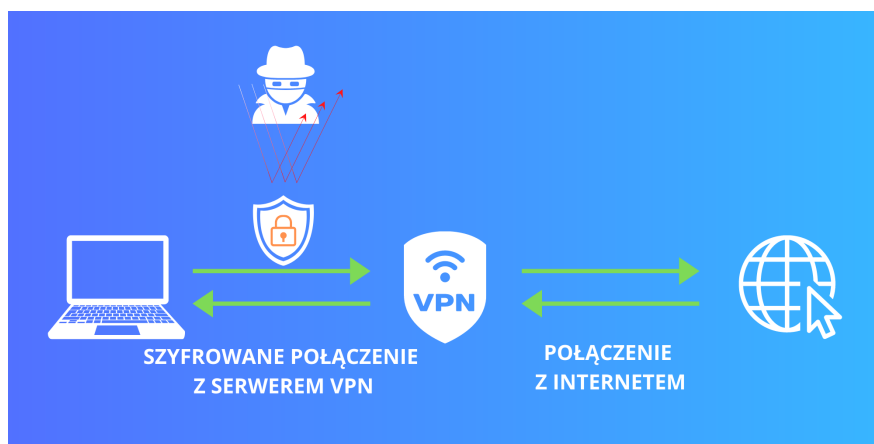
Najprostsze rozwiązania mogą w ogóle nie wspierać ochrony poufności przesyłanych danych, bardziej zaawansowane mogą korzystać z mechanizmu współdzielonego klucza i algorytmów kryptografii symetrycznej do ochrony poufności. Najbardziej zaawansowane rozwiązania korzystają z mechanizmów kryptografii klucza publicznego, certyfikatów cyfrowych.

### Dlaczego warto korzystać z VPN?

Każda wizyta w Internecie pozostawia po sobie mnóstwo informacji, na podstawie których można zidentyfikować użytkownika, zebrać dane o jego zainteresowaniach, odwiedzanych witrynach, a czasami nawet bardziej poufne rzeczy, np. hasła do logowania.

Połączenie VPN maskuje te wszystkie połączenia, kierując je przez serwer VPN. Tworzy to niejako tunel (stąd nazwa tunelowanie), którym poruszają się informacje wymieniane między komputerem użytkownika i Internetem. Jest tylko jedno wyjście i wejście, nie można się do niego dostać „z boku” - zapobiega to atakowi Man in the middle.

Dodatkowo wszystko co wysyłane i odbierane zostaje zaszyfrowane: tunelowanie wychwytuje wszystkie pakiety, szyfruje i każdorazowo umieszcza w nowym pakiecie. Można to porównać do stworzenia pliku archiwum (zip, rar, 7z). Następnie serwer odpakuje sobie te pakiety. Nawet, jeżeli ktoś przechwyciłby te dane, dostanie tylko mnóstwo niezrozumiałych, zakodowanych informacji których nie będzie umiał odczytać, bo nie ma klucza.



Źródło: <https://trybawaryjny.pl/co-to-vpn/>

## Tunel

Tunel to zestawienie połączenia między dwoma odległymi komputerami tak, by stworzyć wrażenie, że są połączone bezpośrednio. Przesłanki do tworzenia tuneli to:

- wygoda
- umożliwienie połączenia między komputerami ukrytymi w sieciach prywatnych (za firewallem, lub z adresami prywatnymi - warunek: jeden z hostów biorących udział w tworzeniu tunelu musi mieć adres publiczny)
- bezpieczeństwo - szyfrowanie połączenia: SSL/TLS, SSH
- możliwość przyspieszenia transmisji – kompresja sprawdza się szczególnie na wolnych łączach

Pojęcia 'tunelowanie' zazwyczaj używa się w odniesieniu do przesyłania poprzez tunel tylko jednego protokołu. Zwykle jest to jeden z typowych protokołów np.: POP3, SMTP, HTTP, które przesyłają dane w sposób jawny (w tym nazwy użytkowników i hasła) za pomocą bezpiecznych protokołów TLS/SSL lub SSH. Wirtualna sieć prywatna – umożliwia transmisję w sposób bezpieczny wielu protokołów poprzez publiczną sieć. Hosty będące po obu stronach VPN-u „widzą się” tak jakby były w jednej sieci. Takie sieci przeważnie działają w warstwie 3 modelu TCP/IP i tunelują warstwę 3 i wyższe - niektóre z nich umożliwiają także transmisję 2 warstwy (np tworzą wirtualną kartę sieciową). Kompresja powinna spełniać dwa podstawowe kryteria: niskie zapotrzebowanie na moc procesora i szybkość działania. Kompresji może zostać poddana całość transmisji lub tylko ładunek użyteczny pakietów IP.

## OpenVPN

Jest to program umożliwiający tworzenie wirtualnych sieci prywatnych. Oparty jest na SSL/TLS i bibliotece OpenSSL, dostępny dla systemów uniksowych i Windows. OpenVPN zapewnia nam komplet mechanizmów, który pozwala nam zbudować bezpieczną sieć VPN – autoryzuje, uwierzytelnia, szyfruje i zapewnia integralność przesłanych danych. Ponadto chroni serwer (koncentrator) przed atakami DoS oraz sieć wirtualną przed wstrzykiwaniem obcych danych.

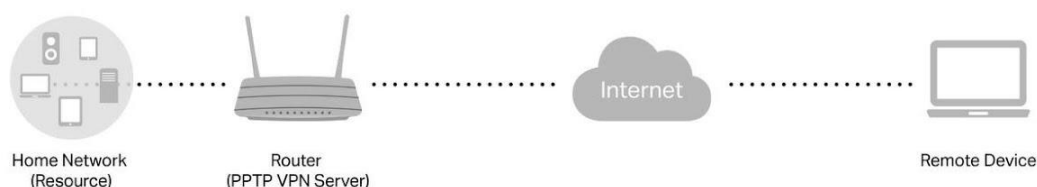
OpenVPN oferuje dwa rodzaje tuneli, których używamy w zależności od konkretnych potrzeb. W laboratorium skorzystam z rodzaju tunelu „tun”. Hosty mają przydzielony po obu stronach adres IP i odbywa się klasyczny routing: wystarczy w większości przypadków takich jak dostęp do sieci firmowej czy do Internetu.

### **Zabezpieczenia OpenVPN** (generowane klucze, podpisy, szyfrowanie połączeń)

- Implementacja tunelu punkt-punkt z wykorzystaniem SSLv3/TLSv1, w odróżnieniu od wielu innych implementacji VPN nie bazuje na IPsec ale wykorzystuje wirtualny interfejs sieciowy tun/tap (tun – warstwa 3 protokół IP, tap – warstwa druga – np ethernet) dostępny jest m.in. dla Linux-a, BSD (w tym Mac OS X), Windows2000 w górę, Solaris.
- Autentykacja może odbywać się za pomocą współdzielonego klucza, certyfikatów oraz nazwy użytkownika i hasła.
- Pakiet składa się z jednego programu binarnego, dodatkowo można użyć pliku konfiguracyjnego oraz plików z kluczem współdzielonym lub certyfikatów.
- Szyfrowanie bazujące na SSL może być wspomagane przez mechanizm haszujący HMAC (Hash Message Authentication Code) w celu zwiększenia bezpieczeństwa.
- Pracuje w przestrzeni użytkownika i używa do komunikacji protokołu UDP lub TCP wykorzystując oficjalnie przyznany temu protokołowi przez IANA port 1194.
- Nie wymaga tak jak IPsec tworzenia 2 kanałów – w OpenVPN tunel jest dwukierunkowy i łatwiej przechodzi przez firewalle.
- Pozwala na równoległe tworzenie wielu tuneli i wspomaga routing między nimi.
- Do kompresji używany jest szybki algorytm LZO podobny do gzip.
- Wspiera akcelerację sprzętową oraz karty procesorowe.

## PPTP

PPTP (Point-to-Point Tunneling Protocol) jest jednym z protokołów, w oparciu o który mogą działać virtualne sieci prywatne. To protokół komunikacyjny umożliwiający tworzenie virtualnych sieci prywatnych wykorzystując technologię tunelowania. Typową topologię protokołu PPTP w tunelu VPN przedstawia poniższy schemat.



Źródło: <https://www.youtube.com/watch?v=Wv42x6Gr4Yc>

Protokół PPTP umożliwia zdalne połączenie się do stacji roboczej lub sieci za pośrednictwem Internetu oraz tworzenie wirtualnego połączenia z lokalną siecią. Ma na celu zapewnienie bezpieczeństwa przy zdalnym przesyłaniu danych. Inicjalizacja połączenia wykonywana jest na port 1723.

Spośród wszystkich protokołów VPN, PPTP jest jednym z najczęstszych używanych, najłatwiejszych do skonfigurowania i najszybszych. Z tego powodu, protokół PPTP jest przydatny w aplikacjach, w których prędkość jest najważniejsza, jak streaming audio lub wideo oraz na starszych, wolniejszych urządzeniach z bardziej ograniczonymi procesorami.

PPTP wykazuje poważne luki w zabezpieczeniach. Podstawowe protokoły uwierzytelniania są zasadniczo niezabezpieczone i były wielokrotnie łamane w analizach bezpieczeństwa od czasu ich wprowadzenia. Z tego powodu PPTP nie jest zalecany z wyjątkiem przypadków, w których prędkość transmisji jest ważniejsza od jej bezpieczeństwa.

## Użyte narzędzia i aplikacje

- **Windows 10 Home** w wersji 21H1
- **Virtualbox** - maszyna wirtualna Oracle (u mnie z systemem Linux)
- **Ubuntu 20.04.3 LTS** (Focal Fossa) - dystrybucja Linuxa
- **Wireshark** - sniffer pozwalający "podsluchiwać" sieć także w czasie rzeczywistym; analizuje ruch sieciowy; umożliwia przechwytywanie i nagrywanie pakietów danych, a także ich dekodowanie
- **Npcap 1.55** - biblioteka windowsowa do Wiresharka; pozwala w czasie rzeczywistym śledzić dane, które przechodzą przez kartę sieciową komputera, dzięki niej widzimy wszystkie aktualnie przechwytywane interfejsy sieciowe
- **OpenVPN** - protokół wykorzystywany do tworzenia tuneli
- **PPTP** - protokół komunikacyjny umożliwiający tworzenie virtualnych sieci prywatnych wykorzystujących technologię tunelowania
- **dostęp do sieci bezprzewodowej**

# Przebieg ćwiczenia

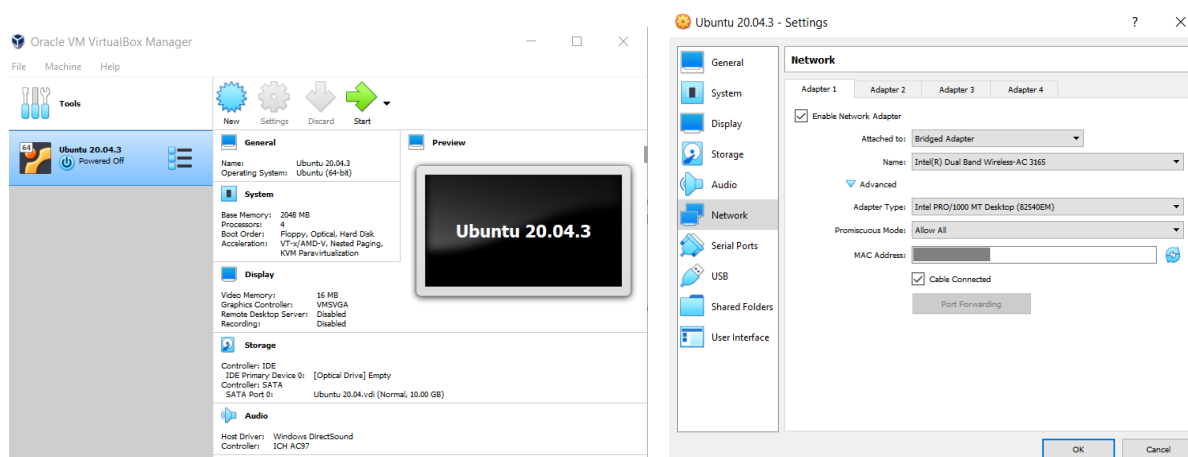
Laboratorium wykonywałam na komputerze z systemem Windows 10 oraz wirtualnej maszynie z Ubuntu. Na Ubuntu na wirtualnej maszynie zainstalowałam serwery PPTP oraz OpenVPN aby ustawić tunelowanie. Na jednym z komputerów (klient z Windowsem) uruchomiłam program Wireshark, którym następnie śledziłam działania komputera atakowanego (serwera).

Laboratorium podzieliłam na dwie części: najpierw badałam tunelowanie z użyciem protokołu PPTP, a następnie z OpenVPN. We wnioskach na końcu zestawiałam obie technologie połączeniowe oraz porównałam ich poziomy skomplikowania konfiguracji - także z innymi systemami tunelowania.

## CZĘŚĆ I: Badanie ruchu w niezabezpieczonym tunelu

### 1. Konfiguracja VirtualBox (z Ubuntu)

Dostosowałam konfigurację wirtualnej maszyny do potrzeb laboratorium zapewniając odpowiednią ilość miejsca, RAM oraz połączenie sieciowe niezbędne do ustanowienia tunelowania.



### 2. Instalacja serwera PPTP (Ubuntu)

Do uruchomienia tunelowania użyłam distro Linuxa (Ubuntu) jako serwera, na którym wykonałam instalację i konfigurację technologii PPTP. Wprowadzając do terminala poszczególne polecenia pomyślnie zainstalowałam serwer PPTP.

```
sudo apt-get update  
sudo apt-get upgrade  
sudo apt-get install pptpd
```

aktualizacja oprogramowania  
instalacja serwera PPTP

```
yaviena@yaviena-VirtualBox: ~  
yaviena@yaviena-VirtualBox:~$ sudo apt-get install pptpd  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following additional packages will be installed:  
  bcrelay  
The following NEW packages will be installed:  
  bcrelay pptpd  
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.  
Need to get 87,3 kB of archives.  
After this operation, 299 kB of additional disk space will be used.  
Do you want to continue? [Y/n] y  
Get:1 http://pl.archive.ubuntu.com/ubuntu focal/main amd64 bcrelay amd64 1.4.0-11build1 [12,0 kB]  
Get:2 http://pl.archive.ubuntu.com/ubuntu focal/main amd64 pptpd amd64 1.4.0-11build1 [75,3 kB]  
Fetched 87,3 kB in 0s (505 kB/s)  
Selecting previously unselected package bcrelay.  
(Reading database ... 178128 files and directories currently installed.)  
Preparing to unpack .../bcrelay_1.4.0-11build1_amd64.deb ...  
Unpacking bcrelay (1.4.0-11build1) ...  
Selecting previously unselected package pptpd.  
Preparing to unpack .../pptpd_1.4.0-11build1_amd64.deb ...  
Unpacking pptpd (1.4.0-11build1) ...  
Setting up bcrelay (1.4.0-11build1) ...  
Setting up pptpd (1.4.0-11build1) ...  
update-rc.d: warning: start and stop actions are no longer supported; falling back to defaults  
Processing triggers for man-db (2.9.1-1) ...  
Processing triggers for systemd (245.4-4ubuntu3.15) ...  
yaviena@yaviena-VirtualBox:~$
```

<code>sudo nano /etc/pptpd.conf</code> <code>sudo nano /etc/ppp/pptpd-option</code>	konfiguracja pliku serwera PPTP modyfikacja wpisu (dodanie swoich serwerów DNS)
--	--

Edytując plik serwera PPTP ustawiam:

- **localip** - adres po stronie serwera PPTP
- **remoteip** - pulę adresów, które zostały mi przyznane przy podłączeniu się do serwera VPN).

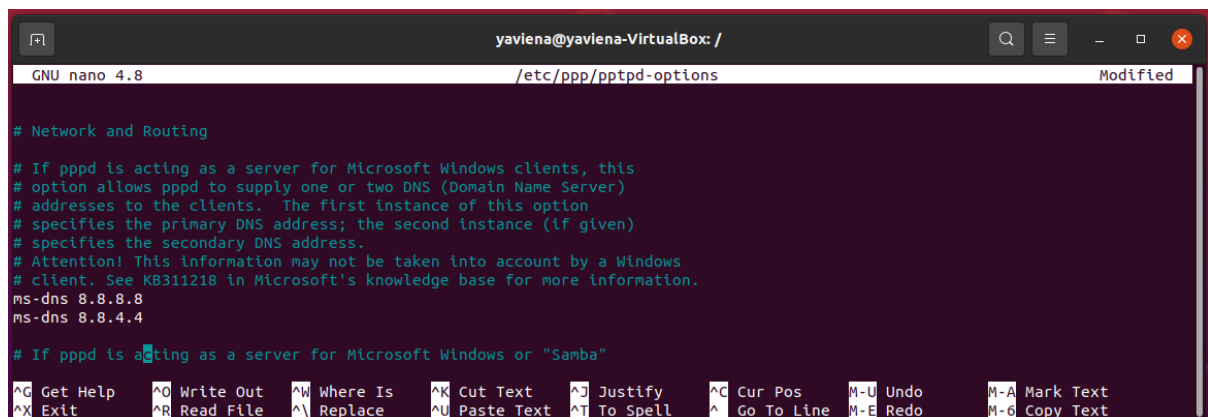
```
GNU nano 4.8 /etc/pptpd.conf Modified  
#  
# 2. If you give more IP addresses than the value of connections,  
# it will start at the beginning of the list and go until it  
# gets connections IPs. Others will be ignored.  
#  
# 3. No shortcuts in ranges! ie. 234-8 does not mean 234 to 238,  
# you must type 234-238 if you mean this.  
#  
# 4. If you give a single localIP, that's ok - all local IPs will  
# be set to the given one. You MUST still give at least one remote  
# IP for each simultaneous client.  
#  
# (Recommended)  
localip 192.168.1.19  
remoteip 192.168.1.1-254  
# or  
#localip 192.168.0.234-238,192.168.0.245  
#remoteip 192.168.1.234-238,192.168.1.245  
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos M-U Undo  
^X Exit ^R Read File ^_ Replace ^U Paste Text ^T To Spell ^_ Go To Line M-E Redo
```



Zmodyfikowałam wpis dodając swoje serwery DNS.

**sudo nano /etc/ppp/pptpd-option**

modyfikacja wpisu i dodanie swoich serwerów DNS



```
GNU nano 4.8 /etc/ppp/pptpd-options Modified

# Network and Routing

# If pptpd is acting as a server for Microsoft Windows clients, this
# option allows pptpd to supply one or two DNS (Domain Name Server)
# addresses to the clients. The first instance of this option
# specifies the primary DNS address; the second instance (if given)
# specifies the secondary DNS address.
# Attention! This information may not be taken into account by a Windows
# client. See KB311218 in Microsoft's knowledge base for more information.
ms-dns 8.8.8.8
ms-dns 8.8.4.4

# If pptpd is acting as a server for Microsoft Windows or "Samba"

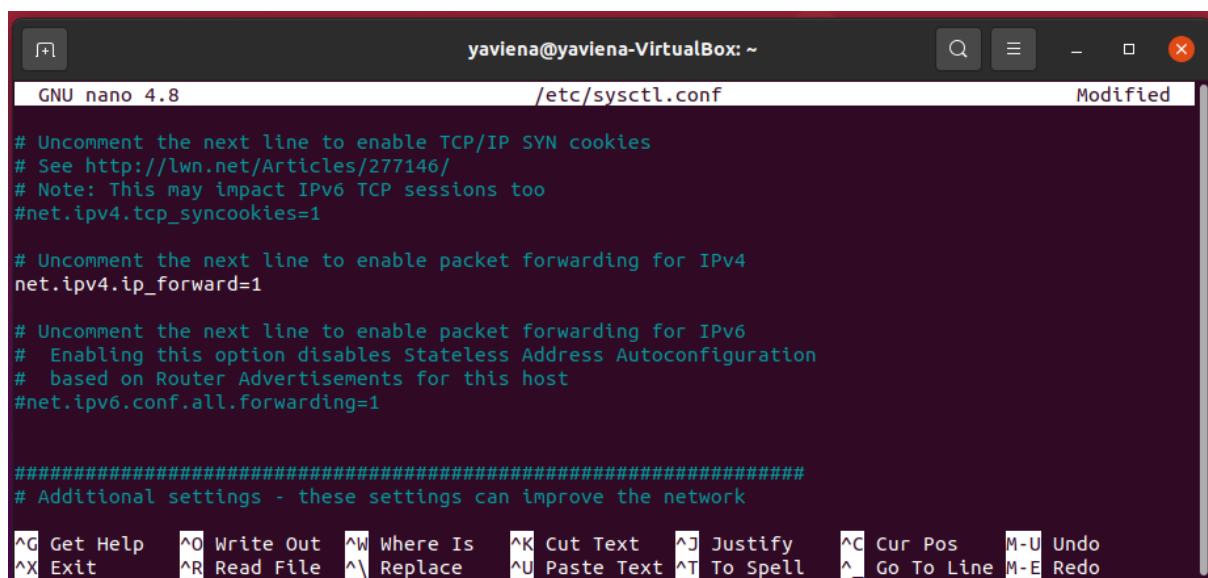
^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos      M-U Undo        M-A Mark Text
^X Exit          ^R Read File    ^_ Replace      ^U Paste Text   ^T To Spell     ^_ Go To Line   M-E Redo        M-G Copy Text
```

Domyślnie, w ubuntu przekazywanie pakietów IP (IP forwarding) jest wyłączone. Muszę to zmienić, aby przez moje połączenie VPN można było przeglądać Internet.

**sudo nano /etc/sysctl.conf**

modyfikacja wpisu z przekazywaniem pakietów

Odszukałam wpis "net.ipv4.ip\_forward=1" i usunęłam znak komentarza # przed wpisem.



```
GNU nano 4.8 /etc/sysctl.conf Modified

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1

#####
# Additional settings - these settings can improve the network

^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos      M-U Undo        M-A Mark Text
^X Exit          ^R Read File    ^_ Replace      ^U Paste Text   ^T To Spell     ^_ Go To Line   M-E Redo        M-G Copy Text
```

Potwierdziłam zmiany poleceniem:

**sudo sysctl -p**

potwierdzenie zmian w zmodyfikowanym wpisie

```
yaviena@yaviena-VirtualBox: ~  
yaviena@yaviena-VirtualBox:~$ sudo nano /etc/sysctl.conf  
[sudo] password for yaviena:  
yaviena@yaviena-VirtualBox:~$ sudo sysctl -p  
net.ipv4.ip_forward = 1  
yaviena@yaviena-VirtualBox:~$
```

Przystąpiłam do konfiguracji użytkowników. Zedytowałam plik, w którym umieściłam użytkowników oraz hasła. Plik miał postać:

<b>sudo nano /etc/ppp/chap-secrets</b>	konfiguruję użytkowników
--	--------------------------

```
GNU nano 4.8 /etc/ppp/chap-secrets Modified  
# Secrets for authentication using CHAP  
# client      server  secret          IP addresses  
yaviena pptpd 1234psd *  
  
^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos      M-U Undo        M-A Mark Text  
^X Exit          ^R Read File    ^_ Replace      ^U Paste Text  ^T To Spell    ^G Go To Line   M-E Redo        M-6 Copy Text
```

Dodałam kilka reguł do firewalla:

<b>iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE iptables -A FORWARD -i eth0 -o ppp0 -m state --state RELATED, ESTABLISHED -j ACCEPT iptables -A FORWARD -i ppp0 -o eth0 -j ACCEPT</b>
---

```
yaviena@yaviena-VirtualBox: /etc/ppp  
yaviena@yaviena-VirtualBox:/$ iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE  
Fatal: can't open lock file /run/xtables.lock: Permission denied  
yaviena@yaviena-VirtualBox:/$ sudo /etc/ppp  
[sudo] password for yaviena:  
sudo: /etc/ppp: command not found  
yaviena@yaviena-VirtualBox:/$ cd /etc/ppp  
yaviena@yaviena-VirtualBox:/etc/ppp$ iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE  
Fatal: can't open lock file /run/xtables.lock: Permission denied  
yaviena@yaviena-VirtualBox:/etc/ppp$ sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE  
yaviena@yaviena-VirtualBox:/etc/ppp$ sudo iptables -A FORWARD -i eth0 -o ppp0 -m state --state RELATED, ESTABLISHED -j ACCEPT  
yaviena@yaviena-VirtualBox:/etc/ppp$ sudo iptables -A FORWARD -i ppp0 -o eth0 -j ACCEPT  
yaviena@yaviena-VirtualBox:/etc/ppp$
```

Następnie zrestartowałam serwer PPTP.

<b>sudo /etc/init.d/pptpd restart</b>	restart serwera
---------------------------------------	-----------------

```
yaviena@yaviena-VirtualBox: ~  
yaviena@yaviena-VirtualBox:~$ sudo /etc/init.d/pptpd restart  
[sudo] password for yaviena:  
Restarting ptpd (via systemctl): ptpd.service.  
yaviena@yaviena-VirtualBox:~$
```

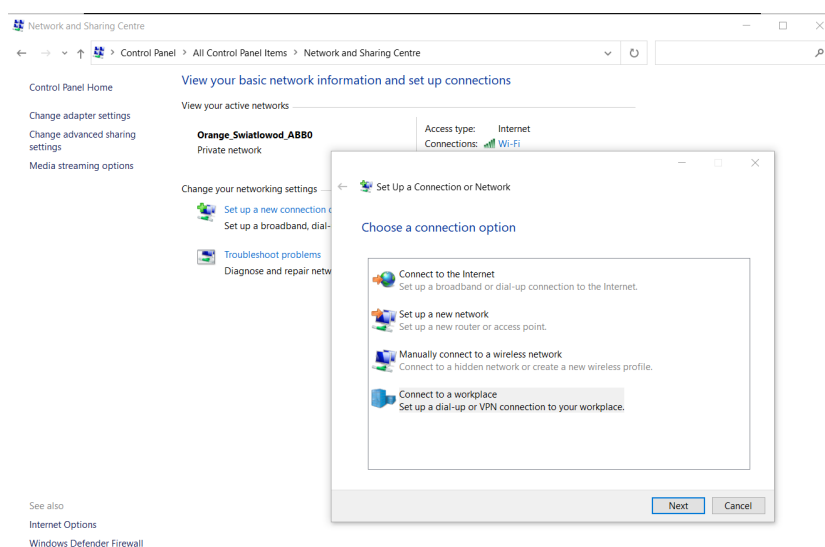
Na koniec sprawdziłam status zainstalowanego serwera PPTP.

service ptpd status	konfiguruję użytkowników
---------------------	--------------------------

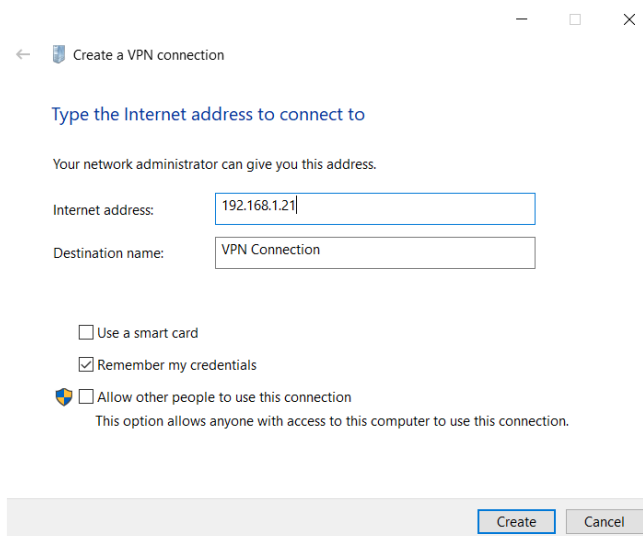
```
yaviena@yaviena-VirtualBox:~$ sudo service ptpd start  
yaviena@yaviena-VirtualBox:~$ sudo service ptpd status  
● ptpd.service - PoPToP Point to Point Tunneling Server  
   Loaded: loaded (/lib/systemd/system/pptpd.service; disabled; vendor preset: enabled)  
   Active: active (running) since Tue 2022-02-15 22:14:59 CET; 12min ago  
     Docs: man:pptpd(8)  
           man:pptpctrl(8)  
           man:pptpd.conf(5)  
  Main PID: 6734 (pptpd)  
    Tasks: 1 (limit: 4632)  
   Memory: 228.0K  
    CGroup: /system.slice/pptpd.service  
            └─6734 /usr/sbin/pptpd --fg  
  
lut 15 22:14:59 yaviena-VirtualBox systemd[1]: Started PoPToP Point to Point Tunneling Server.  
lut 15 22:14:59 yaviena-VirtualBox ptpd[6734]: MGR: Maximum of 100 connections reduced to 6, not enough  
lut 15 22:14:59 yaviena-VirtualBox ptpd[6734]: MGR: Manager process started  
lut 15 22:14:59 yaviena-VirtualBox ptpd[6734]: MGR: Maximum of 6 connections available  
yaviena@yaviena-VirtualBox:~$
```

### 3. Konfiguracja klienta PPTP (Windows 10)

W kliencie na systemie Windows 10 w ustawieniach sieci i połączeń utworzyłam nowe połączenie VPN.



Podaję adres mojego serwera VPN.



← Create a VPN connection

Type the Internet address to connect to


Your network administrator can give you this address.

Internet address: 192.168.1.21

Destination name: VPN Connection

☐ Use a smart card

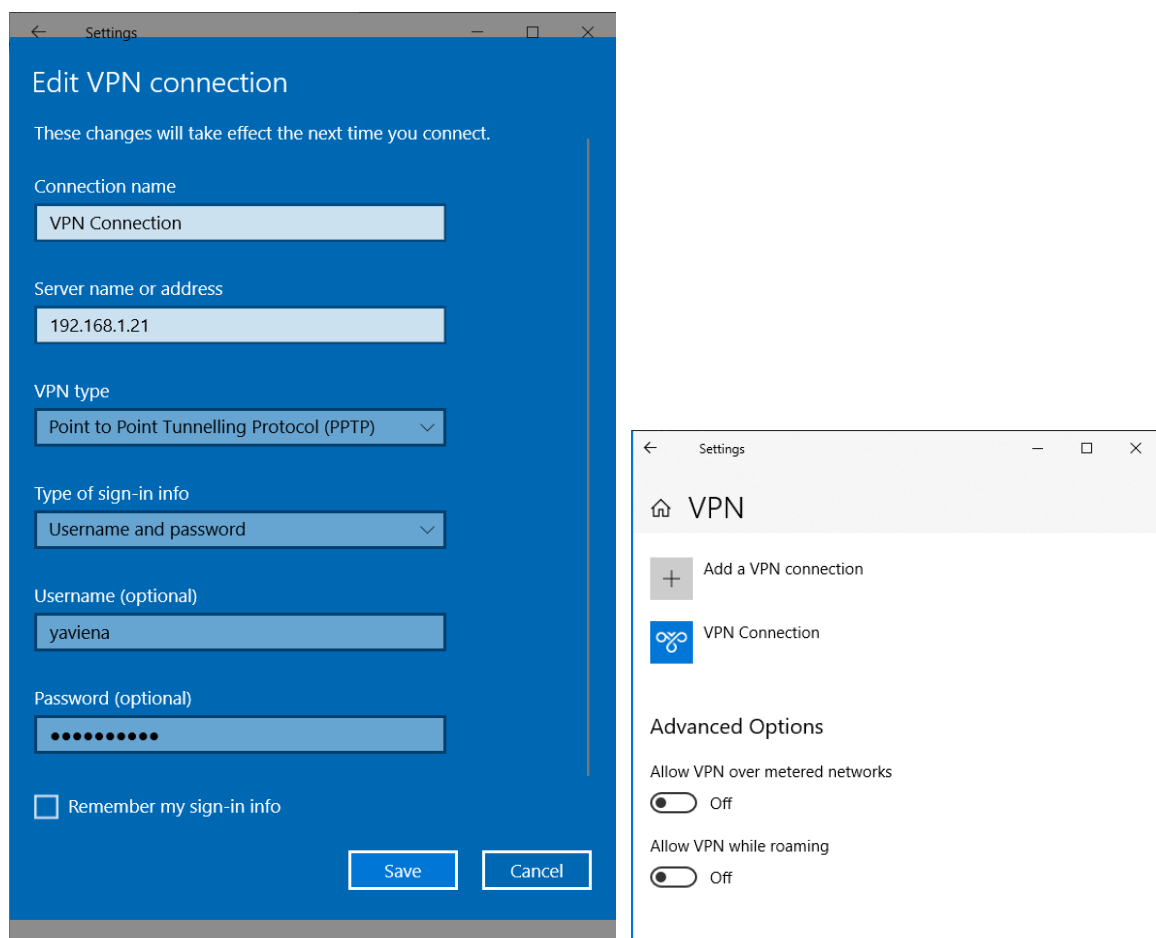
☒ Remember my credentials

 ☐ Allow other people to use this connection

This option allows anyone with access to this computer to use this connection.

Create Cancel

Zalogowałam się do nowego połączenia VPN podając utworzone wcześniej na serwerze hasło.



← Settings

### Edit VPN connection

These changes will take effect the next time you connect.

Connection name: VPN Connection

Server name or address: 192.168.1.21

VPN type: Point to Point Tunneling Protocol (PPTP)

Type of sign-in info: Username and password

Username (optional): yaviena

Password (optional): ••••••••

☐ Remember my sign-in info

Save Cancel

← Settings

### VPN

+ Add a VPN connection

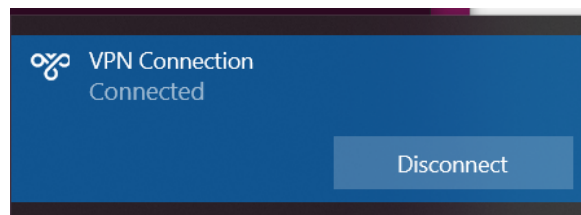
VPN Connection

#### Advanced Options

Allow VPN over metered networks: Off

Allow VPN while roaming: Off

Po konfiguracji nawiązałam połączenie z wcześniej utworzonym kontem (*yaviena*) logując się tym samym do serwera PPTP.



#### 4. Analiza Wiresharka

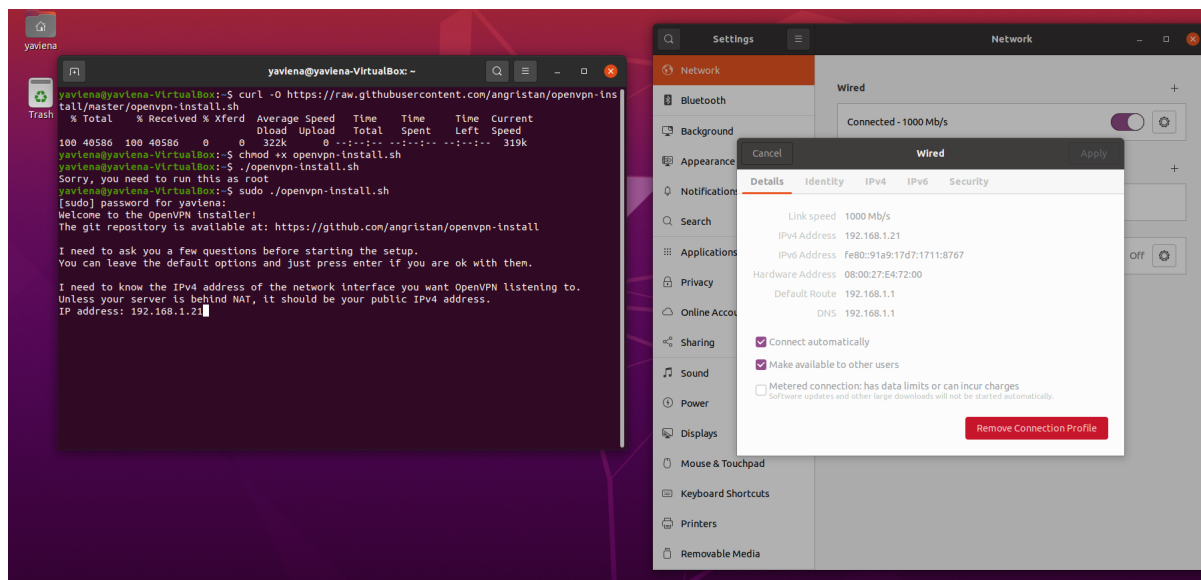
Po nawiązaniu połączenia przeanalizowałam ruch sieciowy z wykorzystaniem programu Wireshark. Przeprowadziłam analizę pakietu odpowiadającego za połączenie z serwerem PPTP. Widać gołym okiem przesłany zarówno login jak i hasło w analizowanym pakiecie. Oznacza to, że protokół ten nie jest bezpieczny.

20 10.406459	▼ Data
21 10.406521	Peer-ID-Length: 7
22 10.409103	Peer-ID: yaviena
23 10.409162	Password-Length: 7
24 10.409681	Password: 1234psd
25 10.908759	
<	0000 00 09 e9 55 c0 1
	0010 00 36 18 d3 00 0
> Frame 24: 64 bytes	0020 00 01 30 81 88 0
> Ethernet II, Src:	0030 00 02 ff 03 c0 2
> Internet Protocol	0040 69 78 69 61
> Generic Routing En	
> Point-to-Point Pro	
0000 00 14 00 0	
0010 00 32 1a e	
0020 00 02 30 8	
0030 00 06 ff 0	

## CZĘŚĆ II: Badanie ruchu w zabezpieczonym tunelu

### 1. Instalacja serwera OpenVPN (Ubuntu)

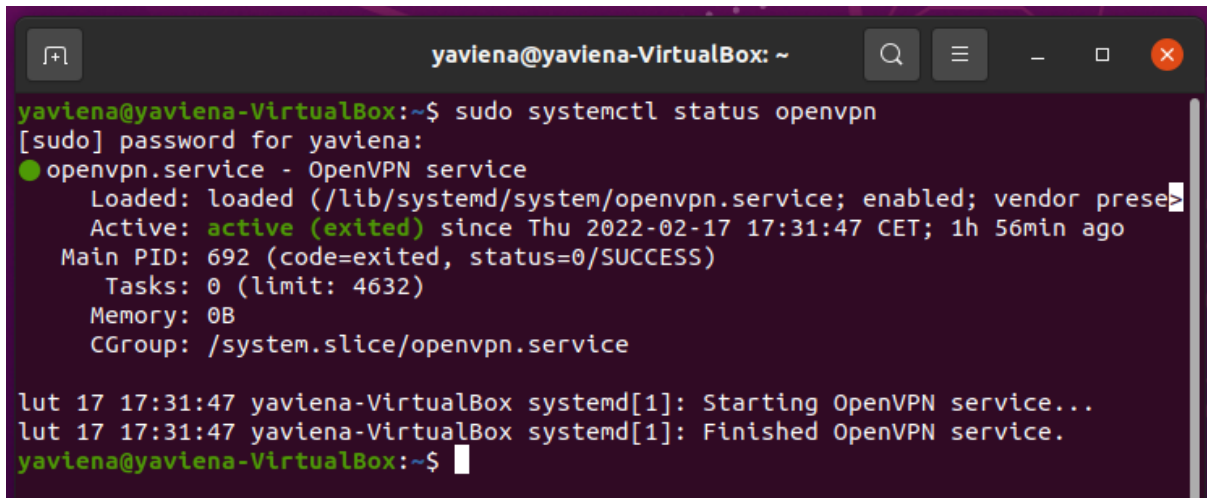
W drugiej części ćwiczenia zainstalowałam i skonfigurowałam na systemie Ubuntu w VirtualBoxie protokół OpenVPN.



W jego ustawieniach wybrałam maksymalną ochronę, tj. każdy z użytkowników posiada własne klucze autoryzacyjne (z certyfikatem), zaś transmisja jest chroniona kierunkowym kluczem TLS.

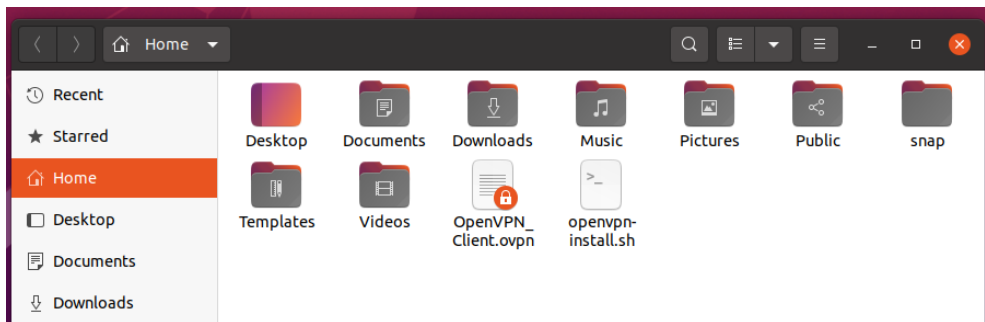
```
yaviena@yaviena-VirtualBox: ~  
* Applying /etc/sysctl.d/99-openvpn.conf ...  
net.ipv4.ip_forward = 1  
* Applying /etc/sysctl.d/99-sysctl.conf ...  
* Applying /usr/lib/sysctl.d/protect-links.conf ...  
fs.protected_fifos = 1  
fs.protected_hardlinks = 1  
fs.protected_regular = 2  
fs.protected_symlinks = 1  
* Applying /etc/sysctl.conf ...  
Created symlink /etc/systemd/system/multi-user.target.wants/openvpn@server.service → /etc/systemd/system/openvpn@server.service.  
Created symlink /etc/systemd/system/multi-user.target.wants/lptables-openvpn.service → /etc/systemd/system/lptables-openvpn.service.  
  
Tell me a name for the client.  
The name must consist of alphanumeric character. It may also include an underscore or a dash.  
Client name: OpenVPN_Client  
  
Do you want to protect the configuration file with a password?  
(e.g. encrypt the private key with a password)  
1) Add a passwordless client  
2) Use a password for the client  
Select an option [1-2]: 2  
⚠ You will be asked for the client password below ⚠  
  
Note: using Easy-RSA configuration from: /etc/openvpn/easy-rsa/vars  
Using SSL: openssl OpenSSL 1.1.1f 31 Mar 2020  
Generating an EC private key  
writing new private key to '/etc/openvpn/easy-rsa/pki/easy-rsa-15948.W5qdRL/tmp.8F8aNH'  
Enter PEM pass phrase:  
Verifying - Enter PEM pass phrase:  
-----  
Using configuration from /etc/openvpn/easy-rsa/pki/easy-rsa-15948.W5qdRL/tmp.Pvr4DM  
Check that the request matches the signature  
Signature ok  
The Subject's Distinguished Name is as follows  
commonName                :ASN.1 12:'OpenVPN_Client'  
Certificate is to be certified until May 22 16:59:02 2024 GMT (825 days)  
  
Write out database with 1 new entries  
Data Base Updated  
  
Client OpenVPN_Client added.  
  
The configuration file has been written to /home/yaviena/OpenVPN_Client.ovpn.  
Download the .ovpn file and import it in your OpenVPN client.  
yaviena@yaviena-VirtualBox:~$
```

Po ukończonej z sukcesem instalacji sprawdziłam poprawność zainstalowanego protokołu OpenVPN.



```
yaviena@yaviena-VirtualBox: ~  
yaviena@yaviena-VirtualBox:~$ sudo systemctl status openvpn  
[sudo] password for yaviena:  
● openvpn.service - OpenVPN service  
   Loaded: loaded (/lib/systemd/system/openvpn.service; enabled; vendor prese  
   Active: active (exited) since Thu 2022-02-17 17:31:47 CET; 1h 56min ago  
   Main PID: 692 (code=exited, status=0/SUCCESS)  
     Tasks: 0 (limit: 4632)  
    Memory: 0B  
    CGroup: /system.slice/openvpn.service  
  
lut 17 17:31:47 yaviena-VirtualBox systemd[1]: Starting OpenVPN service...  
lut 17 17:31:47 yaviena-VirtualBox systemd[1]: Finished OpenVPN service.  
yaviena@yaviena-VirtualBox:~$
```

Utworzony plik konfiguracyjny z nowym kontem użytkownika stworzonym na serwerze skopiowałam na komputer (z systemem Windows), który będzie klientem łączącym się do tegoż serwera.



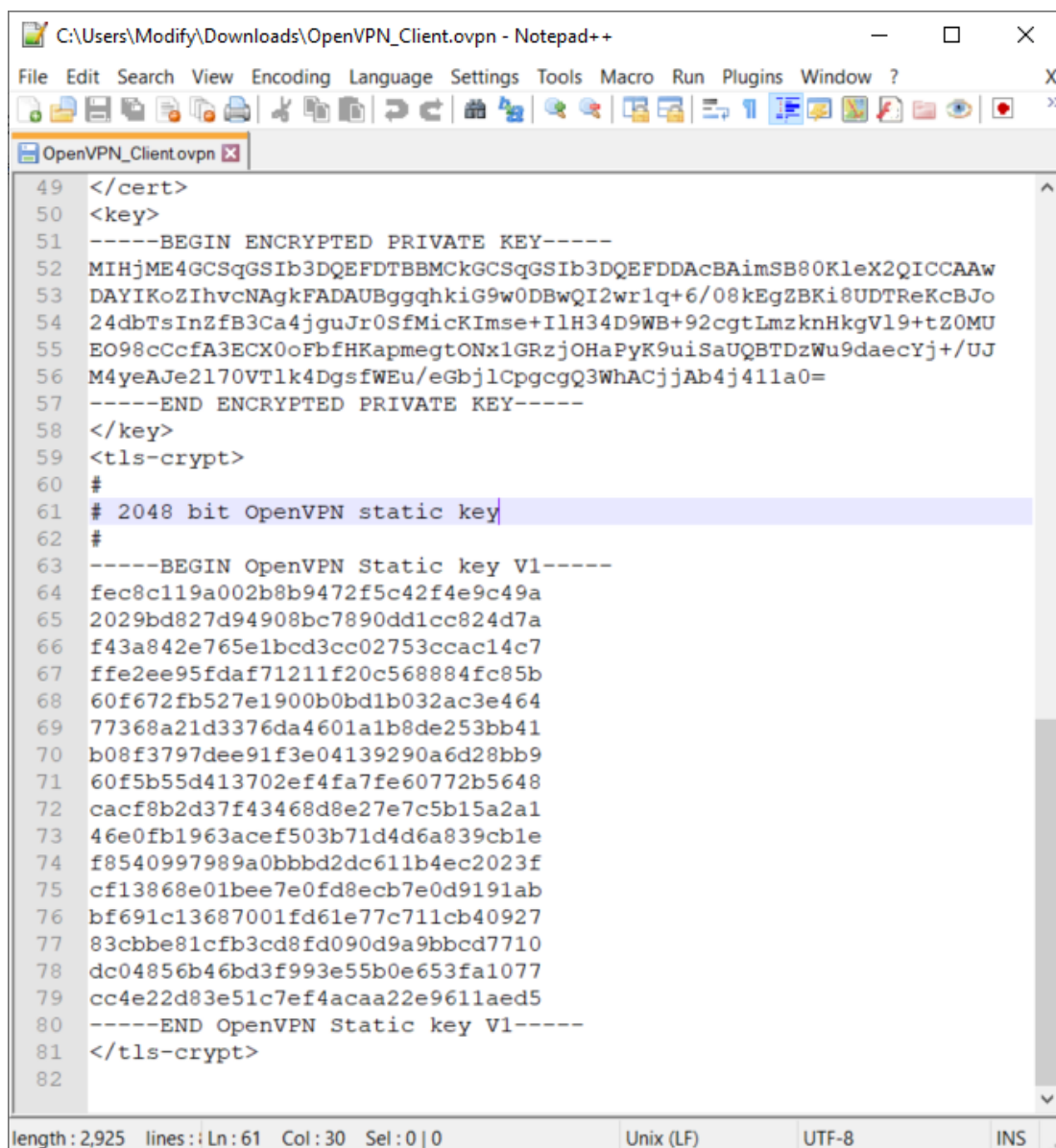
Skontrolowałam jak wygląda wygenerowany plik konfiguracyjny z kontem użytkownika. Widać w nim wyraźnie takie informacje jak m.in.:

- protokół połączenia,
- IP oraz port serwera, do którego można się połączyć,
- rodzaj szyfrowania,
- akceptowalną wersję protokołu TLS
- klucze certyfikatu.



```
C:\Users\Modify\Downloads\OpenVPN_Client.ovpn - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
OpenVPN_Client.ovpn
1 client
2 proto udp
3 explicit-exit-notify
4 remote 192.168.1.21 1194
5 dev tun
6 resolv-retry infinite
7 nobind
8 persist-key
9 persist-tun
10 remote-cert-tls server
11 verify-x509-name server_puHZ8hR967V3eLFJ name
12 auth SHA256
13 auth-nocache
14 cipher AES-128-GCM
15 tls-client
16 tls-version-min 1.2
17 tls-cipher TLS-ECDSA-WITH-AES-128-GCM-SHA256
18 ignore-unknown-option block-outside-dns
19 setenv opt block-outside-dns # Prevent Windows 10 DNS leak
20 verb 3
21 <ca>
22 -----BEGIN CERTIFICATE-----
23 MIIB2DCCAX2gAwIBAgIUQpfyBKcndNHltPNKk9CmyOpgxjEwCgYIKoZIzj0EAwIw
24 HjEcMBoGA1UEAwTY25fZ0g2UVd6SlBMcGpQMfplODAEfw0yMjAyMTcxNjUxMDda
25 Fw0zMjAyMTUxNjUxMDdaMB4xHDAaBgNVBAMME2NuX2dINlFXekpQTHBqUDBadTgw
26 WTATBgcqhkJOPQIBBggqhkJOPQMBBwNCAAQn09Zwd8bIYACofYHxY06Qapsi8+YP
27 zLqci4zTeYQlJPIiwvAq2kn51JcRHudOf/fEcNhDiiUoTSGJ/pQDGALXo4GYMIGV
28 MB0GA1UdDgQWBBSBSCpN9kTTONffc3t5BkgjwZGOMX7jBZBgNVHSMEUjBQgBSCpN9k
29 TTONffc3t5BkgjwZGOMX7qEipCAwHjEcMBoGA1UEAwTY25fZ0g2UVd6SlBMcGpQ
30 MFplOIIUQpfyBKcndNHltPNKk9CmyOpgxjEwDAYDVROTBAAUwAwEB/zALBgNVHQ8E
31 BAMCAQYwCgYIKoZIzj0EAwIDSQAARgIhAIL3LIwm4GXdx119Ss+VmZpDYFMxJy/o
32 dwOdSEEl4vw6AiEA2SZgfObegRwWmTO+7S8ccUCfKhJTea461IaMrSBkjZg=
33 -----END CERTIFICATE-----
34 </ca>
35 <cert>
36 -----BEGIN CERTIFICATE-----
37 MIIB4TCCAYegAwIBAgIRAME2ABOazWkph5xc3koKYyggCgYIKoZIzj0EAwIwHjEc
38 MBoGA1UEAwTY25fZ0g2UVd6SlBMcGpQMfplODAEfw0yMjAyMTcxNjU5MDJaFw0y
39 NDA1MjIxNjU5MDJaMBkxZzAVBgNVBAMMDk9wZW5WUE5fQ2xpZW50MFkwEwYHKoZI
40 zj0CAQYIKoZIzj0DAQcDQgAE0jc2MIOm2h/gGstbQkFpDXq5QwPyfhlUES/Aya88
41 LvRRQ3UzpXkqDCKc35A4Rr8IM7wZtXl9rh49U5CMhXqug6OBqjCBpzAJBgNVHRME
42 AjaAMB0GA1UdDgQWBQBQHzLeMM13DhDuaSZyK7KlQiJFRtBZBgNVHSMEUjBQgBSC
43 pN9kTTONffc3t5BkgjwZGOMX7qEipCAwHjEcMBoGA1UEAwTY25fZ0g2UVd6SlBM
44 cGpQMfplOIIUQpfyBKcndNHltPNKk9CmyOpgxjEwEwYDVROlBAwCgYIKwYBBQUH
45 AwIwCwYDVROlBAQDAgeAMAOGCCqGSM49BAMCA0gAMEUCIQc8RPbBT0UJ4eQfhLIb
46 psAQbh4+ChIMCzMdsd0Yf/nD2QIgInnaPvlyEyJ9VV8Erg4E76lJCPeIjLlKd6ds
47 88DfQB4=
48 -----END CERTIFICATE-----
49 </cert>
50 <key>
51 -----BEGIN ENCRYPTED PRIVATE KEY-----
length: 2,925 lines: Ln: 1 Col: 1 Sel: 0 | 0 Unix (LF) UTF-8 INS
```

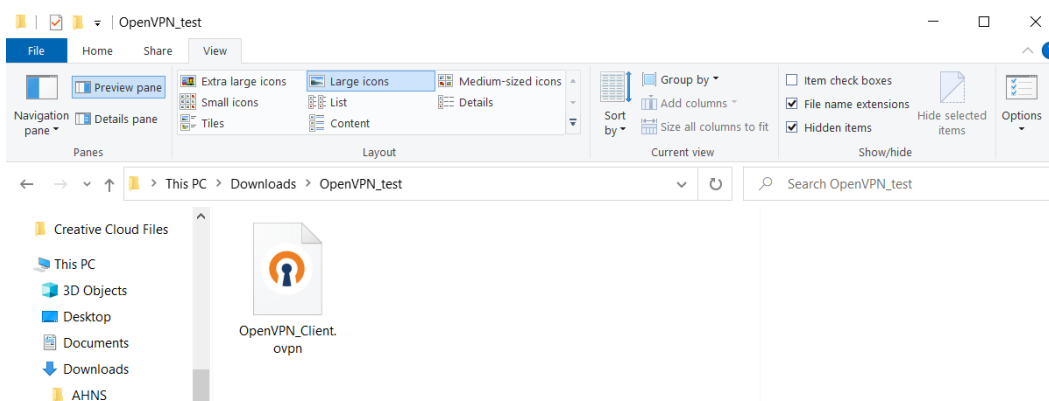




```
49 </cert>
50 <key>
51 -----BEGIN ENCRYPTED PRIVATE KEY-----
52 MIHjME4GCSqGSib3DQEFDTBBMCKGCSqGSib3DQEFDDAcBAImSB80KleX2QICCAAw
53 DAYIKoZIhvcNAGkFADAUBggghkiG9w0DBwQI2wrlq+6/08kEgZBKisUDTreKcBJo
54 24dbTsInZfB3Ca4jguJr0SfMicKImse+IlH34D9WB+92cgtLmzknHkgVl9+tZ0MU
55 EO98cCcfA3ECX0oFbfHKapmegtONx1GRzjOHaPyK9uiSaUQBTdZwU9daecYj+/UJ
56 M4yeAJe2l70VTlk4DgsfWEu/eGbjlCpgcgQ3WhACjjAb4j411a0=
57 -----END ENCRYPTED PRIVATE KEY-----
58 </key>
59 <tls-crypt>
60 #
61 # 2048 bit OpenVPN static key
62 #
63 -----BEGIN OpenVPN Static key V1-----
64 fec8c119a002b8b9472f5c42f4e9c49a
65 2029bd827d94908bc7890dd1cc824d7a
66 f43a842e765elbcd3cc02753ccac14c7
67 ffe2ee95daf71211f20c568884fc85b
68 60f672fb527e1900b0bd1b032ac3e464
69 77368a21d3376da4601a1b8de253bb41
70 b08f3797dee91f3e04139290a6d28bb9
71 60f5b55d413702ef4fa7fe60772b5648
72 cacf8b2d37f43468d8e27e7c5b15a2a1
73 46e0fb1963acef503b71d4d6a839cble
74 f8540997989a0bbbd2dc611b4ec2023f
75 cf13868e01bee7e0fd8ecb7e0d9191ab
76 bf691c13687001fd61e77c711cb40927
77 83cbbe81cfb3cd8fd090d9a9bbcd7710
78 dc04856b46bd3f993e55b0e653fa1077
79 cc4e22d83e51c7ef4acaa22e9611aed5
80 -----END OpenVPN Static key V1-----
81 </tls-crypt>
82
```

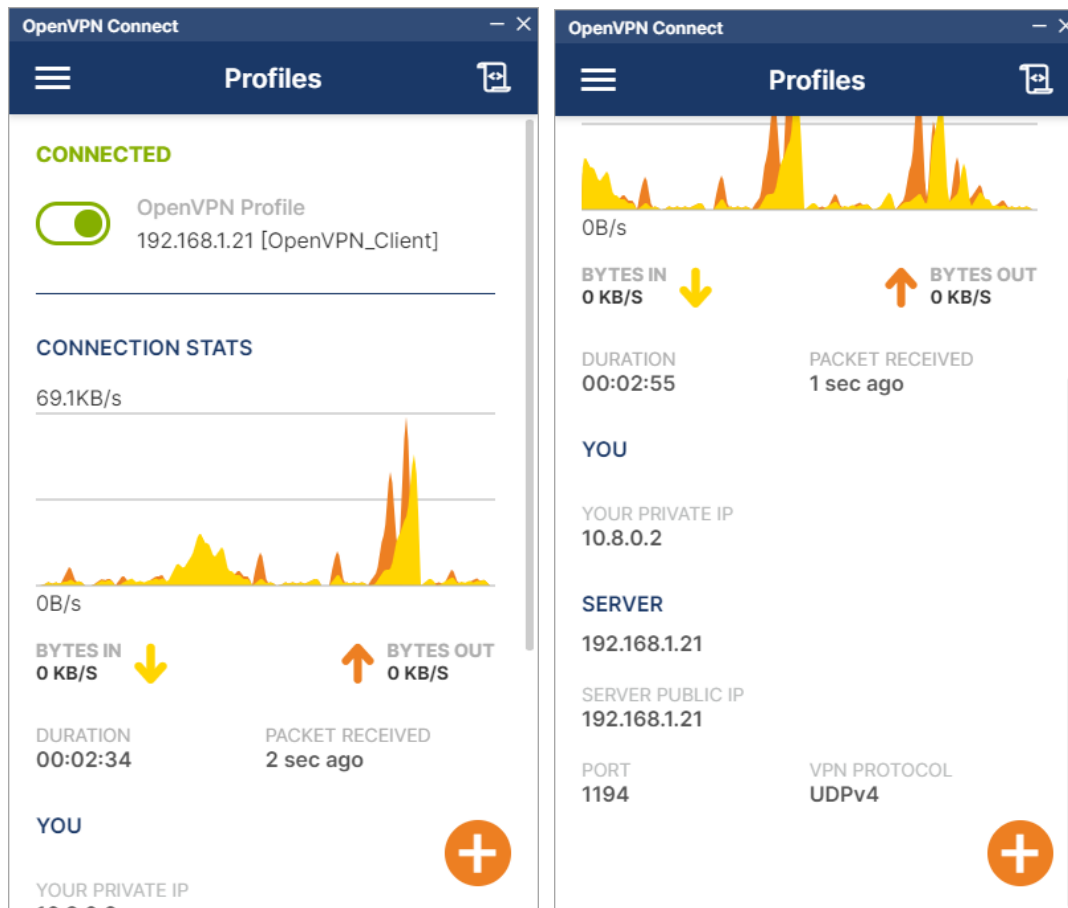
length: 2,925 lines: 61 Ln: 61 Col: 30 Sel: 0 | 0 Unix (LF) UTF-8 INS

Skopiowałam plik konfiguracyjny na komputer - klienta (z systemem Windows), który będzie łączył się do serwera.



## 2. Konfiguracja klienta OpenVPN (Windows 10)

Klient OpenVPN będzie łączył się z serwerem. Po dodaniu do klienta pliku konfiguracyjnego wykonałam połączenie i sprawdziłam jego status.



Upewniłam się, że mój adres IP na urządzeniu klienta został po połączeniu zmieniony na adres przypisany z serwera (10.8.0.2).

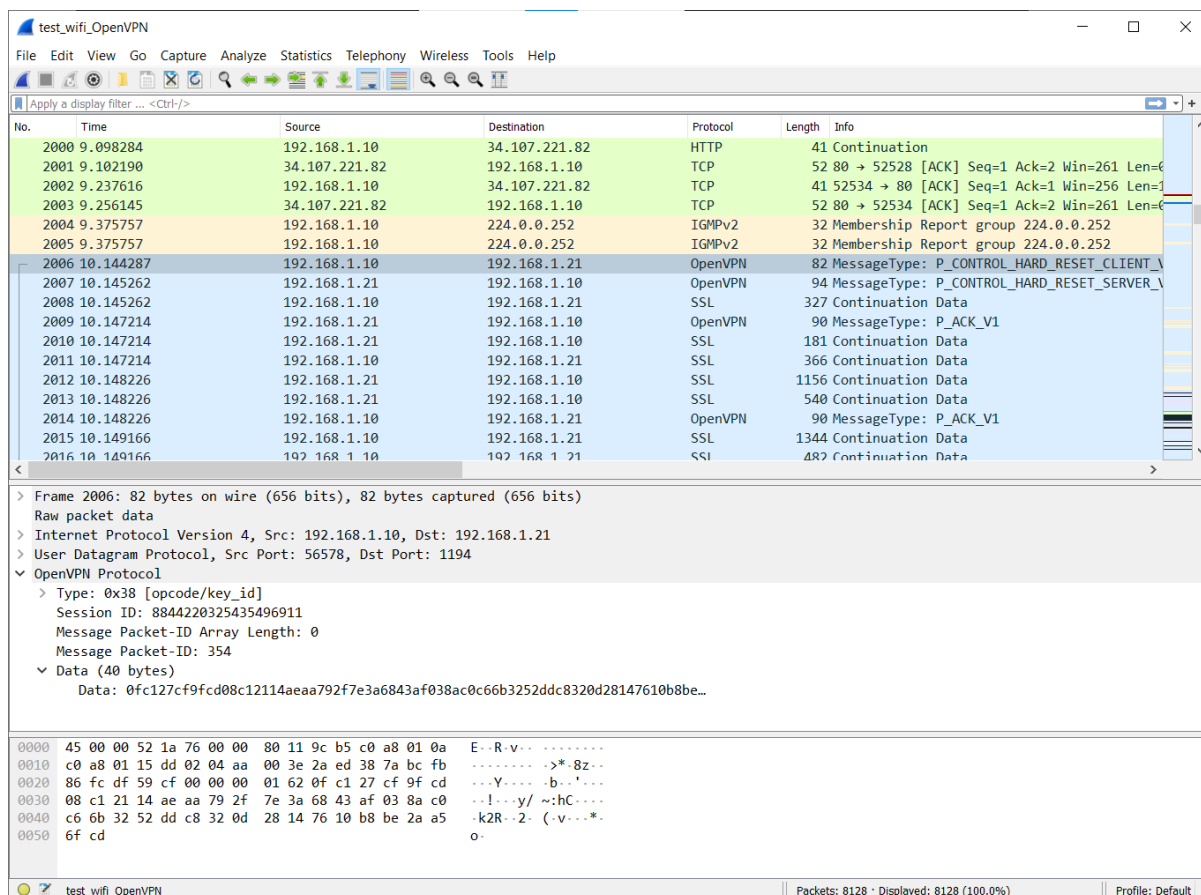
```
Unknown adapter Połączenie lokalne:

Connection-specific DNS Suffix  . : 
Description . . . . . : TAP-Windows Adapter V9 for OpenVPN Connect
Physical Address. . . . . : 00-FF-72-FB-EB-4E
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::2c8f:9fef:370b:a839%56(Preferred)
IPv4 Address. . . . . : 10.8.0.2(Preferred)
Subnet Mask . . . . . : 255.255.255.0
```

Upewniłam się, że pakiety danych na serwerze OpenVPN oraz ruch są przekierowywane poprawnie.

```
yaviena@yaviena-VirtualBox: ~  
yaviena@yaviena-VirtualBox:~$ sudo systemctl status openvpn  
[sudo] password for yaviena:  
● openvpn.service - OpenVPN service  
   Loaded: loaded (/lib/systemd/system/openvpn.service; enabled; vendor prese  
   Active: active (exited) since Thu 2022-02-17 17:31:47 CET; 1h 56min ago  
   Main PID: 692 (code=exited, status=0/SUCCESS)  
     Tasks: 0 (limit=4632)  
    Memory: 0B  
    CGroup: /system.slice/openvpn.service  
  
lut 17 17:31:47 yaviena-VirtualBox systemd[1]: Starting OpenVPN service...  
lut 17 17:31:47 yaviena-VirtualBox systemd[1]: Finished OpenVPN service.  
yaviena@yaviena-VirtualBox:~$
```

Programem Wireshark ponownie sprawdziłam jak będzie wyglądać proces logowania się do sieci poprzez protokół OpenVPN. Jak widać na poniższym zrzucie ekranu, klient jest podłączony do serwera i to przez niego przekierowywany jest cały ruch. Dodatkowo, wyraźnie widać kiedy nastąpiło połączenie poprzez OpenVPN. Nie można w tym wypadku odczytać danych logowania do serwera OpenVPN w żadnym z analizowanych pakietów.



# Wnioski

Celem laboratorium było przetestowanie działania tuneli sieciowych z wykorzystaniem technologii VPN.

Spośród wszystkich protokołów VPN, protokół PPTP jest jednym z najczęstszych, najłatwiejszych do skonfigurowania i najszybszych protokołów tuneli sieciowych. Z tego powodu, protokół PPTP jest przydatny w aplikacjach, w których prędkość jest najważniejsza, takich jak streaming audio lub wideo oraz na starszych, wolniejszych urządzeniach z bardziej ograniczonymi procesorami. Implementacja i konfiguracja protokołu PPTP jest prosta i wymaga zaangażowania o wiele mniejszych zasobów niż pozostałe protokoły VPN. Jednak, PPTP również wykazuje poważne luki w zabezpieczeniach. Podstawowe protokoły uwierzytelniania (zazwyczaj MS-CHAP-v1/v2) są zasadniczo niezabezpieczone i były wielokrotnie łamane w analizach bezpieczeństwa od czasu ich wprowadzenia. Z tego powodu PPTP nie jest zalecany z wyjątkiem przypadków, w których bezpieczeństwo nie jest absolutnie konieczne.

OpenVPN jest nieco bardziej skomplikowany w konfiguracji zarówno po stronie serwera jak i po stronie klienta, który będzie łączył się do serwera. Komunikacja za pośrednictwem OpenVPN ma jednak nieporównywalnie wiele zalet:

- obsługuje różne systemy operacyjne (np. Linux, Solaris, różne BSD, OS X, Windows oraz iOS i Android),
- stabilność,
- skalowalność (tysiące klientów),
- relatywnie łatwa instalacja,
- dynamiczne adresy IP i NAT,
- model zabezpieczeń OpenSSL,
- SSL/TLS oraz X509 PKI (Public Key Infrastructure) do uwierzytelniania sesji,
- protokół IPsec ESP do bezpiecznego transportu tunelowego przez UDP.

OpenVPN wydaje się idealnym rozwiązaniem, jednak można i tu wskazać kilka wad. Przede wszystkim mniejsza szybkość połączenia oraz proces konfiguracji technicznej - szczególnie po stronie klienta. To właśnie tam (klient) należy obsłużyć wygenerowany przez serwer plik konfiguracyjny i odpowiednio połączyć go z klientem.

Reasumując, PPTP to najszybszy protokół VPN za sprawą słabszego szyfrowania, któremu poddawane są przesyłane dane. Straty prędkości połączenia z siecią są przez to minimalne. Najczęściej stosowane w protokole PPTP jest szyfrowanie 128-bitowe (algorytmy RC4 i RSA ), które w porównaniu do 256-bitowego szyfrowania w nowszych rozwiązaniach nie zapewnia pełnego bezpieczeństwa.

Zamiast PPTP lepiej jest korzystać z OpenVPN, WireGuard, IKEv2 lub L2TP/IPSec. Są to bezpieczne protokoły, które nie mają poważnych luk bezpieczeństwa.