OVPN Server

Pertemuan 11
Fitri Setyorini
D3 PSDKU Sumenep
Semester Genap
2023-2024

What is VPN ?

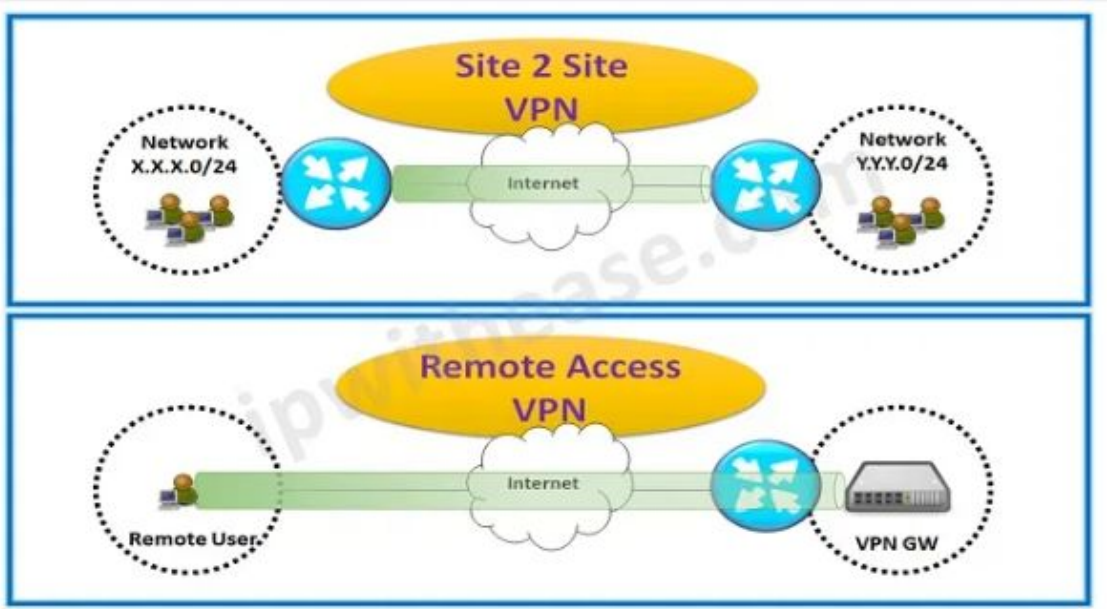
- VPN stands for "Virtual Private Network"
- VPN can establish a protected network connection when using public networks.
- VPNs encrypt your internet traffic and disguise your online identity.
- This makes it more difficult for third parties to track your activities online and steal data.
- The encryption takes place in real time.

How does a VPN work?

- A VPN hides your IP address by letting the network redirect it through a specially configured remote server run by a VPN host.
- This means that if you surf online with a VPN, the VPN server becomes the source of your data.
- This means your Internet Service Provider (ISP) and other third parties cannot see which websites you visit or what data you send and receive online.
- A VPN works like a filter that turns all your data into "gibberish". Even if someone were to get their hands on your data, it would be useless.

Type VPN :

- Remote Access VPN:
  - Remote access VPN digunakan oleh user yang ingin terkoneksi ke jaringan privat dan mengakses semua layanan dan resourcenya secara remote.
  - Untuk koneksi antara user dan privat network dilakukan via Internet dan koneksi bersifat secure dan privat
  - Misal : dosen PENS yang sedang di rumah menggunakan VPN untuk mengakses mis milik PENS. MIS PENS pada awalnya hanya dapat diakses secara internal via jaringan kampus. Namun semenjak kebijakan WFH, MIS dapat diakses via internet dari luar kampus. Selainitu, VPN dapat pula digunakan oleh user untuk mengakses website yang diblok. VPN juga dapat digunakan oleh user yang menginginkan privacy dan sekuritas
- Site to Site VPN:
  - Site-to-Site VPN disebut juga Router-to-Router VPN, biasa digunakan oleh perusahaan besar yang memiliki beberapa cabang di beberapa kota yang berbeda . Ada 2 tipe site to site:
  - Intranet based VPN: Beberapa cabang dari perusahaan yang sama saling terkoneksi lewat tipe Site-to-Site VPN
  - Extranet based VPN: Ketika beberapa perusahaan yang berbeda menggunakan tipe Site-to-site VPN


Site 2 Site
VPN
Network
X.X.X.0/24
Internet
Network
Y.Y.Y.0/24
Remote Access
VPN
Internet
Remote User
VPN GW

Teknologi VPN :

- Internet Protocol Security (IPSec):
  - Internet Protocol Security, known as IPSec, is used to secure Internet communication across an IP network. IPSec secures Internet Protocol communication by verifying the session and encrypts each data packet during the connection.
  - IPSec runs in 2 modes: (i) Transport mode (ii) Tunneling mode
  - The work of transport mode is to encrypt the message in the data packet and the tunneling mode encrypts the whole data packet. IPSec can also be used with other security protocols to improve the security system.
- Layer 2 Tunneling Protocol (L2TP):
  - L2TP or Layer 2 Tunneling Protocol is a tunneling protocol that is often combined with another VPN security protocol like IPSec to establish a highly secure VPN connection. L2TP generates a tunnel between two L2TP connection points and IPSec protocol encrypts the data and maintains secure communication between the tunnel.
- Point–to–Point Tunneling Protocol (PPTP):
  - PPTP or Point-to-Point Tunneling Protocol generates a tunnel and confines the data packet. Point-to-Point Protocol (PPP) is used to encrypt the data between the connection. PPTP is one of the most widely used VPN protocol and has been in use since the early release of Windows. PPTP is also used on Mac and Linux apart from Windows.

- SSL and TLS:
  - SSL (Secure Sockets Layer) and TLS (Transport Layer Security) generate a VPN connection where the web browser acts as the client and user access is prohibited to specific applications instead of entire network. Online shopping websites commonly uses SSL and TLS protocol. It is easy to switch to SSL by web browsers and with almost no action required from the user as web browsers come integrated with SSL and TLS. SSL connections have “https” in the initial of the URL instead of “http”.
- OpenVPN:
  - OpenVPN is an open source VPN that is commonly used for creating Point-to-Point and Site-to-Site connections. It uses a traditional security protocol based on SSL and TLS protocol.
- Secure Shell (SSH):
  - Secure Shell or SSH generates the VPN tunnel through which the data transfer occurs and also ensures that the tunnel is encrypted. SSH connections are generated by a SSH client and data is transferred from a local port on to the remote server through the encrypted tunnel

Opensource VPN Software

- Algo
- WireGuard
- PPTP
- SoftEther
- OpenVPN 2.x
- eduVPN - Let's Connect!
- OpenConnect
- LibreSwan
- dll

OpenVPN

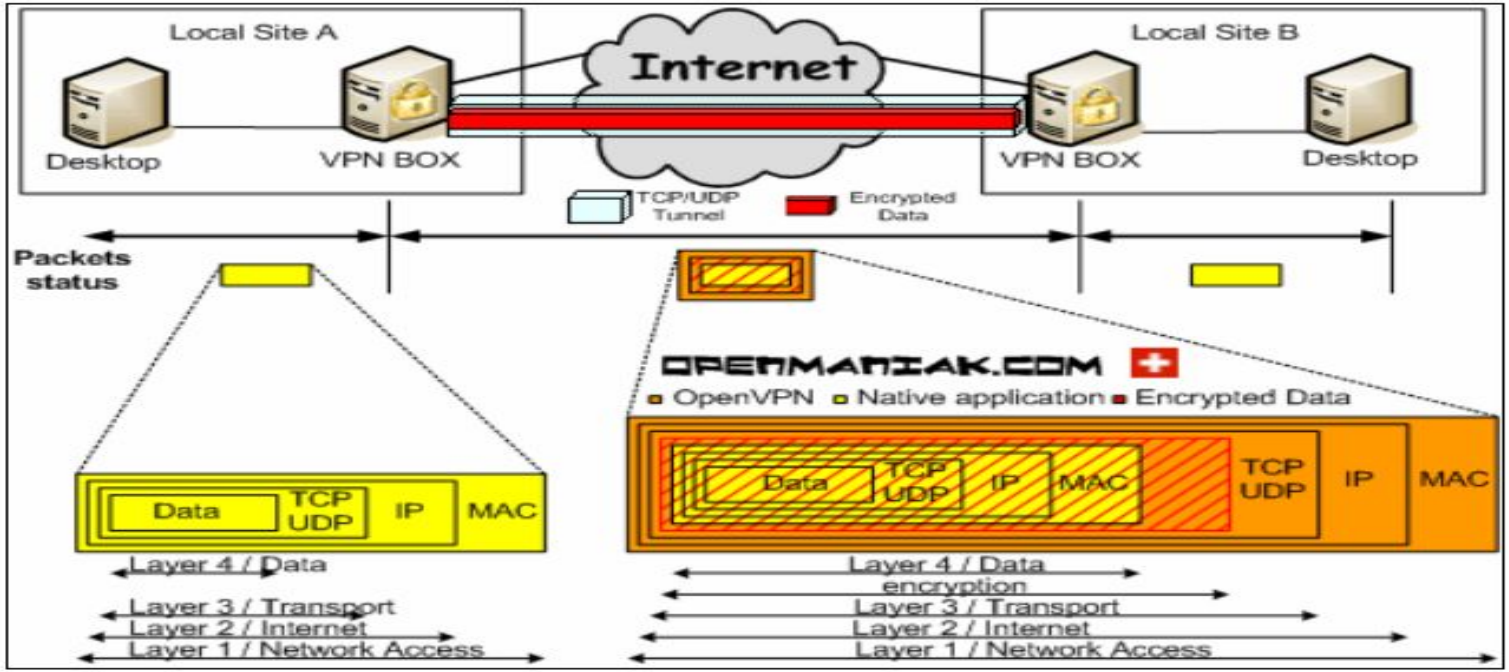
- OpenVPN is a virtual private network (VPN) system that implements techniques to create secure point-to-point or site-to-site connections in routed or bridged configurations and remote access facilities.
- It implements both client and server applications.
- OpenVPN allows peers to authenticate each other using pre-shared secret keys, certificates or username/password.
- When used in a multiclient-server configuration, it allows the server to release an authentication certificate for every client, using signatures and certificate authority.
- It uses the OpenSSL encryption library extensively, as well as the TLS protocol, and contains many security and control features. It uses a custom security protocol[11] that utilizes SSL/TLS for key exchange. It is capable of traversing network address translators (NATs) and firewalls.
- OpenVPN has been ported and embedded to several systems. For example, DD-WRT has the OpenVPN server function. SoftEther VPN, a multi-protocol VPN server, also has an implementation of OpenVPN protocol.
- It was written by James Yonan and is free software, released under the terms of the GNU General Public License version 2 (GPLv2).[12] Additionally, commercial licenses are available.[13]

OpenVPN: Pros and cons

You've looked through a lot of explanations, now let me put it simply. What will you get, and what will you sacrifice if you use OpenVPN?

OpenVPN's pros	OpenVPN's cons
Better security	Slower speed
Strong encryption	Manual setup
Reliable connection	May require 3rd party applications

If the pros outweigh the cons for you, let's see how you can use it!


Local Site A
Internet
Local Site B
Desktop
VPN BOX
VPN BOX
Desktop
TCP/UDP Tunnel
Encrypted Data
Packets status
OPENMANIAK.COM
OpenVPN
Native application
Encrypted Data
Data
TCP UDP
IP
MAC
Layer 4 / Data
Layer 3 / Transport
Layer 2 / Internet
Layer 1 / Network Access
Data
TCP UDP
IP
MAC
TCP UDP
IP
MAC
Layer 4 / Data encryption
Layer 3 / Transport
Layer 2 / Internet
Layer 1 / Network Access

Komponen OpenVPN

- PKI
- CA

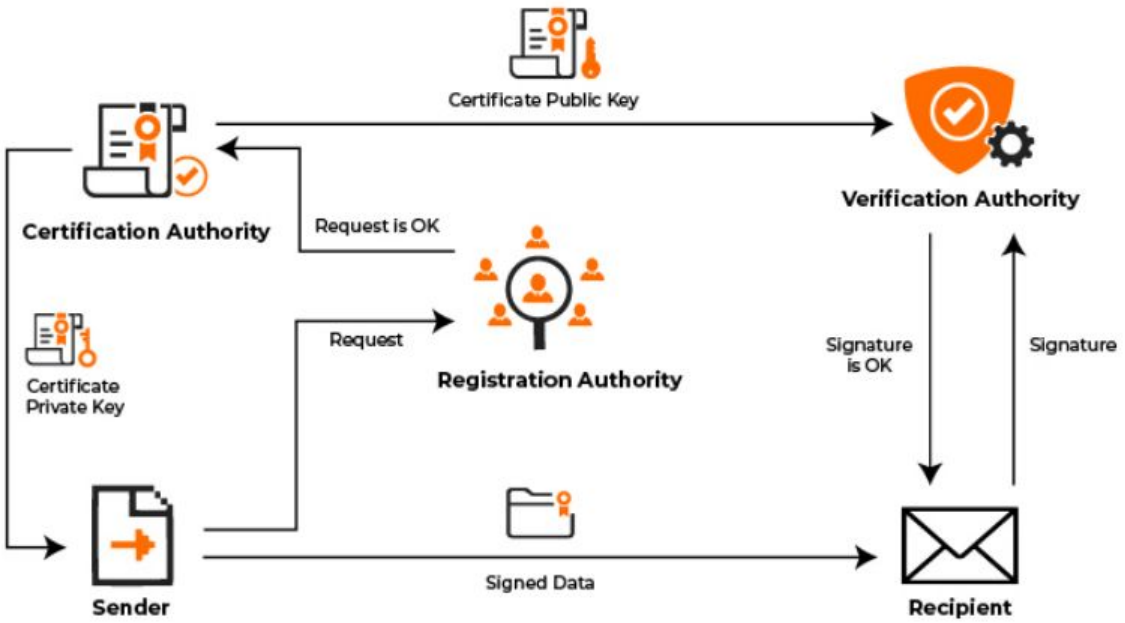
PKI

- PKI (Public Key Infrastructure), is a framework that enables the encryption of public keys and includes their affiliated crypto-mechanisms.
- The underlying purpose of any PKI setup is to manage the keys and certificates associated with it, thereby creating a highly secure network environment for use by applications and hardware.
- X.509 certificates and public keys form the cornerstone of PKI, acting as the mechanism through which cryptography can be established for an endpoint
    - – consequently, PKI may refer to any software, policy, process, or procedure that may be employed while configuring and managing those certificates and keys.

What is PKI used for?

- In a nutshell, PKI is responsible for making online interactions more secure, and it does this by:
  - Establishing the identity of endpoints on a network
  - Encrypting the flow of data via the network's communication channels
  - It does this by using private keys and public keys for encryption and decryption respectively, which are facilitated in turn by digital certificates

Public Key Infrastructure



What is a certificate authority (CA)?

- A certificate authority (CA) is a trusted entity that issues Secure Sockets Layer (SSL) certificates.
- These digital certificates are data files used to cryptographically link an entity with a public key.
- Web browsers use them to authenticate content sent from web servers, ensuring trust in content delivered online.
- As providers of these certificates, CAs are a reliable and critical trust anchor of the internet's public key infrastructure (PKI).
- They help secure the internet for both organizations and users.
- The main goal of a CA is to verify the authenticity and trustworthiness of a website, domain and organization so users know exactly who they're communicating with online and whether that entity can be trusted with their data.
- When a CA issues a digital certificate for a website, users know they are connected with an official website, not a fake or spoofed website created by a hacker to steal their information or money.

Key roles of a certificate authority

- issues digital certificates;
- helps establish trust between communicating entities over the internet;
- verifies domain names and organizations to validate their identities; and
- maintains certificate revocation lists.

Praktikum OpenVPN Server

Requirement

- 3 VM Linux :
  - 1 CA
  - 1 OpenVPN Server
  - 1 OpenVPN Client
- Gunakan Bridged Network

1. CA: Konfigurasi

- Ubah hostname anda agar lebih mudah dikenali

  #hostnamectl set-hostname DEB12CA

- Update database Linux

  #apt update

- Install easy-rsa

  #apt install easy-rsa

```
root@DEB12-CA2:~# apt install easy-rsa
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libccid opensc opensc-pkcs11 pcscd
Suggested packages:
  pcmciautils
The following NEW packages will be installed:
```

- Masuk sebagai user biasa

  #su - fitri
- Buat direktory bernama easy-rsa, di homedir user (fitri), yaitu: /home/fitri

  $mkdir ~/easy-rsa
- Buat link directory /usr/share/easy-rsa/* ke direktory easy-rsa

  $ ln -s /usr/share/easy-rsa/* ~/easy-rsa/

```
root@DEB12-CA2:~# su - fitri
fitri@DEB12-CA2:~$ mkdir easy-rsa
fitri@DEB12-CA2:~$ ln -s /usr/share/easy-rsa/* ~/easy-rsa/
fitri@DEB12-CA2:~$ chmod 700 /home/fitri/easy-rsa
```

- Cek link di /home/fitri/easyrsa

  $cd ~/easyrsa

  $ls -l

```
fitri@DEB12-CA2:~$ cd easy-rsa/
fitri@DEB12-CA2:~/easy-rsa$ ls -l
total 4
lrwxrwxrwx 1 fitri fitri   27 May 16 07:04 easyrsa -> /usr/share/easy-rsa/easyrsa
lrwxrwxrwx 1 fitri fitri   39 May 16 07:04 openssl-easyrsa.cnf -> /usr/share/easy-rsa/o
penssl-easyrsa.cnf
drwx------ 7 fitri fitri 4096 May 16 07:32 pki
lrwxrwxrwx 1 fitri fitri   32 May 16 07:04 vars.example -> /usr/share/easy-rsa/vars.exa
mple
lrwxrwxrwx 1 fitri fitri   30 May 16 07:04 x509-types -> /usr/share/easy-rsa/x509-types
```

- Batasi akses ke directory /home/fitri/easy-rsa. Hanya user yang memiliki hak read, write dan execute

  $ chmod 700 /home/fitri/easy-rsa

- Cek hak akses /home/fitri/easy-rsa. Hak akses sudah berubah jadi 700 (rwx— —)

  $ cd ~

  $ ls -l

```
fitri@DEB12-CA2:~/easy-rsa$ cd ..
fitri@DEB12-CA2:~$ ls -l
total 40
drwxr-xr-x 2 fitri fitri 4096 Sep 15  2023 Desktop
drwxr-xr-x 2 fitri fitri 4096 Sep 15  2023 Documents
drwxr-xr-x 2 fitri fitri 4096 Sep 15  2023 Downloads
drwx------ 3 fitri fitri 4096 May 16 07:05 easy-rsa
drwxr-xr-x 2 fitri fitri 4096 Sep 15  2023 Music
drwxr-xr-x 2 fitri fitri 4096 Sep 15  2023 Pictures
drwxr-xr-x 2 fitri fitri 4096 Sep 15  2023 Public
drwxr-xr-x 2 fitri fitri 4096 Sep 15  2023 Templates
drwxr-xr-x 2 fitri fitri 4096 May 16 07:30 tmp
drwxr-xr-x 2 fitri fitri 4096 Sep 15  2023 Videos
```

- Inisialisasi PKI

  $ cd ~/easy-rsa

  $ ./easyrsa init-pki

```
fitri@DEB12-CA2:~$ cd easy-rsa/
fitri@DEB12-CA2:~/easy-rsa$ ./easyrsa init-pki
* Notice:

  init-pki complete; you may now create a CA or requests.

  Your newly created PKI dir is:
  * /home/fitri/easy-rsa/pki

* Notice:
  IMPORTANT: Easy-RSA 'vars' file has now been moved to your PKI above.
```

- Sekarang kita akan membuat CA (Certificate Authority)
- Sebelum membuat CA private key dan certificate, anda harus setting variabel telah dibuat di /pki/vars.
- Edit file konfigurasi sebagai berikut

- Masuk ke directory easy-rsa, edit file pki

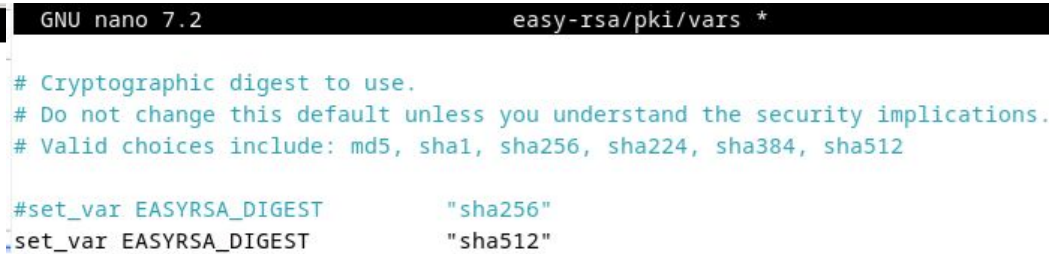
  $ cd ~/easy-rsa

  $ nano pki/vars

- Ubah baris dibawah sebagai berikut

  set_var EASYRSA_ALGO ec

  set_var EASYRSA_DIGEST "sha512"



- Simpan dan exit
- Jalankan perintah dibawah untuk menciptakan private dan public key untuk CA anda
- Gunakan opsi nopass agar anda tidak perlu memasukkan password

  $ ./easyrsa build-ca nopass

  Perintah ini akan menciptakan 2 file :

  - ~/easy-rsa/pki/ca.crt : CA public key file.
    - Semua user dan OpenVPN Server membutuhkan salinan file ini.
  - ~/easy-rsa/pki/private/ca.key : CA private key file
    - yang dipakai untuk menandatangani sertifikat dari OpenVPN server.

- Saat ditanya Common Name, kosongkan saja, karena kita akan menggunakan nilai default yaitu Easy-RSA
-

```
fitri@DEB12-CA2:~/easy-rsa$ ./easyrsa build-ca nopass
* Notice:
Using Easy-RSA configuration from: /home/fitri/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.9 30 May 2023 (Library: OpenSSL 3.0.9 30 May 2023)

Using configuration from /home/fitri/easy-rsa/pki/f1ff2a48/temp.fd61ba58
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Common Name (eg: your user, host, or server name) [Easy-RSA CA]:

* Notice:

CA creation complete and you may now import and sign cert requests.
Your new CA certificate file for publishing is at:
/home/fitri/easy-rsa/pki/ca.crt
```

2. OpenVPNServer : Instalasi Openvpn

- Ubah hostname anda agar lebih mudah dikenali

  #hostnamectl set-hostname DEB12VPN
- Update database Linux

  #apt update
- Sekarang, kita install dulu openvpn dan easy-rsa

  #apt install openvpn easy-rsa
- Masuk sebagai user biasa

  #su - fitri

- Buat direktory bernama easy-rsa, di homedir user (fitri), yaitu: /home/fitri

  $mkdir ~/easy-rsa

- Buat link directory /usr/share/easy-rsa/* ke direktory easy-rsa

  $ ln -s /usr/share/easy-rsa/* ~/easy-rsa/

- Batasi akses ke directory /home/fitri/easy-rsa. Hanya user yang memiliki hak read, write dan execute

  $ chmod 700 /home/fitri/easy-rsa

3. OpenVPNServer : Membuat PKI

- Sekarang masuk ke /easy-rsa dan lakukan inisiasi pki. PKI di ServerVPN digunakan untuk sentral penyimpanan certificate requests dan sertifikat publik

  $ cd ~/easy-rsa

  $ ./easyrsa init-pki
- Buka file /easy-rsa/pki/vars untuk kita edit

  $nano pki/vars
- Ubah baris dibawah sebagai berikut

  set_var EASYRSA_ALGO ec

  set_var EASYRSA_DIGEST "sha512"
- Simpan dan Exit

- Karena kita tidak menggunakan untuk CA, maka hanya 2 value ini yg kita butuhkan.
- VPN server dan CA menggunakan ECC, yang artinya, ketika server dan client hendak membuat key simetrik, akan digunakan algoritma Elliptic Curve. Ini lebih cepat dari Diffie-Hellman dengan algoritma RSA, karena angka yang dipakai lebih sedikit dan komputasinya lebih cepat .

4. OpenVPNServer : Membuat Certificate Request and Private Key

- Buka direktory easy-rsa

  $ cd ~/easy-rsa
- Sekarang, generate private key dan Certificate Signing Request di VPN Server.
- Gunakan perintah gen-req yang diikuti oleh Common NAme (CN) dari server VPN.
- Kita akan menggunakan server sebagai CN untuk VPN Server
- Kita juga menggunakan nopass untuk menghindara memasukkan password

  $ ./easyrsa gen-req server nopass

  Disini anda memasukkan common name : server

- Ketika anda diminta memasukkan CN, kosongkan saja

```
* Notice:
Using Easy-RSA configuration from: /home/<username>/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.9 30 May 2023 (Library: OpenSSL 3.0.9 30 May 2023)

-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Common Name (eg: your user, host, or server name) [server]:
* Notice:

Keypair and certificate request completed. Your files are:
req: /home/<username>/easy-rsa/pki/reqs/server.req
key: /home/<username>/easy-rsa/pki/private/server.key
```

- Kopikan server key ke /etc/openvpn/server directory

  $su - root

  #cp /home/fitri/easy-rsa/pki/private/server.key /etc/openvpn/server/
- Certificate Signing Request (CSR) siap ditandatangani CA.

5. Menandatangani CSR dari VPN Server

- Tahap berikutnya adalah membuat CA menandatangani CSR dari VPN Server
- CSR dikirim ke CA lewat perintah scp
- Baik pada VPN Server dan CA sebelumnya, install terlebih dahulu openssh-server dan lakukan konfigurasi
- Lakukan instalasi openssh-server di VPN Server dan CA

  #apt install openssh-server
- Cek nomor IP dari VPN Server dan CA

  #ip address
- Buka file /etc/ssh/sshd_config. Edit ListenAddress

  #nano /etc/ssh/sshd_config.

```
fitri@DEB12-VPN2:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOW
1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_code
ult qlen 1000
    link/ether 00:0c:29:74:8e:0b brd ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.220.136/24 brd 192.168.220.255 scope global dynamic noprefixroute ens3
```

```
  GNU nano 7.2                          /etc/ssh/sshd_config
# default value.

Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
ListenAddress 192.168.220.136
#ListenAddress ::
```

```
fitri@DEB12-CA2:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen
1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq
ult qlen 1000
    link/ether 00:0c:29:06:0a:7d brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.220.133/24 brd 192.168.220.255 scope global dynamic noprefixroute ens3
3
       valid_lft 1602sec preferred_lft 1602sec
    inet6 fe80::20c:29ff:fe06:a7d/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

```
  GNU nano 7.2                          /etc/ssh/sshd_config

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
ListenAddress 192.168.220.133
#ListenAddress ::
```

- Simpan dan exit
- Jalankan ssh server di OpenVPN Server dan CA

  #systemctl restart ssh

  #systemctl status ssh

```
root@DEB12-CA2:~# systemctl restart ssh
root@DEB12-CA2:~# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; preset: enabled)
     Active: active (running) since Thu 2024-05-16 10:22:27 PDT; 5s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
```

```
root@DEB12-VPN2:~# systemctl restart ssh
root@DEB12-VPN2:~# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; enabled; preset: enabled)
     Active: active (running) since Thu 2024-05-16 07:17:27 PDT; 7s ago
       Docs: man:sshd(8)
             man:sshd_config(5)
```

- Buat direktory tmp di /home/fitri untuk VPN Server dan CA

  $su - fitri

  $mkdir tmp

- Lakukan scp dari VPN Server. IP 192.168.220.133 adalah IP CA

  $ scp /home/fitri/easy-rsa/pki/reqs/server.req fitri@192.168.220.133:/home/fitri/tmp

```
fitri@DEB12-VPN2:~$ scp /home/fitri/easy-rsa/pki/reqs/server.req fitri@192.168.220.133:
/home/fitri/tmp
The authenticity of host '192.168.220.133 (192.168.220.133)' can't be established.
ED25519 key fingerprint is SHA256:UrYM4Hcb5TZY9ZpvA5vJPKThaME1Y4fmI93NPeoqtT4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.220.133' (ED25519) to the list of known hosts.
fitri@192.168.220.133's password:
server.req                                    100%  436   144.7KB/s   00:00
```

- SCP akan mengirim CSR (server.req) dari VPN Server ke CA
- Di CA, lakukan import CSR

$ cd ~/easy-rsa

$ ./easyrsa import-req /home/fitri/tmp/server.req server

```
fitri@DEB12-CA2:~/easy-rsa$ ./easyrsa import-req /home/fitri/tmp/server.req server
* Notice:
Using Easy-RSA configuration from: /home/fitri/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.11 19 Sep 2023 (Library: OpenSSL 3.0.11 19 Sep 2023)

* Notice:

The request has been successfully imported with a short name of: server
You may now use this name to perform signing operations on this request.
```

- Sekarang, di CA, lakukan signing (tandatangan) CSR yang telah diimport

  $ ./easyrsa sign-req server server

- Pada saat ditanya

  Confirm request details : yes

```
fitri@DEB12-CA2:~/easy-rsa$ ./easyrsa sign-req server server
* Notice:
Using Easy-RSA configuration from: /home/fitri/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.11 19 Sep 2023 (Library: OpenSSL 3.0.11 19 Sep 2023)


You are about to sign the following certificate.
Please check over the details shown below for accuracy. Note that this request
has not been cryptographically verified. Please be sure it came from a trusted
source or that you have verified the request checksum with the sender.

Request subject, to be signed as a server certificate for 825 days:

subject=
    commonName                = server


Type the word 'yes' to continue, or any other input to abort.
  Confirm request details: yes


Using configuration from /home/fitri/easy-rsa/pki/6d57df1b/temp.9cd6ea13
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'server'
Certificate is to be certified until Aug 19 14:21:16 2026 GMT (825 days)

Write out database with 1 new entries
Database updated

* Notice:
Certificate created at: /home/fitri/easy-rsa/pki/issued/server.crt
```

- Setelah ditandatangani CA, maka kembalikan file sertifikat server.crt dan ca.crt ke VPN Server
- Di CA, lakukan scp sbb:

  $cd easy-rsa

  $ scp pki/issued/server.crt fitri@192.168.220.136:/home/fitri/tmp

-
```
fitri@DEB12-CA2:~/easy-rsa$ scp pki/issued/server.crt fitri@192.168.220.136:/home/fitri
/tmp
The authenticity of host '192.168.220.136 (192.168.220.136)' can't be established.
ED25519 key fingerprint is SHA256:y1Uf6u+r+JtuapReKerRxkDXpr+mRi3X+g2qbbu/pY0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.220.136' (ED25519) to the list of known hosts.
fitri@192.168.220.136's password:
server.crt                                    100% 2912     1.9MB/s   00:00
```

- Di CA : file ca.crt juga harus dikembalikan ke VPN Server

  $ scp pki/ca.crt fitri@192.168.220.136:/home/fitri/tmp

```
fitri@DEB12-CA2:~/easy-rsa$ scp pki/ca.crt fitri@192.168.220.136:/home/fitri/tmp
fitri@192.168.220.136's password:
ca.crt                                        100%  749   549.5KB/s   00:00
```

- Pada VPN Server, taruh 2 file sertifikat tersebut di /etc/openvpn/server

  $su - root

  #cp /tmp/server.crt /etc/openvpn/server

  #cp /tmp/ca.crt /etc/openvpn/server

6. VPN Server : Konfigurasi Kriptografi

- Kita akan tambahkan secret key tambahan sehinggaserver dan client bisa menggunakan enkripsi tls-crypt
- Dengan adanya enkripsi, VPN Server dapat menghindari adanya unauthenticated traffic, port scans, dan Denial of Service attacks.
- Selain itu, dengan enkripsi, trafik VPN Server akan susah diidentifikasi
- Masuk ke direktory easy-rsa

  $su - fitri

  $ cd ~/easy-rsa
- Generate key dengan Diffie Hellman

  $ ./easyrsa gen-dh
- Waktu pembangkitan key ini agak lama, tunggulah dengan sabar

```
fitri@DEB12-VPN2:~$ cd easy-rsa/
fitri@DEB12-VPN2:~/easy-rsa$ ./easyrsa gen-dh
* Notice:
Using Easy-RSA configuration from: /home/fitri/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.11 19 Sep 2023 (Library: OpenSSL 3.0.11 19 Sep 2023)

Generating DH parameters, 2048 bit long safe prime
.....................................................+.............................
.................+.....................................................................+.
.....+.........................+.....................................................
.....................................................++*++*++*++*++*++*++*++*++*++*++*
++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*
++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*++*

* Notice:

DH parameters of size 2048 created at /home/fitri/easy-rsa/pki/dh.pem
```

- Generate tls-crypt pre-shared key. Ini akan menciptakan file ta.key

  $su - root

  #openvpn --genkey secret ta.key

```
root@DEB12-VPN2:~# openvpn --genkey secret ta.key
root@DEB12-VPN2:~# ls
ta.key
```

- Kopikan file ta.key ke /etc/openvpn/server directory.

  #cp ta.key /etc/openvpn/server

  #cp /home/fitri/easy-rsa/pki/dh.pem /etc/openvpn/server

```
root@DEB12-VPN2:~# cp ta.key /etc/openvpn/server
root@DEB12-VPN2:~# cp /home/fitri/easy-rsa/pki/dh.pem /etc/openvpn/server
```

7. VPN Server : Generate Client Certificate dan Key Pair

- Buat direktory baru di VPN Server untuk menyimpan client certificate dan key files.

  #su - fitri

  $ mkdir -p ~/client-configs/keys

- Batasi hak akses dire ktory hanya untuk user untuk memproteksi file-file didalamnya

  $ chmod -R 700 ~/client-configs

```
root@DEB12-VPN2:~# su - fitri
fitri@DEB12-VPN2:~$ mkdir -p client-configs/keys
fitri@DEB12-VPN2:~$ chmod -R 700 client-configs/
```

- Masuk ke direktory /easy-rsa directory.

  $ cd ~/easy-rsa
- Generate client key dengan Common Name (CN) : client1

  $ ./easyrsa gen-req client1 nopass
- Ketika anda ditanya Common Name, cukup di enter saja.

```
fitri@DEB12-VPN2:~/easy-rsa$ ./easyrsa gen-req client1 nopass
* Notice:
Using Easy-RSA configuration from: /home/fitri/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.11 19 Sep 2023 (Library: OpenSSL 3.0.11 19 Sep 2023)

-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Common Name (eg: your user, host, or server name) [client1]:
* Notice:

Keypair and certificate request completed. Your files are:
req: /home/fitri/easy-rsa/pki/reqs/client1.req
key: /home/fitri/easy-rsa/pki/private/client1.key
```

- Sekarang, kopikan file client1.key ke direktory ~/client-configs/keys

  $ cp pki/private/client1.key ~/client-configs/keys/

- Sekarang pindahkan file client1.req ke CA

  $ scp pki/reqs/client1.req fitri@192.168.220.133:/home/fitri/tmp

```
fitri@DEB12-VPN2:~/easy-rsa$ cp pki/private/client1.key ~/client-configs/keys
fitri@DEB12-VPN2:~/easy-rsa$ scp pki/reqs/client1.req fitri@192.168.220.133:/home/fitri
/tmp
fitri@192.168.220.133's password:
client1.req                                   100%  436   219.5KB/s   00:00
```

- Buka CA anda dan lakukan import file client1.req

  #su - fitri

  $ cd ~/easy-rsa

  $ ./easyrsa import-req /tmp/client1.req client1

```
fitri@DEB12-CA2:~/easy-rsa$ ./easyrsa import-req /home/fitri/tmp/client1.req client1
* Notice:
Using Easy-RSA configuration from: /home/fitri/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.11 19 Sep 2023 (Library: OpenSSL 3.0.11 19 Sep 2023)

* Notice:

The request has been successfully imported with a short name of: client1
You may now use this name to perform signing operations on this request.
```

- Request telah berhasil diimport dengan nama client1
- Gunakan nama tersebut untuk menandatangani request .

  $ ./easyrsa sign-req client client1

  Saat anda ditanya :

  Confirm request details : yes

```
fitri@DEB12-CA2:~/easy-rsa$ ./easyrsa sign-req client client1
* Notice:
Using Easy-RSA configuration from: /home/fitri/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.11 19 Sep 2023 (Library: OpenSSL 3.0.11 19 Sep 2023)


You are about to sign the following certificate.
Please check over the details shown below for accuracy. Note that this request
has not been cryptographically verified. Please be sure it came from a trusted
source or that you have verified the request checksum with the sender.

Request subject, to be signed as a client certificate for 825 days:

subject=
    commonName                = client1

Type the word 'yes' to continue, or any other input to abort.
  Confirm request details: yes

Using configuration from /home/fitri/easy-rsa/pki/14921d33/temp.be96e6d3
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'client1'
Certificate is to be certified until Aug 19 14:32:19 2026 GMT (825 days)

Write out database with 1 new entries
Database updated

* Notice:
Certificate created at: /home/fitri/easy-rsa/pki/issued/client1.crt
```

- Kirim kembali client certificate ke VPN Server

  $ scp pki/issued/client1.crt fitri@192.168.220.136:/home/fitri/tmp
- Di VPN Server, ambil client certificate dan kirim ke /home/fitri/client-configs/keys/

  $ cp /home/fitri/tmp/client1.crt ~/client-configs/keys/
- Kopikan ca.crt dan ta.key ke directory ~/client-configs/keys/

  $ su - root

  #cp ~/ta.key /home/fitri/client-configs/keys/

  #cp /etc/openvpn/server/ca.crt /home/fitri/client-configs/keys/

  #chown fitri:fitri /home/fitri/client-configs/keys/*

8. Konfigurasi VPN Server

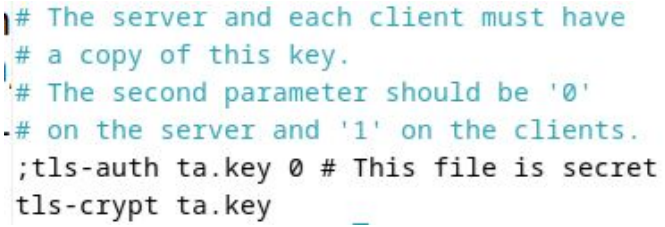
- Kopikan file server.conf dari file sample /usr/share/doc/openvpn/examples/sample-config-files/server.conf untuk dasar konfigurasi VPN, ke /etc/openvpn/server/

  #cp/usr/share/doc/openvpn/examples/sample-config-files/server.conf /etc/openvpn/server/
- Buka file /etc/openvpn/server/server.conf

  #nano /etc/openvpn/server/server.conf
- Cari kata berikut : tls-auth ta.key 0 . Beri tanda ; di depannya. Tambahkan baris tls-crypt di kata dibawahnya.

  ;tls-auth ta.key 0 # This file is secret

  tls-crypt ta.key



- Cari kata berikut : AES-256-CBC . Beri tanda ; atau # di depannya. Tambahkan baris AES-256-GCM di kata dibawahnya.
- Tambahkan baris : auth SHA256 dibawahnya

```
# Select a cryptographic cipher.
# This config item must be copied to
# the client config file as well.
# Note that v2.4 client/server will automatically
# negotiate AES-256-GCM in TLS mode.
# See also the ncp-cipher option in the manpage
#cipher AES-256-CBC
cipher AES-256-GCM
auth SHA256
```

```
# Diffie hellman parameters.
# Generate your own with:
#   openssl dhparam -out dh2048.pem 2048
;dh dh2048.pem
dh none
```

- Karena kita menggunakan Elliptic Curve Cryptography (ecc), kita tidak akan menggunakan enkripsi Diffie-Hellman.
- Beri tanda ; didepan dh dh2048.pem line dan tambahkan dh none dibawahnya

  ;dh dh2048.pem

  dh none
- VPN Server seharusnya dijalankan tanpa privileges ketika dijalankan. Untuk itu, tambahkan baris user nobody dan grup nogroup. Pastikan bahwa user openvpn dan group openvpn lines diberi tanda ; di depannya

```
# You can uncomment this on non-Windows
# systems after creating a dedicated user.
;user openvpn
;group openvpn
user nobody
group nogroup
```

- Cari baris : push "redirect-gateway def1 bypass-dhcp". Baris ini akan memberitahu client untuk meredireksi semua trafik leway VPN Server. Hilangkan tanda ; agar baris tersebut bekerja

```
# or bridge the TUN/TAP interface to the internet
# in order for this to work properly).
push "redirect-gateway def1 bypass-dhcp"
```

- Cari baris berikut dan hilangkan tanda ; di depannya. Baris ini memberitahu client untuk menggunakan OpenDNS

```
# The addresses below refer to the public
# DNS servers provided by opendns.com.
push "dhcp-option DNS 208.67.222.222"
push "dhcp-option DNS 208.67.220.220"
```

- OpenVPN menggunakan port 1194 dan protokol UDP untuk menerima koneksi Client.
- Anda dapat mengubah sesuai kebutuhan anda.
- Gunakan port 443 dan protokol TCP Change Port and tcp protocol

```
# Which TCP/UDP port should OpenVPN listen on?
# If you want to run multiple OpenVPN instances
# on the same machine, use a different port
# number for each one.  You will need to
# open up this port on your firewall.
;port 1194
port 443
# TCP or UDP server?
proto tcp
;proto udp
```

- Karena kita menggunakan protokol TCP, maka kita harus mengubah baris berikut dari bernilai 1 ke 0

```
# Notify the client that when the server restarts so it
# can automatically reconnect.
explicit-exit-notify 0
```

- Ketika anda menggunakan perintah ./easy-rsa gen-req server, anda menggunakan CN default yaitu: server. Maka pastikan anda memiliki baris berikut. Jika anda menggunakan CN lain, sesuaikan baris dibawah

```
# Any X509 key management system can be used.
# OpenVPN can also use a PKCS #12 formatted key file
# (see "pkcs12" directive in man page).
ca ca.crt
cert server.crt
key server.key
```

- Selesai ! Anda dapat menyimpan file server.conf dan Exit.

9. VPN Server: Setting Konfigurasi Network

- Langkah berikutnya adalah mengkonfigurasi variabel jaringan dari VPN Server.
- Buka file /etc/sysctl.conf

  $su - root

  #nano /etc/sysctl.conf
- Carilah baris net.ipv4.ip_forward. Hilangkan tanda # di depannya

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

- Save dan Exit

- Lakukan testing dengan dengan perintah sysctl -p untuk memastikan variabel net.ipv4.ip_forward berisi 1

```
root@DEB12-VPN2:/etc/openvpn/server# sysctl -p
net.ipv4.ip_forward = 1
```

- Konfigurasi ini akan merutekan semua trafik web dari client menuju IP address server, dan akibatnya alamat publik client akan tersembunyi

10. VPN Server : Jalankan Layanan VPN Server

- Jalankan layanan VPN Server

  #systemctl start openvpn-server@server.service
- Cek status layanan VPN Server

  #systemctl status openvpn-server@server.service

```
root@DEB12-VPN2:/etc/openvpn/server# systemctl start openvpn-server@server.service
root@DEB12-VPN2:/etc/openvpn/server# systemctl status openvpn-server@server.service
● openvpn-server@server.service - OpenVPN service for server
     Loaded: loaded (/lib/systemd/system/openvpn-server@.service; disabled; preset: en>
     Active: active (running) since Thu 2024-05-16 17:55:38 PDT; 15s ago
       Docs: man:openvpn(8)
             https://community.openvpn.net/openvpn/wiki/Openvpn24ManPage
             https://community.openvpn.net/openvpn/wiki/HOWTO
   Main PID: 3545 (openvpn)
     Status: "Initialization Sequence Completed"
      Tasks: 1 (limit: 2244)
     Memory: 3.3M
        CPU: 124ms
     CGroup: /system.slice/system-openvpn\x2dserver.slice/openvpn-server@server.service
             └─3545 /usr/sbin/openvpn --status /run/openvpn-server/status-server.log ->

May 16 17:55:38 DEB12-VPN2 systemd[1]: Starting openvpn-server@server.service - OpenVP>
May 16 17:55:38 DEB12-VPN2 systemd[1]: Started openvpn-server@server.service - OpenVPN>
lines 1-16/16 (END)
```

11. VPN Server: Konfigurasi VPN Client

- Sebelum melakukan testing dengan VPN Client, anda harus membuat file konfigurasi untuk Client.
- Kita akan menggunakan sample file yang dikustomisasi sesuai kebutuhan VPN Client
- Sebelumnya, buat direktory files di direktory client-config

  $su - fitri

  $ mkdir -p ~/client-configs/files
- Kopikan file /usr/share/doc/openvpn/examples/sample-config-files/client.conf berikut ke /home/fitri/client-configs/base.conf

  $ cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf ~/client-configs/base.conf

- Edit file /home/fitri/client-configs/base.conf

  $ nano client-configs/base.conf

- Cari baris berikut dan set menuju IP VPN Server dan juga ubah port menjadi 443.
- IP VPN Server adalah 192.168.220.136

```
# The hostname/IP and port of the server.
# You can have multiple remote entries
# to load balance between the servers.
;remote my-server-1 1194
;remote my-server-2 1194
remote 192.168.220.136 443
```

- Berikutnya set protokol untuk VPN Client menggunalan tcp, seperti setting protokol yang ada di VPN Server. Beri tanda ; di depan udp dan hilangkan tanda ; dari tcp

```
# Are we connecting to a TCP or
# UDP server?  Use the same setting as
# on the server.
proto tcp
;proto udp
```

- Turunkan privilege user dan group menjadi nobody dan nogroup

```
# Downgrade privileges after initialization (non-Windows only)
;user openvpn
;group openvpn
user nobody
user nogroup
```

- Cari baris ca, cert dan key. Beri tanda ; di ketiga baris tersebut. Kita menonaktifkan ketiganya karena nanti sertifikat client akan disertakan dalam file konfigurasi client.

```
# SSL/TLS parms.
# See the server config file for more
# description.  It's best to use
# a separate .crt/.key file pair
# for each client.  A single ca
# file can be used for all clients.
;ca ca.crt
;cert client.crt
;key client.key
```

- Beri tanda ; pada baris tls-auth karena file ta.key akan dicantumkan pada file konfigurasi

```
# If a tls-auth key is used on the server
# then every client must also have the key.
;tls-auth ta.key 1
```

- Pada file base.conf, sesuaikan setting untuk cipher pada file `/etc/openvpn/server/server.conf', yaitu AES-256-GCM.
- Tambahkan setting auth pada bagian bawah file :

```
# Select a cryptographic cipher.
# If the cipher option is used on the server
# then you must also specify it here.
# Note that v2.4 client/server will automatically
# negotiate AES-256-GCM in TLS mode.
# See also the data-ciphers option in the manpage
;cipher AES-256-CBC
cipher AES-256-GCM
auth SHA256
```

- Tambahkan setting key-direction menjadi 1, pada bagian paling bawah file base.conf, agar VPN dapat berjalan dengan baik. Save dan Exit

```
# Silence repeating messages
;mute 20

key-direction 1
```

12. VPN Server: Kompilasi File Client

- Sekarang, buat script untuk VPN Server untuk mengkompilasi file base.conf dengan sertifikat, key dan file enkripsi.
- Buat file berikut di VPN Server

  $ nano client-configs/make_config.sh

- Tambahkan baris berikut dan sesuaikan home direktory dengan user anda
- Simpan dan Exit.

```
  GNU nano 7.2                    client-configs/make_config.sh
#!/bin/bash

# First argument: Client identifier

KEY_DIR=/home/fitri/client-configs/keys
OUTPUT_DIR=/home/fitri/client-configs/files
BASE_CONFIG=/home/fitri/client-configs/base.conf

cat ${BASE_CONFIG} \
    <(echo -e '<ca>') \
    ${KEY_DIR}/ca.crt \
    <(echo -e '</ca>\n<cert>') \
    ${KEY_DIR}/${1}.crt \
    <(echo -e '</cert>\n<key>') \
    ${KEY_DIR}/${1}.key \
    <(echo -e '</key>\n<tls-crypt>') \
    ${KEY_DIR}/ta.key \
    <(echo -e '</tls-crypt>') \
    > ${OUTPUT_DIR}/${1}.ovpn
```


```
#!/bin/bash

# First argument: Client identifier

KEY_DIR=/home/fitri/client-configs/keys
OUTPUT_DIR=/home/fitri/client-configs/files
BASE_CONFIG=/home/fitri/client-configs/base.conf

cat ${BASE_CONFIG} \
    <(echo -e '<ca>') \
    ${KEY_DIR}/ca.crt \
    <(echo -e '</ca>\n<cert>') \
    ${KEY_DIR}/${1}.crt \
    <(echo -e '</cert>\n<key>') \
    ${KEY_DIR}/${1}.key \
    <(echo -e '</key>\n<tls-crypt>') \
    ${KEY_DIR}/ta.key \
    <(echo -e '</tls-crypt>') \
    > ${OUTPUT_DIR}/${1}.ovpn
```

- Buat file tersebut executable dan batasi hak aksesnya.

  $ chmod 700 ~/client-configs/make_config.sh

  Skrip make_config.sh akan mengkopikan base.conf, mengumpulkan semua file sertifikat dan file key, mengekstraksi isinya dan menambahkan ke file base.conf.
- Masuk ke directory ~/client-configs

  $ cd ~/client-configs

- Jalankan skrip sebagai berikut :

  $ ./make_config.sh client1

- Perintah diatas akan menciptakan file baru bernama client1.ovpn.
- Setiap kali anda menambahkan client baru, anda harus meng-generate key baru dan sertifikat baru.
- Kemudian menjalankan skrip make_config.sh untuk menciptakan file konfigurasi VPN Client yang baru

- Cek isi direktory /home/fitri/client-configs/files . Ternyata file client1.ovpn telah terbentuk di VPN Server

  $ ls ~/client-configs/files client1.ovpn

```
fitri@DEB12-VPN2:~$ ls ~/client-configs/files/
client1.ovpn
```

13. VPN Client : Instal VPN Client

- Install VPN client dengan perintah berikut :

  # apt install openvpn

- Download dulu file client1.ovpn dari VPN server dengan perintah scp

  #su - fitri

  $mkdir tmp

  $ scp fitri@192.168.220.136:/home/fitri/client-configs/files/client1.ovpn /home/fitri/tmp

- Pastikan VPN server dalam keadaan menyala

  #systemctl status openvpn-server@server.service

  Jika belum, restart VPN Server

  #systemctl restart openvpn-server@server.service
- Koneksi client ke server

  #openvpn --config /home/fitri/tmp/client1.ovpn –user nobody –group nogroup

```
root@debian12:~# openvpn --config /home/fitri/tmp/client1.ovpn --user nobody --group no
group
2024-05-17 13:09:57 Note: Kernel support for ovpn-dco missing, disabling data channel o
ffload.
2024-05-17 13:09:57 OpenVPN 2.6.3 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOL
L] [PKCS11] [MH/PKTINFO] [AEAD] [DCO]
2024-05-17 13:09:57 library versions: OpenSSL 3.0.11 19 Sep 2023, LZO 2.10
2024-05-17 13:09:57 DCO version: N/A
2024-05-17 13:09:57 TCP/UDP: Preserving recently used remote address: [AF_INET]192.168.
220.134:443
2024-05-17 13:09:57 Socket Buffers: R=[131072->131072] S=[16384->16384]
```

- Testing dari sisi VPN server, coba cek ip address dari VPN server. Ada baris baru yaitu tun0

```
fitri@DEB12-VPN2:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen
1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group defa
ult qlen 1000
    link/ether 00:0c:29:74:8e:0b brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.220.136/24 brd 192.168.220.255 scope global dynamic noprefixroute ens3
3
       valid_lft 1308sec preferred_lft 1308sec
    inet6 fe80::20c:29ff:fe74:8e0b/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOW
N group default qlen 500
    link/none
    inet 10.8.0.1 peer 10.8.0.2/32 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::b31a:a959:c9bf:d528/64 scope link stable-privacy
```

- Testing dari sisi VPN server, coba cek semua proses dan portnya dengan ss. Cari protokol tcp dengan Local Address:Port = 0.0.0.0:443 dengan status Listen

```
fitri@DEB12-VPN2:~$ ss -nltpu
Netid  State   Recv-Q   Send-Q     Local Address:Port      Peer Address:Port  Process
udp    UNCONN  0        0               0.0.0.0:5353           0.0.0.0:*
udp    UNCONN  0        0               0.0.0.0:50475          0.0.0.0:*
udp    UNCONN  0        0               0.0.0.0:631            0.0.0.0:*
udp    UNCONN  0        0                  [::]:5353              [::]:*
udp    UNCONN  0        0                  [::]:41860             [::]:*
tcp    LISTEN  0        32              0.0.0.0:443            0.0.0.0:*
tcp    LISTEN  0        128           127.0.0.1:631            0.0.0.0:*
tcp    LISTEN  0        128              [::1]:631              [::]:*
```

https://www.howtoforge.com/how-to-install-and-configure-openvpn-server-on-debian-12/