ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Đồ Án Quản trị mạng – NT132.N22.MMCL

Triển khai OpenVPN - Linux

Giảng Viên Hướng Dẫn: Trần Thị Dung

Nhóm Sinh Viên Thực Hiện:

Đỗ Thế Danh – 21520685

Lê Hoàng Khánh – 21522205

Trần Nhựt Linh – 21521081

Trường Đại Học Công Nghệ Thông Tin, 2023

MỤC LỤC

DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG

Lời cảm ơn

Các thành viên trong nhóm 01 muốn gửi lời cảm ơn chân thành sâu sắc tới giảng viên hướng dẫn Trần Thị Dung đã tận tình hỗ trợ nhóm để hoàn thành đồ án này. Qua đồ án này, nhóm đã tích lũy được nhiều kiến thức bổ ích về chuyên môn cũng như rèn luyện kĩ năng làm việc nhóm.

Mặc dù các thành viên trong nhóm đã cố gắng hoàn thành đồ án một cách hoàn thiện nhất nhưng do thiếu sót về kinh nghiệm cũng như một số hạn chế khác nên đồ án này cũng không thể tránh khỏi các sai sót. Nhóm rất mong nhận được sự cảm thông, chia sẻ và góp ý từ quý thầy cô cũng như các bạn sinh viên.

Nhóm 01 xin chân thành cảm ơn!

CHƯƠNG I - Tổng quan về OpenVPN

1.1 Giới thiệu về OpenVPN

1.1.1 Giới thiệu VPN

VPN là viết tắt của Virtual Private Network, là một công nghệ được sử dụng để kết nối các thiết bị mạng với nhau thông qua một kết nối an toàn và mã hóa. Kết nối VPN cho phép truy cập tới các tài nguyên mạng từ bất kỳ đâu trên thế giới thông qua một mạng riêng ảo (VPN). Các giao thức VPN thông dụng OpenVPN, IPSec, Wireguard, L2TP...

1.1.2 OpenVPN

OpenVPN là một phần mềm mã nguồn mở được sử dụng để tạo kết nối mạng riêng ảo (VPN) an toàn và mã hóa thông qua giao thức SSL/TLS. OpenVPN triển khai đầy đủ tính năng an toàn của SSL dưới OSI layer 2 hoặc 3 bằng cách sử dụng giao thức SSL/TLS, hỗ trợ các phương pháp xác thực Client linh hoạt dựa trên certificates, smart cards hoặc tài khoản mật khẩu và cho phép điều khiển quyền truy cập cụ thể của người dùng hoặc nhóm bằng cách sử dụng các chính sách kiểm soát truy cập (access control policies) dựa trên tường lửa được áp dụng cho interface VPN ảo. OpenVPN có thể chạy trên nhiều nền tảng khác nhau, bao gồm Windows, macOS, Linux, Android và iOS. OpenVPN được coi là một trong những phần mềm VPN phổ biến nhất hiện nay.

1.2 Thành phần của OpenVPN

1.2.1 Thành phần của OpenVPN

- **OpenVPN Server:**

OpenVPN Server là thành phần trung tâm của hệ thống VPN, chịu trách nhiệm cho việc xử lý yêu cầu kết nối từ các máy khách (clients) OpenVPN. Server sẽ lắng nghe kết nối trên một cổng (Port) được chỉ định và đợi Client kết nối. Khi một kết nối được thiết lập, server sẽ thiết lập kênh ảo (virtual tunnel) cho việc truyền dữ liệu giữa Client và server. Ngoài ra máy chủ OpenVPN cung cấp các dịch vụ như xác thực người dùng, quản lý chứng chỉ số, thiết lập kênh ảo (virtual tunnel) và định tuyến giữa các máy khách.

- **OpenVPN Client:**

OpenVPN Client đây là phần mềm được cài đặt trên máy tính hoặc thiết bị kết nối đến mạng riêng ảo OpenVPN. OpenVPN Client thiết lập một kết nối đến OpenVPN Server thự hiện quá trình xác thực, mã hóa (Encrypt) và giải mã (Decrypt) dữ liệu truyền tải giữa Client và Server qua kênh ảo. Client có thể được cài đặt trên các hệ điều hành khác nhau, bao gồm Windows, macOS, Linux, iOS và Android.

- **Digital Certificates (Chứng chỉ số):**

Chứng chỉ số là một thành phần quan trọng để xác thực danh tính trong OpenVPN. Chứng chỉ số được sử dụng để xác định tính hợp lệ của máy chủ và máy khách. Máy chủ và máy khách sử dụng chứng chỉ số để xác thực lẫn nhau và tạo một kênh ảo an toàn.

- **Encryption Algorithms (Thuật toán mã hóa):**

OpenVPN sử dụng một số thuật toán mã hóa, bao gồm AES (Advanced Encryption Standard), Blowfish và RSA (Rivest-Shamir-Adleman Các thuật toán này được sử dụng để mã hóa dữ liệu trước khi truyền và giải mã dữ liệu khi nhận được. Việc sử dụng mã hoá đảm bảo tính bảo mật và ngăn chặn bên không được uỷ quyền có thể đọc được thông tin truyền đi. Thuật toán mã hóa được sử dụng có thể được chỉ định trong tệp cấu hình.

- **Transport Protocols (Giao thức vận chuyển):**

OpenVPN có thể sử dụng giao thức TCP (Transmission Control Protocol) hoặc UDP (User Datagram Protocol) để truyền dữ liệu qua mạng.

Giao thức TCP đảm bảo tính tin cậy trong truyền dữ liệu bằng cách xác nhận và tái gửi các gói tin tuy nhiên TCP có độ trễ cao hơn và tốc độ truyền thấp hơn so với UDP. Giao thức này sử dụng cơ chế xác nhận và tái gửi, làm gia tăng độ trễ và giảm tốc độ truyền dữ liệu. Cùng theo đó là khả năng bị quá tải do TCP có nhiều cơ chế kiểm soát luồng và xác nhận, dẫn đến tốn nhiều tài nguyên mạng. Điều này có thể gây ra hiện tượng quá tải trong mạng có khả năng xử lý hạn chế.

Giao thức UDP sẽ có tốc độ truyền dữ liệu nhanh do UDP không sử dụng cơ chế xác nhận và tái gửi như TCP, do đó, nó có độ trễ thấp và tốc độ truyền nhanh hơn. Điều này phù hợp cho các ứng dụng yêu cầu truyền dữ liệu liên tục như streaming media hoặc VoIP. UDP cũng tiêu thụ ít tài nguyên mạng và không tạo ra quá tải cho hệ

thống. Tuy nhiên thay vào đó UDP sẽ không đảm bảo được tính toàn vẹn của dữ liệu và tính tin cậy trong việc truyền dữ liệu.

- **Virtual Tunnels (Kênh ảo):**

OpenVPN tạo ra các kênh ảo (virtual tunnels) để đưa dữ liệu từ máy khách đến máy chủ thông qua mạng public. Các kênh ảo này tạo một kết nối an toàn và được mã hóa để bảo vệ dữ liệu khi truyền qua mạng. Kênh ảo có thể được thiết lập bằng giao thức TCP hoặc UDP, việc này có thể xác định đươc trong tệp cấu hình.

1.2.2 Tệp cấu hình

- **Tệp cấu hình máy chủ (Server Configuration File):**

Tệp cấu hình máy chủ trong OpenVPN chứa các thiết lập và thông số quan trọng để cấu hình máy chủ OpenVPN. Các thành phần chính trong tệp cấu hình máy chủ bao gồm:

Mode (Chế độ): Xác định chế độ hoạt động của máy chủ OpenVPN, có thể là chế độ server hoặc chế độ point-to-point.

Protocol (Giao thức): Xác định giao thức vận chuyển được sử dụng cho kết nối VPN, ví dụ: TCP hoặc UDP.

Port (Cổng): Xác định số cổng mà máy chủ OpenVPN lắng nghe để chấp nhận các kết nối đến.

Server Address (Địa chỉ máy chủ): Xác định địa chỉ IP hoặc tên miền của máy chủ OpenVPN.

Authentication (Xác thực): Xác định phương thức xác thực được sử dụng để xác minh danh tính của người dùng hoặc máy khách. Điều này có thể bao gồm xác thực qua tên người dùng và mật khẩu hoặc sử dụng chứng chỉ số.

Encryption (Mã hóa): Xác định thuật toán mã hóa và khóa được sử dụng để bảo vệ dữ liệu trong quá trình truyền. Điều này bao gồm việc cung cấp các khóa mã hóa và chọn thuật toán mã hóa như AES, Blowfish, RSA.

Network Configuration (Cấu hình mạng): Xác định cấu hình mạng riêng ảo và tùy chọn định tuyến cho máy chủ OpenVPN. Bao gồm việc xác định địa chỉ IP, mạng con, và quy tắc định tuyến.

- **Tệp cấu hình máy khách (Client Configuration File):**

Tệp cấu hình máy khách trong OpenVPN chứa thông tin cần thiết để máy khách OpenVPN có thể kết nối và thiết lập kết nối VPN với máy chủ. Các thành phần chính trong tệp cấu hình máy khách bao gồm:

Mode (Chế độ): Xác định chế độ hoạt động của máy khách OpenVPN, thường là chế độ client.

Protocol (Giao thức): Xác định giao thức vận chuyển được sử dụng cho kết nối VPN, tương ứng với giao thức được cấu hình trên máy chủ.

Remote Server (Máy chủ): Xác định địa chỉ IP hoặc tên miền của máy chủ OpenVPN mà máy khách sẽ kết nối.

Port (Cổng): Xác định số cổng mà máy chủ OpenVPN đang lắng nghe.

Authentication (Xác thực): Xác định phương thức xác thực được sử dụng để xác minh danh tính của máy khách. Điều này phải phù hợp với cấu hình xác thực trên máy chủ.

Encryption (Mã hóa): Xác định thuật toán mã hóa và khóa được sử dụng để bảo vệ dữ liệu trong quá trình truyền. Điều này phải phù hợp với cấu hình mã hóa trên máy chủ.

Network Configuration (Cấu hình mạng): Xác định cấu hình mạng riêng ảo và tùy chọn định tuyến cho máy khách OpenVPN. Bao gồm việc xác định địa chỉ IP và mạng con được sử dụng trên máy khách.

1.2.3 Các thành phần bổ sung

- **EasyRSA:**

EasyRSA là một công cụ dòng lệnh giúp đơn giản hóa việc quản lý Hạ tầng Khóa công khai (PKI - Public Key Infrastructure) cần thiết cho OpenVPN. Nó cung cấp một cách đơn giản để tạo và quản lý các chứng chỉ và khóa cần thiết cho hạ tầng VPN.

Với EasyRSA, bạn có thể dễ dàng tạo và quản lý "nhà cung cấp chứng thực số" (CA - Certificate Authority), chứng chỉ máy chủ và chứng chỉ khách hàng cần thiết cho việc giao tiếp an toàn trong OpenVPN.

EasyRSA cung cấp các lệnh để tạo các yêu cầu ký chứng chỉ (CSR - Certificate signing request) cần thiết, ký và thu hồi chứng chỉ, và quản lý danh sách thu hồi chứng chỉ (CRL - Certificate revocation list).

Việc sử dụng EasyRSA giúp tối ưu quy trình thiết lập và quản lý Hạ tầng Khóa công khai, làm cho việc triển khai và duy trì OpenVPN dễ dàng hơn.

- **OpenSSL:**

OpenSSL là một thư viện phần mềm mã nguồn mở cung cấp các chức năng mật mã, bao gồm việc tạo và quản lý chứng chỉ và khóa SSL/TLS.

OpenVPN phụ thuộc vào OpenSSL để thực hiện mã hóa, xác thực và trao đổi khóa an toàn cần thiết để thiết lập kết nối VPN an toàn.

OpenSSL cung cấp các thuật toán mã hóa cần thiết như AES, RSA và HMAC để đảm bảo việc truyền thông an toàn trong OpenVPN.

OpenSSL cũng có thể sử dụng để tạo "nhà cung cấp chứng thực số" (CA - Certificate Authority), chứng chỉ máy chủ và chứng chỉ khách hàng, cũng như thực hiện các hoạt động mật mã học khác cần thiết trong OpenVPN.

1.3 Hoạt động của OpenVPN

- **Authentication and Verification:**

Khi máy khách (client) muốn kết nối đến máy chủ (server) thông qua mạng VPN, quá trình xác thực xảy ra. Ở phần xác thực OpenVPN sẽ sử dụng má hoá bất đối xứng để xác thực thông tin đăng nhập. Máy khách gửi yêu cầu xác thực đến máy chủ. Máy chủ sử dụng sử dụng chứng chỉ SSL/TLS và cặp khoá private-key và public-key để xác thực danh tính của máy khách. Nếu xác thực thành công thì quá trình tiếp tục.

- **Key Exchange:**

Ở bước này OpenVPN sử dụng giao thức TLS (Transport Layer Security) để thoả thuận khoá dùng để mã hoá dữ liệu qua tunnel giữ máy chú và máy khách. Máy chủ và máy khách sẽ trao đổi thông tin về khoá bằng mã hoá bất đối xứng với cặp khoá private-key và public-key từ bước thứ nhất nhằm đảm bảo an toàn về thông tin của khoá. Kết quả máy khách và máy chủ thoả thuật được khoá dùng chung cho việc mã hoá dữ liệu qua kênh ảo (tunnel) được mã hoá bằng kỹ thuật đối xứng nhằm đảm bảo hiệu năng truyền dữ liệu.

- **Connection Establishment:**

Sau khi thoả thuận khoá thành công OpenVPN sẽ tạo ra một kênh ảo (tunnel) giữa máy chủ và máy khác. Kênh ảo này có thể sử dụng giao thức TCP (Transmission Control Protocol) hoặc UDP (User Datagram Protocol) để truyền dữ liệu qua mạng public. Trước khi dữ liệu được gửi đi qua kênh ảo, nó sẽ được mã hóa sử dụng ký thuật đối sứng với khoá có được từ bước trên và thêm các tiêu đề (header) để định tuyến đến máy chủ.

- **Packet Transmission and Rouing:**

Các gói tin dữ liệu sẽ được gửi qua kênh ảo đã tạo, thông qua giao thức mạng chuẩn, chẳng hạn như TCP hoặc UDP. Khi gói tin đến máy chủ, OpenVPN sẽ giải mã nó bằng khoá đã được thoả thuận với máy khác từ trước sau và chuyển nội dung dữ liệu đến đích. OpenVPN cũng cho phép người dùng tùy chỉnh các quy tắc định tuyến, cho phép chỉ định lưu lượng dữ liệu nào được gửi qua kết nối VPN và lưu lượng dữ liệu nào được gửi qua kết nối Internet thông thường.

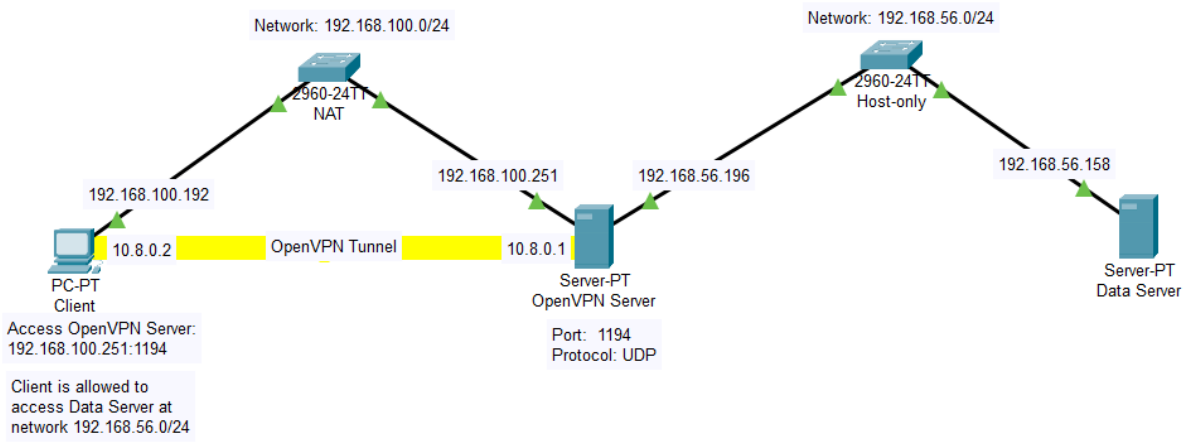
- **Disconnect:**

Khi quá trình kết nối kết thúc, dữ liệu cuối cùng được gửi và kết nối được ngắt.

CHƯƠNG II – Triển khai OpenVPN

2.1 Mô hình OpenVPN

2.1.1 Hình vẽ



Hình 1. Mô hình

2.1.2 Thành phần

Interface	Devices	IP Address
NAT	Client	192.168.100.192
	OpenVPN Server	192.168.100.251
Host-Only	Data Server	192.168.56.158
	OpenVPN Server	192.168.56.196
Tunnel	Client	10.8.0.2
	OpenVPN Server	10.8.0.1

Hình 2. IP Table

2.1.3 Hoạt động

- Giả sử, gửi một gói tin từ phía Client đến Data Server.
- Ban đầu, gói tin được khởi tạo ở Client với IP nguồn là 192.168.100.192 và IP đích là 192.168.100.251 và địa chỉ 192.168.56.158 sẽ được thêm vào phần Header của gói tin. OpenVPN được bật giữa Client và OpenVPN Server sẽ tạo ra một kết nối ảo (tunnel). Gói tin sẽ được OpenVPN mã hóa (encrypt) và được gửi đi (route) qua tunnel. Gói tin được OpenVPN nhận sẽ được giải mã (decrypt) thành gói tin gốc ban đầu, ở phần header chưa địa chỉ IP cần forward đến là 192.168.56.158 và được chuyển tiếp (foward) qua interface Host-Only của OpenVPN Server với IP nguồn là 192.168.56.196 và IP đích là 192.168.56.158. Gói tin sau đó được gửi đi đến Data Server.

2.2 Cài đặt OpenVPN

2.2.1 easy-rsa

- **Cấu hình**

- Cài đặt easy-rsa:

```
[tuilakhanh@server ~]$ sudo dnf install easy-rsa
Last metadata expiration check: 0:00:31 ago on Sun 18 Jun 2023 02:04:21 PM UTC.
Dependencies resolved.
================================================================================
 Package              Architecture        Version              Repository    Size
================================================================================
Installing:
 easy-rsa             noarch              3.1.4-1.fc37         updates       64 k

Transaction Summary
================================================================================
Install  1 Package

Total download size: 64 k
Installed size: 213 k
Is this ok [y/N]: y
Downloading Packages:
easy-rsa-3.1.4-1.fc37.noarch.rpm                 47 kB/s |  64 kB     00:01
--------------------------------------------------------------------------------
Total                                            26 kB/s |  64 kB     00:02
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                        1/1
  Installing       : easy-rsa-3.1.4-1.fc37.noarch                           1/1
  Verifying        : easy-rsa-3.1.4-1.fc37.noarch                           1/1

Installed:
  easy-rsa-3.1.4-1.fc37.noarch

Complete!
```

Hình 3. Cài đặt easy-rsa

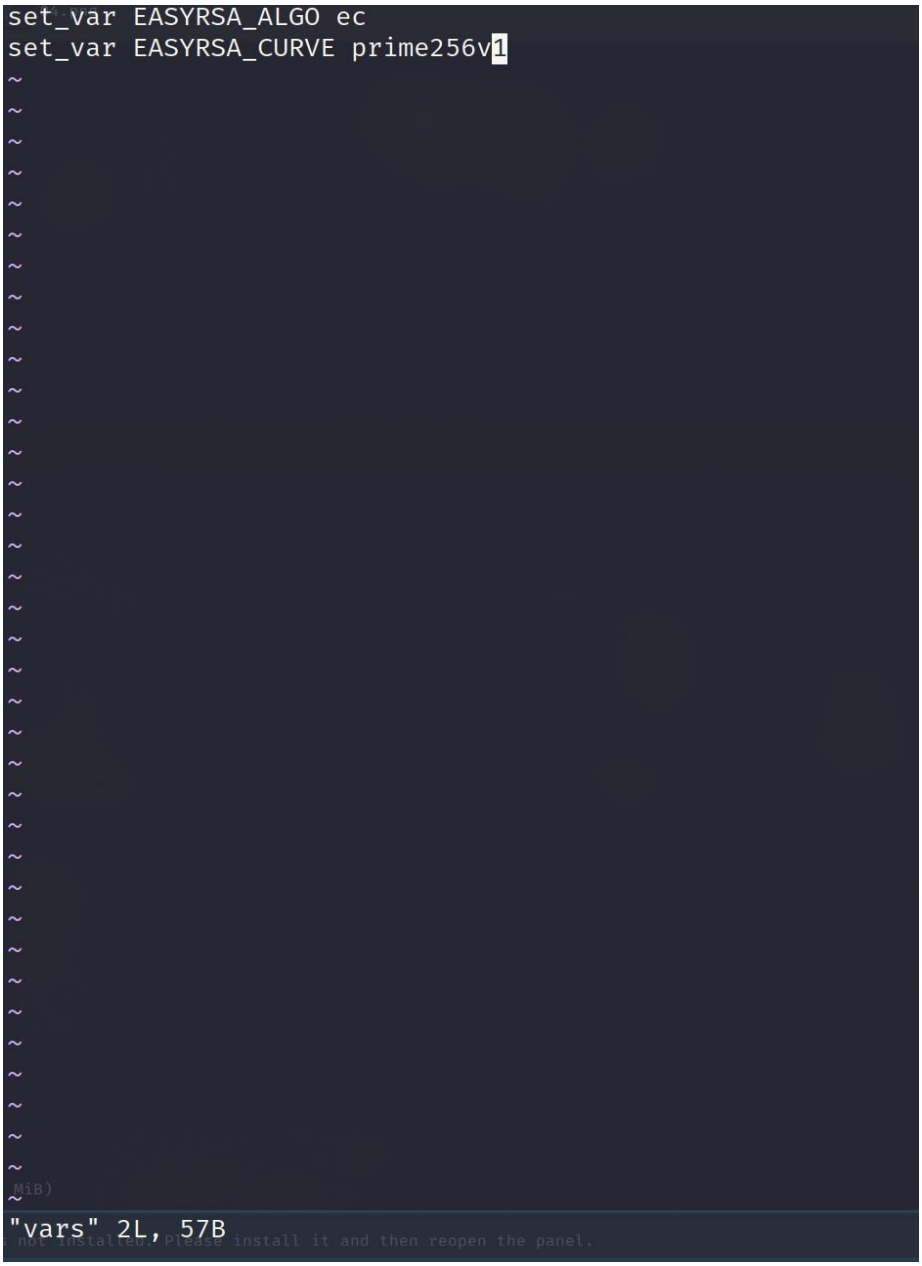
* Sử dụng người dùng root để thực hiện các bước tiếp theo

- Khởi tạo PKI: tạo ra directory và files cần thiết cho việc thiết lập và lưu trữ CA, certificates và keys:

```
[root@server openvpn]# mkdir /etc/openvpn/easy-rsa; cp -rai /usr/share/easy-rsa/3/* /etc/openvpn/easy-rsa/
[root@server openvpn]# cd /etc/openvpn/easy-rsa/
[root@server easy-rsa]# ls
easyrsa  openssl-easyrsa.cnf  x509-types
```

Hình 4. Tạo directory

+ Tạo directory /etc/openvpn/easy-rsa, và di chuyển tất cả files từ dir mà package easy-rsa được tải xuống (/etc/share/easy-rsa/3/*) sang dir vừa mới tạo.
+ Cấu hình file "vars":



Hình 5. Cấu hình file "vars"

Đặt giá trị của biến EASYRSA_ALGO là ec để chỉ định thuật toán mã hóa được dùng trong việc khởi tạo các certificates và keys. (ec bảo mật hơn rsa bởi vì với cùng 1 độ phức tạp của thuật toán, dùng ít bits hơn).Đặt giá trị của biến EASYRSA_CURVE là prime256v1 để chỉ định elliptic curve. Với prim256v1, nó có tính cân bằng trong việc bảo mật và tính hiệu quả.

+ Sau đó, khởi tạo PKI:

```
[root@server easy-rsa]# ./easyrsa init-pki

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

Your newly created PKI dir is:
* /etc/openvpn/easy-rsa/pki

* Using Easy-RSA configuration: /etc/openvpn/easy-rsa/vars

* The preferred location for 'vars' is within the PKI folder.
  To silence this message move your 'vars' file to your PKI
  or declare your 'vars' file with option: --vars=<FILE>

* Using x509-types directory: /etc/openvpn/easy-rsa/x509-types
```

Hình 6. Khởi tạo PKI

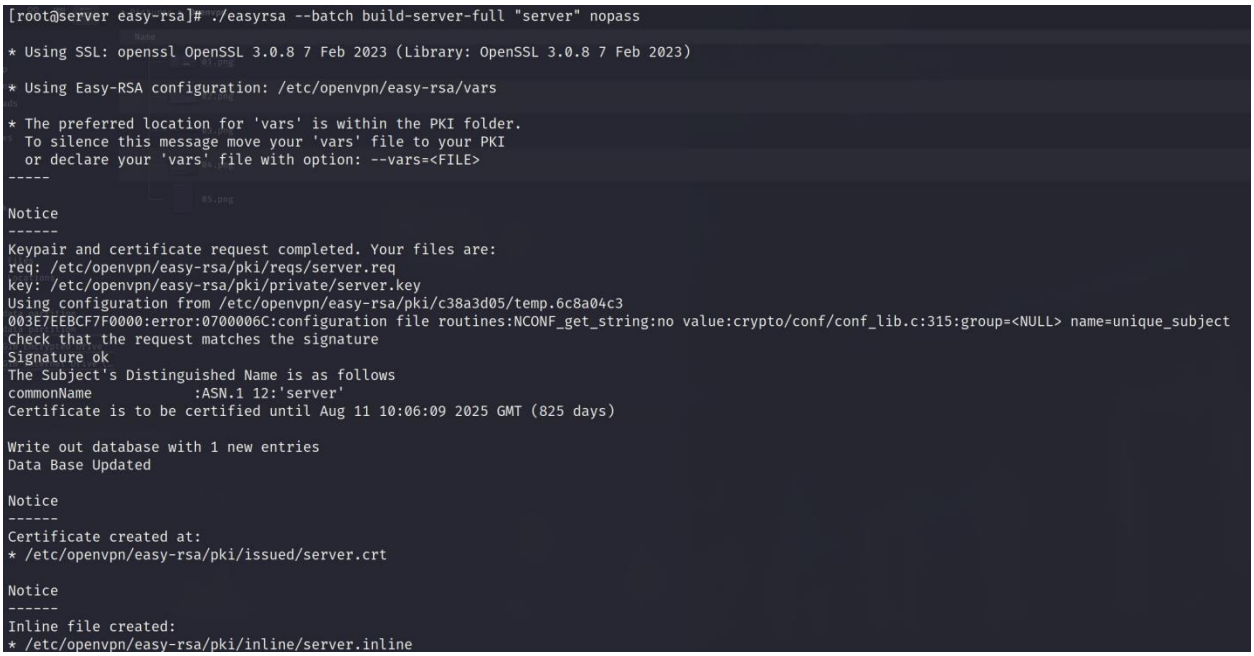
- Xây dựng CA

```
[root@server easy-rsa]# ./easyrsa --batch build-ca nopass

* Using SSL: openssl OpenSSL 3.0.8 7 Feb 2023 (Library: OpenSSL 3.0.8 7 Feb 2023)

* Using Easy-RSA configuration: /etc/openvpn/easy-rsa/vars

* The preferred location for 'vars' is within the PKI folder.
  To silence this message move your 'vars' file to your PKI
  or declare your 'vars' file with option: --vars=<FILE>
Using configuration from /etc/openvpn/easy-rsa/pki/179e9b7e/temp.794fb2d1
-----

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

Hình 7. Xây dựng CA

Ở đây, tùy chọn "nopass" không đặt mật khẩu cho CA để đơn giản cho các quá trình sau.

- Tạo certificate và key:



Hình 8. Tạo certificate và key cho Server

```
[root@server easy-rsa]# ./easyrsa --batch build-client-full "client" nopass

* Using SSL: openssl OpenSSL 3.0.8 7 Feb 2023 (Library: OpenSSL 3.0.8 7 Feb 2023)

* Using Easy-RSA configuration: /etc/openvpn/easy-rsa/vars

* The preferred location for 'vars' is within the PKI folder.
  To silence this message move your 'vars' file to your PKI
  or declare your 'vars' file with option: --vars=<FILE>
-----

Notice
------
Keypair and certificate request completed. Your files are:
req: /etc/openvpn/easy-rsa/pki/reqs/client.req
key: /etc/openvpn/easy-rsa/pki/private/client.key
Using configuration from /etc/openvpn/easy-rsa/pki/07bd4d3f/temp.ac52080b
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'client'
Certificate is to be certified until Aug 11 10:19:58 2025 GMT (825 days)

Write out database with 1 new entries
Data Base Updated

Notice
------
Certificate created at:
* /etc/openvpn/easy-rsa/pki/issued/client.crt

Notice
------
Inline file created:
* /etc/openvpn/easy-rsa/pki/inline/client.inline
```

Hình 9. Tạo certificate và key cho Client

Sử dụng tùy chọn "build-server-full" và "build-client-full" để khởi tạo certificate và key cho server và client chỉ trong 1 bước mà không cần đến bước ký yêu cầu.

2.2.2 Cấu hình OpenVPN Server

- Copy các file certificate và key đã tạo được từ bước trên vào cùng folder ở có tên server ở folder /etc/openvpn đây sẽ là folder chưa cấu hình của OpenVPN server và các file liên quan.

```
[root@server easy-rsa]# cp pki/ca.crt pki/private/ca.key pki/issued/server.crt pki/private/server.key /etc/openvpn/server
[root@server easy-rsa]# ls /etc/openvpn/server/
ca.crt  ca.key  server.crt  server.key
```

Hình 10. Copy các file certificate và key

- Tạo khoá bí mật (secret key) cho mã hoá TLS-Crypt

```
[root@server server]# openvpn --genkey secret /etc/openvpn/server/tls-crypt.key
[root@server server]# ls
ca.crt  ca.key  server.crt  server.key  tls-crypt.key
[root@server server]#
```

Hình 11. Tạo khóa bí mật

- Tạo file server.conf và "server" cũng là tên của OpenVPN server.

```
[root@server ~]# cd /etc/openvpn/server/
[root@server server]# touch server.conf
[root@server server]# ls
ca.crt  ca.key  server.conf  server.crt  server.key  tls-crypt.key
```

Hình 12. Tạo file server.conf

- Cấu hình OpenVPN server.

```
port 1194
proto udp
dev tun
user nobody
group nobody
persist-key
persist-tun
keepalive 10 120
topology subnet
server 10.8.0.0 255.255.255.0
status /var/log/openvpn/status.log
ifconfig-pool-persist ipp.txt
push "redirect-gateway def1 bypass-dhcp"
push "route 192.168.56.0 255.255.255.0"
```

Hình 13. Cấu hình OpenVPN server

- Giải thích cấu hình:

port 1194: Xác định số cổng mà máy chủ OpenVPN lắng nghe để chấp nhận kết nối từ các máy khách. Trong trường hợp này, cổng 1194 được sử dụng.

proto udp: Xác định giao thức vận chuyển được sử dụng cho kết nối VPN. Trong trường hợp này, giao thức UDP (User Datagram Protocol) được sử dụng. UDP thường được ưu tiên trong OpenVPN vì tốc độ truyền dữ liệu nhanh hơn và hiệu suất tốt hơn đối với các kết nối thời gian thực. Tại sao lại không là TCP (Transmission Control Protocol) vì tốc độ chậm và độ trễ cao dễ gây ra quá tải mạng, quan trọng nhất việc truyền TCP qua TCP không phải một ý kến hay.

dev tun: Xác định giao diện (interface) mạng ảo được sử dụng cho kết nối VPN. Trong trường hợp này, giao diện TUN được sử dụng. Giao diện TUN được sử dụng cho kết nối mạng IP. Ngoài ra còn có giao diện TAP hoạt động ở Layer 2.

user nobody, group nobody: Sử dụng các tùy chọn user nobody và group nobody trong tệp cấu hình làm loại bỏ quyền root đối với OpenVPN sau khi thiết lập kết nối nhằm đảm bạo tính bảo mật và an toàn cho server. Tuỳ vào các bản phân phối của linux user và group có thể là **nobody** hoặc là **nouser.** Để kiểm tra điều này sử dụng lệnh sau.

```
[root@server server]# cat /etc/group | grep no
nobody:x:65534:
```

Hình 14. Kiểm tra nobody hoặc nouser

persist-key: Đảm bảo rằng khóa mã hóa được thoả thuận với Client được giữ nguyên khi OpenVPN khởi động lại.

persist-tun: Đảm bảo rằng giao diện (interface) mạng ảo được lưu giữ nguyên khi quá trình OpenVPN khởi động lại.

keepalive 10 120: Xác định tần suất kiểm tra kết nối giữa máy chủ và máy khách. Trong trường hợp này, mỗi 10 giây, máy chủ sẽ gửi một tin nhắn keepalive đến máy khách. Nếu máy khách không phản hồi sau 120 giây, kết nối sẽ được coi là đã bị mất.

server 10.8.0.0 255.255.255.0: Xác định mạng con được sử dụng cho mạng ảo. Trong trường hợp này, mạng con 10.8.0.0/24 được sử dụng và máy chủ OpenVPN sẽ quản lý việc phân phối các địa chỉ IP cho các máy khách kết nối.

topology subnet: được sử dụng để xác định loại cấu hình mạng ảo trong quá trình thiết lập kết nối VPN. Khi tùy chọn này được đặt thành "subnet", mạng ảo sẽ được thiết lập dưới dạng một mạng con (subnet), nghĩa là mỗi máy khách sẽ nhận được một địa chỉ IP từ mạng con 10.8.0.0/24.

status /var/log/openvpn/status.log: Xác định đường dẫn tới tệp log. Trong trường hợp này, tệp status.log sẽ được tạo ra tại đường dẫn /var/log/openvpn/ và sẽ chứa thông tin về log của máy chủ OpenVPN.

ifconfig-pool-persist ipp.txt: Xác định đường dẫn tới tệp lưu trữ thông tin cấp phát địa chỉ IP cho máy khách. Tức mỗi máy khác sau khi tham gia vào mạng ảo

và được cấp phát ip máy chủ sẽ lưu ip này lại là sẽ cấp phát đúng ip đó khi máy khác tham gia lại vào mạng ảo.

push "redirect-gateway defl bypass-dhcp": Tùy chọn này đẩy thông báo cho máy khách để chuyển hướng toàn bộ lưu lượng mạng của nó thông qua kết nối VPN. Với các tham số. **redirect-gateway** đây là một tùy chọn chuyển hướng gateway mạng. Nó yêu cầu máy khách đẩy gói tin qua kết nối VPN thay vì sử dụng gateway mạng mặc định. **defl** Tùy chọn này chỉ định rằng máy khách sẽ chuyển hướng tất cả các dải địa chỉ mạng (0.0.0.0/0) thông qua kết nối VPN. **bypass-dhcp** Tùy chọn này cho phép máy khách tránh việc sử dụng DHCP (Dynamic Host Configuration Protocol) để cấu hình gateway mạng. Thay vào đó, nó sẽ sử dụng cấu hình được đưa ra trong tệp cấu hình OpenVPN. Tùy chọn này có ý nghĩa là máy khách sẽ sử dụng kết nối VPN làm gateway mặc định và tất cả lưu lượng mạng của nó sẽ đi qua kết nối VPN.

push "route 192.168.56.0 255.255.255.0": Tùy chọn này đẩy thông báo cho máy khách để thêm một route vào bảng định tuyến của máy khách. Với các tham số sau. **route** Đây là một tùy chọn để thêm một route vào bảng định tuyến của máy khách. **192.168.56.0** Đây là địa chỉ mạng đích mà router sẽ áp dụng cho đó cũng sẽ là lớp mạng chưa DataServer. **255.255.255.0** Đây là subnet mask của địa chỉ mạng đích. Tùy chọn này có ý nghĩa là máy khách sẽ được thông báo để thêm một route cho mạng 192.168.56.0/24 vào bảng định tuyến của nó thông qua kết nối VPN. Điều này cho phép máy khách truy cập vào mạng con 192.168.56.0/24 tức mạng chưa DataServer thông qua kết nối VPN.

- **Cấu hình bảo mật cho OpenVPN:**

Khi máy khách kết nối đến máy chủ OpenVPN, quá trình xác thực diễn ra như sau:

+ Máy khách tạo một yêu cầu kết nối đến máy chủ OpenVPN.
+ Máy chủ phản hồi yêu cầu của máy khách bằng cách gửi chứng chỉ máy chủ cho máy khách. Chứng chỉ máy chủ bao gồm khóa công khai của máy chủ và thông tin về máy chủ.
+ Máy khách kiểm tra chứng chỉ máy chủ bằng cách so sánh nó với các chứng chỉ trong file CA. Nếu chứng chỉ hợp lệ, quá trình xác thực tiếp tục. Nếu chứng chỉ không hợp lệ hoặc không tìm thấy, kết nối sẽ bị từ chối do lúc này máy chủ đã bị bên thứ ba can thiệp.

+ Sau khi xác thực chứng chỉ máy chủ, máy khách sử dụng cặp public và private. Máy khách gửi yêu cầu chứng chỉ máy khách cùng với khóa public của nó đến máy chủ.
+ Máy chủ sử dụng khóa private của mình để ký và trả về chứng chỉ máy khách cho máy khách.
+ Máy khách kiểm tra chứng chỉ của mình bằng cách so sánh nó với các chứng chỉ tròn file CA. Nếu chứng chỉ hợp lệ, quá trình xác thực hoàn tất. Nếu chứng chỉ không hợp lệ hoặc không tìm thấy, kết nối sẽ bị từ chối.

Sau khi quá trình xác thực hoàn tất, máy khách và máy chủ đã xác định danh tính của nhau và có thể bắt đầu truyền dữ liệu qua mạng VPN một cách an toàn và bảo mật.

Các file cần thiết ở bước này:

- Chứng chỉ CA.
- Cặp khoá Public và Private của Server.
- Static Key.

Các file cần thiết đã được tạo ở bước trên và có đường dẫn ở.

- /etc/openvpn/server/ca.crt
- /etc/openvpn/server/server.crt và /etc/openvpn/server/server.key
- /etc/openvpn/server/tls-crypt.key

```
dh none
ecdh-curve prime256v1
tls-crypt tls-crypt.key
ca ca.crt
cert server.crt
key server.key
auth SHA256
cipher AES-128-GCM
ncp-ciphers AES-128-GCM
tls-server
tls-version-min 1.2
tls-cipher TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256
verb 3
```

Hình 15. Cấu hình bảo mật cho OpenVPN Server

dh: none: Vì sử dụng TLS với elliptic curves nên Diffie-Hellman là không cần thiết.

ecdh-curve prime256v1: Tùy chọn này xác định đường cong elliptic curve Diffie-Hellman (ECDH) được sử dụng để trao đổi khóa. Trong trường hợp này, "prime256v1" đề cập đến đường cong NIST P-256, một đường cong thông dụng được sử dụng cho việc thỏa thuận khóa an toàn.

tls-crypt tls-crypt.key: Do sử dụng TLS với elliptic curves nên tùy chọn này sẽ được sử dụng. Tuỳ chọn cho phép sử dụng một tệp static key để bổ sung mã hóa các gói tin kênh điều khiển. Tệp "tls-crypt.key" chứa khóa bí mật được sử dụng cho mục đích này.

ca ca.crt: Tùy chọn này xác định đường dẫn đến tệp chứng chỉ "nhà cung cấp chứng thực số" (CA - Certificate Authority). Chứng chỉ CA được sử dụng để xác minh tính xác thực của chứng chỉ máy chủ trong quá trình xác thực TLS.

cert server.crt: Tùy chọn này xác định đường dẫn đến tệp chứng chỉ máy chủ. Chứng chỉ máy chủ được sử dụng bởi máy chủ OpenVPN trong quá trình xác thực TLS để chứng minh danh tính của nó. Và trong trứng chỉ có đình kèm thêm Public Key của server.

key server.key: Tùy chọn này xác định đường dẫn đến tệp Private Key tương ứng với chứng chỉ máy chủ. Private Key được sử dụng để giải gói tin TLS đến và xác thực máy chủ.

auth SHA256: Tùy chọn này xác định thuật toán xác thực thông điệp (message authentication) được sử dụng để kiểm tra tính toàn vẹn dữ liệu. Trong trường hợp này, SHA256 được sử dụng, đây là một thuật toán băm an toàn.

cipher AES-128-GCM: Tùy chọn này xác định thuật toán mã hóa đối xứng được sử dụng để mã hóa dữ liệu.

ncp-ciphers AES-128-GCM: Tùy chọn này xác định các tập hợp cipher suites có sẵn để thương lượng trong quá trình xác thực TLS. Trong trường hợp này, chỉ có AES-128-GCM được cho phép.

tls-server: Tùy chọn này xác định rằng tls đang chạy theo kiểu máy chủ.

tls-version-min 1.2: Tùy chọn này xác định phiên bản TLS tối thiểu yêu cầu. Trong trường hợp này, phiên bản tối thiểu yêu cầu là TLS 1.2.

tls-cipher TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256: Tùy chọn này xác định tập hợp cipher suites ưu tiên cho quá trình xác thực TLS. Đây là bao gồm tất cả các mã hoá được sử dụng ở các bước trên.

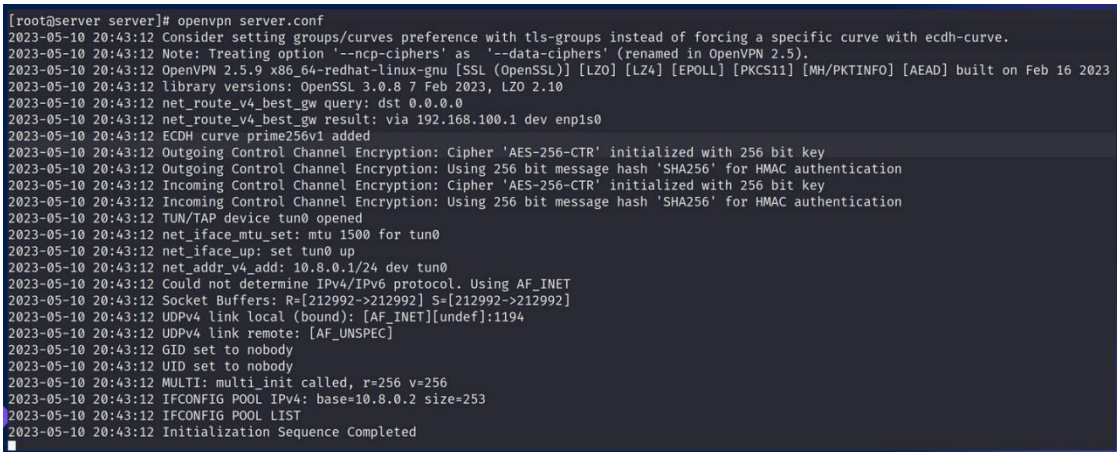
- **File cấu hình hoàn chỉnh:**

```
port 1194
proto udp
dev tun
user nobody
group nobody
persist-key
persist-tun
keepalive 10 120
topology subnet
server 10.8.0.0 255.255.255.0
status /var/log/openvpn/status.log
ifconfig-pool-persist ipp.txt
push "redirect-gateway def1 bypass-dhcp"
push "route 192.168.56.0 255.255.255.0"
dh none
ecdh-curve prime256v1
tls-crypt tls-crypt.key
ca ca.crt
cert server.crt
key server.key
auth SHA256
cipher AES-128-GCM
ncp-ciphers AES-128-GCM
tls-server
tls-version-min 1.2
tls-cipher TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256
verb 3
```

Hình 16. File cấu hình hoàn chỉnh

- **Kiểm tra cấu hình OpenVPN Server**

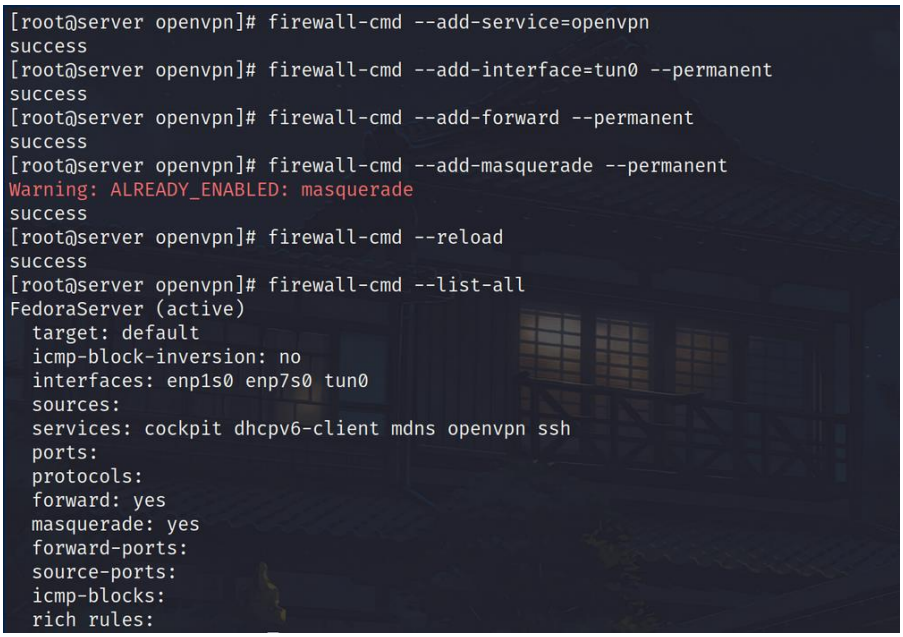
Chạy **openvpn /etc/openvpn/server/server.conf** (với quyền root) trên máy chủ. Kết quả OpenVPN Server đã chạy thành công.



Hình 17. Kiểm tra cấu hình OpenVPN Server

- **Cấu hình Firewall cho OpenVPN Server**

Mặc định đối với Server chúng ta nên cài đặt Firewall để đảm an toàn cho Server. Đối với bản phối phân phối Linux đang sử dụng là Fedora đã có sẵn Firewall đó là **firewalld**. Mặc định firewalld sẽ chặn các kết nối không được tin cậy, vì vậy nên tiếp theo thực hiện cấu hình firewalld cho phép OpenVPN Server và Client có thể giao tiếp với nhau.



Hình 18. Cấu hình Firewall cho OpenVPN Server

firewall-cmd --add-service=openvpn: Lệnh này thêm một quy tắc dựa trên dịch vụ để cho phép lưu lượng của OpenVPN Server. Nó mở các cổng và giao thức cần thiết cho dịch vụ OpenVPN Server.

firewall-cmd --add-interface=tun0 --permanent: Lệnh này thêm một quy tắc để cho phép lưu lượng trên giao diện (interface) cụ thể là **tun0**. Tùy chọn --permanent làm cho quy tắc này tồn tại sau khi khởi động lại hệ thống.

firewall-cmd --add-forward --permanent: Lệnh này thêm một quy tắc chuyển tiếp (forward) để cho phép chuyển tiếp lưu lượng mạng. Nó cho phép máy chủ OpenVPN chuyển tiếp lưu lượng giữa các máy khách VPN và mạng đang kết nối tới.

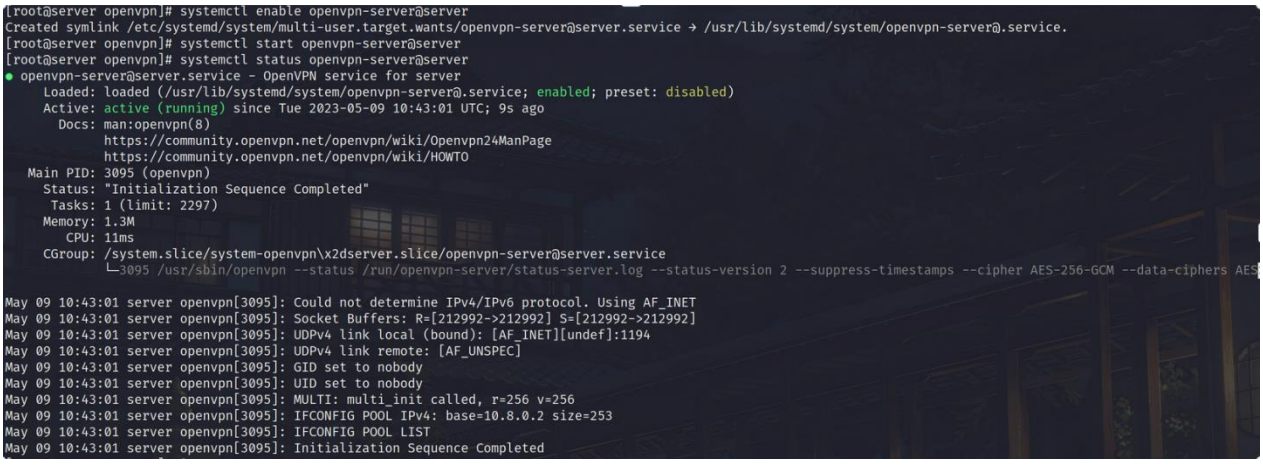
firewall-cmd --add-masquerade --permanent: Lệnh này kích hoạt chế độ masquerade (NAT). Nó cho phép máy chủ OpenVPN chuyển đổi địa chỉ IP nguồn của các gói tin ra ngoại mạng. Điều này giúp máy chủ OpenVPN dịch các địa chỉ IP nguồn của các gói tin gửi đi từ các máy khách VPN.

firewall-cmd --reload: Lệnh này tải lại cấu hình tường lửa để áp dụng các thay đổi mới. Sau khi tải lại, các quy tắc tường lửa mới sẽ được áp dụng và có hiệu lực.

firewall-cmd --list-all: Lệnh này hiển thị tất cả các quy tắc hiện có trong tường lửa để xác nhận lại mọi quy tắc ở trên đã được thêm vào firewalld.

- **Thêm OpenVPN server vào system.**

Tác dụng của việc sử dụng systemd để quản lý OpenVPN server là tạo sự tiện lợi trong việc khởi động, dừng và quản lý quá trình hoạt động của máy chủ OpenVPN. Có thể dễ dàng thực hiện các thao tác quản lý thông qua các lệnh systemctl, chẳng hạn như khởi động tự động khi hệ thống khởi động hoặc tắt máy chủ OpenVPN khi không cần thiết. Điều này giúp đơn giản hóa việc quản lý và duy trì máy chủ OpenVPN một cách hiệu quả.



Hình 19. Thêm OpenVPN server vào system

Sử dụng lệnh **systemctl enable openvpn-server@{server name}**. Với server name là tên của file cấu hình nằm tròn folder **/etc/openvpn/server** ở đây server name có tên là server. Sau khi thực hiện lệnh ta đã cấu hình OpenVPN Server sẽ khởi chạy cùng với Server khi khởi động.

Tiếp theo sử dụng lệnh **systemctl start openvpn-server@server** để khởi động OpenVPN server dưới system.

Sử dụng **systemctl status openvpn-server@server** để kiểm tra trạng thái server ở đây server đã khởi chạy thành công.

2.2.3 Cấu hình OpenVPN Client

- **Tạo file cấu hình cho OpenVPN Client**

Việc tao file cấu hình cho client được thực hiện ở OpenVPN Server sau đó file cấu hình được chuyển đến cho Client.

Tạo file **client.opvn** có nội dug sau:

```
client
remote 192.168.100.251 1194
proto udp
explicit-exit-notify
dev tun
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
verify-x509-name server name
auth SHA256
auth-nocache
cipher AES-128-GCM
tls-client
tls-version-min 1.2
tls-cipher TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256
verb 3
```

Hình 20. Tạo file cấu hình client.opvn

remote 192.168.100.251 1194: Xác định địa chỉ IP và số cổng của máy chủ OpenVPN mà máy khách sẽ kết nối đến.

proto udp: Giao thức sẽ là UDP giống như máy chủ.

explicit-exit-notify: Yêu cầu máy khách gửi thông báo cho máy chủ khi kết nối bị ngắt.

dev tun: Xác định thiết bị mạng được sử dụng cho kết nối VPN. Trong trường hợp này, sử dụng giao diện (interface) tun.

nobind: yêu cầu máy khách không ràng buộc (bind) vào một địa chỉ IP và cổng cụ thể trên máy tính hoặc thiết bị đó. Khi không có chỉ thị này, máy khách sẽ cố gắng ràng buộc (bind) kết nối VPN của mình đến một địa chỉ IP và cổng cụ thể, làm cho nó chỉ có thể hoạt động trên đó.

persist-key: Bật tính bền vững (persistency) cho khóa cá nhân của máy khách qua các phiên kết nối VPN.

persist-tun: Bật tính bền vững cho tunnel VPN qua các phiên kết nối.

remote-cert-tls server: Cấu hình máy khách xác minh chứng chỉ của máy chủ bằng giao thức TLS.

verify-x509-name server name: Xác định Common Name (CN) hoặc Subject Alternative Name (SAN) mà chứng chỉ máy chủ phải khớp. Dùng để xác nhận tên máy chủ đúng như trong cấu hình không.

auth SHA256: Thuật toán cho quá trình xác thực giống như cấu hình của server.

auth-nocache: Vô hiệu hóa việc lưu trữ tạm thời thông tin xác thực để đảm bảo bảo mật.

cipher AES-128-GCM: Xác định thuật toán mã hóa giống như cấu của server.

tls-client: Cấu hình máy khách như một TLS Client, cho phép sử dụng TLS cho kết nối VPN.

tls-version-min 1.2: Xác định phiên bản TLS tối thiểu cho phép cho kết nối giống như cấu hình của server.

tls-cipher TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256: Xác định thuật toán mã hóa TLS sẽ được sử dụng cho kết nối giống như cấu hình của server.

verb 3: Thiết lập mức độ chi tiết của logging của máy khách OpenVPN. Mức độ 3 cung cấp đầu ra chi tiết hơn.

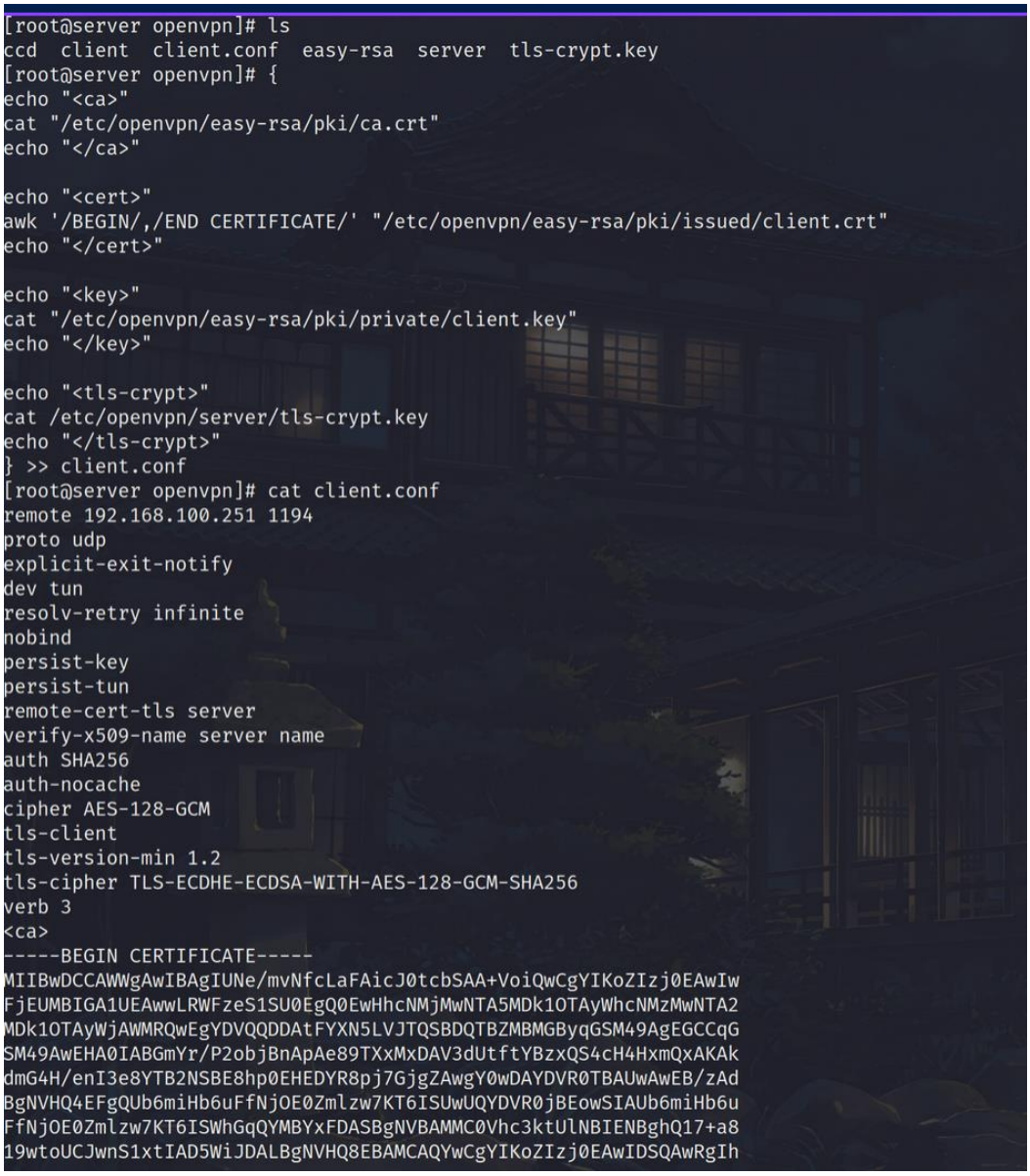
- **Cấu hình bảo mật cho OpenVPN Client**.

Các file cần thiết ở bước này:

+ Chứng chỉ CA
+ Cặp khoá Public và Private của Client
+ Static Key

Các file cần thiết đã được tạo ở bước trên và có đường dẫn ở.

+ /etc/openvpn/easy-rsa/pki/ca.crt
+ /etc/openvpn/easy-rsa/pki/issued/client.crt và /etc/openvpn/easy-rsa/pki/private/client.key
+ /etc/openvpn/server/tls-crypt.key

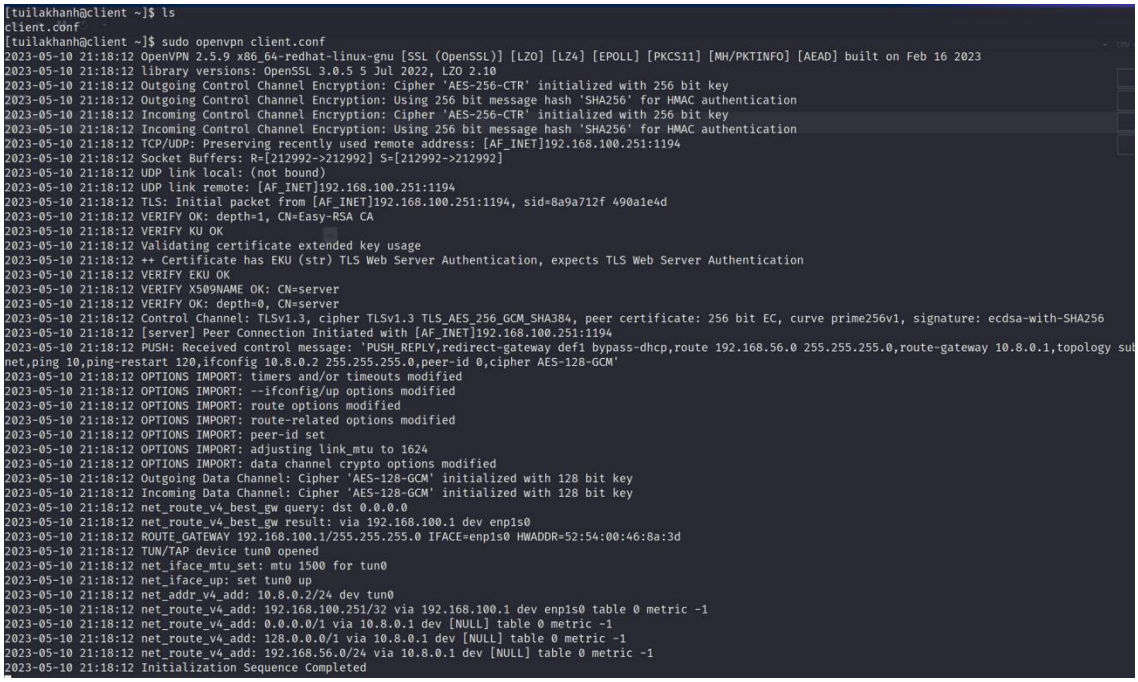


Hình 21. Cấu hình bảo mật cho OpenVPN Client

Thực hiện cat trực tiếp nội dung của các khoá và chứng chỉ bảo mật vào file client.opvn để thuận tiện trong việc gửi file .opvn đến máy Client. Sau khi hoàn thành file .opvn đã sẵn sàng để sử dụng.

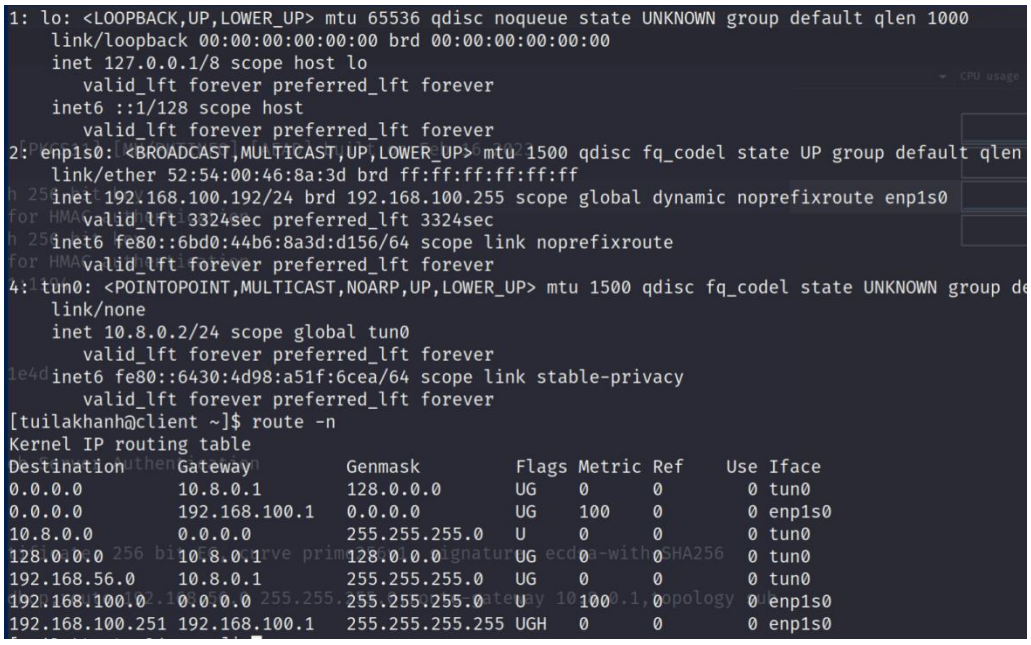
- **Chạy OpenVPN Client trên máy Client**

  Thực hiện truyền file cấu hình đã tạo được ở Server qua máy khách. Chạy openvpn client.conf để chạy OpenVPN ở máy khách. Kết quả trả về như hình dưới thì OpenVPN đã kết nối đến Server thành công.



Hình 22. Chạy openvpn client.conf

- **Kiểm tra IP và Routing.**

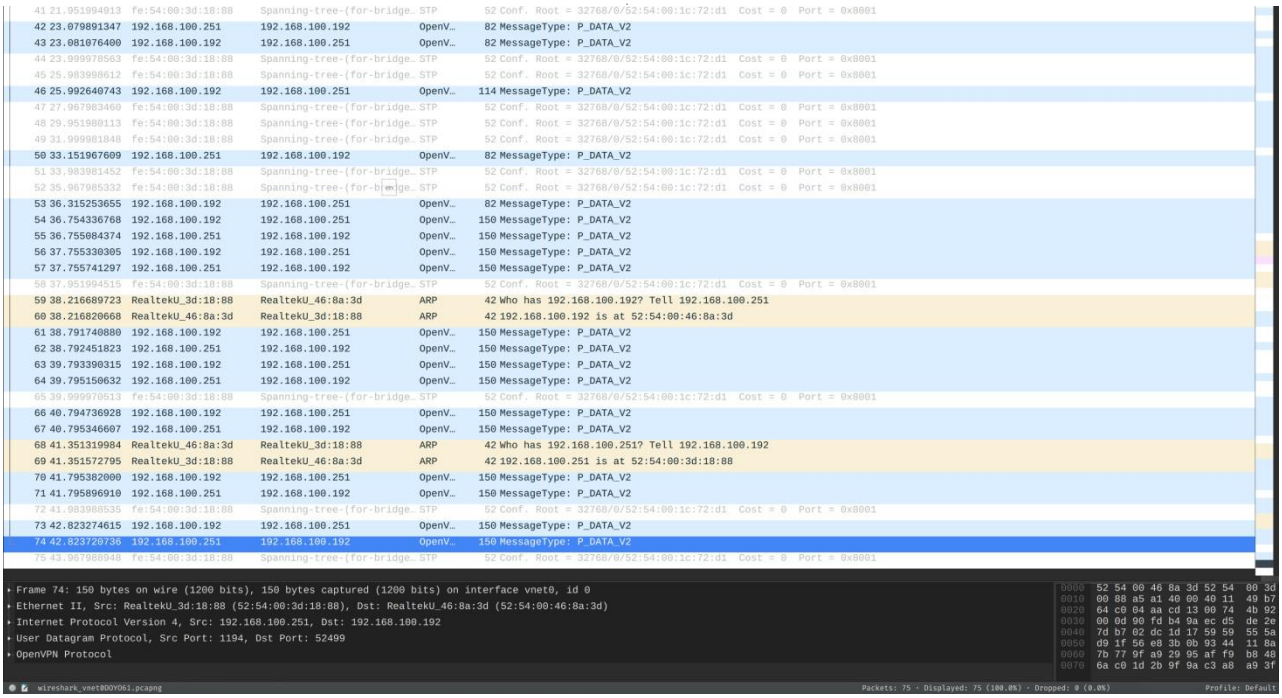


Hình 23. Kiểm tra IP và Routing

Các Route và IP đã được cấu hình đúng như mong muốn.

2.2.4 Gói tin giao tiếp giữa Server và Client

Sử dụng wireshark bắt gói tin ta thấy Server và Client đàng giao tiếp với nhau bằng protocol OpenVPN riêng sử dụng UDP và mọi dữ liệu cũng đã được mã hóa.



Hình 24. Gói tin giao tiếp giữa Server và Client

CHƯƠNG III – KẾT QUẢ VÀ KẾT LUẬN

3.1 Demo

Link demo OpenVPN: https://drive.google.com/drive/folders/1hfNtZtDeT4boyH3-ZEriXGuXEA_1EVAT?usp=sharing

3.2 Tài liệu tham khảo

- OpenVPN Community: OpenVPN Community
- OpenVPN wiki: OpenVPN - Fedora Project Wiki
- Easy-RSA: Easy-RSA - ArchWiki (archlinux.org)

3.3 Phụ lục

3.3.1 Bảng phân công

Tên	Task	Tiến độ hoàn thành
Đỗ Thế Danh	Cài đặt easy-rsa, thiết kế mô hình, viết báo cáo	100%
Lê Hoàng Khánh	Cấu hình OpenVPN,viết báo cáo, hiệu đính báo cáo, quay demo	100%
Trần Nhựt Linh	Tìm hiểu tổng quan, giới thiệu, thiết kế slide, viết báo cáo, trả lời câu hỏi	100%

Bảng 1. Bảng phân công

3.3.2 Bảng tự đánh giá

Báo cáo (1)	1
Thuyết trình (1)	0.75
Cơ sở lý thuyết (2)	2
Demo (5)	4.75
Tổng	8.5/9

Bảng 2. Bảng tự đánh giá

3.3.3 Bảng trả lời câu hỏi

Câu hỏi	Câu trả lời
Những lỗi, rắc rối dễ gặp khi cấu hình và sử dụng OpenVPN là gì?	1. Lỗi cấu hình: Sai sót trong cấu hình mạng, VPN, hoặc máy chủ VPN có thể gây lỗi kết nối hoặc không thể kết nối. 2. Lỗi quản lý chứng chỉ SSL/TLS: Sai sót trong cấu hình chứng chỉ SSL/TLS hoặc không nhận diện được chứng chỉ có thể gây lỗi kết nối. 3. Lỗi tường lửa: Tường lửa không được cấu hình đúng có thể chặn lưu lượng VPN và gây lỗi kết nối. Cần mở các cổng và cho phép lưu lượng VPN đi qua tường lửa. 4. Lỗi mạng: Sự cố hoặc hạn chế trong mạng có thể ảnh hưởng đến kết nối VPN, chẳng hạn như mạng không ổn định, khoảng cách xa giữa máy chủ và máy khách, hoặc giao tiếp không ổn định giữa các máy chủ và máy khách.
Nhược điểm của việc triển khai OpenVPN là gì, đề xuất một số triển khai khác ngoài OpenVPN	- Nhược điểm của việc triển khai OpenVPN: 1. Phức tạp trong cấu hình và quản lý. 2. Yêu cầu kiến thức kỹ thuật cao về mạng và bảo mật. 3. Hiệu suất có thể bị ảnh hưởng do mã hóa và giải mã dữ liệu.

	4. Khả năng mở rộng hạn chế, đặc biệt khi số lượng người dùng tăng. 5. Đòi hỏi cấu hình tường lửa và quản lý chứng chỉ SSL/TLS phức tạp. - Các triển khai VPN khác: 1. IPsec: Một giao thức VPN phổ biến khác với hiệu suất cao và tính bảo mật mạnh mẽ. 2. WireGuard: Một giao thức VPN mới nhưng nhanh, đơn giản và dễ cấu hình. 3. SoftEther VPN: Một phần mềm mã nguồn mở hỗ trợ nhiều giao thức VPN, bao gồm OpenVPN và L2TP/IPsec.
Làm thế nào để cài đặt và sử dụng Open VPN trên các hệ điều hành khác nhau?	- Windows: Tải xuống gói cài đặt OpenVPN từ trang web chính thức. Chạy tệp cài đặt và tuân thủ các hướng dẫn trên màn hình. Sao chép tệp cấu hình (.ovpn) từ máy chủ VPN hoặc tạo mới.Sử dụng ứng dụng OpenVPN để kết nối và quản lý kết nối VPN. - macOS: Tải xuống và cài đặt ứng dụng OpenVPN từ trang web chính thức hoặc thông qua Homebrew. Sao chép tệp cấu hình (.ovpn) từ máy chủ VPN hoặc tạo mới. Sử dụng ứng dụng OpenVPN để kết nối và quản lý kết nối VPN.

Chức năng "Change default route to this VPN tunnel" nghĩa là gì ?	Chức năng "Change default route to this VPN tunnel" có nghĩa là thay đổi tuyến đường mặc định để chuyển hướng toàn bộ lưu lượng mạng qua kết nối VPN. Điều này đảm bảo rằng tất cả các yêu cầu kết nối internet và gói tin mạng sẽ được định tuyến qua kết nối VPN, thay vì đi qua tuyến đường mạng thông thường. Chức năng này thường được sử dụng để truy cập vào tài nguyên mạng nội bộ hoặc tăng tính bảo mật bằng cách mã hóa toàn bộ lưu lượng mạng.
Làm thế nào để tạo một chứng chỉ SSL/TLS tự ký cho OpenVPN trên Linux	1. Cài đặt OpenSSL (nếu chưa có). 2. Sử dụng lệnh openssl để tạo cặp khóa riêng tư và chứng chỉ tự ký. 3. Cấu hình tệp cấu hình OpenVPN để sử dụng chứng chỉ mới tạo. 4. Khởi động dịch vụ OpenVPN và kiểm tra kết nối.

Bảng 3. Trả lời câu hỏi

HẾT