# Implementasi VPN Pada VPS Server Menggunakan OpenVPN dan Raspberry Pi

**Taufik Rahman[1*], Giovanni Maria Vianney Tobia Mariatmojo[2], Hafis Nurdin[3], Herman Kuswanto[4]**

[1,2] Program Studi Teknologi Komputer, Universitas Bina Sarana Informatika, Jakarta Pusat, DKI Jakarta
[3] Program Studi Sistem Informasi, Universitas Nusa Mandiri. Jakarta Timur, DKI Jakarta
[4] Program Studi Teknik Informatika, Universitas Nusa Mandiri, Jakarta Timur, DKI Jakarta
Email: [1*] taufik@bsi.ac.id, [2] giovan13180051@bsi.ac.id, [3] hafis.nnr@nusamandiri.ac.id, [4] herman.hko@nusamandiri.ac.id

(Naskah masuk: 23 Mei 2022, direvisi: 22 Jun 2022, diterima: 22 Jun 2022)

**Abstrak**

Komunikasi jaringan internet butuh keamanan, kemudahan, dan kecepatan transfer data yang baik. Hal ini harus diperhatikan oleh setiap pengguna dalam melakukan kegiatan di dunia maya atau internet, sehingga kerahasiaan informasi bisa terjaga dengan baik dan kemudahan, kecepatan pertukaran data bisa di implementasikan sehingga dapat menjadi suatu nilai lebih. VPN dapat terjadi antara dua PC atau bisa juga antara dua atau lebih jaringan yang berbeda. VPN dapat dibentuk dengan menggunakan teknologi *tunneling* dan *encryption*. *Server OpenVPN* menghasilkan profil jaringan terenkripsi, kemudian semua pengguna dapat membentuk jaringan VPN dan mereka dapat menggunakan layanan organisasi seolah-olah mereka secara geografis berada di tempat yang sama, konfigurasi ini juga memungkinkan penggunaan layanan atau aplikasi eksternal apa pun tanpa memengaruhi keamanan organisasi, VPN dapat berjalan pada IPv6 dan IPv4, dengan kelebihan yang banyak *OpenVPN* melintasi NAT (*Network Address Translation*). Jaringan VPN ini dibuat untuk memudahkan pekerja yang sedang melaksanakan aktivitas WFH (*Work From Home*) VPN sebuah koneksi virtual yang bersifat *private* dan tidak semua orang bisa mengaksesnya. Implementasi VPN dapat mempermudah komunikasi data jarak jauh tanpa khawatir ada tabrakan data, karena dengan menggunakan jaringan virtual ini dapat terhubung secara bersamaan. Dengan VPN, pengguna mempunyai keamanan data yang lebih dibandingkan dengan menggunakan jaringan lokal biasa. Ketika pengguna terhubung ke jaringan VPN maka akan mempunyai IP yang berbeda dengan IP *Physical* dibuktikan dengan tool *tracert* terlihat IP dan *hoop*. *Raspberry Pi* berfungsi untuk membuat sertifikat VPN *client*, karena fungsi tersebut tidak digabungkan ke dalam VPS *Server* karena masalah keamanan jaringan.

**Kata Kunci:** VPN, VPS, *OpenVPN*, *Raspberry Pi*, WFH.

# *VPN Implementation on VPS Server using OpenVPN and Raspberry Pi*

***Abstract***

*Internet network communication requires security, convenience, and good data transfer speed. This must be considered by every user in carrying out activities in cyberspace or the internet, so that the confidentiality of information can be maintained properly and easily, the speed of data exchange can be implemented so that it can be better. VPN can happen between two PCs or it can be between two or more different networks. VPNs can be established using tunneling and encryption technologies. The OpenVPN server generates an encrypted network profile, then all users can form a VPN network and can use the organization's services as if they were geographically in the same place, configuring this also allows the use of any external service or application without affecting the organization's security, VPN can run on both IPv6 and IPv4, with many advantages OpenVPN traverses NAT (Network Address Translation). This VPN network was created to make it easier for workers who are carrying out WFH (Work From Home) vpn activities, a virtual connection that is private and not everyone can access it. VPN implementation can facilitate long-distance data communication without worrying about data collisions, because this virtual network can be connected simultaneously. With vpn, users have more security data compared to using a regular local network. When a user connects to a VPN network, they will have a different IP from the Physical IP as evidenced by the tracert tool that looks ip and*

*hoop. The Raspberry Pi is working to create a VPN client certificate, because that function is not integrated into the VPS Server due to network security issues.*

***Keywords:*** *VPN, VPS, OpenVPN, Raspberry Pi, WFH.*

## I. PENDAHULUAN

Masalah keamanan, kecepatan transfer data adalah salah satu aspek yang penting dari suatu komunikasi jaringan internet, terutama untuk perusahaan besar yang memiliki cabang, universitas, bahkan juga penting untuk perusahaan UMKM(Usaha Mikro, Kecil, dan Menengah).

Komunikasi jaringan internet pasti membutuhkan keamanan, kemudahan, dan kecepatan transfer data yang baik. Hal ini harus diperhatikan oleh pemilik dan juga oleh *staff admin* maupun akuntansi sistem suatu perusahaan dalam melakukan kegiatan di dunia maya atau internet, sehingga kerahasiaan informasi bisa terjaga dengan baik dan kemudahan, kecepatan pertukaran data bisa di implementasikan sehingga dapat menjadi suatu nilai lebih.

*Server OpenVPN* menghasilkan profil sehingga pengguna dapat membuat terowongan terenkripsi, kemudian semua pengguna membentuk jaringan pribadi virtual dan mereka dapat menggunakan layanan organisasi seolah-olah mereka secara geografis berada di tempat yang sama, konfigurasi ini juga memungkinkan penggunaan layanan atau aplikasi eksternal apa pun tanpa memengaruhi keamanan organisasi, berfungsi untuk IPv6 dan IPv4, dengan keuntungan besar *OpenVPN* melintasi NAT. Penelitian terkait keamanan komunikasi di antara nya; Perangkat *VPNoT* didasarkan pada *OpenVPN* yang memberikan kerahasiaan dan integritas, juga berdasarkan *Raspberry Pi* sebagai perangkat keras dan *Linux* sebagai sistem operasi, keduanya menyediakan konektivitas menggunakan berbagai jenis media untuk mengakses Internet dan manajemen jaringan [1]. Untuk satu organisasi, *OpenVPN* dan *Squid Proxy* diimplementasikan di *Server* Utama, untuk ketersediaan tinggi, *Server* Sekunder harus ditambahkan. Klien memerlukan dua terowongan yang dikonfigurasi, satu untuk setiap *server* [2].

*Virtual Private Networks* membuat 'terowongan' terenkripsi antara komputer yang digunakan dan *server host*, dengan lalu lintas internet masuk dan keluar dari *server host*. ISP atau pemerintah hanya dapat melihat bahwa komputer telah terhubung ke *server* VPN dan tidak selebihanya, alamat IP yang telah dikunjungi, dll. semuanya sepenuhnya tersembunyi di balik enkripsi minimum 128-bit [3].

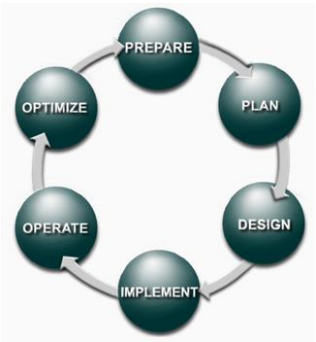
Kemudian protokol keamanan biasa diuji dalam skala kecil, misal dengan *Raspberry Pi*. Penelitian terkait *Raspberry Pi* seperti Implementasi *Efficientnet-Lite* dan *Hybrid* CNN-KNN untuk Pengenalan Ekspresi Wajah pada *Raspberry Pi* [4]. *Raspberry Pi* memainkan peran pemrosesan data dan melakukan kode selain menyimpan data di kartu memori yang terpasang pada *raspberry* [5]. Prototipe pendeteksi tingkat stres menggunakan *Galvanic Skin Response*, DS18B20, dan *Raspberry Pi* [6]. Mengidentifikasi daun tanaman herbal menggunakan metode kecerdasan buatan, yaitu *Convolutional Neural Network* (CNN) yang ditanam pada *Raspberry Pi* [7].

CV. Garuda Kirana merupakan toko penyedia *frozen food* (makanan cepat saji yang tidak mudah basi jika disimpan dalam keadaan beku dan bersuhu dingin) yang cukup berkembang, karena menyediakan berbagai jenis daging berkualitas dan ketersediaanya barangnya terjamin.

Penelitian dilakukan dimasa pandemi *COVID-19*, dimana para pekerja, karyawan diharuskan untuk bekerja dari rumah atau WFH (*Work from Home).* Pencatatan transaksi penjualan, penerimaan harus tetap berjalan walaupun pekerja sedang tidak ada ditempat. Dengan demikian, bagaimana mengimplementasikan VPN pada *Virtual Private Server* (VPS) menggunakan *OpenVPN* digabungkan dengan *Raspberry Pi* pada jaringan *existing* CV. Garuda Kirana yang membutuhkan sebuah sistem keamanan yang baik sehingga jaringan privat tersebut tidak dapat diakses oleh pengguna yang tidak berwenang.

## II. METODOLOGI PENELITIAN

Pada penelitian ini menggunakan model perancangan jaringan PPDIOO dengan 6 fase: *Prepare* (persiapan), *Plan* (Perencanaan), *Design* (Desain), *Implement* (Implementasi), *Operate* (Operasi) dan *Optimize* (Optimasi) [8].




Gambar 1. Model PPDIOO

Metode perancangan jaringan PPDIOO mempunyai fase sebagai berikut:

a. Fase *Prepare*
   Menetapkan kebutuhan apa saja yang dibutuhkan oleh PT.ICC *Export* dalam mengembangkan jaringan, dan mengusulkan konsep arsitektur yang dibutuhkan yang disesuaikan dengan kemampuan finansial pada perusahaan tersebut.
b. Fase *Plan*

Merancang konsep kebutuhan jaringan berdasarkan kepentingan dan kebutuhan pengguna. Fase ini mendeskripsikan karakteristik kebutuhan jaringan, yang memiliki tujuan untuk menilai *gap* analisis pada perancangan pada sebuah arsitektur.

c. Fase *Design*
Desain jaringan dikembangkan berdasarkan persyaratan teknis, dan persiapan yang diperoleh dari kondisi sebelumnya. Hasil desain termasuk didalamnya *flow* jaringan, dan daftar peralatan jaringan.

d. *Implement* (Implementasi)
Perangkat-perangkat akan disesuaikan dengan yang ada di CV. Garuda Kirana. Setiap langkah dalam implementasi, akan menyertakan deskripsi, perkiraan waktu untuk penerapan, evaluasi, dan informasi lainnya sebagai referensi tambahan. Setelah di lakukan implementasi, dalam fase ini juga dilakukan pengujian untuk memastikan bahwa sistem telah berjalan.

e. *Operate* (Operasi)
Memastikan jaringan baru yang sudah terimplementasi di CV. Garuda Kirana telah beroperasi dengan normal. Pengelolaan jaringan, pemeliharaan *routing*, dan mengelola kinerja. Tahapan ini akan dipantau untuk stabilitas dan kinerja jaringan, koreksi konfigurasi, dan kegiatan pemantauan kinerja.

f. *Optimize* (Optimasi)
Fase optimasi, memungkinkan untuk memodifikasi desain jaringan, jika terlalu banyak masalah jaringan yang ditimbulkan, dan untuk memperbaiki masalah kinerja.

Setelah sistem VPN ini selesai kemudian akan di implementasikan pada jaringan komputer di CV. Garuda Kirana, bertujuan untuk mengamankan akses data.

Ada tiga kategori *server online*, yaitu *dedicated*, *virtual private*, dan *cloud server*. Menyewa *server online* memungkinkan seseorang untuk menggunakan komputer dari jarak jauh tanpa memilikinya. Berbeda dengan *server* khusus yang menawarkan kinerja komputasi mentah pada mesin yang terisolasi secara fisik yang didedikasikan untuk satu klien, *Virtual Private Server* (VPS) mendapat bagian dari mesin fisik yang sumber daya perangkat kerasnya diisolasi secara virtual ke beberapa *server* independen kecil yang telah dikonfigurasi sebelumnya. Dengan kata lain, *server* independen kecil yang telah dikonfigurasi sebelumnya merupakan VPS, dan sumber dayanya eksklusif untuk klien VPS. Dari sudut pandang klien, VPS, ketika diakses dari jarak jauh, hampir tidak berbeda dari *server* khusus. Namun, dari sudut pandang *server*, VPS menampilkan kelincahan dalam memvirtualisasikan berbagai konfigurasi *server* dengan harga murah. Perhatikan bahwa *server* cloud yang disediakan oleh perusahaan seperti *Amazon* dan *Alibaba* juga didasarkan pada virtualisasi. Namun, dibandingkan dengan VPS, mereka memiliki banyak manfaat lanjutan terkait keandalan, skalabilitas, dan elastisitas [9].

*Raspberry Pi* adalah *Broad com* BCM2835 SOC (sistem pada papan chip). Muncul dilengkapi dengan 700 MHz, 512 MB SDRAM dan CPU inti ARM1176JZF-S. Port USB 20 dari *raspberry pi boar* hanya menggunakan opsi konektivitas data eksternal. *Raspberry Pi* mendapatkan dayanya dari *adaptor micro* USB, dengan kisaran minimum 2,5 watt (500MA). Grafik, chip khusus dirancang untuk mempercepat manipulasi perhitungan gambar. Ini terintegrasi dengan kabel *Broad com video core* IV yang berguna untuk menjalankan game atau video melalui *raspberry pi* [5].

*OpenVPN* adalah solusi VPN berbasis TLS yang banyak digunakan, tidak ada spesifikasi resmi dari protokol tersebut, yang menjadikannya target yang sangat menarik untuk dianalisis. *OpenVPN* menyediakan *tunneling* untuk memberikan kerahasiaan, otentikasi, dan integritas untuk data yang dikirimkan. Seluruh pesan yang akan ditransmisikan (paket IP atau bingkai *Ethernet*, termasuk meta-datanya seperti pengirim dan penerima) dienkapsulasi dalam pesan *OpenVPN* [10].

*Easy-RSA* adalah utilitas CLI untuk membangun dan mengelola CA PKI. Artinya membuat otoritas sertifikat *root*, dan meminta dan menandatangani sertifikat, termasuk CA perantara dan daftar pencabutan sertifikat (CRL). *Easy-RSA* yang digabungkan dengan *OpenVPN* digunakan untuk menghasilkan dan memperbarui sertifikat [11].

## III. HASIL DAN PEMBAHASAN

CV. Garuda Kirana mempunyai skema jaringan yang sederhana, dengan koneksi internet yang diperoleh dari ISP Indihome serta menggunakan 2 buah *router*, 1 *server*, 3 *laptop* menggunakan *wireless* dan 1 PC dengan menggunakan kabel LAN untuk bisa terhubung. Pada konfigurasi jaringan CV. Garuda Kirana, kita lihat bahwa Kelas IP yang digunakan adalah kelas C.

*Gateway* yang digunakan pada skema jaringan CV. Garuda Kirana adalah 192.168.3.20/24 dan *Subnet Mask* 255.255.255.0 = 24 dimana alamat *broadcast* ialah 192.168.3.255, *Host Minimal* 192.168.3.1 sampai *Host Maximal* di 192.168.3.254. Konfigurasi IP yang digunakan adalah DHCP (*Dynamic Host Configuration Protocol*).

Pada *laptop* 1 menggunakan *IP Address* 192.168.3.101 dengan *Subnet Mask* 255.255.255.0 dan *Default Gateway* 192.168.3.20/24. *Laptop* 2 menggunakan *IP Address* 192.168.3.102 *Subnet Mask* 255.255.255.0 *Default Gateway* 192.168.3.20. *Laptop* 3 dengan *IP Address* 192.168.3.103 *Subnet Mask* 255.255.255.0 *Default Gateway* 192.168.3.20 *dan yang terakhir PC 1 menggunakan IP Address* 192.168.3.104 *Subnet Mask* 255.255.255.0 *Default Gateway* 192.168.3.20, seperti pada Tabel 1.

Tabel 1. *Network ID* dan *Host ID*

| No | Perangkat | *Network ID* | *Host ID* | *IP Address* |
|---|---|---|---|---|
| 1 | *Server* | 192.168.3. | 20 | 192.168.3.20 |
| 2 | *Laptop* 1 | 192.168.3. | 101 | 192.168.3.101 |
| 3 | *Laptop* 2 | 192.168.3. | 102 | 192.168.3.102 |
| 4 | *Laptop* 3 | 192.168.3. | 103 | 192.168.3.103 |
| 5 | PC 1 | 192.168.3. | 104 | 192.168.3.104 |

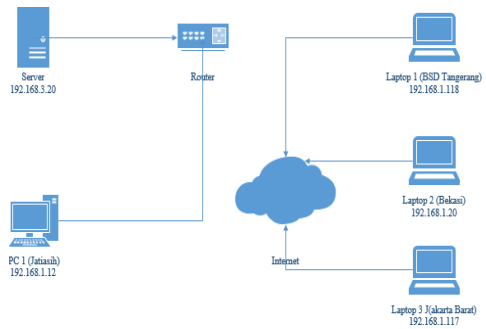
Perangkat lunak yang digunakan untuk mengimplementasikan *Virtual Private Network* (VPN)

menjadi sarana untuk kebutuhan.Spesifikasi perangkat lunak pada Tabel 2 sebagai berikut:

Tabel 2. Spesifikasi Perangkat Lunak

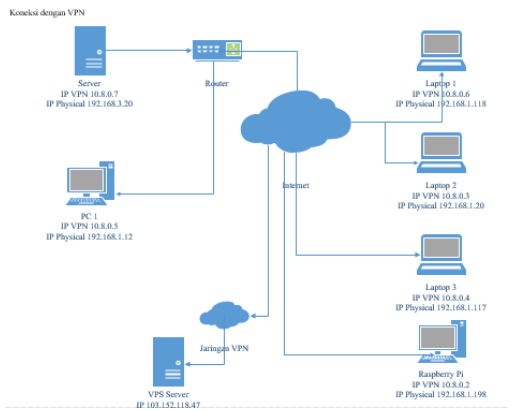
| No | Perangkat Lunak | Jenis |
|---|---|---|
| 1 | Sistem Operasi | *Windows* 10 Pro |
| 2 | Aplikasi VPN | OpenVPN GUI 2.5.0 |
| 3 | Sewa IP *Public* VPS *Server* | *herza.id* |
| 4 | Aplikasi *Remote Access* | *PuTTy* (64-bit), *WinSCP*, *Rebex TinySftpServer* |
| 5 | Antivirus | *Bitdefender* |

Pada penelitian ini, mencoba membuat suatu rancangan jaringan usulan dalam bentuk simulasi menggunakan 1 *router*, 1 PC dan 3 *laptop*, 1 perangkat *raspberry pi*, 1 *server* kantor dan 1 *server* VPS, dimana masalah yang terjadi ketika *staff accounting* atau *staff admin* ingin mengakses kedalam *server*, harus datang ke kantor untuk bisa mengakses file yang ada dalam *server*. Maka dibuatnya Implementasi *Virtual Private Network* ini supaya *staff accounting*, administrasi maupun *staff* lainya dapat mengakses kedalam *server* hanya dari rumah saja dengan koneksi internet.




Gambar 2. Alamat IP Sebelum Terhubung ke VPN

Topologi jaringan pada CV.Garuda Kirana dan topologi internet di rumah masing masing staf, sebelum adanya koneksi VPN seperti pada Gambar 2. Pada CV. Kirana terdapat *Server*, PC1 (Jatiasih) yang terkoneksi ke router. Kemudian laptop staf dari rumah terkoneksi ke internet.




Gambar 3. Alamat IP Setelah Terkoneksi Dengan VPN

Pada skema jaringan usulan pada Gambar 3. terlihat ada perangkat tambahan yaitu VPS *Server* dan juga *Raspberry Pi*, fungsi VPS *Server* adalah sebagai *Server* VPN dan yang memberikan IP *Public* kepada setiap *client* yang mempunyai sertifikat VPN, lalu fungsi *Raspberry Pi* adalah untuk membuat konfigurasi sertifikat *client* yang nantinya akan dipakai tiap *client* yang ingin terhubung ke toko CV. Garuda Kirana. Pembuatan sertifikat klien tidak digabungkan kedalam VPS *Server* karena masalah keamanan.

Dengan menggunakan jaringan VPN, maka *software* VPN tersebut mempunyai kemampuan untuk menambahkan *tunneling* sehingga setiap *user* akan mempunyai IP yang berbeda dengan IP *Physical*. Berikut adalah hasil *screenshot* alamat IP yang berbeda dalam satu *device*.

```
C:\Users\TBID Tech>ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : LAPTOP-P8054LIM
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No

Unknown adapter OpenVPN TAP-Windows6:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : TAP-Windows Adapter V9
   Physical Address. . . . . . . . . : 00-FF-9D-5C-30-2B
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::e1d6:8233:2f14:df19%16(Preferred)
   IPv4 Address. . . . . . . . . . . : 10.8.0.3(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : 03 July 2021 19:24:13
   Lease Expires . . . . . . . . . . : 03 July 2022 19:24:13
   Default Gateway . . . . . . . . . :
   DHCP Server . . . . . . . . . . . : 10.8.0.254
   DHCPv6 IAID . . . . . . . . . . . : 167837597
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-26-29-A0-81-7C-8A-E1-35-CB-03
   DNS Servers . . . . . . . . . . . : 94.140.14.14
                                       94.140.15.15
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

Gambar 4. IP *Address* dari Jaringan VPN

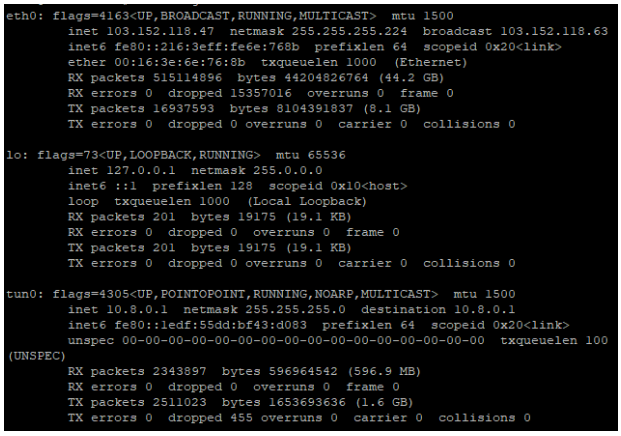
Dalam Gambar 4, dapat dilihat bahwa IP *Address* yang terdapat pada laptop *client* adalah 10.8.0.3 dengan *Subnet Mask* 255.255.255.0 dimana *Network* ID nya 10. dan *Host* ID nya adalah 8.0.3.

```
Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Qualcomm Atheros QCA9377 Wireless Network Adapter
   Physical Address. . . . . . . . . : E4-AA-EA-F3-AF-C7
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::a14f:49d6:c2d1:8d4b%11(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.1.118(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : 07 September 2021 11:12:00
   Lease Expires . . . . . . . . . . : 08 September 2021 11:11:59
   Default Gateway . . . . . . . . . : 192.168.1.1
   DHCP Server . . . . . . . . . . . : 192.168.1.1
   DHCPv6 IAID . . . . . . . . . . . : 165980906
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-26-29-A0-81-7C-8A-E1-35-CB-03
   DNS Servers . . . . . . . . . . . : 118.136.64.5
                                       111.95.141.4
                                       61.247.0.130
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

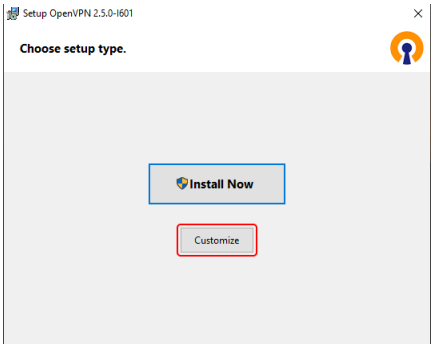
Gambar 5. *Physical Address*

Pada Gambar 5 terlihat alamat IP dari laptop *client* berbeda dengan alamat IP VPN pada laptop tersebut. *Physical* IP *Address* adalah 192.168.1.118 dengan *Subnet Mask* 255.255.255.0 dan alamat *gateway* 192.168.1.1 dimana *network ID* nya adalah 192.168.1 dan *host* ID ya 118.



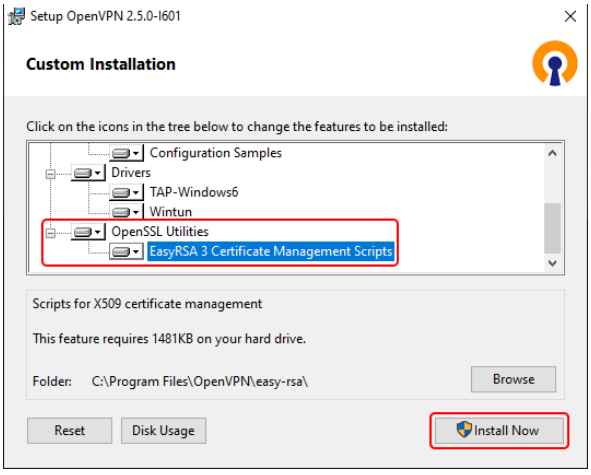

Gambar 6. IP *Address* VPN *Server*

Pada Gambar 6 terlihat bahwa *server* ataupun *client* akan mempunyai dua alamat IP karena *software* VPN tersebut mempunyai kemampuan untuk menambahkan *tunneling* sehingga setiap *user* akan mempunyai IP yang berbeda dengan IP *Physical.* Instalasi Perangkat Lunak OpenVPN. Pada Gambar 7, sebelum memulai proses instalasi, klik '*Customize'.*




Gambar 7. Tampilan Awal Setup *OpenVPN*

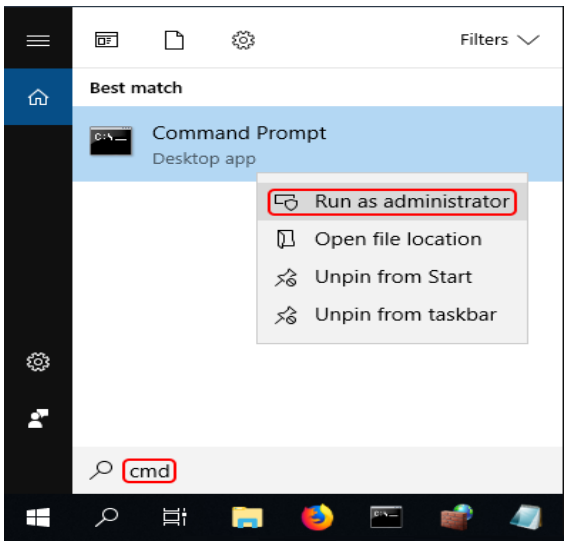
Saat berada di jendela 'Instalasi Kustom' pada Gambar 8, gulir ke bawah untuk menemukan *OpenSSL Utilities* → *EasyRSA 3 Certificate Management Scripts*, pastikan diinstal bersama dengan *OpenVPN* dan klik *'Install Now'.*




Gambar 8. Menu *Custom Installation OpenVPN*

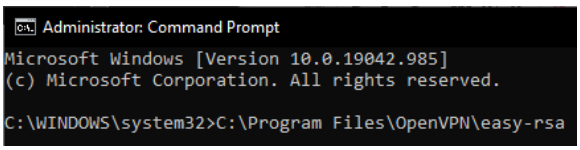
**Mempersiapkan *Easy-RSA***

Sekarang kita mulai membuat sertifikat dan kunci. Kita akan menggunakan aplikasi *Easy-RSA 3* yang diinstal bersama dengan *OpenVPN*. Perintah *Easy-RSA* dijalankan melalui *Command Prompt Windows*. Dapat dibuka dengan mengetik *cmd* di bilah pencarian *Windows* ( tombol *Windows* + S ) dan menjalankannya sebagai *administrator* seperti Gambar 9.




Gambar 9. *Run Command Prompt*

Perintah pada Gambar 10, dilakukan untuk mengubah direktori saat ini ke folder *EasyRSA.*




Gambar 10. Mengubah Direktori pada CMD

Sebelum menghasilkan *file* dengan *Easy-RSA*, kita harus terlebih dahulu menginisialisasi direktori untuk Infrastruktur Kunci Publik (PKI). Ini dapat dilakukan dengan perintah pada Gambar 11.

```
./easyrsa init-pki
```

Gambar 11. Inisialisasi Direktori

Buka *file vars.bat* dengan *editor* teks *Notepad* dengan perintah pada Gambar 12.

```
notepad vars.bat
```

Gambar 12. Membuka *File Editor* Teks *Notepad*

Pada Gambar 13 adalah *file template* untuk membuat sertifikat, informasi yang disimpan di sini diberikan sebagai

nilai default selama pembuatan sertifikat. Ubah baris berikut sesuai kebutuhan saja.

```
setel KEY_COUNTRY=US
setel KEY_PROVINCE=CA
atur KEY_CITY=SanFrancisco
atur KEY_ORG=OpenVPN
atur KEY_EMAIL=mail@host.domain
```

Gambar 13. *File Template* Sertifikat

Untuk mengatur ukuran kunci untuk parameter *Diffie Hellman* pada Gambar 14.

```
atur DH_KEY_SIZE=2048
```

Gambar 14. Parameter *Diffie Hellman*

Setelah semua selesai, simpan *file* dan tutup *file editor* dengan perintah pada Gambar 15.

```
vars.bat
./easyrsa clean-all
```

Gambar 15. Perintah Tutup *File Editor*

**Membuat Sertifikat dan Kunci**

Membuat sertifikat dan kunci. Mulai dengan otoritas sertifikat (CA) seperti pada Gambar 16.

```
./easyrsa build-ca nopass
```

Gambar 16. Membuat Otoritas Sertifikat

Selanjutnya membuat sertifikat dan kunci *server* seperti pada Gambar 17.

```
./easyrsa build-server-full server nopass
```

Gambar 17. Membuat Sertifikat dan Kunci *Server*

Selanjutnya membuat sertifikat dan kunci *clients* seperti pada Gambar 18.

```
./easyrsa build-client-full Client1 nopass
```

Gambar 18. Membuat Sertifikat dan Kunci *Client*

Dan yang terakhir, membuat parameter *Diffie Hellman* seperti pada Gambar 19.

```
./easyrsa gen-dh
```

Gambar 19. Membuat parameter *Diffie Hellman*

*File* yang dihasilkan dan ditandatangani akan muncul di direktori berikut ini (secara *default*) :

*CA certificate (C:\Program Files\OpenVPN\easy-rsa\pki). Diffie Hellman parameters (C:\Program Files\OpenVPN\easy-rsa\pki). Client and Server keys (C:\Program Files\OpenVPN\easy-rsa\pki\private). Client and Server certificates (C:\Program Files\OpenVPN\easy-rsa\pki\issued).*

**Konfigurasi *OpenVPN***

Menggunakan skrip instalasi dari *angristan* di *GitHub*. Perintah ini untuk mengunduhnya seperti pada Gambar 20.

```
wget https://raw.githubusercontent.com/angristan/openvpn-install/master/openvpn-install.sh -O openvpn-install.sh
```

Gambar 20. Skrip Instalasi dari *Angristan*

Kemudian dijalankan dengan perintah seperti pada Gambar 21.

```
sudo bash openvpn-install.sh
```

Gambar 21. Perintah Untuk Instalasi *OpenVPN*

Setelah itu akan tampil seperti pada Gambar 22.

```
Do you want to enable IPv6 support (NAT)? [y/n]: n

What port do you want OpenVPN to listen to?
   1) Default: 1194
   2) Custom
   3) Random [49152-65535]
Port choice [1-3]: 1

What protocol do you want OpenVPN to use?
UDP is faster. Unless it is not available, you shouldn't use TCP.
   1) UDP
   2) TCP
Protocol [1-2]: 1

What DNS resolvers do you want to use with the VPN?
   1) Current system resolvers (from /etc/resolv.conf)
   2) Self-hosted DNS Resolver (Unbound)
   3) Cloudflare (Anycast: worldwide)
   4) Quad9 (Anycast: worldwide)
   5) Quad9 uncensored (Anycast: worldwide)
   6) FDN (France)
   7) DNS.WATCH (Germany)
   8) OpenDNS (Anycast: worldwide)
   9) Google (Anycast: worldwide)
   10) Yandex Basic (Russia)
   11) AdGuard DNS (Anycast: worldwide)
   12) NextDNS (Anycast: worldwide)
   13) Custom
DNS [1-12]: 11
```

Gambar 22. Menu Instalasi *OpenVPN* pada *Raspberry Pi*

Sebagian besar kita akan mempertahankan nilai *default*, jadi cukup tekan *Enter* untuk setiap pertanyaan jika kita tidak tahu. Beberapa detik kemudian, kita akan ditanya beberapa informasi tentang pengguna pertama yang dibuat. Beri nama dan kata sandi jika kita perlu saja.

Setelah selesai, skrip berakhir dan kita akan diberikan akses ke *file* konfigurasi pertama seperti pada Gambar 23.

```
The configuration file has been written to /home/pi/android.ovpn.
Download the .ovpn file and import it in your OpenVPN client.
```

Gambar 23. Skrip Akses ke *File* Konfigurasi Pertama

**Mendapatkan *File OpenVPN***

Untuk mendapatkan *file* klien, kita harus mendapatkanya di *Raspberry Pi* oleh *wizard OpenVPN*, yaitu di

*/home/pi/ovpn.ovpn*. Setelah itu kita hubungkan *Raspberry Pi* dengan *FileZilla* atau *WinSCP*, dan mentransfer *file* dari komputer satu ke komputer lain.

***Edit Config OpenVPN***

Supaya bisa terkoneksi dengan IP *Public* atau *Hosting*, kita akan ubah beberapa konfigurasi yang ada di dalam *file client OpenVPN* seperti pada Gambar 24.

```
client
proto udp
explicit-exit-notify
remote 103.152.118.47  7070
dev tun
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
verify-x509-name server_y8K4KXKNoj5fagdI 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
ignore-unknown-option block-outside-dns
setenv opt block-outside-dns # Prevent Windows 10 DNS leak
verb 3
```

Gambar 24. Konfigurasi *File OpenVPN*

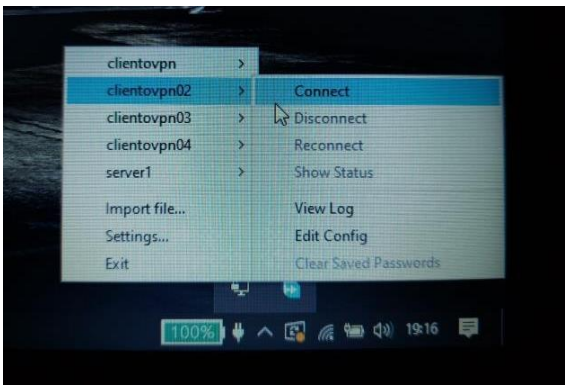
**Hasil Pengujian Jaringan**

***Connect* VPN dengan *OpenVPN* GUI**

Di dalam *step* ini, akan diuji apakah konfigurasi VPN yang telah dibuat berhasil atau tidak. Langkah pertama, buka aplikasi *OpenVPN GUI* dan ketika klik aplikasi tersebut, akan muncul logo bahwa *OpenVPN GUI* telah aktif, namun belum terhubung seperti pada Gambar 25.


Gambar 25. Logo *OpenVPN* GUI Belum Tersambung

Selanjutnya, kita klik kanan pada logo *OpenVPN* yang muncul di *taskbar* tadi lalu akan muncul pilihan seperti pada Gambar 26. Kita tinggal pilih file klien yang telah di *import* agar bisa terkoneksi dengan jaringan VPN.




Gambar 26. Koneksi *Client OpenVPN*

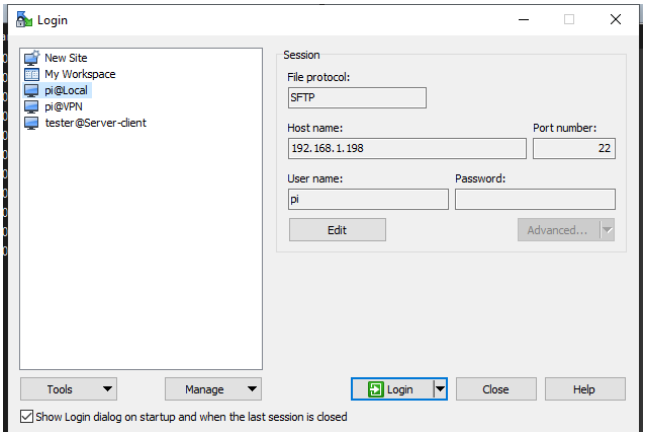
Setelah berhasil *connect* akan muncul tampilan seperti pada Gambar 27, menandakan bahwa kita sudah mendapatkan koneksi *private* dengan *OpenVPN*.


Gambar 27. *Client* Berhasil Terhubung ke Jaringan VPN

***Connect* Ke *Client* Dengan *OpenVPN* Menggunakan *Software WinSCP***

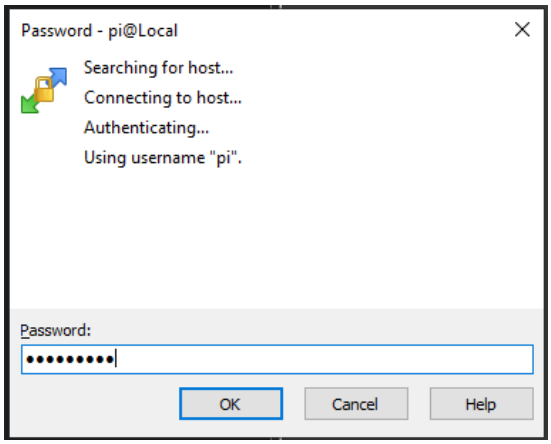
Setelah pengujian koneksi *OpenVPN* berhasil, selanjutnya mencoba *remote* satu arah dari laptop ke *client* yang ada di CV. Garuda Kirana untuk *upload* dan *download file*.

Sebelum *remote* dengan jaringan VPN, mencoba dengan jaringan lokal terlebih dahulu apakah berhasil terkoneksi atau belum. Gambar 28 adalah tampilan untuk *login* dengan memasukan *IP Address*, *Password* dan *Port Number*.



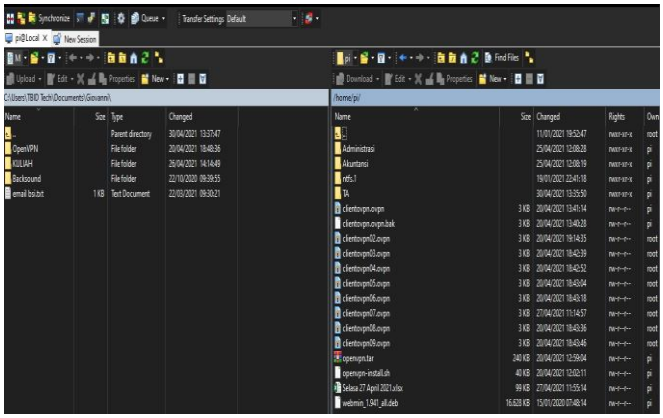

Gambar 28. Tampilan *Login* dengan *WinSCP*

Setelah klik *Login*, akan muncul tampilan seperti pada Gambar 29, karena tidak menyimpan *password* di *home* dari *WinSCP*. Setelah memasukan *password*, klik OK.



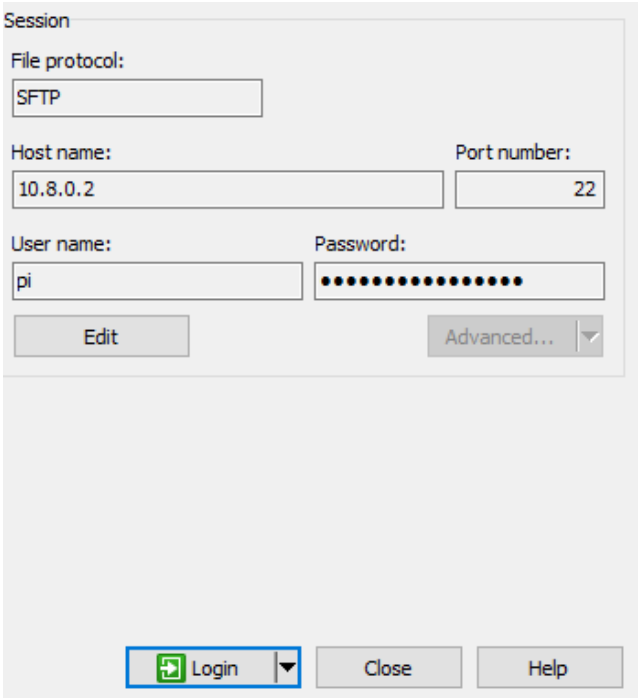

Gambar 29. Tampilan *Input Password*

Setelah proses *Login* berhasil, akan tampil seperti Gambar 30 artinya koneksi menggunakan jaringan lokal berhasil, dan sekarang tes dengan jaringan VPN.



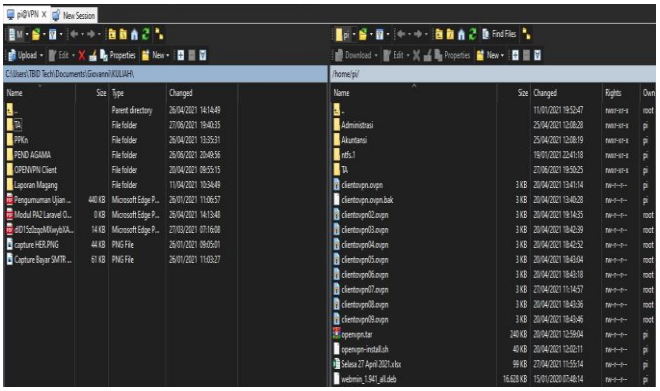
Gambar 30. Tampilan *Client* Saling Terhubung Dengan Jaringan Lokal

Selanjutnya, langsung buat *New Session* untuk menguji apakah *laptop*/ PC yang dipakai dapat terhubung dengan melalui jaringan VPN yang telah dibuat seperti pada Gambar 31.



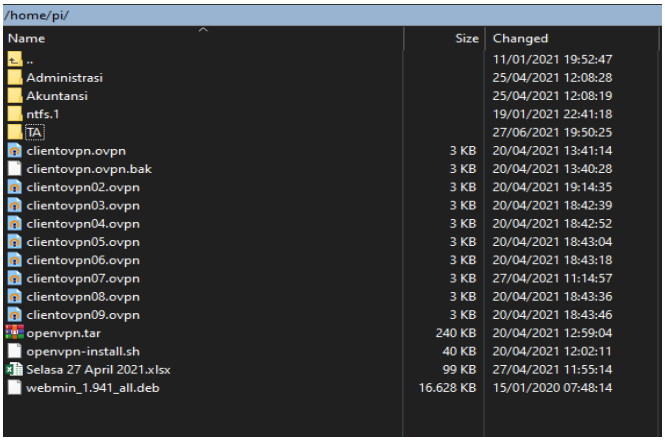

Gambar 31. *New Session* Melalui Jaringan VPN

Tampilan pada Gambar 32 menunjukan bahwa dapat terhubung ke *client pi* dengan jaringan VPN, dan sekarang coba *upload* atau *download* file.



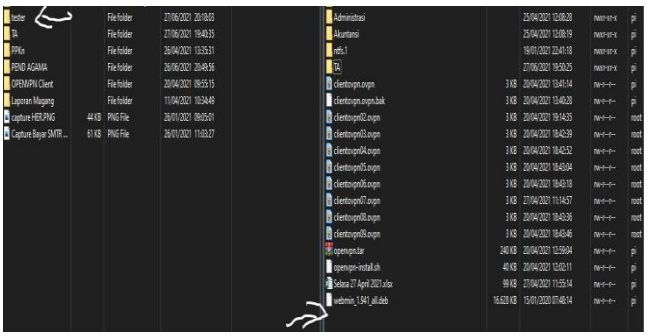
Gambar 32. Tampilan *Client* Berhasil Terhubung Dengan VPN

Pada Gambar 33, belum melakukan *upload* file, dan sekarang akan dicoba dengan membuat *folder* bernama "*tester*" dengan tujuan direktori "*/home/pi/"*.



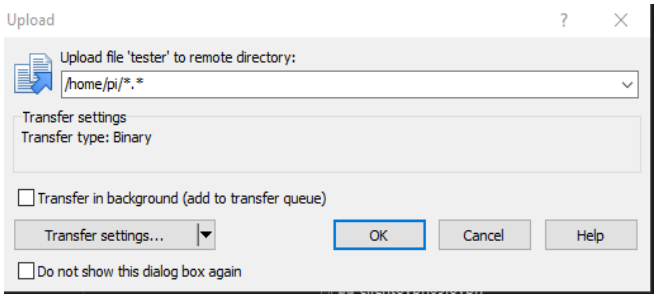

Gambar 33. Tampilan Direktori *"/home/pi/"*

Pada Gambar 34, tanda panah berwarna putih sebelah kiri menunjukan *folder* dibuat di *laptop*, jadi sekarang akan memindahkan *folder* ke panah sebelah kanan atau meng-*upload file folder* ke direktori "*/home/pi/"* dengan menggunakan jaringan VPN.



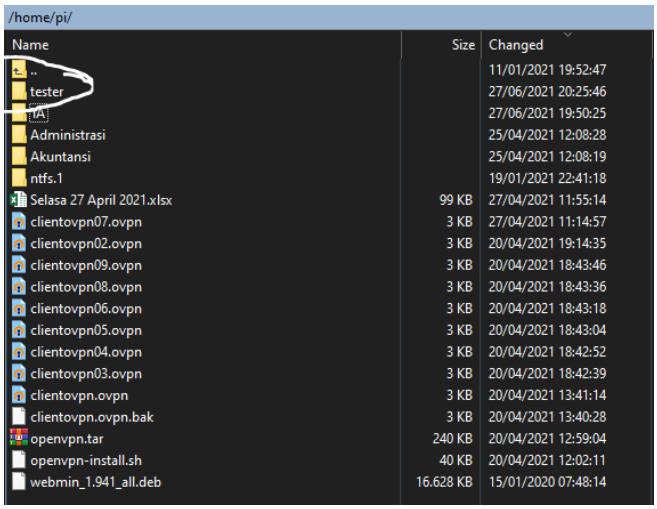
Gambar 34. Tampilan Sebelum *Upload Folder* Melalui VPN

Pada Gambar 35 menunjukan bahwa *folder* atau *file "tester"* akan dipindahkan ke "*/home/pi/"*.



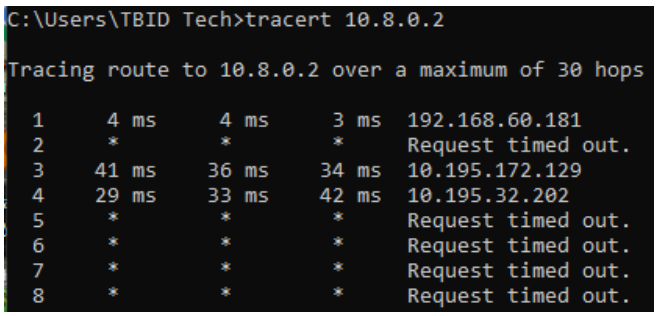

Gambar 35. Tampilan Menu *Upload* File ke Tujuan

Setelah uji koneksi jaringan VPN dengan *upload folder* bernama "*tester*" pada Gambar 36, membuktikan bahwa *remote* VPN yang telah dikonfigurasi dapat dipakai untuk *upload* dan *download file* atau *folder* secara *multiuser*, jadi tidak perlu bergantian untuk bisa terkoneksi karena di jaringan VPN ini dapat saling terhubung dengan beberapa *client* sekaligus.



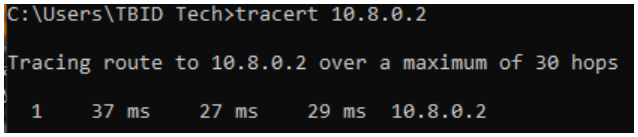

Gambar 36. *Upload File* Melalui Jaringan VPN Berhasil

Pada hasil pengujian Gambar 37 didapat hasil uji *tracert* ke IP *adddress* 10.8.0.2 saat *client* tidak terhubung pada jaringan VPN.




Gambar 37. Hasil *Tracert* Tanpa VPN

Hasil pengujian Gambar 38 didapat hasil uji *tracert* ke IP *adddress* 10.8.0.2 saat *client* terhubung pada jaringan VPN.




Gambar 38. Hasil *tracert* dengan VPN

## IV. KESIMPULAN

Hasil implementasi jaringan *Virtual Private Network* yang dibuat bahwa VPN sebuah koneksi virtual yang bersifat *private* dan tidak semua orang bisa mengaksesnya. VPN mempunyai 4 fungsi utama dalam keamanan datanya yaitu, Transfer Data, *Confidentially* (Kerahasiaan Data), *Data Integrity* (Keutuhan Data) dan *Origin Authentication* (Autentikasi Sumber). Implementasi VPN dapat mempermudah komunikasi data jarak jauh tanpa khawatir ada tabrakan data, karena dengan menggunakan jaringan virtual ini kita dapat terhubung secara bersamaan. Dengan VPN, pengguna mempunyai keamanan data yang lebih dibandingkan dengan menggunakan jaringan lokal biasa. Ketika *client* terhubung ke jaringan VPN maka setiap *user* akan mempunyai IP yang berbeda dengan IP *Physical*. *Raspberry Pi* berfungsi untuk membuat sertifikat VPN *client*, karena fungsi tersebut tidak digabungkan ke dalam VPS *Server* karena masalah keamanan jaringan.

## REFERENSI

[1] C. A. Romero Goyzueta, J. E. Cruz De La Cruz, and C. D. Cahuana, "VPNoT: End to End Encrypted Tunnel Based on OpenVPN and Raspberry Pi for IoT Security," in *2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, 2021, pp. 1–5.

[2] J. E. C. de la Cruz, C. A. R. Goyzueta, and C. D. Cahuana, "OpenVProxy: Low Cost Squid Proxy Based Teleworking Environment with OpenVPN Encrypted Tunnels to Provide Confidentiality, Integrity and Availability," 2020, pp. 20–23.

[3] N. Kumar, S. Sunil, K. Rawani, H. Shankar, P. Tonde, and B. Kishor, "VPN IS SECURE THEN OTHER PROXIES FOR INTERNET FILTARATION," *Int. Educ. Res. J. [IERJ]*, vol. 5, no. 2, pp. 15–17, 2019.

[4] M. N. Ab Wahab, A. Nazir, A. T. Z. Ren, M. H. M. Noor, M. F. Akbar, and A. S. A. Mohamed, "Efficientnet-Lite and Hybrid CNN-KNN Implementation for Facial Expression Recognition on Raspberry Pi," *IEEE Access*, vol. 9, pp. 134065–134080, 2021.

[5] A. M. Abd-Elrahim, A. Abu-Assal, A. A. A. A. Mohammad, A. I. M. Al-Imam, A. H. A. Hassan, and M. A. M. Muhi-Aldeen, "Design and Implementation of Raspberry Pi based Cell phone," *Proc. 2020 Int. Conf. Comput. Control. Electr. Electron. Eng. ICCCEEE 2020*, 2021.

[6] F. P. Sabrina and B. B. Murti, "Implementasi Elastic Stack Pada Sistem Pendeteksi Tingkat Stres Menggunakan Sensor GSR dan DS18B20 Berbasis

Raspberry Pi," *Teknika*, vol. 11, no. 1, pp. 38–44, 2022.
[7] Haryono, Khairul Anam, and Azmi Saleh, "Autentikasi Daun Herbal Menggunakan Convolutional Neural Network dan Raspberry Pi," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 9, no. 3, pp. 278–286, 2020.
[8] T. Rahman, E. Sulistianto, A. Sudibyo, and B. Wijonarko, "Per Connection Classifier Load Balancing dan Failover MikroTik pada Dua Line Internet," pp. 195–209.
[9] X. Ma *et al.*, "One Host with so Many IPs! On the Security Implications of Dynamic Virtual Private *Server*s," *IEEE Commun. Mag.*, vol. 59, no. 2, pp. 64–69, 2021.
[10] L. A. Daniel, E. Poll, and J. De Ruiter, "Inferring OpenVPN State Machines Using Protocol State Fuzzing," *Proc. - 3rd IEEE Eur. Symp. Secur. Priv. Work. EURO S PW 2018*, pp. 11–19, 2018.
[11] A. Ometov *et al.*, "Dynamic Trust Associations over Socially-Aware D2D Technology: A Practical Implementation Perspective," *IEEE Access*, vol. 4, pp. 7692–7702, 2016.