Irfan Dhia Irsyad

PT Validata Teknologi, Bandung, Indonesia

MUM Indonesia 2016, Jakarta

Create Your Own Secure VPN Connection Using OpenVPN

Biodata Pribadi

- Irfan Dhia Irsyad
- Bandung
- 28 Oktober
- Jl. Cihampelas Blk. 16 Komp. Perumahan

Flat Kimia Farma No. B.09 Kota Bandung

- Server Architect at PT Validata Teknologi, Insan Mulia Building Jl. Soekarno Hatta No. 550 Bandung
- IT Consultant based in Bandung
- 085797246688
- MTCNA
- Find me on all social media : irdhirs

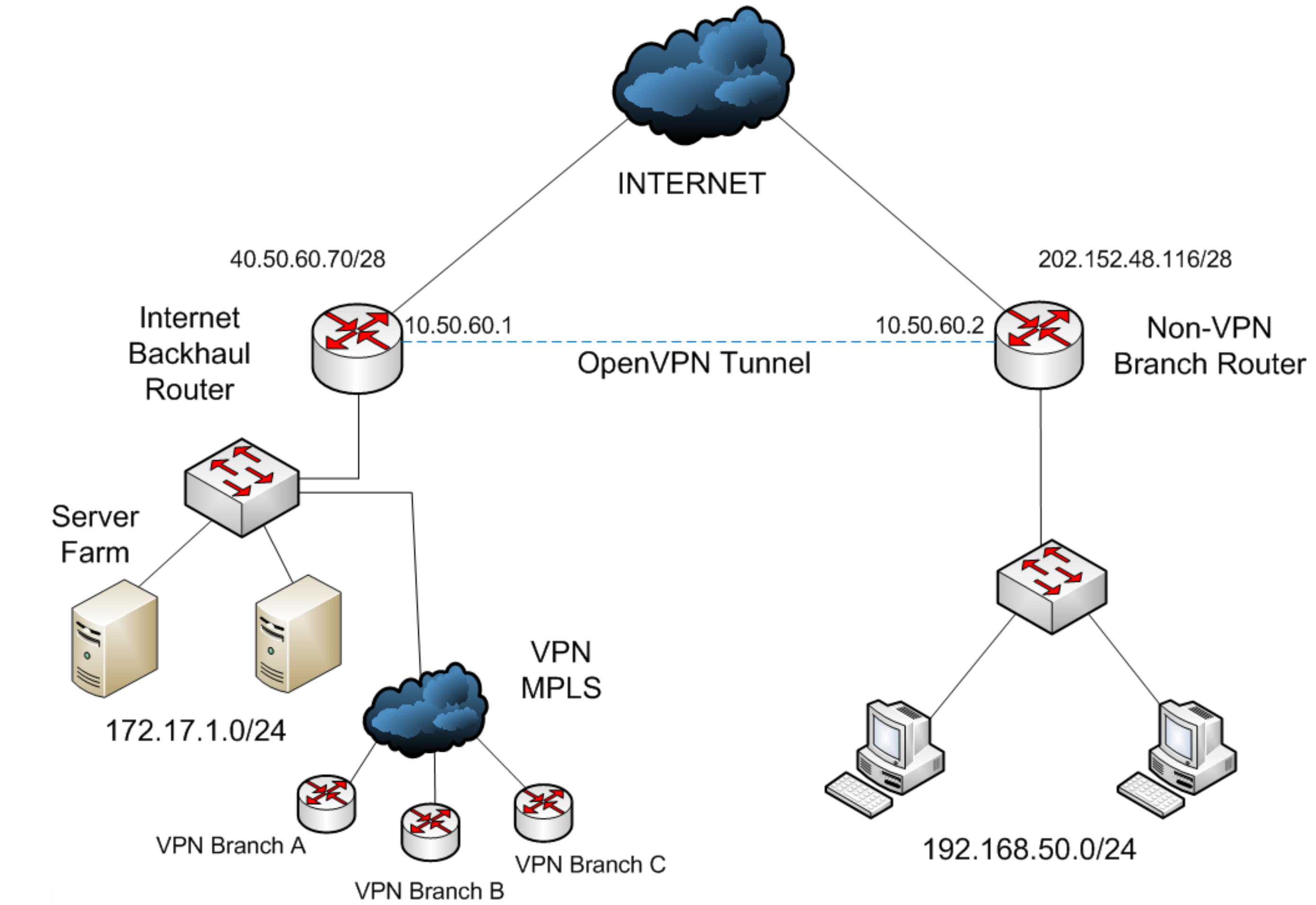
Why?

- All available infrastructures can’t reach existing location
- ISP difference at each spot
- Limited at budgeting ☺
- Free, but secure
- etc.

How To Create It?

- Certificate creation
  - Generate in OpenSSL (linux server required)
  - Generate in MikroTik router
- Import the certificates
- Set OpenVPN server at head office side
- Set OpenVPN client at branch office side
- Add network routing
- Configuration testing

The Topology



CERTIFICATE CREATION USING OPENSSL

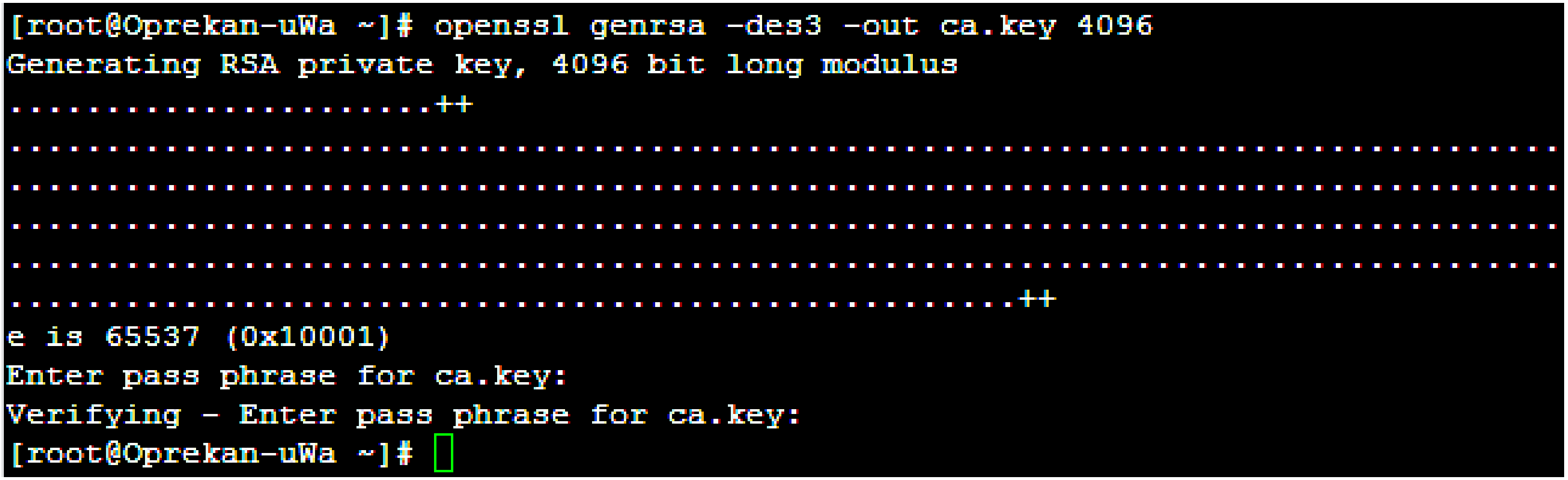
Certificate Creation Using OpenSSL

- Using linux server with OpenSSL installed before
- Certificate creation steps :
  - Generate CA Private Key & CA Certificate Pair
  - Generate Private Key/Certificate Pair for server side
  - Generate Private Key/Certificate Pair for client side
  - Certificate function testing

Generate CA Private Key & CA Certificate Pair

- 1. Use this command in linux server :

  openssl genrsa -des3 -out ca.key 4096



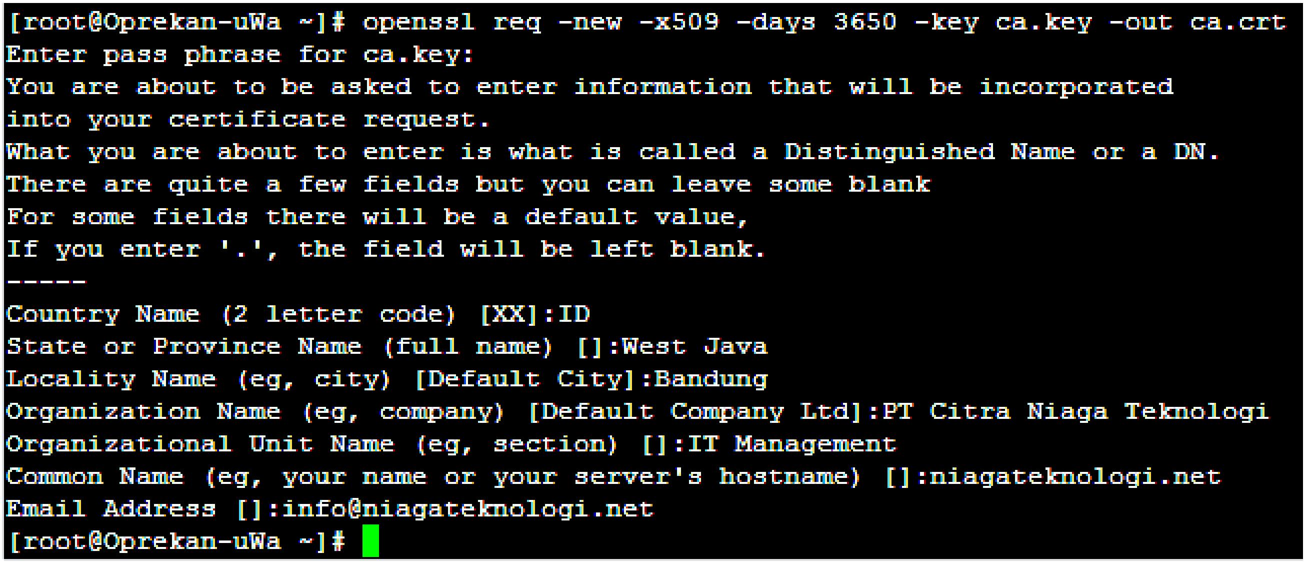
```
[root@Oprekan-uWa ~]# openssl genrsa -des3 -out ca.key 4096
Generating RSA private key, 4096 bit long modulus
......................++
................................................................................
................................................................................
................................................................................
................................................................................
....................................................++
e is 65537 (0x10001)
Enter pass phrase for ca.key:
Verifying - Enter pass phrase for ca.key:
[root@Oprekan-uWa ~]# 
```

- 2. While generating RSA private key, we're asked to provide passphrase. For example, we used **passwordkita** for the passphrase. Then enter.

Generate CA Private Key & CA Certificate Pair

3. Generate CA Certificate file using this command :

openssl req -new -x509 -days 3650 -key ca.key -out ca.crt



```
[root@Oprekan-uWa ~]# openssl req -new -x509 -days 3650 -key ca.key -out ca.crt
Enter pass phrase for ca.key:
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.
-----
Country Name (2 letter code) [XX]:ID
State or Province Name (full name) []:West Java
Locality Name (eg, city) [Default City]:Bandung
Organization Name (eg, company) [Default Company Ltd]:PT Citra Niaga Teknologi
Organizational Unit Name (eg, section) []:IT Management
Common Name (eg, your name or your server's hostname) []:niagateknologi.net
Email Address []:info@niagateknologi.net
[root@Oprekan-uWa ~]#
```

4. Provide the passphrase which is created before : **passwordkita**. Then provide the information needed for *CA Certificate* such as *Common Name (CN)*, *Organization*, *State or province*, etc.

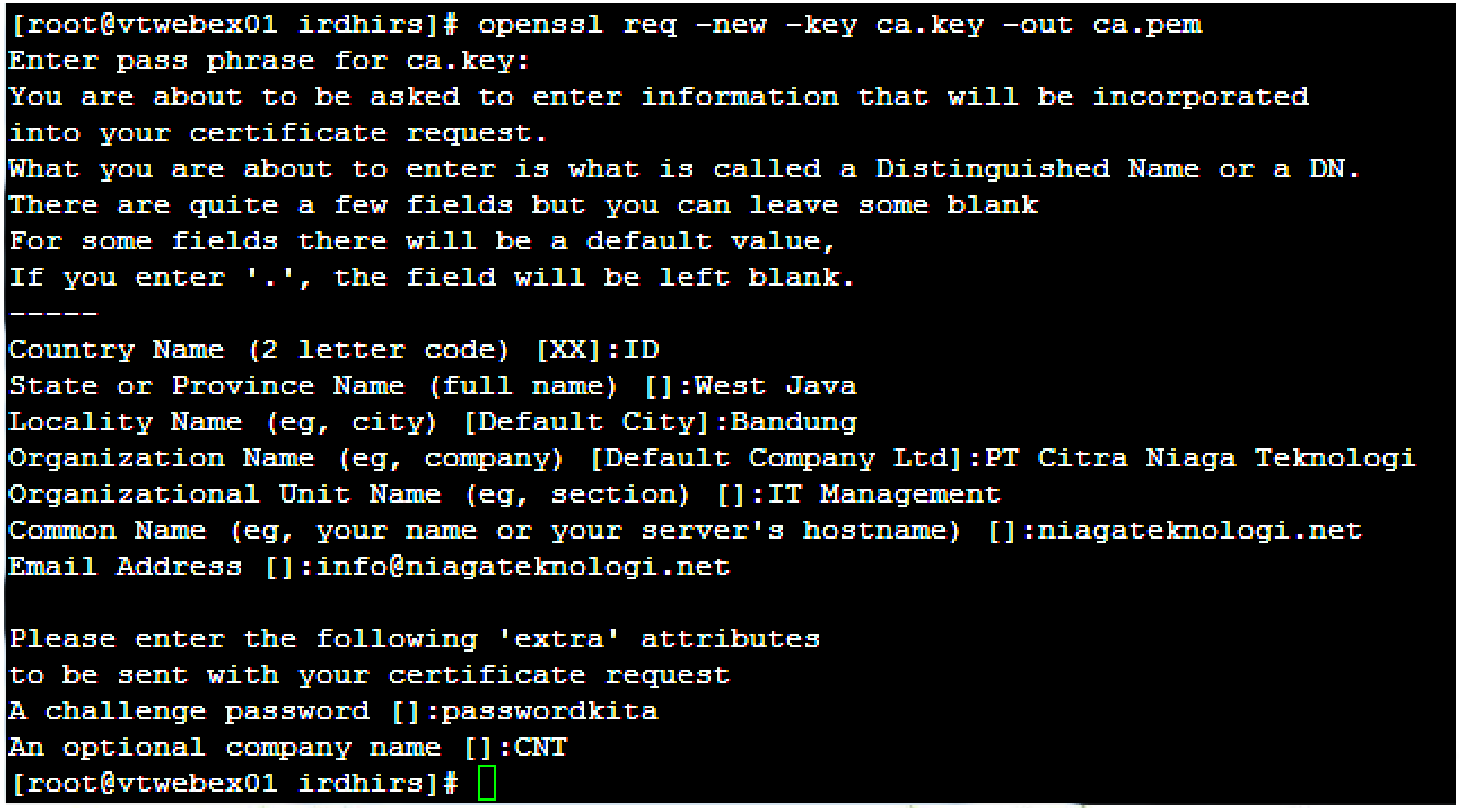
Generate CA Private Key & CA Certificate Pair

- Notes : Several RouterOS version doesn’t support certificate private keys with .key extension, but supports keys with .pem extension. Generate it using this command :

```
openssl req -new -key ca.key -out ca.pem
```

Generate CA Private Key & CA Certificate Pair

- Provide the information needed.

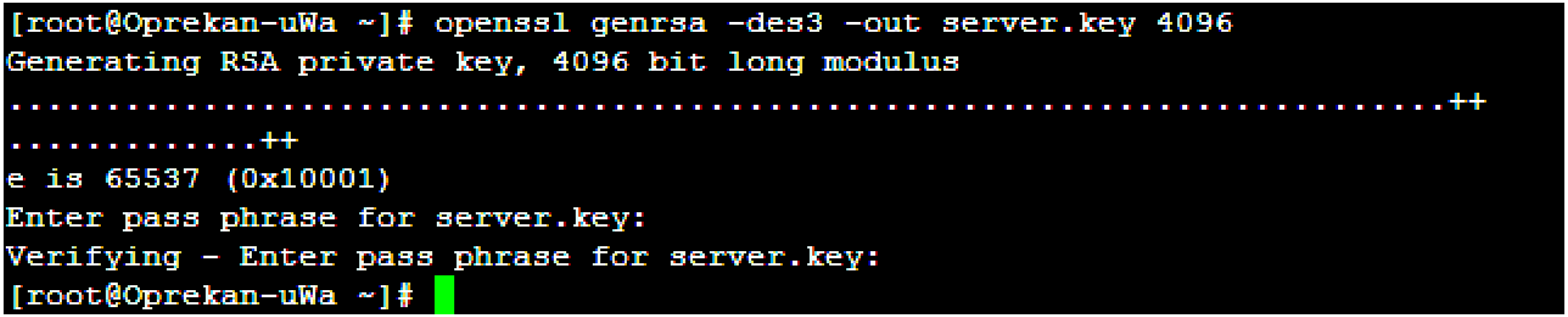


```
[root@vtwebex01 irdhirs]# openssl req -new -key ca.key -out ca.pem
Enter pass phrase for ca.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few 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.
-----
Country Name (2 letter code) [XX]:ID
State or Province Name (full name) []:West Java
Locality Name (eg, city) [Default City]:Bandung
Organization Name (eg, company) [Default Company Ltd]:PT Citra Niaga Teknologi
Organizational Unit Name (eg, section) []:IT Management
Common Name (eg, your name or your server's hostname) []:niagateknologi.net
Email Address []:info@niagateknologi.net

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:passwordkita
An optional company name []:CNT
[root@vtwebex01 irdhirs]#
```

Generate Private Key/Certificate Pair for server side

- 1. Use this command :

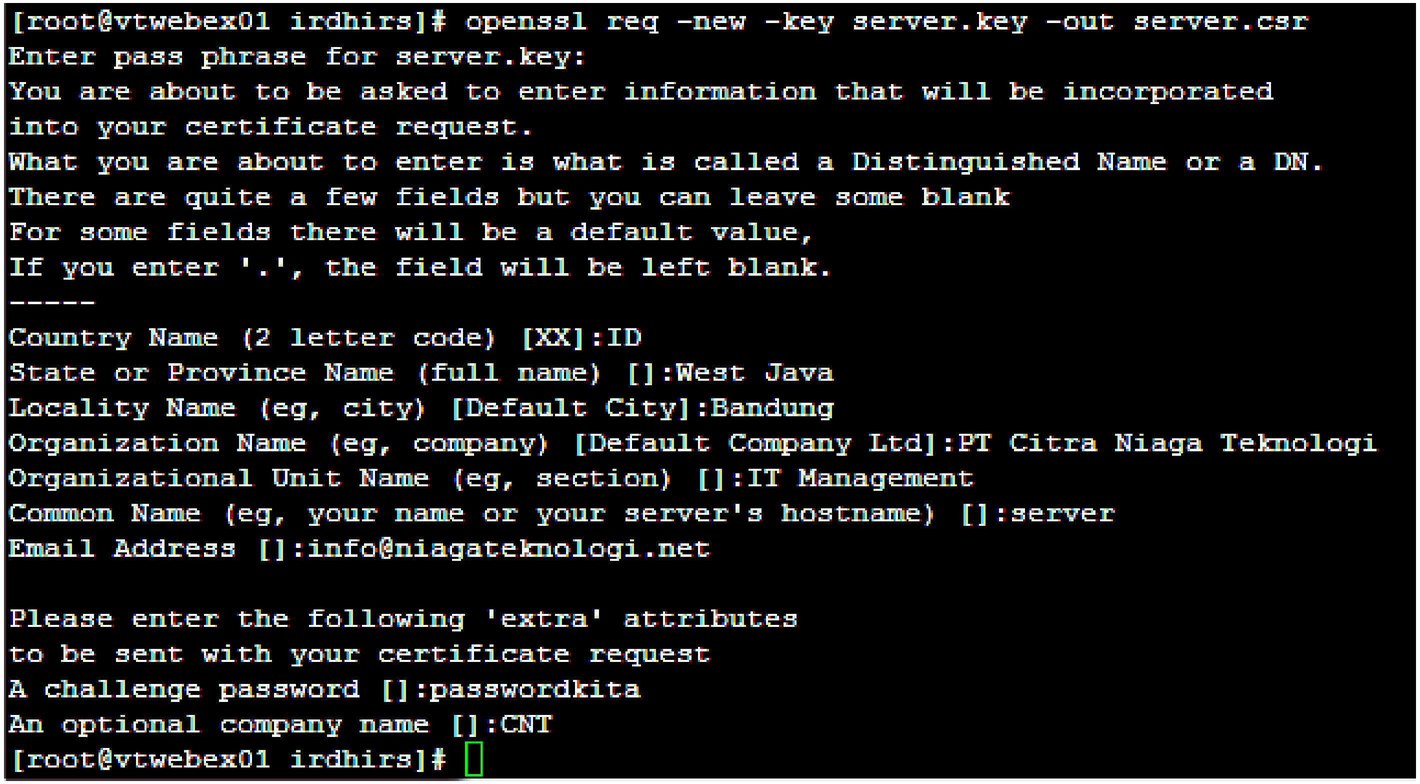
  openssl genrsa -des3 -out server.key 4096



```
[root@Oprekan-uWa ~]# openssl genrsa -des3 -out server.key 4096
Generating RSA private key, 4096 bit long modulus
..........................................................................++
.............++
e is 65537 (0x10001)
Enter pass phrase for server.key:
Verifying - Enter pass phrase for server.key:
[root@Oprekan-uWa ~]#
```

- 2. Provide passphrase for private key in server side, it can be same as the previous passphrase we used, or it can be different. Then press enter.

Generate Private Key/Certificate Pair for server side



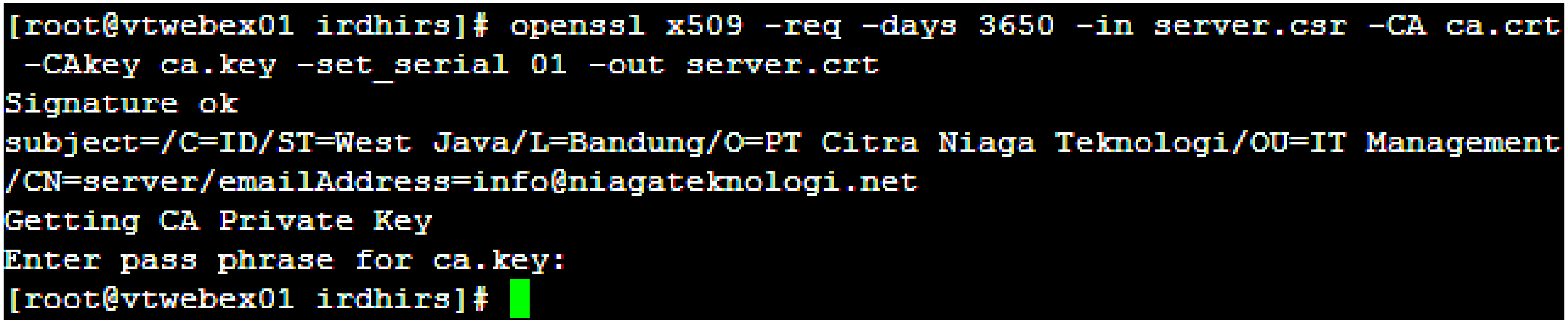
```
[root@vtwebex01 irdhirs]# openssl req -new -key server.key -out server.csr
Enter pass phrase for server.key:
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are 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.
-----
Country Name (2 letter code) [XX]:ID
State or Province Name (full name) []:West Java
Locality Name (eg, city) [Default City]:Bandung
Organization Name (eg, company) [Default Company Ltd]:PT Citra Niaga Teknologi
Organizational Unit Name (eg, section) []:IT Management
Common Name (eg, your name or your server's hostname) []:server
Email Address []:info@niagateknologi.net

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:passwordkita
An optional company name []:CNT
[root@vtwebex01 irdhirs]# 
```

- 3. Provide the information just like before. But for your attention that you must make a difference for **Common Name**. For example, if we provide it with **niagateknologi.net** before, for this time we use different name, such as **server**.

Generate Private Key/Certificate Pair for server side

- 5. Generate the server certificate using :

  openssl x509 -req -days 3650 -in server.csr -CA ca.crt -CAkey ca.key -set_serial 01 -out   server.crt



```
[root@vtwebex01 irdhirs]# openssl x509 -req -days 3650 -in server.csr -CA ca.crt
 -CAkey ca.key -set_serial 01 -out server.crt
Signature ok
subject=/C=ID/ST=West Java/L=Bandung/O=PT Citra Niaga Teknologi/OU=IT Management
/CN=server/emailAddress=info@niagateknologi.net
Getting CA Private Key
Enter pass phrase for ca.key:
[root@vtwebex01 irdhirs]# 
```

- 6. Input the CA key password : **passwordkita**, then press enter. And the server certificates are ready to use.

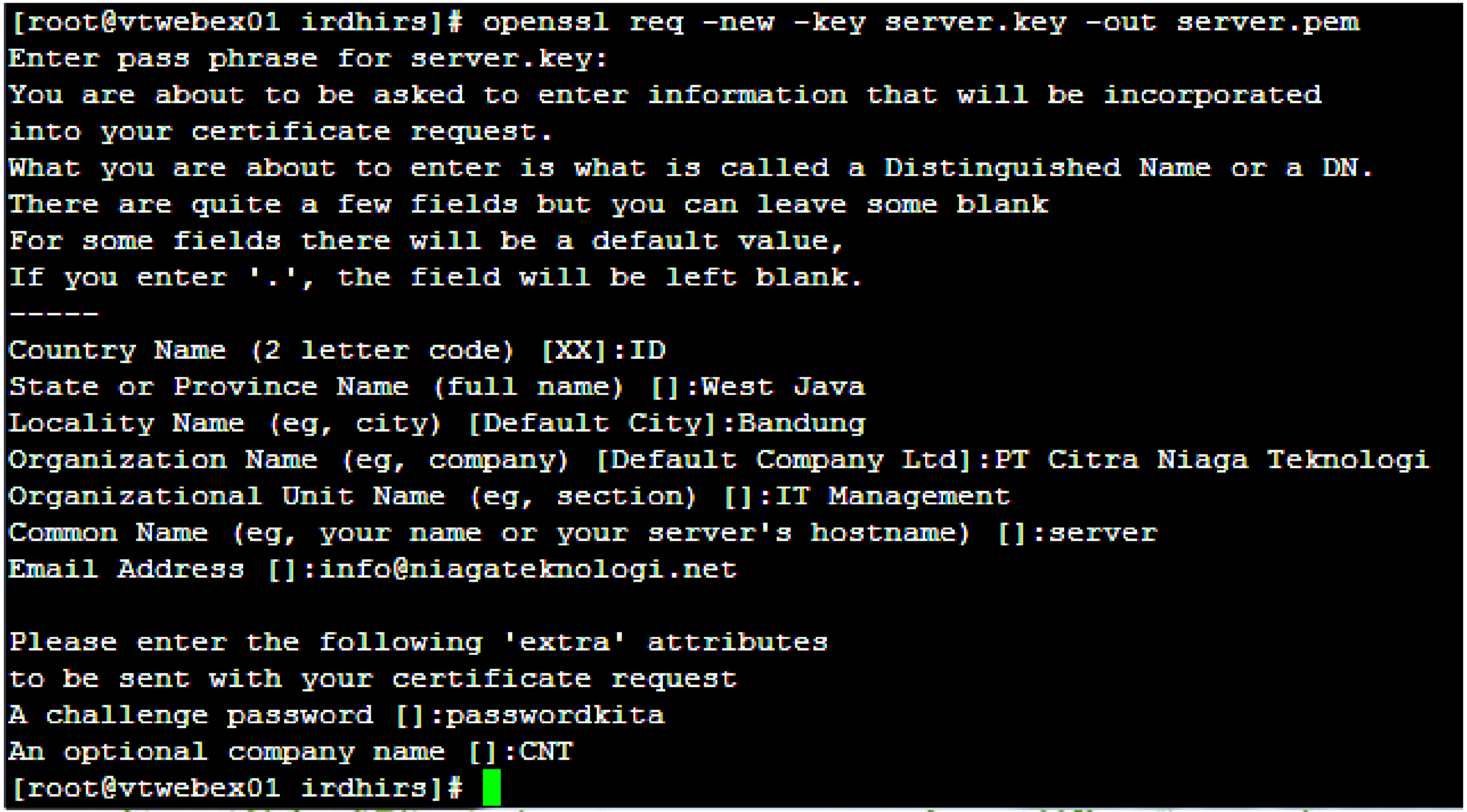
Generate Private Key/Certificate Pair for server side

- Notes : Several RouterOS version doesn’t support certificate private keys with .key extension, but supports keys with .pem extension. Generate it using this command :

```
openssl req -new -key server.key -out server.pem
```

Generate Private Key/Certificate Pair for server side

- Provide the information just like before.



```
[root@vtwebex01 irdhirs]# openssl req -new -key server.key -out server.pem
Enter pass phrase for server.key:
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.
-----
Country Name (2 letter code) [XX]:ID
State or Province Name (full name) []:West Java
Locality Name (eg, city) [Default City]:Bandung
Organization Name (eg, company) [Default Company Ltd]:PT Citra Niaga Teknologi
Organizational Unit Name (eg, section) []:IT Management
Common Name (eg, your name or your server's hostname) []:server
Email Address []:info@niagateknologi.net

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:passwordkita
An optional company name []:CNT
[root@vtwebex01 irdhirs]#
```

Generate Private Key/Certificate Pair for client side

- To generate certificates for client side, just repeat the steps while we generated the server side certificates, but it needs a difference in data and information we provide later, especially for **Common** Name.
- The commands :
  - **openssl genrsa -des3 -out client.key 4096** (*for generating client key*)
  - **openssl x509 -req -days 3650 -in client.csr -CA ca.crt -CAkey ca.key -set_serial 01 -out client.crt** (*for generating Client Certificate*).
  - **openssl req -new -key client.key -out client.pem** (*for generate key with .pem extension*)

Certificate Function Testing

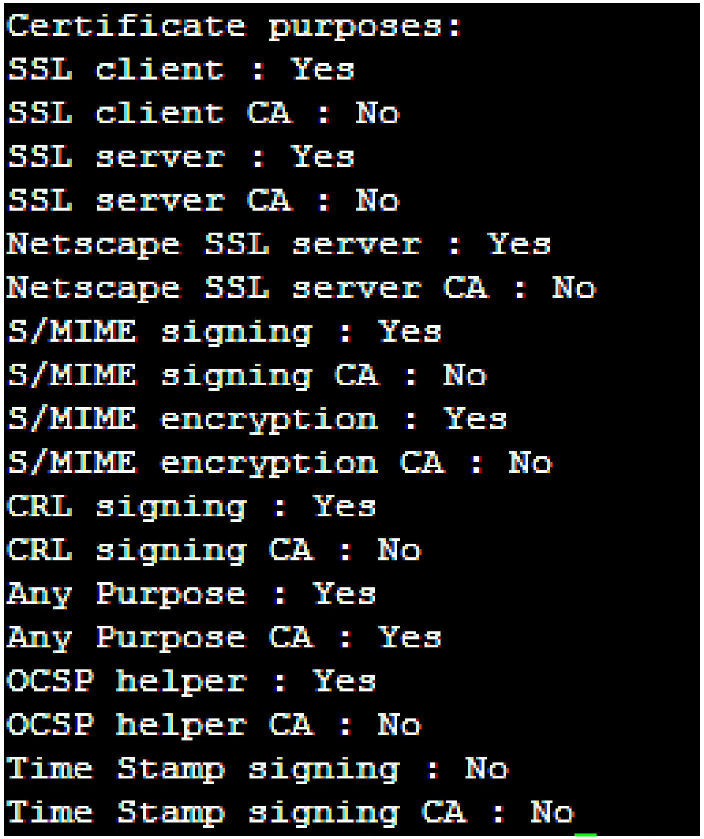
- To test if the certificates we generated before are working or not, use these commands :

  -
    ```
    openssl x509 -noout -text -in server.crt --purpose
    ```

  -
    ```
    openssl x509 -noout -text -in client.crt --purpose
    ```

Certificate Function Testing

- Make sure that there is no **Error** or **Warning** in **Certificate purposes**. It's OK if there is a **No** expression for the certificate.



```
Certificate purposes:
SSL client : Yes
SSL client CA : No
SSL server : Yes
SSL server CA : No
Netscape SSL server : Yes
Netscape SSL server CA : No
S/MIME signing : Yes
S/MIME signing CA : No
S/MIME encryption : Yes
S/MIME encryption CA : No
CRL signing : Yes
CRL signing CA : No
Any Purpose : Yes
Any Purpose CA : Yes
OCSP helper : Yes
OCSP helper CA : No
Time Stamp signing : No
Time Stamp signing CA : No
```

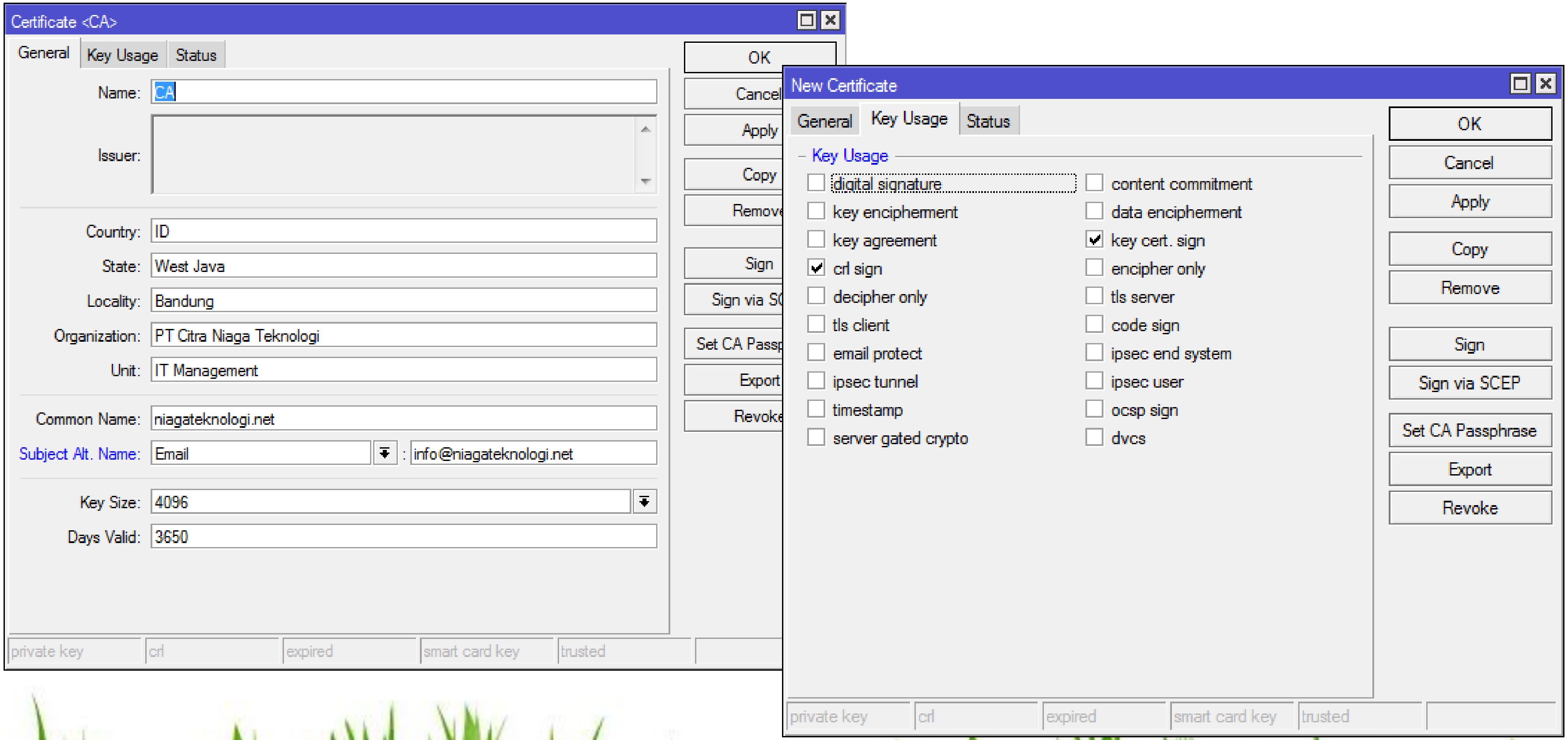
CERTIFICATE CREATION USING MIKROTIK ROUTER

Certificate Creation Using Mikrotik Router

- Using commands or features in MikroTik Terminal or WinBox.
- Certification creation steps
  - Make a template for each certificate (CA, server, client)
  - Certificate signing
  - Set sertifikat to be ‘trusted’
  - Export certificates to each routers (server or client)

Certificate Creation Using Mikrotik Router

- 1. Go to System > Certificates, Then click 'Add' (+) to make CA template.
- 2. Provide the information needed. Then click OK.



Certificate <CA>

General | Key Usage | Status

Name:	CA
Issuer:	
Country:	ID
State:	West Java
Locality:	Bandung
Organization:	PT Citra Niaga Teknologi
Unit:	IT Management
Common Name:	niagateknologi.net
Subject Alt. Name:	Email : info@niagateknologi.net
Key Size:	4096
Days Valid:	3650

New Certificate

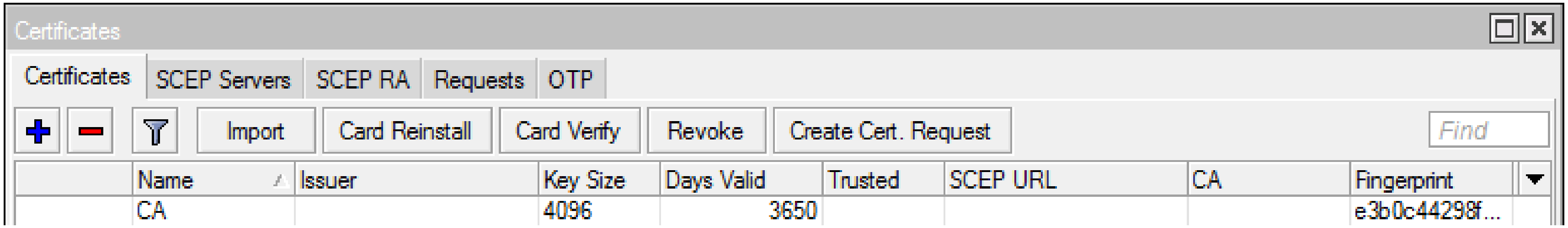
General | Key Usage | Status

Key Usage

☐ digital signature
☐ key encipherment
☐ key agreement
☑ crl sign
☐ decipher only
☐ tls client
☐ email protect
☐ ipsec tunnel
☐ timestamp
☐ server gated crypto
☐ content commitment
☐ data encipherment
☑ key cert. sign
☐ encipher only
☐ tls server
☐ code sign
☐ ipsec end system
☐ ipsec user
☐ ocsp sign
☐ dvcs

Make A Template For Certificates

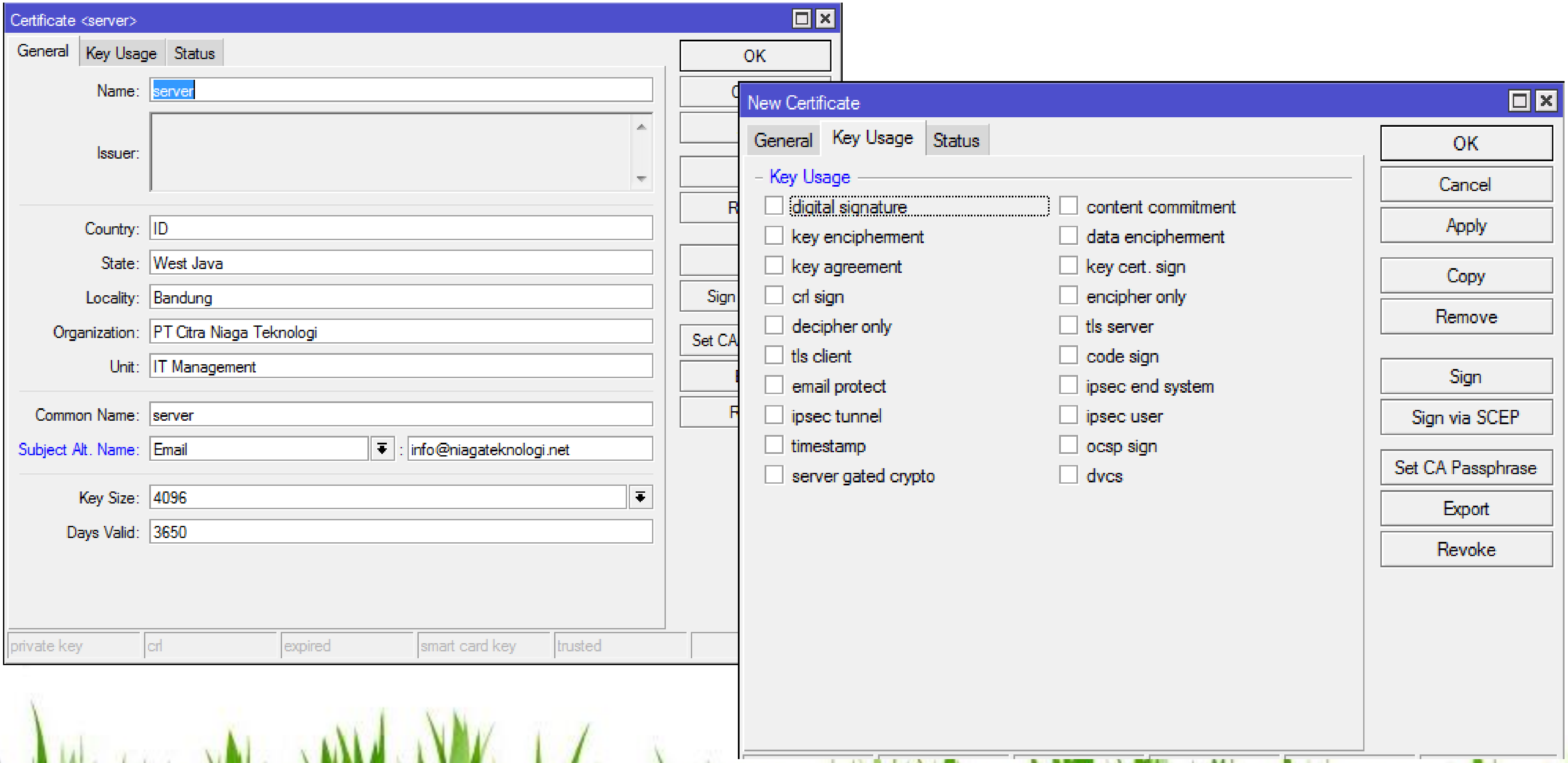
- 3. CA template has been made.



	Name	Issuer	Key Size	Days Valid	Trusted	SCEP URL	CA	Fingerprint
	CA		4096	3650				e3b0c44298f...

Make A Template For Certificates

- 4. Add certificate to make Server and Client templates.



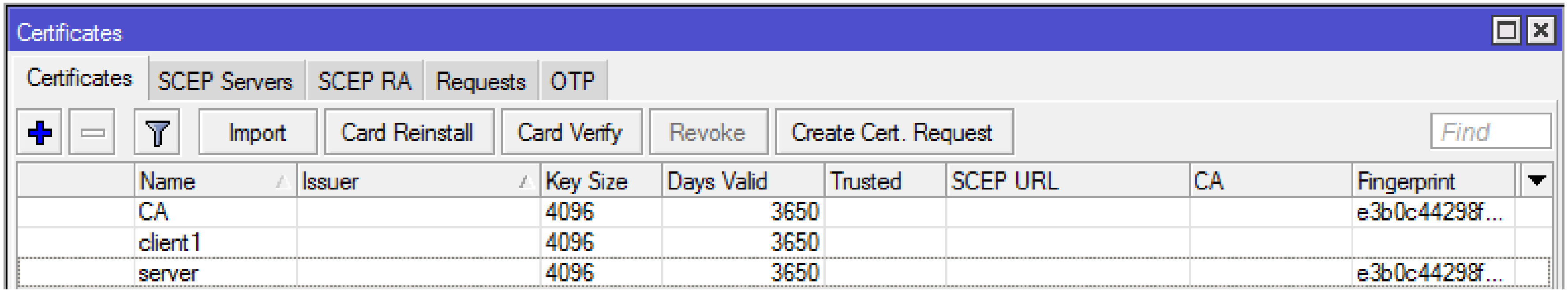
Name:	server
Issuer:	
Country:	ID
State:	West Java
Locality:	Bandung
Organization:	PT Citra Niaga Teknologi
Unit:	IT Management
Common Name:	server
Subject Alt. Name:	Email : info@niagateknologi.net
Key Size:	4096
Days Valid:	3650

Key Usage

- □ digital signature
- □ key encipherment
- □ key agreement
- □ crl sign
- □ decipher only
- □ tls client
- □ email protect
- □ ipsec tunnel
- □ timestamp
- □ server gated crypto
- □ content commitment
- □ data encipherment
- □ key cert. sign
- □ encipher only
- □ tls server
- □ code sign
- □ ipsec end system
- □ ipsec user
- □ ocsp sign
- □ dvcs

Make A Template For Certificates

- 5. Server and Client templates has been made.

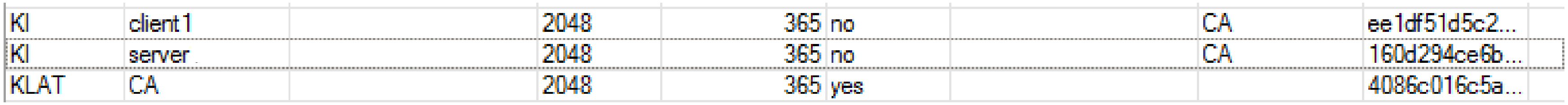


	Name	Issuer	Key Size	Days Valid	Trusted	SCEP URL	CA	Fingerprint
	CA		4096	3650				e3b0c44298f...
	client1		4096	3650				
	server		4096	3650				e3b0c44298f...

Certificate Signing

- Signing process is easier to use terminal because the features is hidden in GUI/WinBox.

```
/certificate
sign CA ca-crl-host=40.50.60.70 name=CA
sign server ca=CA name=server
sign client1 ca=CA name=client1
```

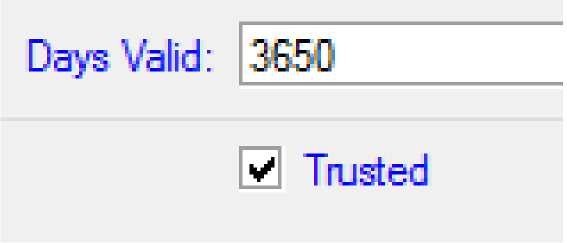


KI	client1		2048	365	no		CA	ee1df51d5c2...
KI	server		2048	365	no		CA	160d294ce6b...
KLAT	CA		2048	365	yes			4086c016c5a...

Set Certificate to be ‘trusted’ & Export Certificates

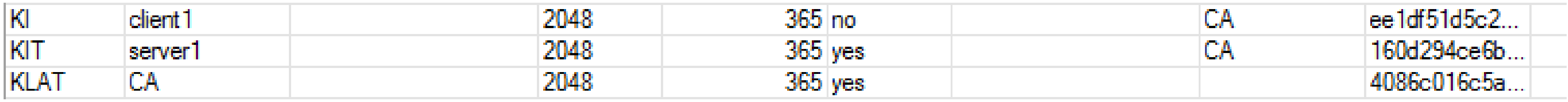
- For certificates which is not ‘trusted’, use this command :

```
/certificate
set server trusted=yes
Set CA trusted=yes
```



Days Valid: 3650

☑ Trusted



KI	client1		2048	365	no		CA	ee1df51d5c2...
KIT	server1		2048	365	yes		CA	160d294ce6b...
KLAT	CA		2048	365	yes			4086c016c5a...

- Export certificate to each router (server and client)

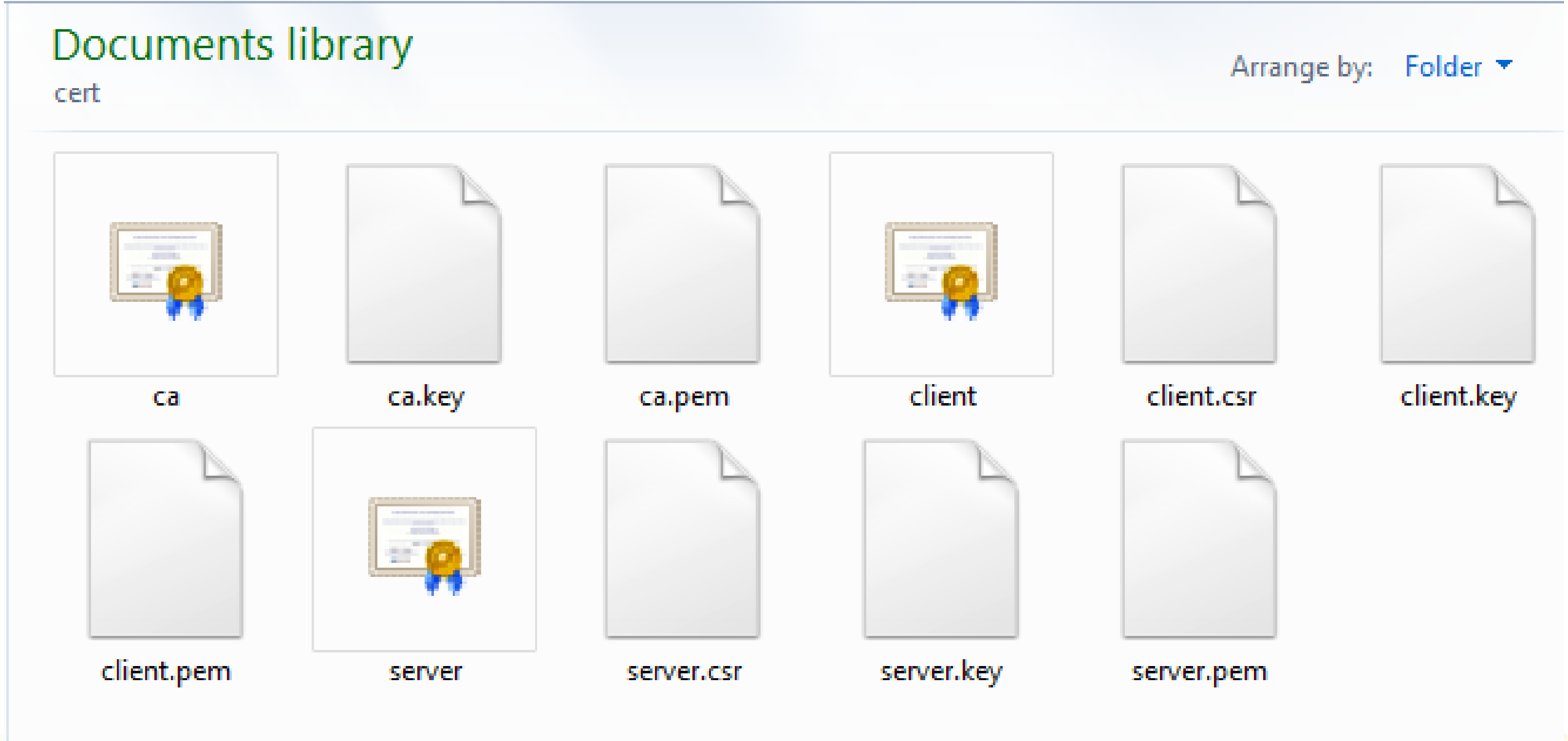
IMPORT CERTIFICATES INTO MIKROTIK ROUTER

Import Certificates

- Can use 2 methods
  - FTP
  - Drag & Drop into Winbox ← we use this ☺
- Files uploaded to server
  - CA (ca.crt & ca.pem/ca.key)
  - Server Certificate (server.crt & server.pem/server.key)
- Files uploaded to client
  - CA (ca.crt & ca.pem/ca.key)
  - Client Certificate (client.crt & client.pem/client.key)

Import Certificates – via Winbox

- First, download the certificate files from linux server (if you used linux server to generate it).

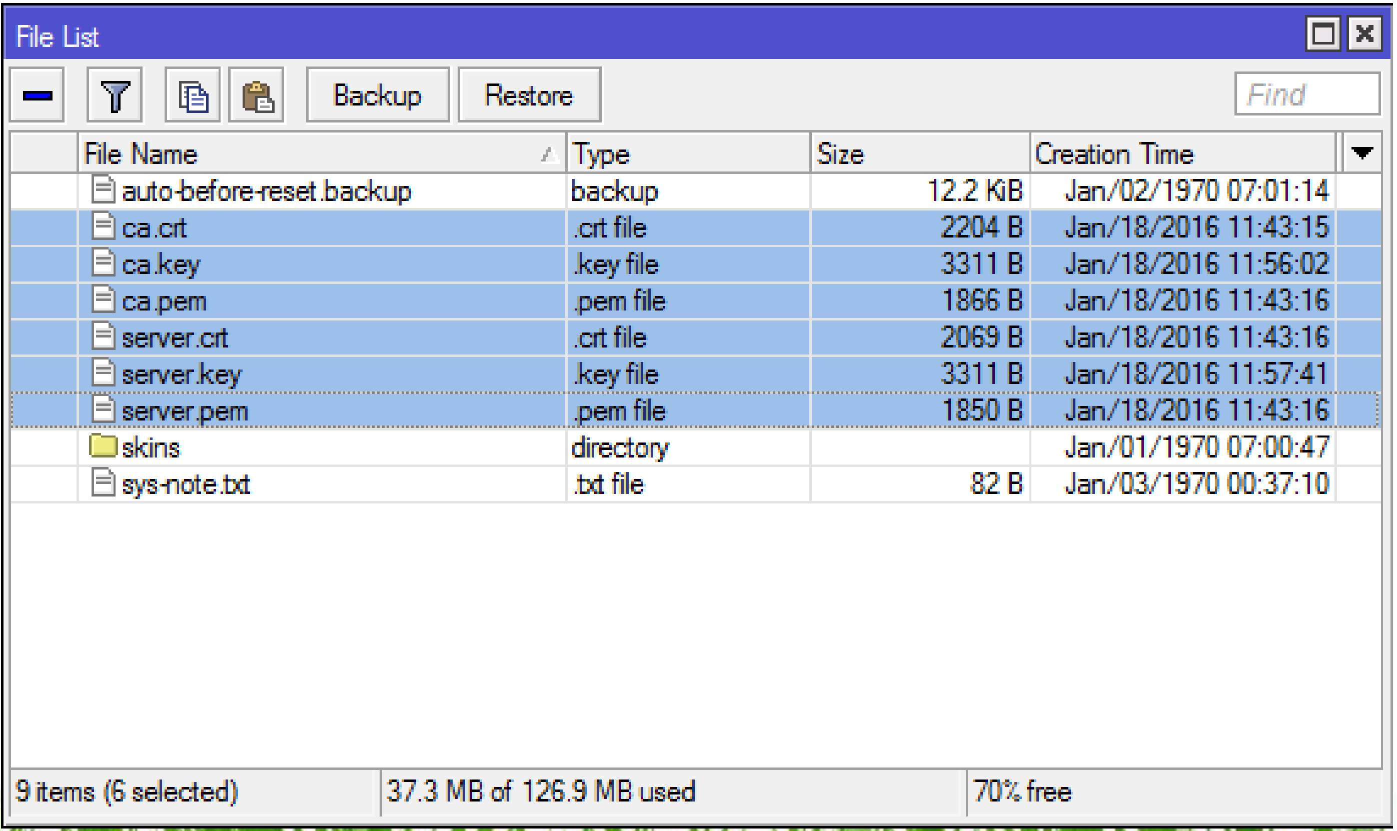


Documents library

cert

Import Certificates – via Winbox

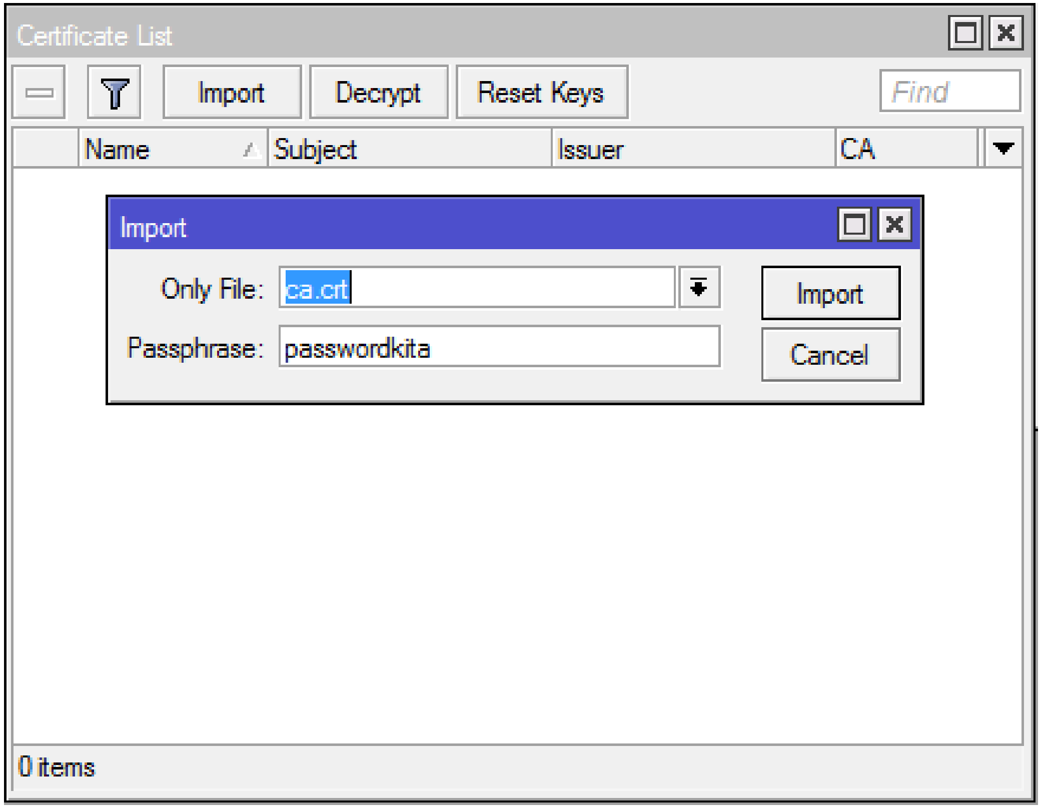
- Drag and Drop CA and Server Certificate files to server side router



File Name	Type	Size	Creation Time
auto-before-reset.backup	backup	12.2 KiB	Jan/02/1970 07:01:14
ca.crt	.crt file	2204 B	Jan/18/2016 11:43:15
ca.key	.key file	3311 B	Jan/18/2016 11:56:02
ca.pem	.pem file	1866 B	Jan/18/2016 11:43:16
server.crt	.crt file	2069 B	Jan/18/2016 11:43:16
server.key	.key file	3311 B	Jan/18/2016 11:57:41
server.pem	.pem file	1850 B	Jan/18/2016 11:43:16
skins	directory		Jan/01/1970 07:00:47
sys-note.txt	.txt file	82 B	Jan/03/1970 00:37:10

9 items (6 selected) | 37.3 MB of 126.9 MB used | 70% free

Import Certificates – via Winbox

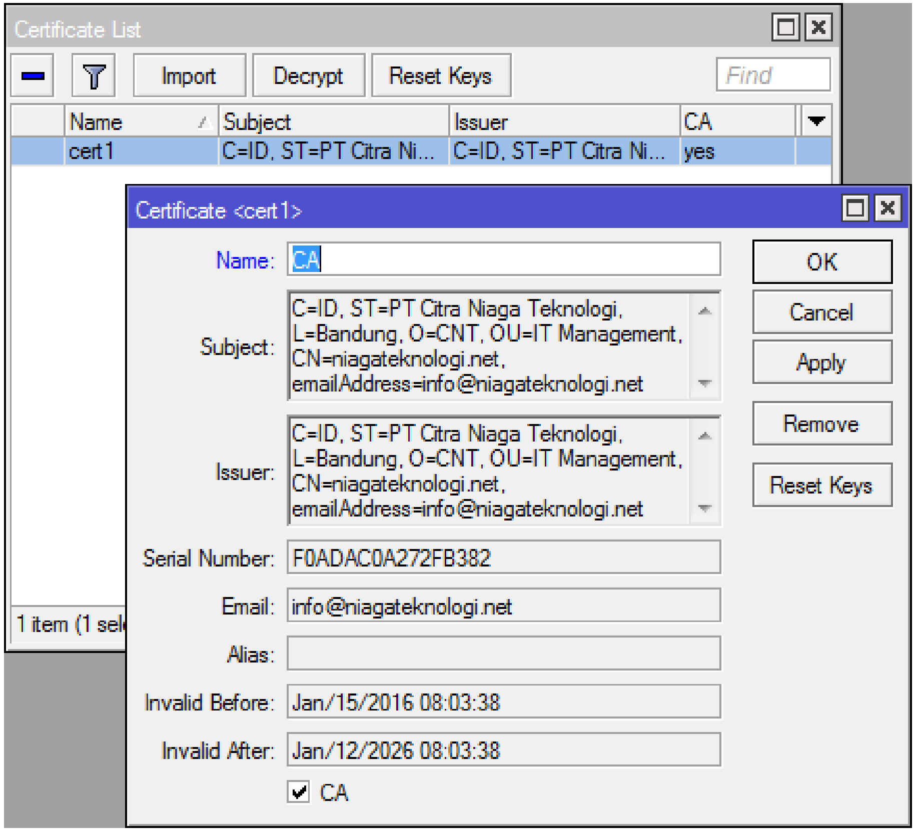


Only File: ca.crt

Passphrase: passwordkita

- To import uploaded certificates, click on **System > Certificates**, then click **Import**. Choose ca.crt file first, then input the passphrase we provided it before, then click **Import**.

Import Certificates – via Winbox



Name	Subject	Issuer	CA
cert1	C=ID, ST=PT Citra Ni...	C=ID, ST=PT Citra Ni...	yes

1 item (1 sel

Certificate <cert1>

Name: CA

Subject: C=ID, ST=PT Citra Niaga Teknologi, L=Bandung, O=CNT, OU=IT Management, CN=niagateknologi.net, emailAddress=info@niagateknologi.net

Issuer: C=ID, ST=PT Citra Niaga Teknologi, L=Bandung, O=CNT, OU=IT Management, CN=niagateknologi.net, emailAddress=info@niagateknologi.net

Serial Number: F0ADAC0A272FB382

Email: info@niagateknologi.net

Alias:

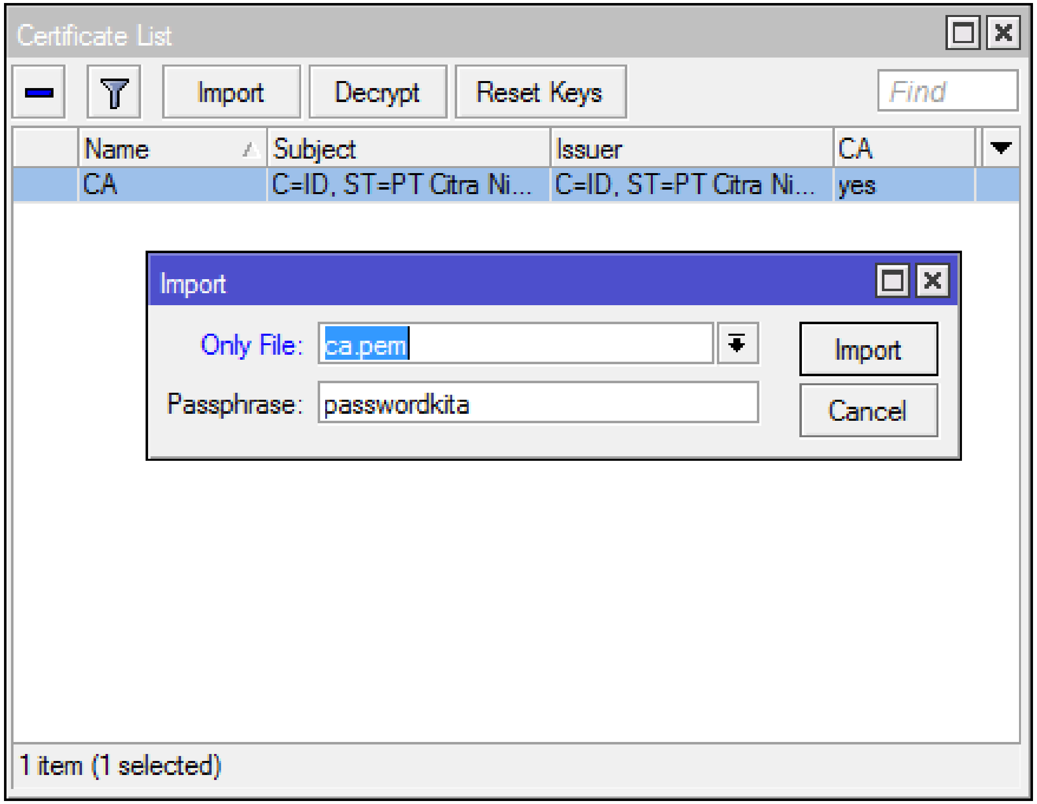
Invalid Before: Jan/15/2016 08:03:38

Invalid After: Jan/12/2026 08:03:38

☑ CA

- If succeeded, **cert1** will show. Double click on it to see the certificate detail and to rename it, such as **CA**. Then click OK.

Import Certificates – via Winbox



Name	Subject	Issuer	CA
CA	C=ID, ST=PT Citra Ni...	C=ID, ST=PT Citra Ni...	yes

Import

Only File: ca.pem

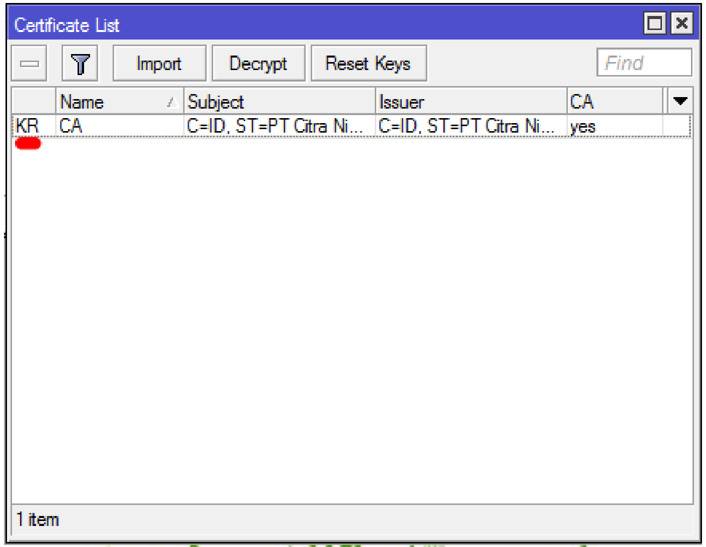
Passphrase: passwordkita

1 item (1 selected)

- Kemudian import file ca.pem/ca.key (tergantung versi RouterOS-nya support untuk ekstensi apa) dan inputkan passphrase-nya. Kemudian klik **Import**.

Import Certificates – via Winbox

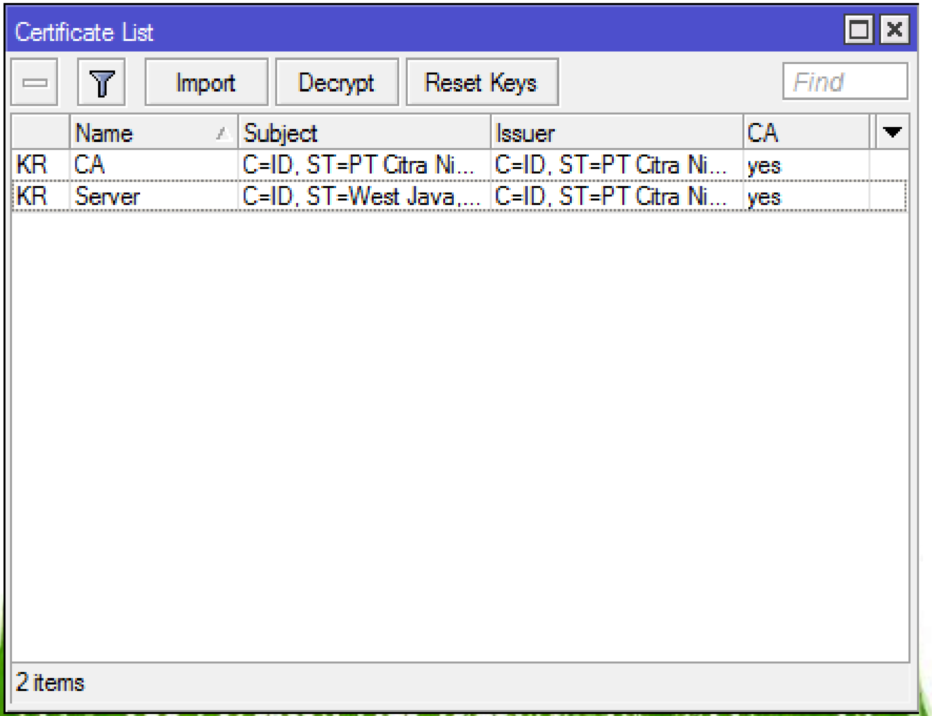
- If the import process succeed, the KR/KT information (depends on RouterOS version) will show.
  - K = private key / decrypted private key
  - R = rsa
  - T = trusted



	Name	Subject	Issuer	CA
KR	CA	C=ID, ST=PT Citra Ni...	C=ID, ST=PT Citra Ni...	yes

Import Certificates – via Winbox

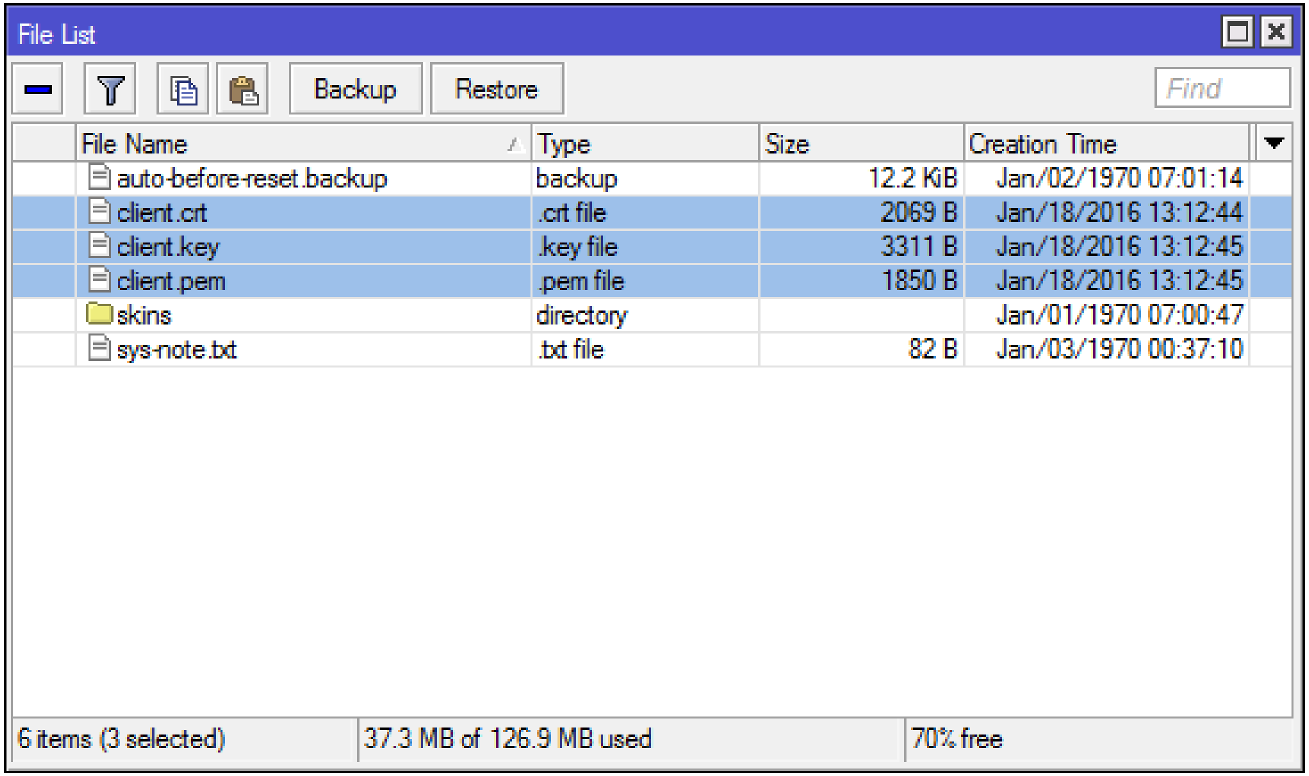
- For server certificates, import it according to CA private key importing steps. Make sure the information is KR/KT.



	Name	Subject	Issuer	CA
KR	CA	C=ID, ST=PT Citra Ni...	C=ID, ST=PT Citra Ni...	yes
KR	Server	C=ID, ST=West Java,...	C=ID, ST=PT Citra Ni...	yes

2 items

Import Certificates – via Winbox



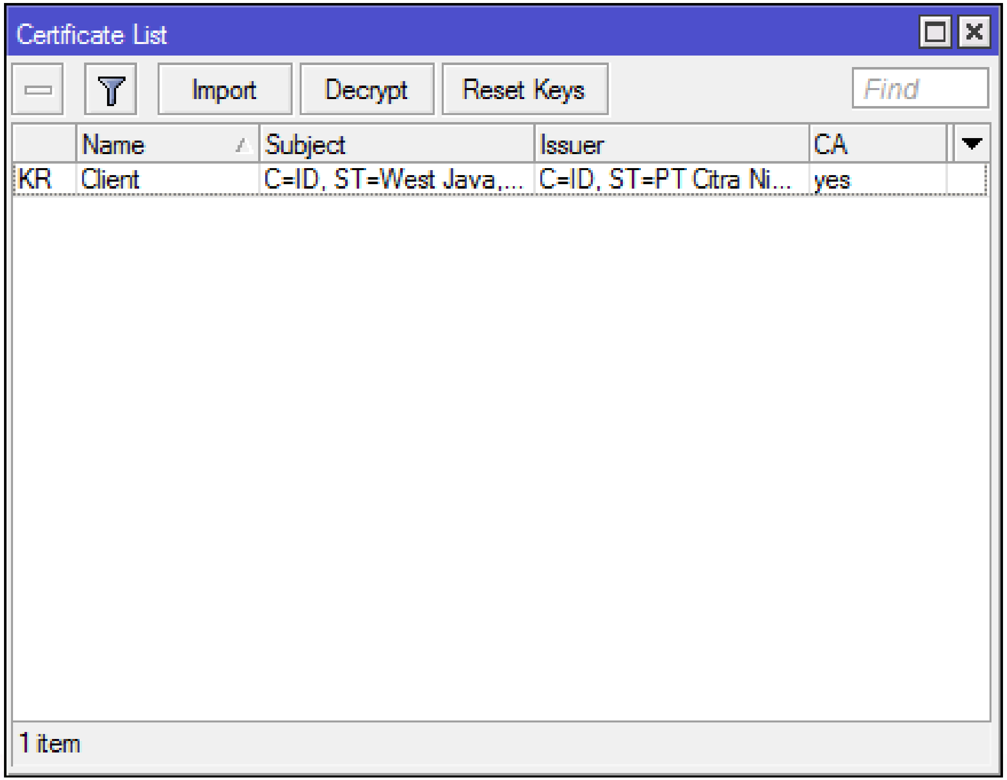
File Name	Type	Size	Creation Time
auto-before-reset.backup	backup	12.2 KiB	Jan/02/1970 07:01:14
client.crt	.crt file	2069 B	Jan/18/2016 13:12:44
client.key	.key file	3311 B	Jan/18/2016 13:12:45
client.pem	.pem file	1850 B	Jan/18/2016 13:12:45
skins	directory		Jan/01/1970 07:00:47
sys-note.txt	.txt file	82 B	Jan/03/1970 00:37:10

6 items (3 selected) | 37.3 MB of 126.9 MB used | 70% free

- For client side, use the same steps just like to import server side certificates. But just upload the client certificates, no need to upload the CA certificates.

Import Certificates – via Winbox

- Make sure the information is KR/KT.



Certificate List

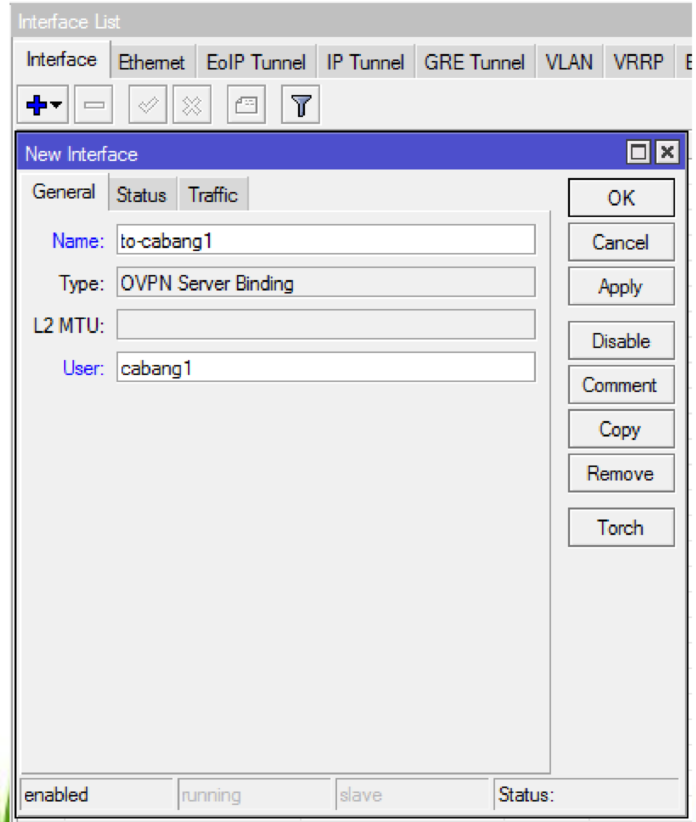
	Name	Subject	Issuer	CA
KR	Client	C=ID, ST=West Java,...	C=ID, ST=PT Citra Ni...	yes

1 item

SET OPENVPN SERVER AT HEAD OFFICE SIDE

Set OpenVPN Server at Head Office Side

- Create accounts for OpenVPN connection in **PPP > Secret** menu. The detail is up to you.
- Create OpenVPN server (OVPN Server/OVPN Server Binding) at server side router



Name: to-cabang1

Type: OVPN Server Binding

L2 MTU:

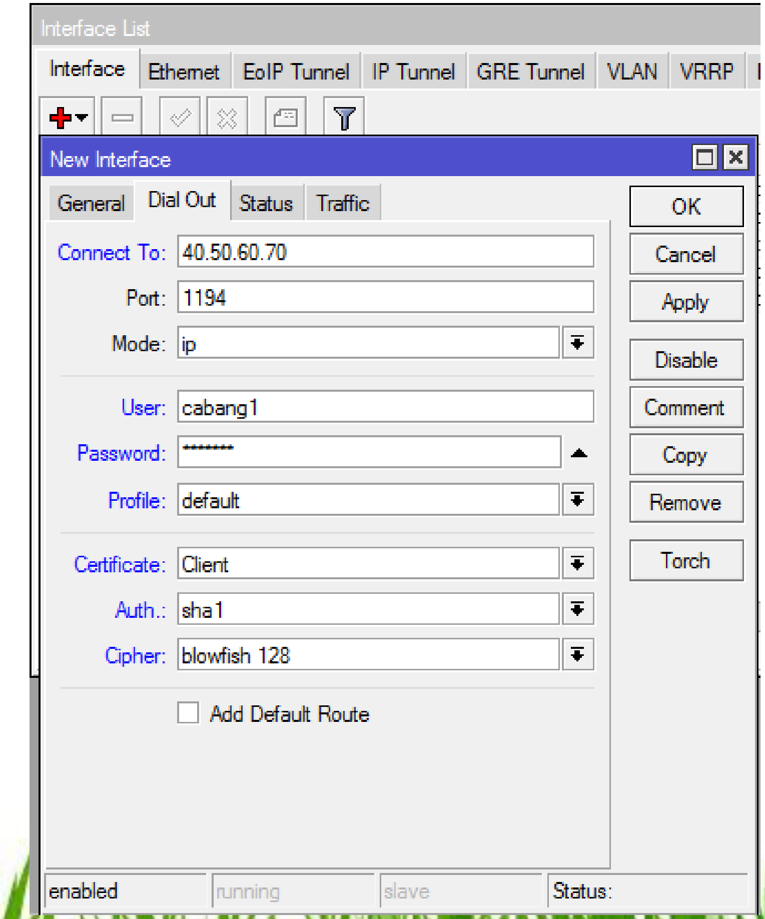
User: cabang1

enabled | running | slave | Status:

SET OPENVPN CLIENT AT BRANCH OFFICE SIDE

Set OpenVPN Client at Branch Office Side

- Set OpenVPN Client at Branch Office Side.



Connect To: 40.50.60.70

Port: 1194

Mode: ip

User: cabang1

Password: ********

Profile: default

Certificate: Client

Auth.: sha1

Cipher: blowfish 128

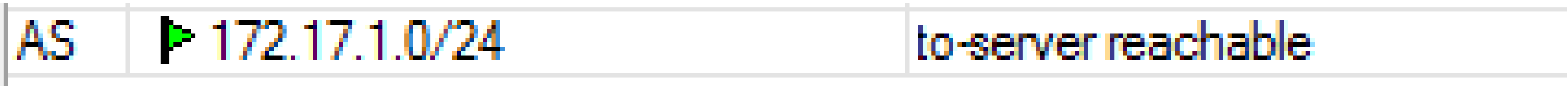
□ Add Default Route

enabled | running | slave | Status:

ADD NETWORK ROUTING

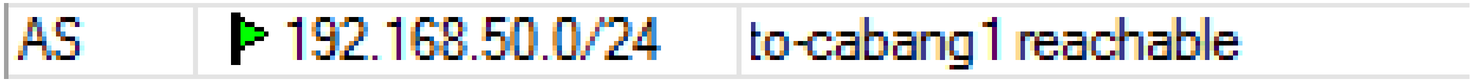
Add Network Routing

- In each side, add network routing to each destination.
- Client :



AS	172.17.1.0/24	to-server reachable

- Server :

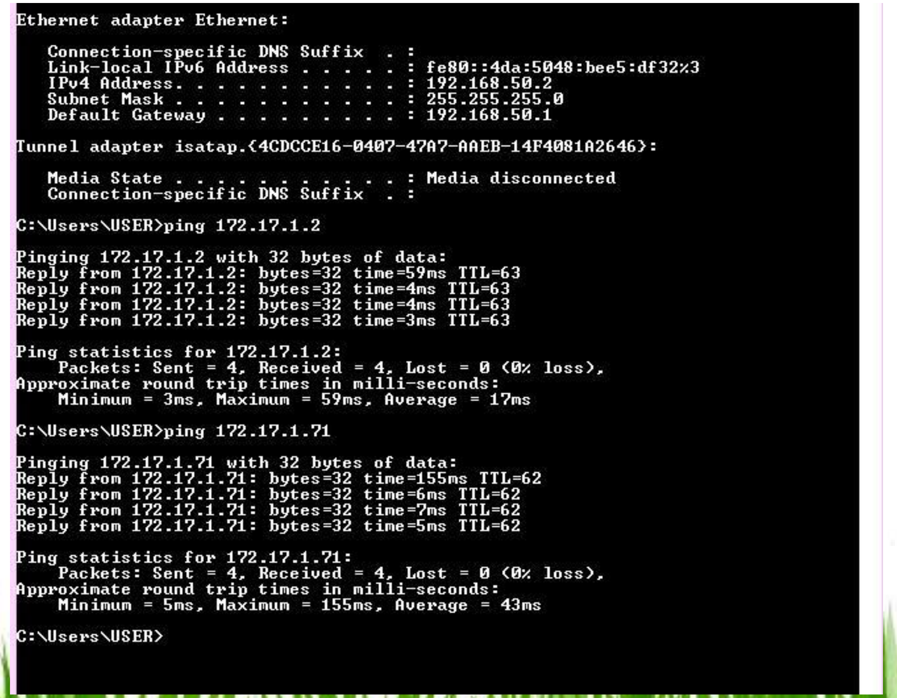


AS	192.168.50.0/24	to-cabang1 reachable

CONFIGURATION TESTING

Configuration Testing

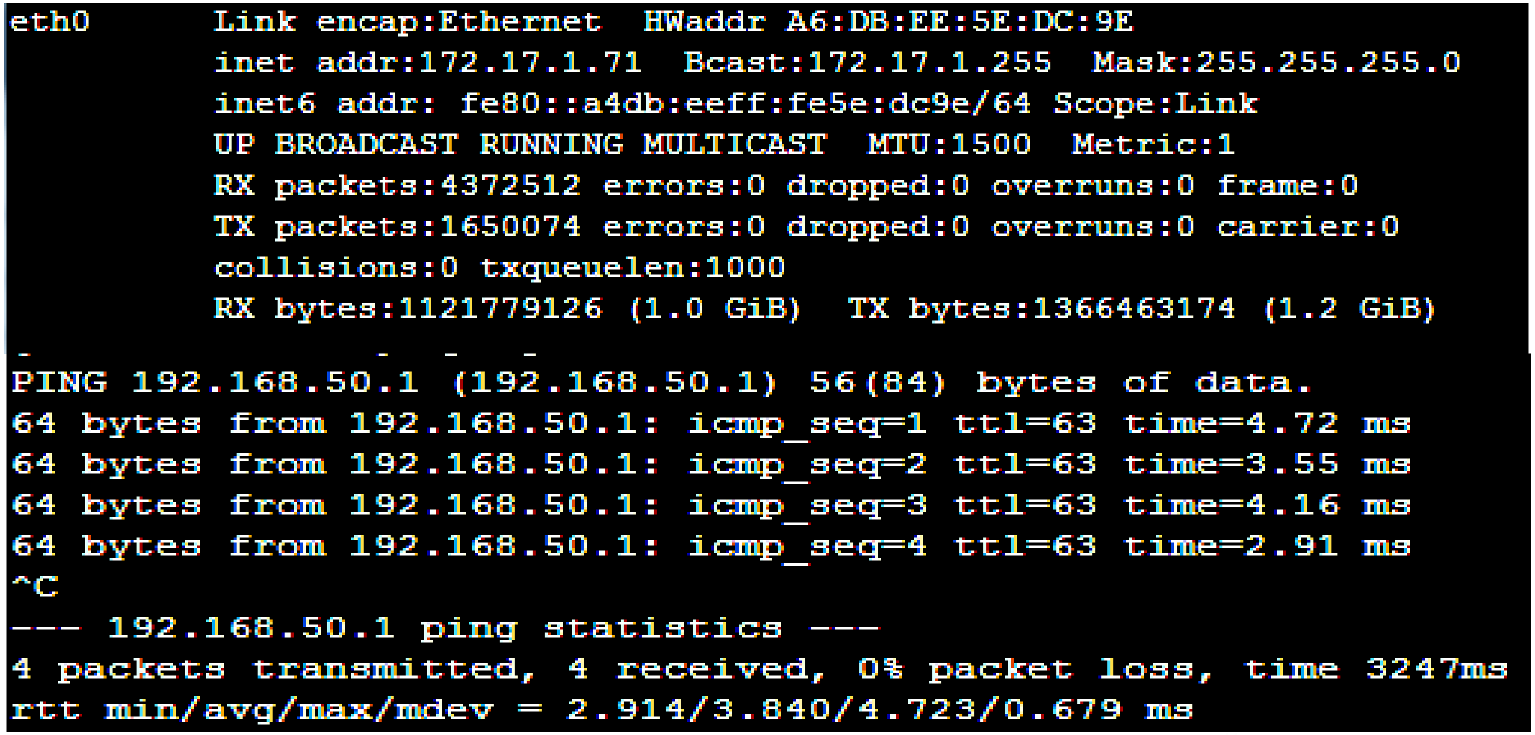
- Do ping tests from each side to other.
- From LAN to server farm :



```
Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::4da:5048:bee5:df32%3
   IPv4 Address. . . . . . . . . . . : 192.168.50.2
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.50.1

Tunnel adapter isatap.{4CDCCE16-0407-47A7-AAEB-14F4081A2646}:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

C:\Users\USER>ping 172.17.1.2

Pinging 172.17.1.2 with 32 bytes of data:
Reply from 172.17.1.2: bytes=32 time=59ms TTL=63
Reply from 172.17.1.2: bytes=32 time=4ms TTL=63
Reply from 172.17.1.2: bytes=32 time=4ms TTL=63
Reply from 172.17.1.2: bytes=32 time=3ms TTL=63

Ping statistics for 172.17.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 3ms, Maximum = 59ms, Average = 17ms

C:\Users\USER>ping 172.17.1.71

Pinging 172.17.1.71 with 32 bytes of data:
Reply from 172.17.1.71: bytes=32 time=155ms TTL=62
Reply from 172.17.1.71: bytes=32 time=6ms TTL=62
Reply from 172.17.1.71: bytes=32 time=7ms TTL=62
Reply from 172.17.1.71: bytes=32 time=5ms TTL=62

Ping statistics for 172.17.1.71:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 5ms, Maximum = 155ms, Average = 43ms

C:\Users\USER>
```

Configuration Testing

- From server to LAN :

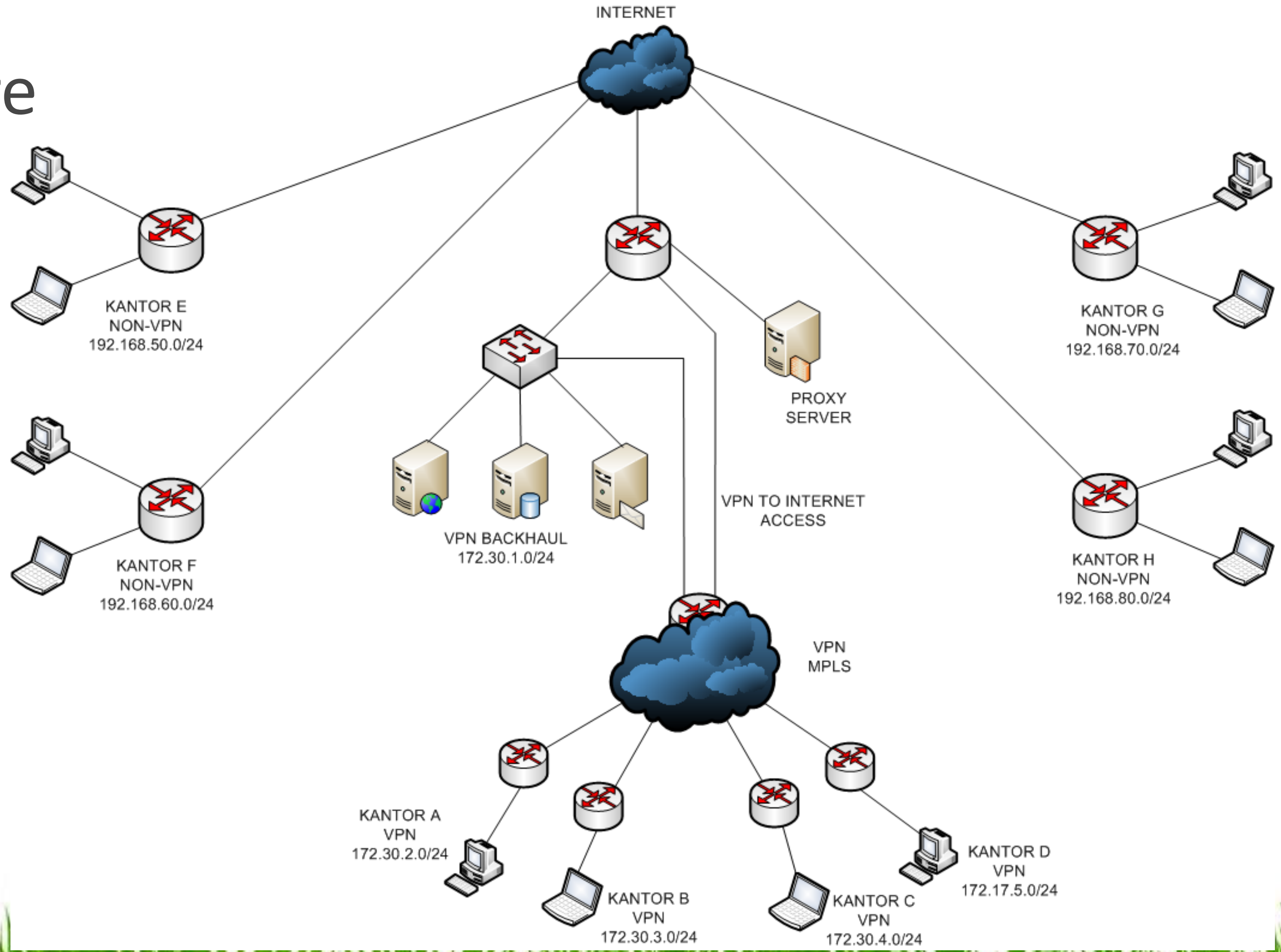


```
eth0      Link encap:Ethernet  HWaddr A6:DB:EE:5E:DC:9E
          inet addr:172.17.1.71  Bcast:172.17.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a4db:eeff:fe5e:dc9e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4372512 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1650074 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1121779126 (1.0 GiB)  TX bytes:1366463174 (1.2 GiB)
PING 192.168.50.1 (192.168.50.1) 56(84) bytes of data.
64 bytes from 192.168.50.1: icmp_seq=1 ttl=63 time=4.72 ms
64 bytes from 192.168.50.1: icmp_seq=2 ttl=63 time=3.55 ms
64 bytes from 192.168.50.1: icmp_seq=3 ttl=63 time=4.16 ms
64 bytes from 192.168.50.1: icmp_seq=4 ttl=63 time=2.91 ms
^C
--- 192.168.50.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3247ms
rtt min/avg/max/mdev = 2.914/3.840/4.723/0.679 ms
```

Easy, isn’t it? ☺

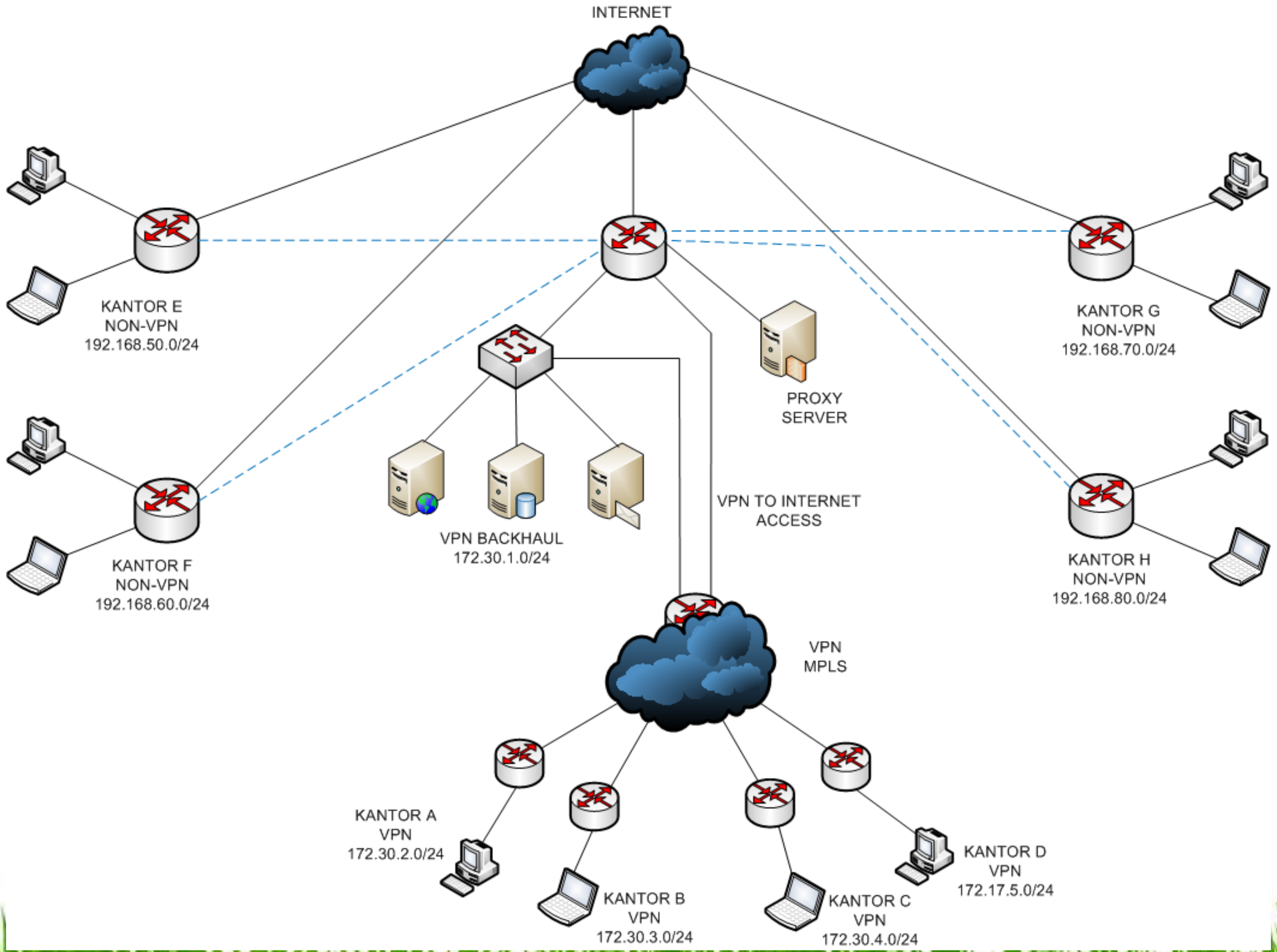
Overview

- Before



Overview

- After



Terima kasih.

====================

Thank You.