Low cost secure VPN MikroTik SSTP over OpenIXP (Indonesian Internet)

About Me

Faisal Reza, ST. (si_faisal)

- Co-founder Asta Informatics
- using MikroTik since early 2008
  applied in Internet Café, ISP, Enterprise Network, Multifinance, Hotel & many more.
- MTCNA, MTCTCE
- member of routing the world with Forum MikroTik Indonesia
  www.forummikrotik.com

Specialities :

Network Solution & Design, Virtualization

Asta Informatics

Established 2011

- Solution Provider with **Green IT** principle
- System Integrator

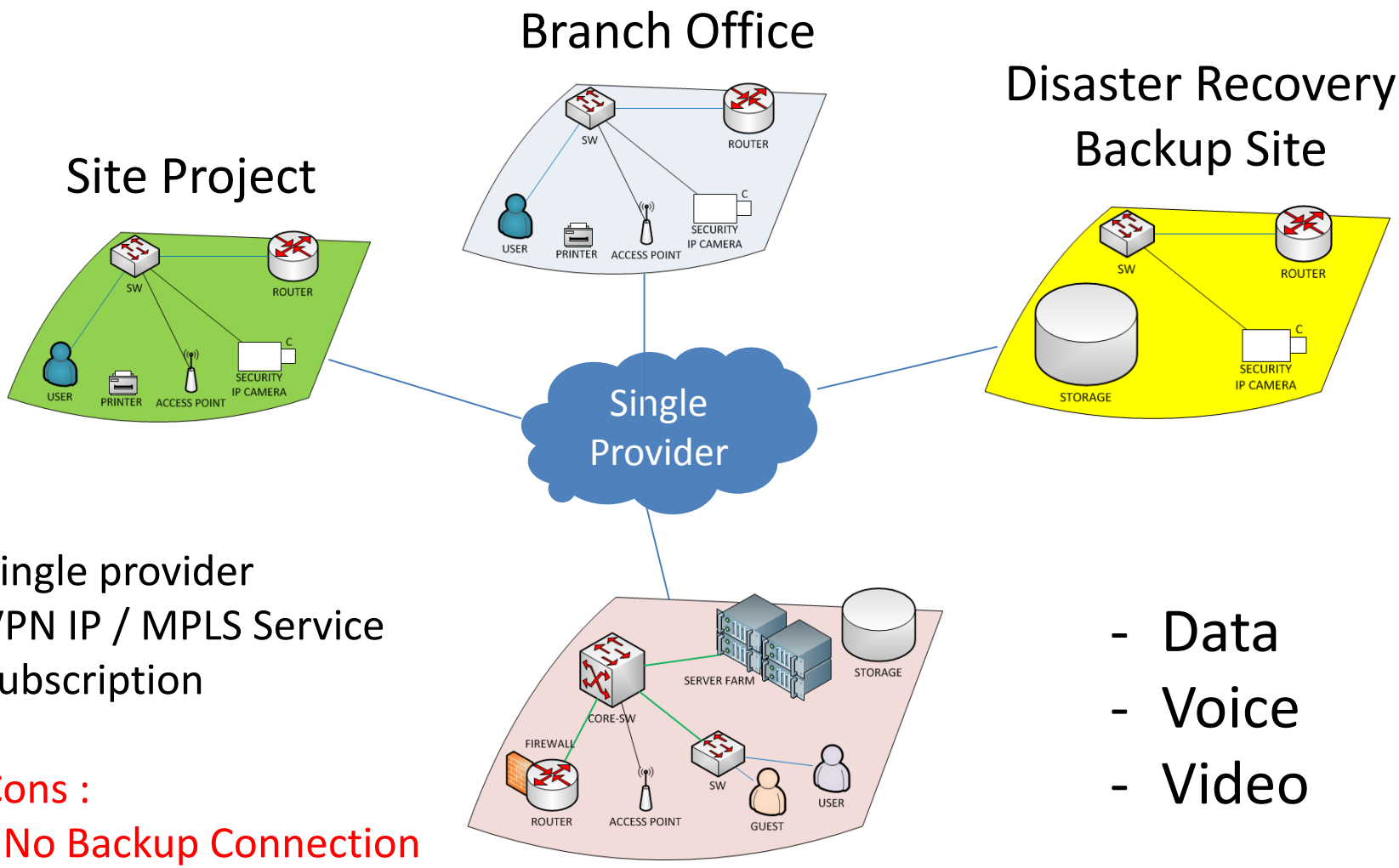
  for Server, Networking, Security and Private Cloud
- IP Surveillance System
- Broadband services
- **Free** Consultation available

More info : www.astainformatics.com

AstaInformatics **@AstaInformatics**

Typical Configuration on Enterprise Network

Single Provider

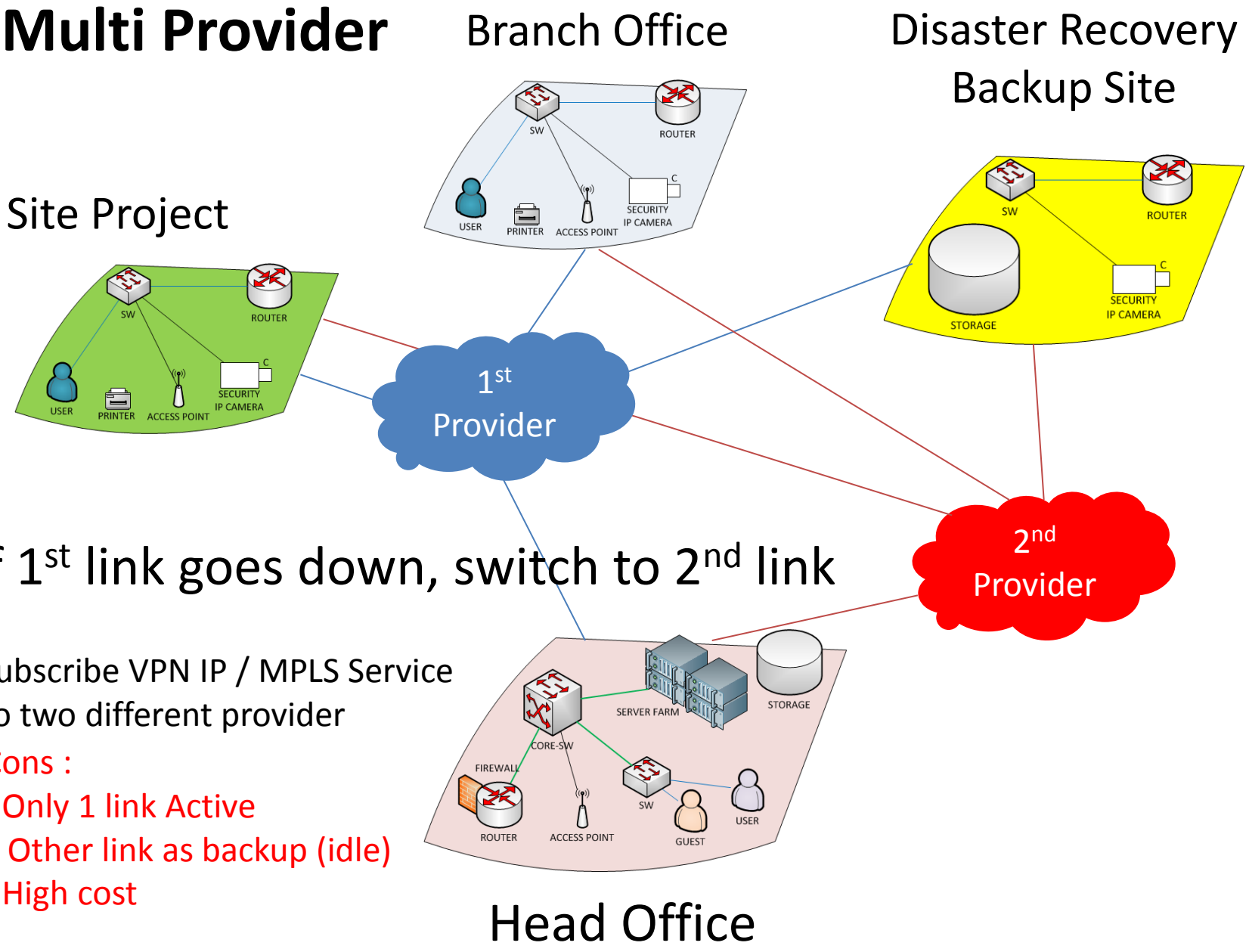


Single provider
VPN IP / MPLS Service subscription

Cons :
- No Backup Connection

- Data
- Voice
- Video

Multi Provider



If 1st link goes down, switch to 2nd link

Subscribe VPN IP / MPLS Service
To two different provider

Cons :
- Only 1 link Active
  Other link as backup (idle)
- High cost

Lets try different solution

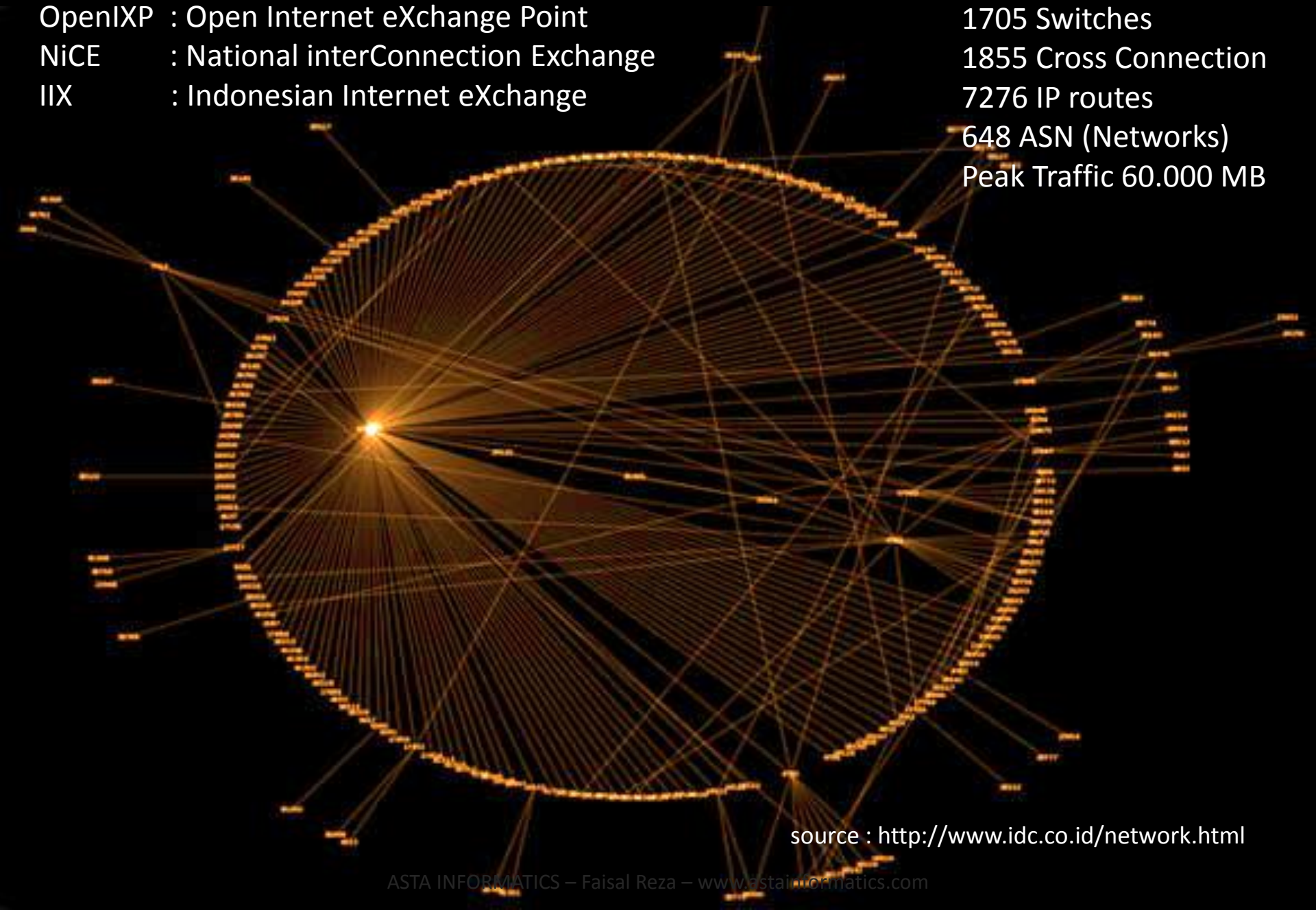
Build your Own VPN on public Infrastructure using

MikroTik

INDONESIAN INTERNET

OpenIXP : Open Internet eXchange Point
NiCE : National interConnection Exchange
IIX : Indonesian Internet eXchange

1705 Switches
1855 Cross Connection
7276 IP routes
648 ASN (Networks)
Peak Traffic 60.000 MB

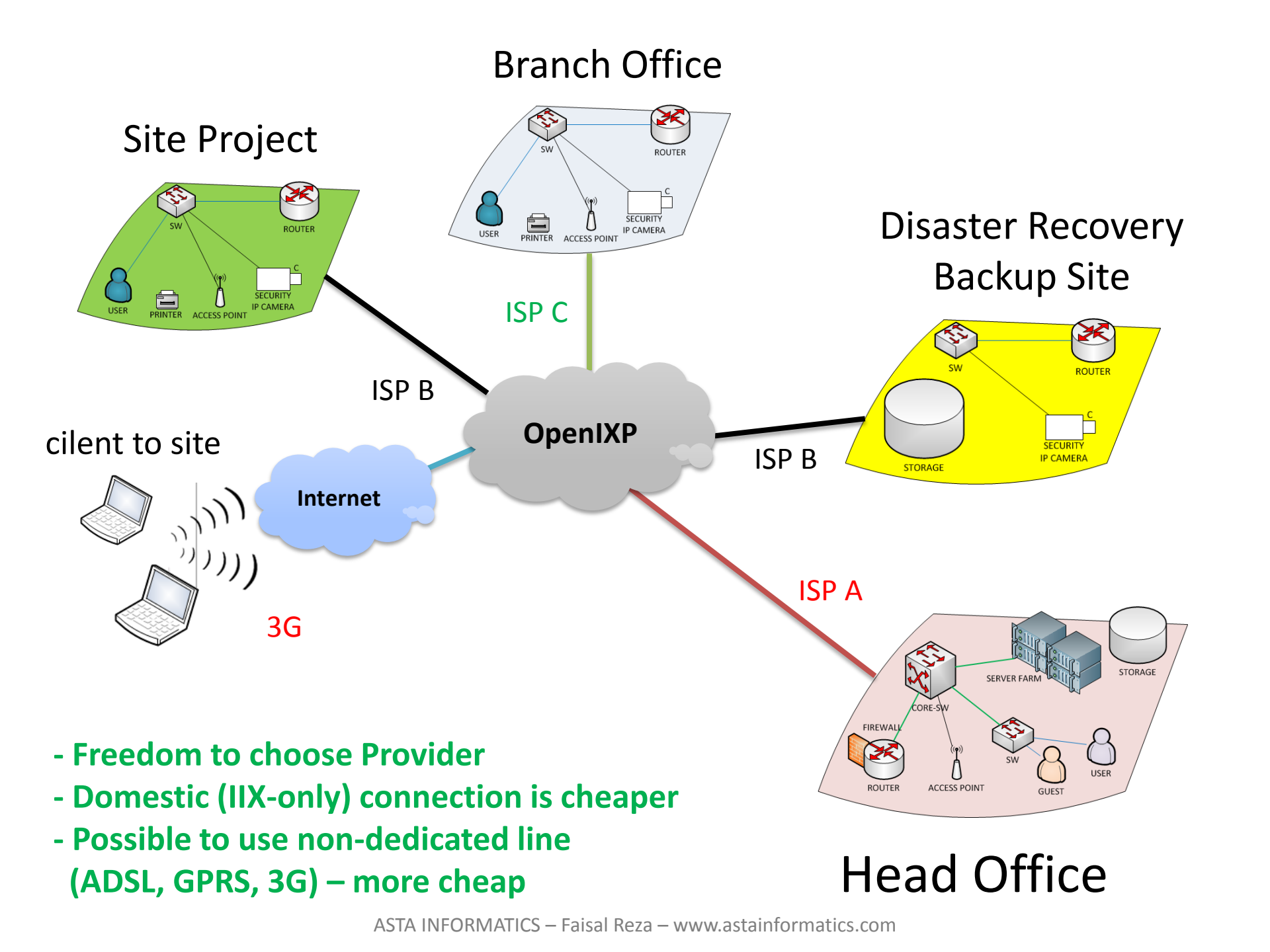


source : http://www.idc.co.id/network.html

OpenIXP / NiCE

Open **I**nternet e**X**change **P**oint
National **i**nter**C**onnection **E**xchange

- All Provider - Connected In one place
- we can buy IIX only connection
  (cheaper than dedicated VPN + get access to indonesian domestic site)
- **Greater** Bandwitdh, **Cheaper** Price
- Low Latency inter provider, average < 10 ms
- Multi Access option (fiber optic, wireless, microwave, ADSL, 3G, Wimax)
- Freedom of choice



- **Freedom to choose Provider**
- **Domestic (IIX-only) connection is cheaper**
- **Possible to use non-dedicated line (ADSL, GPRS, 3G) – more cheap**

MikroTik

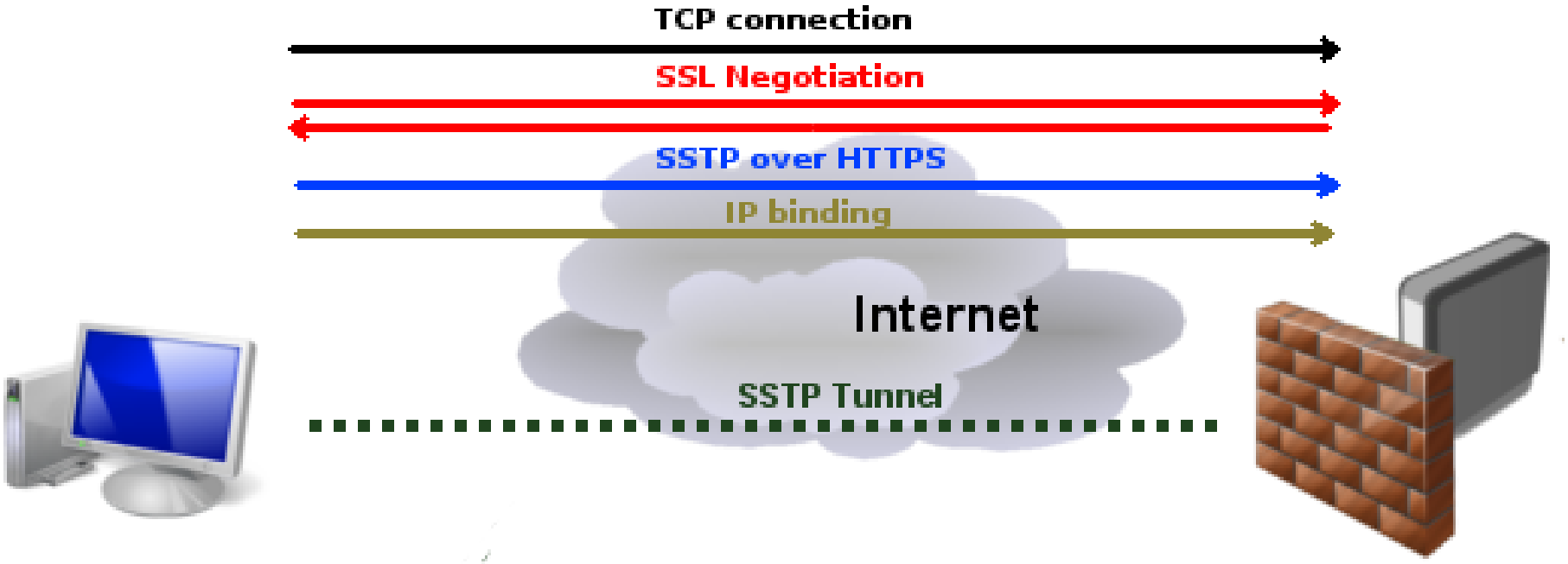
Solution

- MikroTik as VPN Server
- Because we rely on public infrastructure, *security* is more considered.
- Using Secure Socket Tunnel Protocol (SSTP) with self-signed certificate (SSL)
- Site-to-site & Client-to-site are supported
- Multi-provider bandwitdh aggregation supported with eoip-tunnel or MPLS/VPLS over SSTP

SSTP

Secure Socket Tunneling Protocol (SSTP) is the way to transport PPP tunnel over SSL 3.0 channel. The use of SSL over TCP port 443 allows SSTP to pass through virtually all firewalls and proxy servers.

http://wiki.mikrotik.com/wiki/Manual:Interface/SSTP



1. TCP connection is established from client to server (by default on port 443)
2. SSL validates server certificate. If certificate is valid connection is established otherwise connection is torn down.
3. The client sends SSTP control packets within the HTTPS session which establishes the SSTP state machine on both sides.
5. PPP negotiation over SSTP. Client authenticates to the server and binds IP addresses to SSTP interface
5.SSTP tunnel is now established and packet encapsulation can begin.

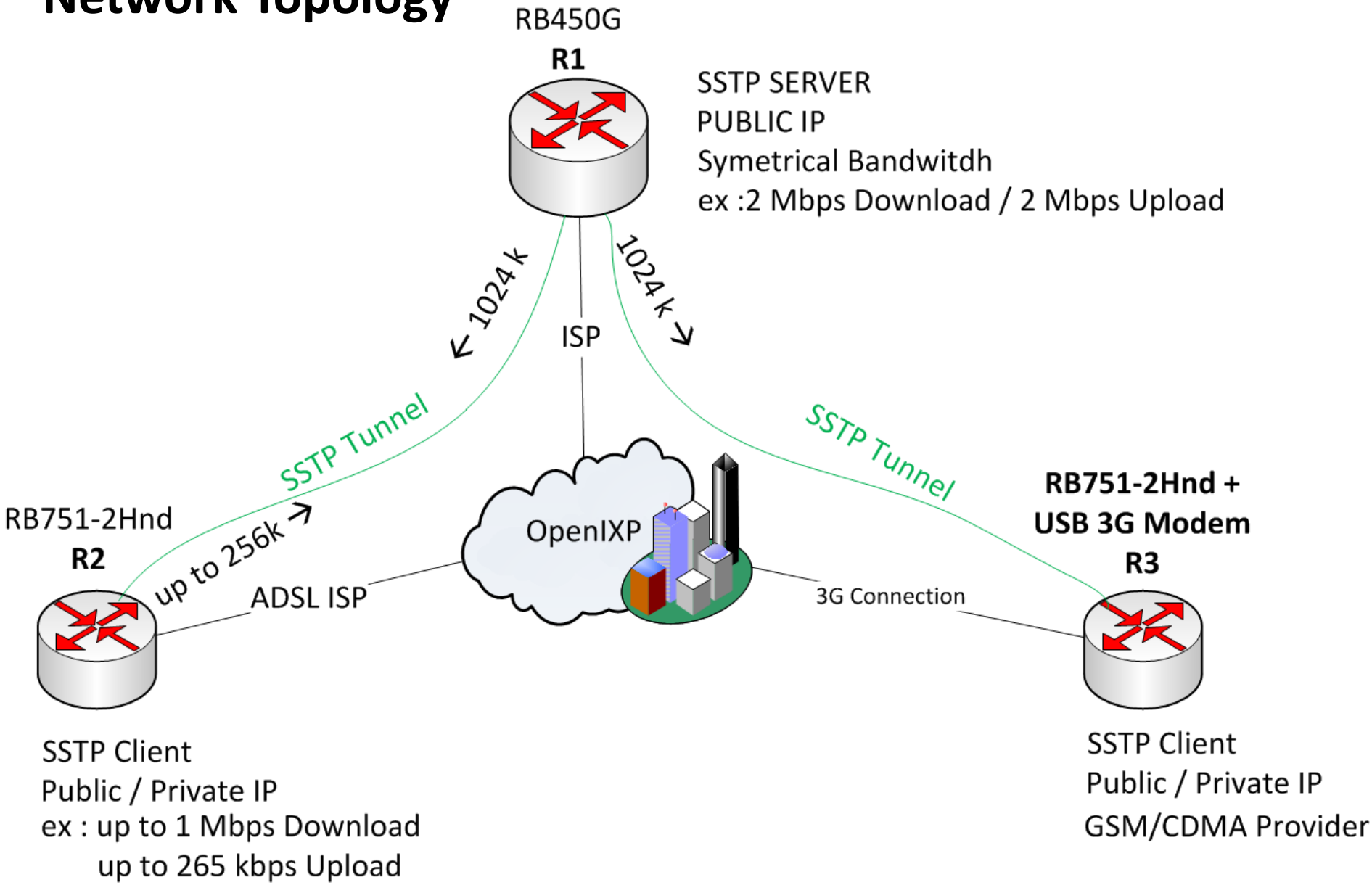
http://wiki.mikrotik.com/wiki/Manual:Interface/SSTP

So, to connect to SSTP Server we need following requirement :

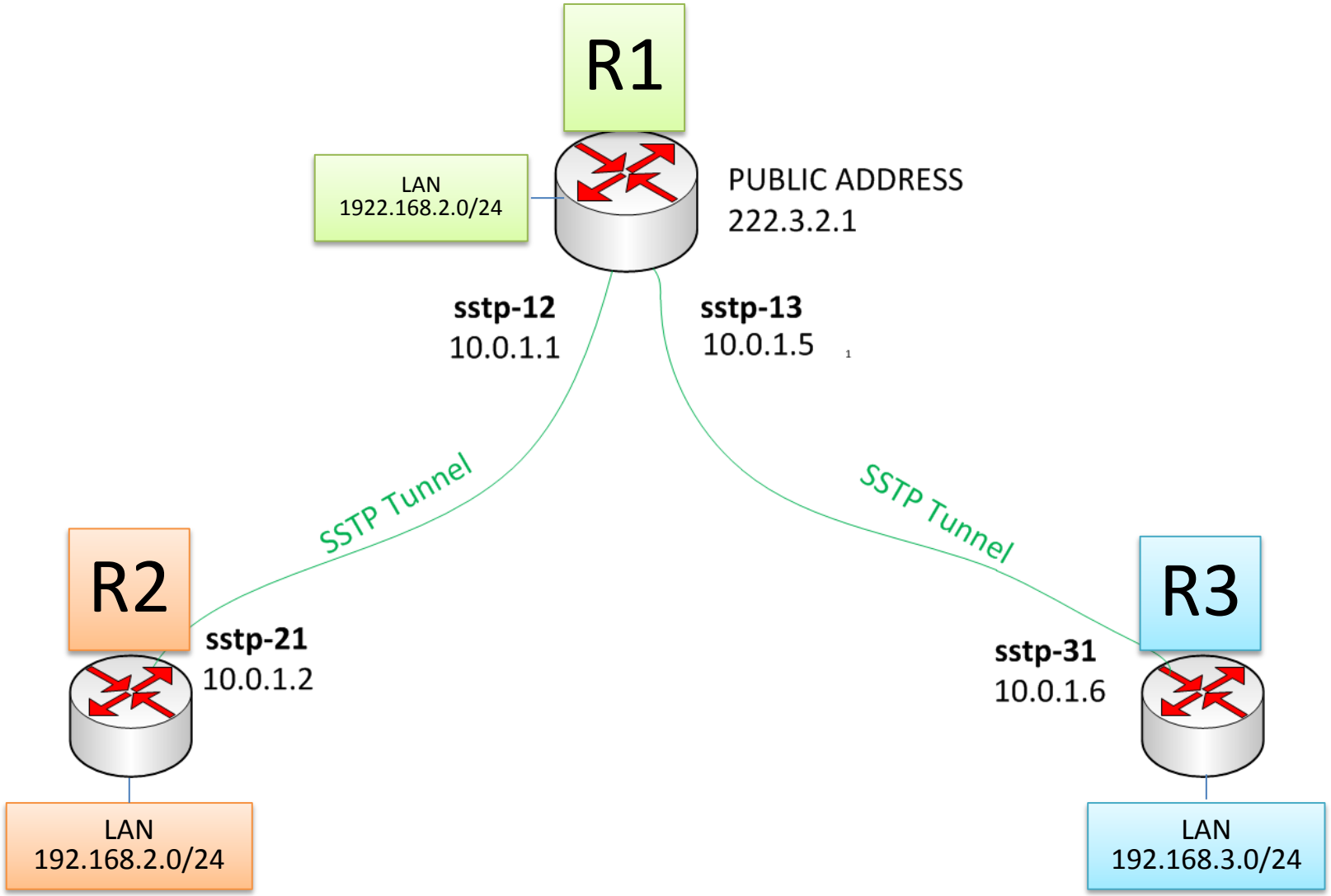
1. TCP Port 443
2. GRE protocol allowed
3. Username
4. Password
5. Valid Certificate (optional)

Later, we will configure
valid certificate **are required**
to establish secure VPN connection!

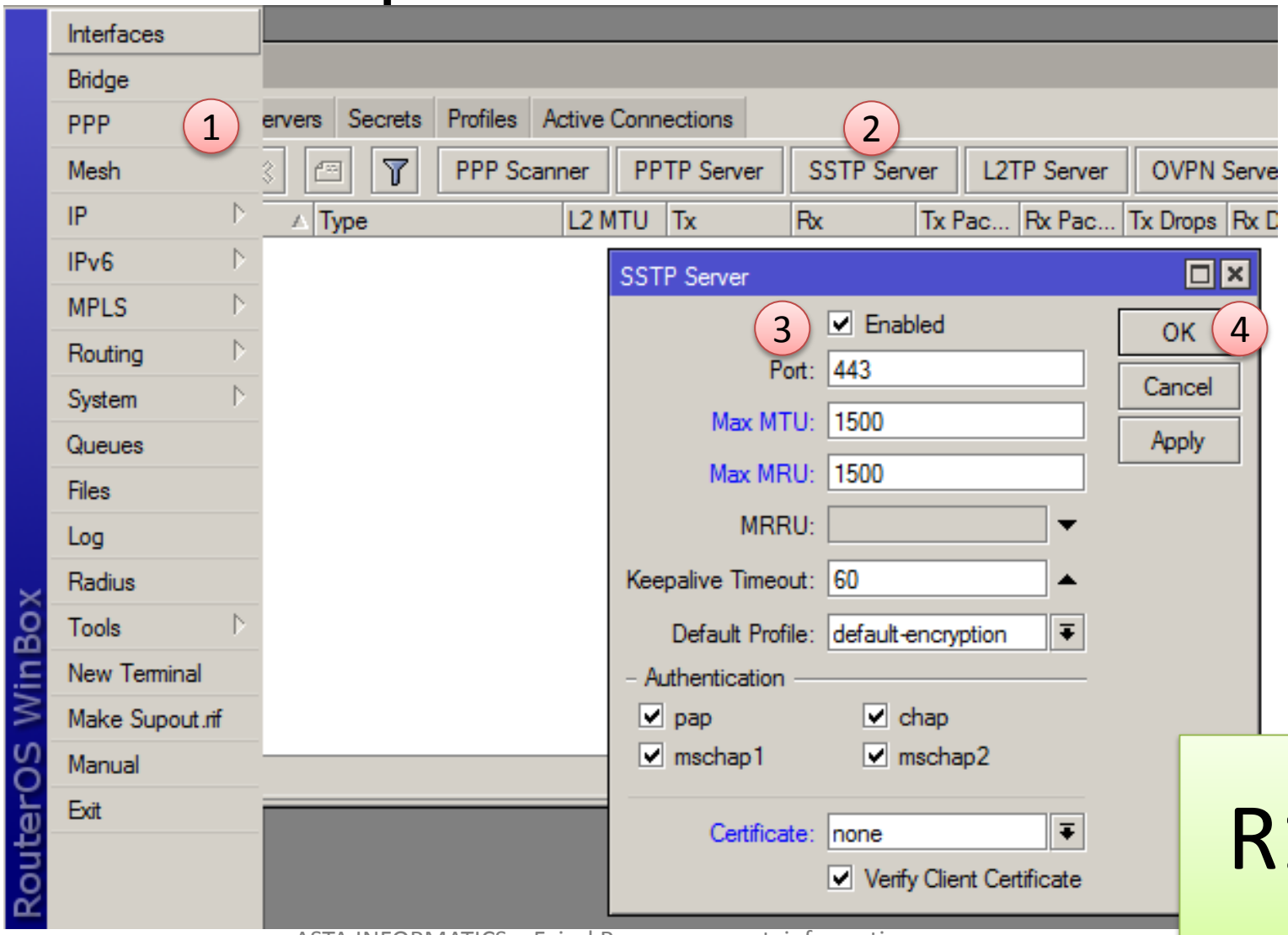
Network Topology



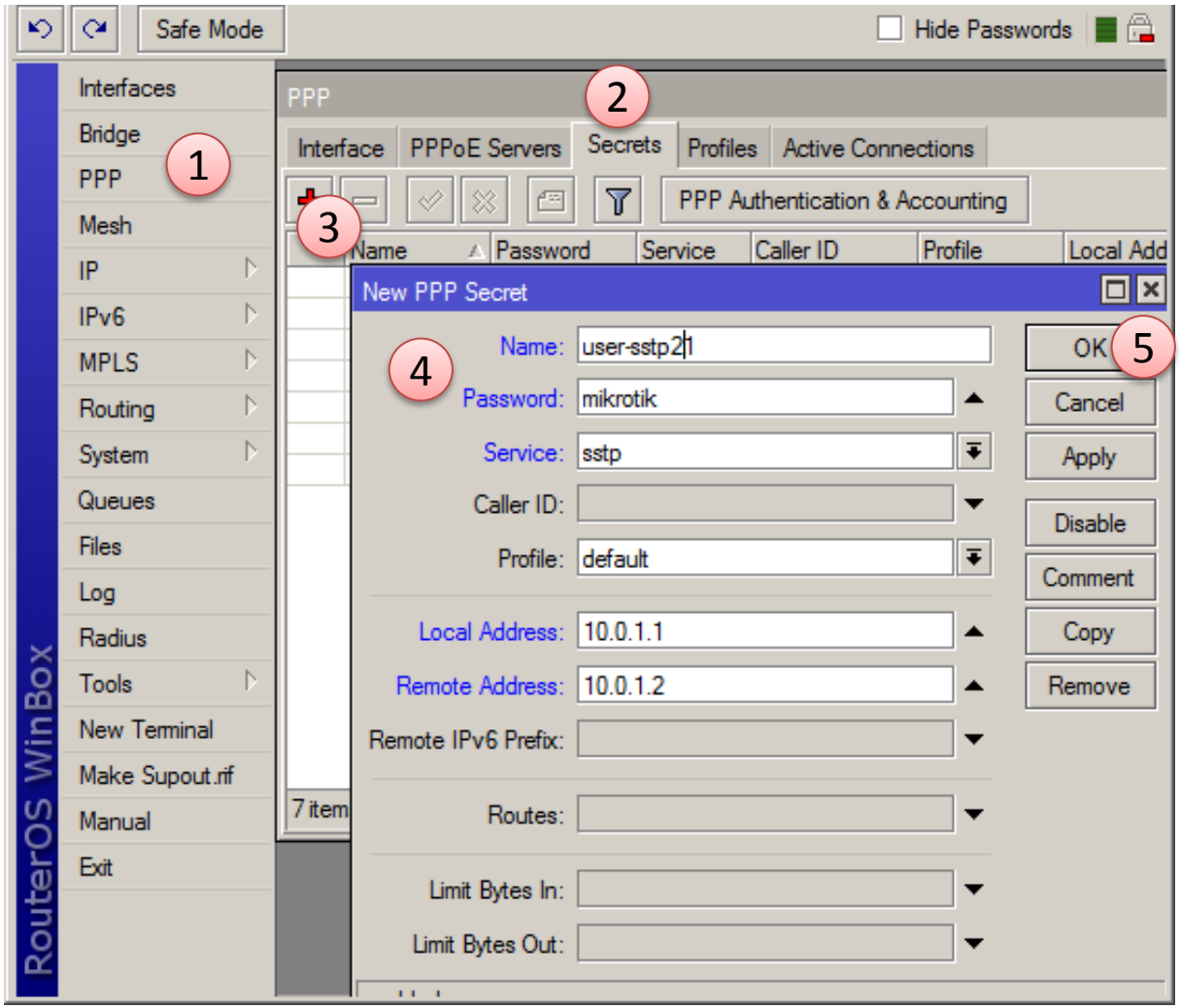
Simple Diagram



SSTP Server Service Setup
Step 1 : Enable SSTP

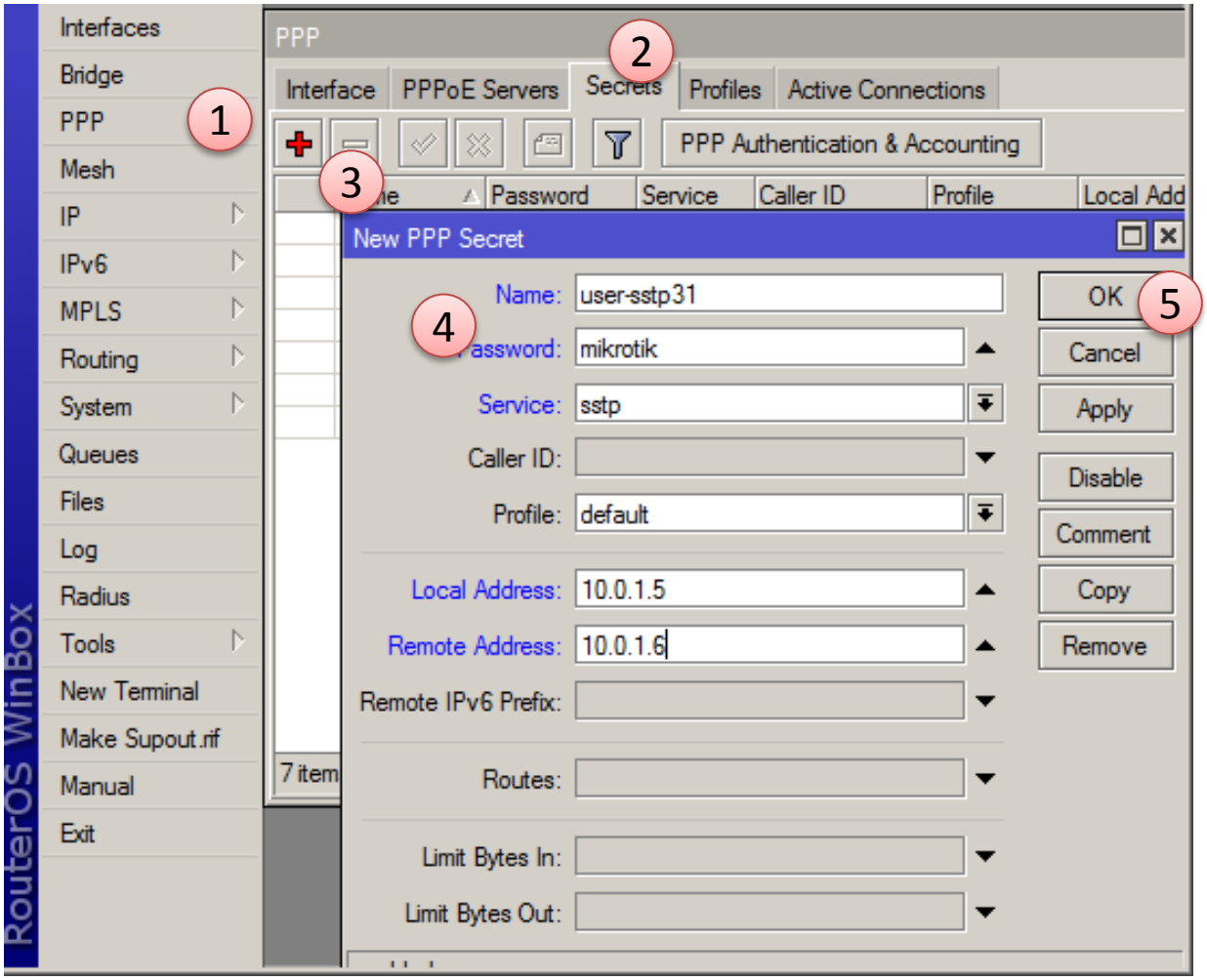


Step 2 : Create user access for R2



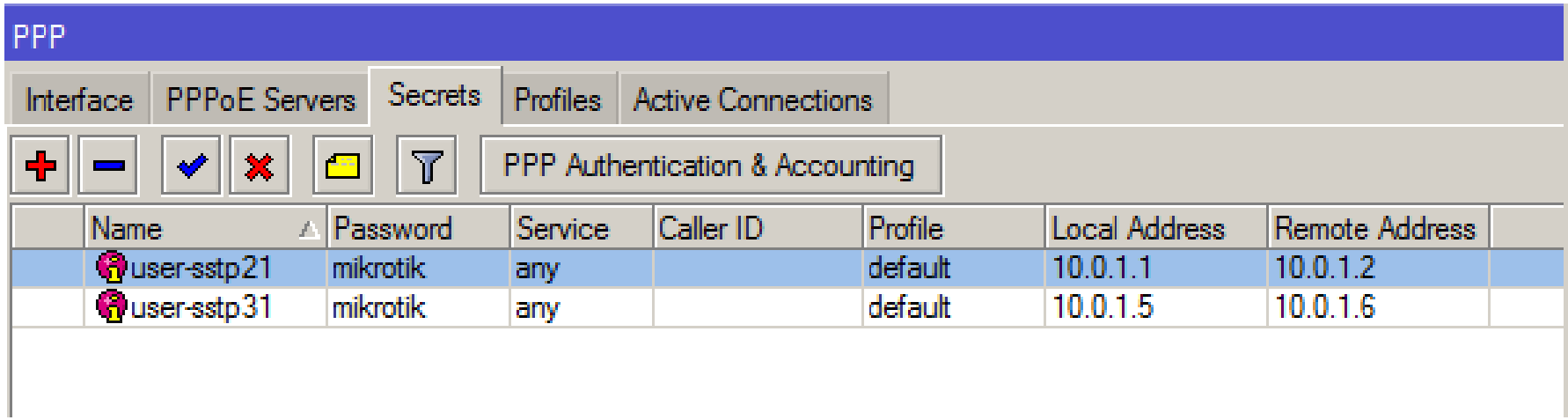
R1

Step 3: Create user access for R3



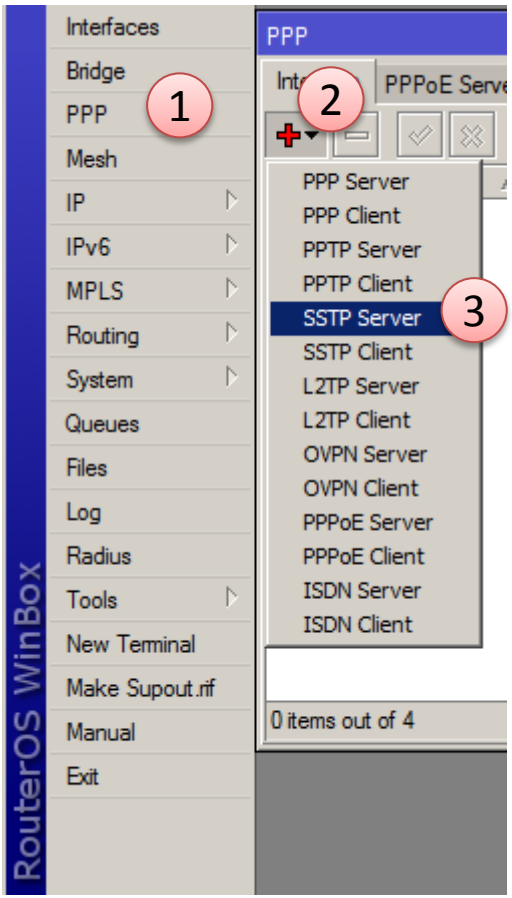
R1

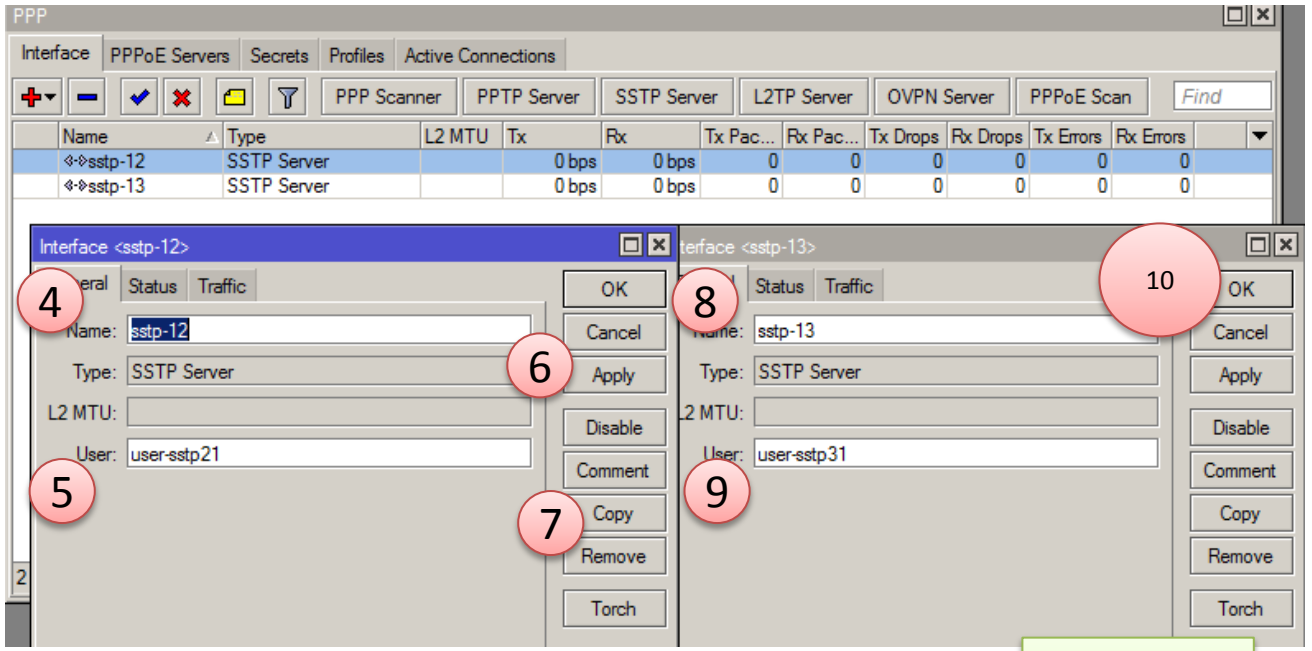
User List


PPP
Interface
PPPoE Servers
Secrets
Profiles
Active Connections
PPP Authentication & Accounting
Name
Password
Service
Caller ID
Profile
Local Address
Remote Address
user-sstp21
mikrotik
any
default
10.0.1.1
10.0.1.2
user-sstp31
mikrotik
any
default
10.0.1.5
10.0.1.6

R1

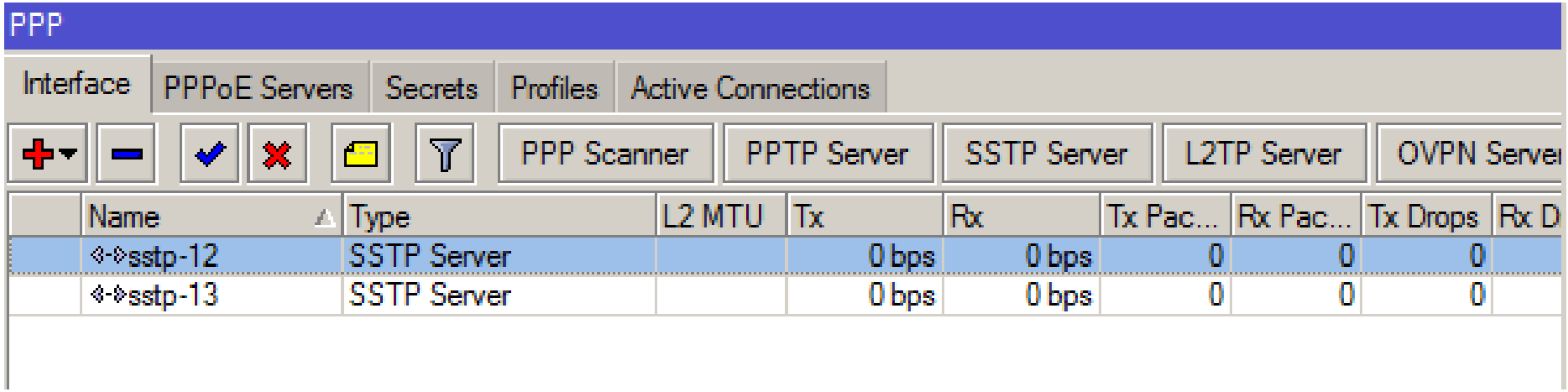
Step 3 : Create SSTP Server static interface





R1

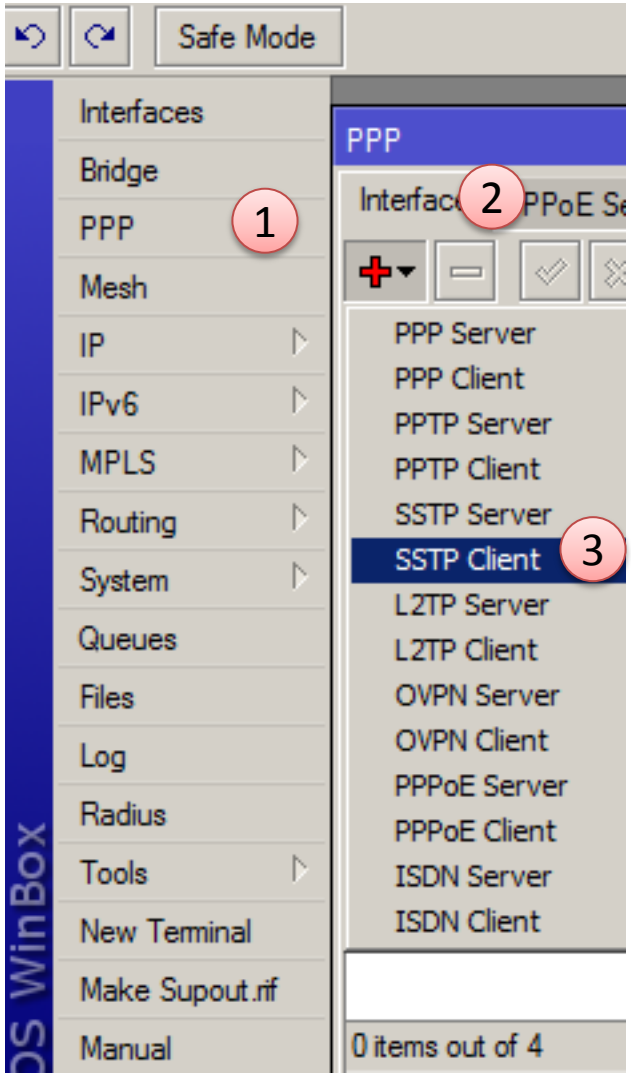
SSTP server interface list

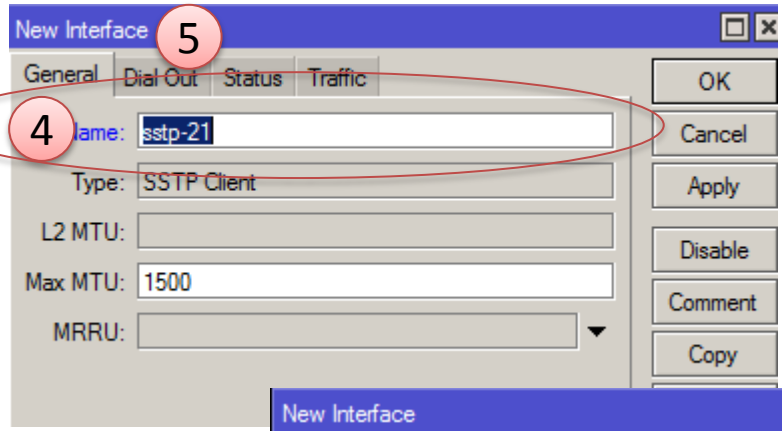


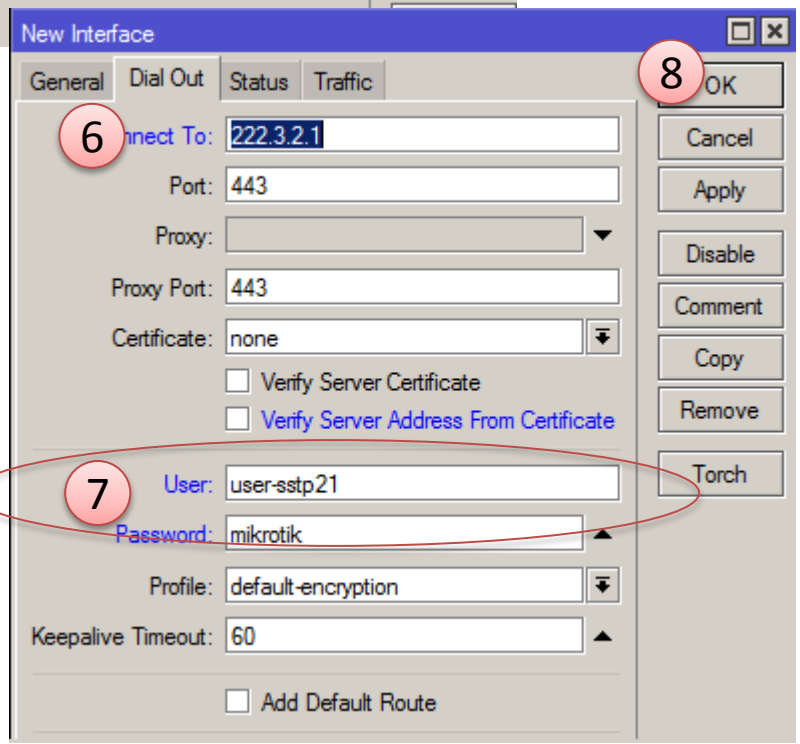
R1

SSTP Client Setup

R2

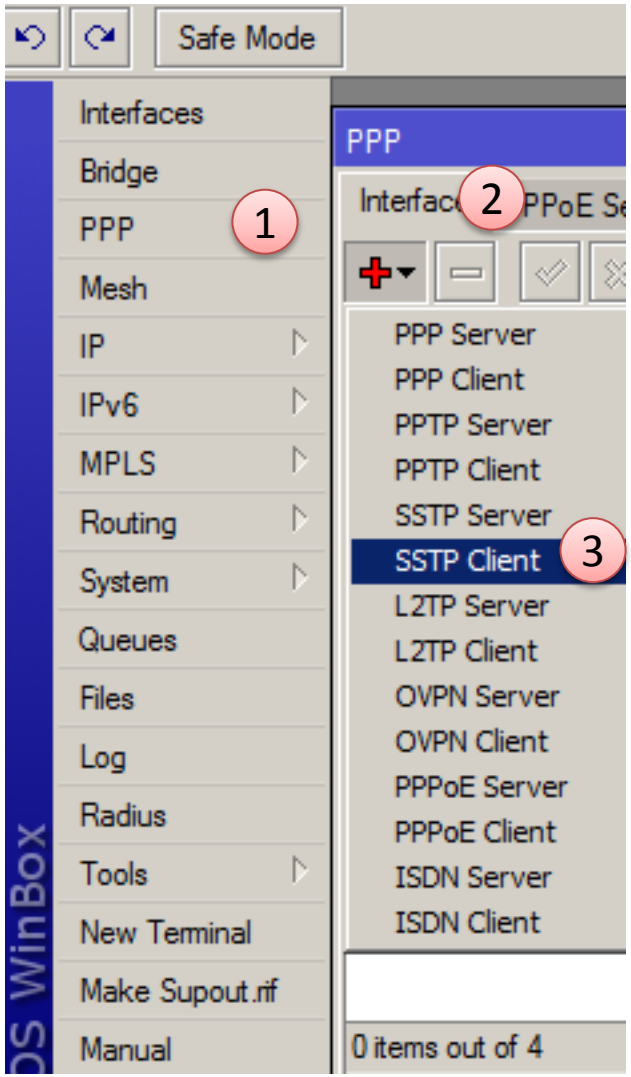


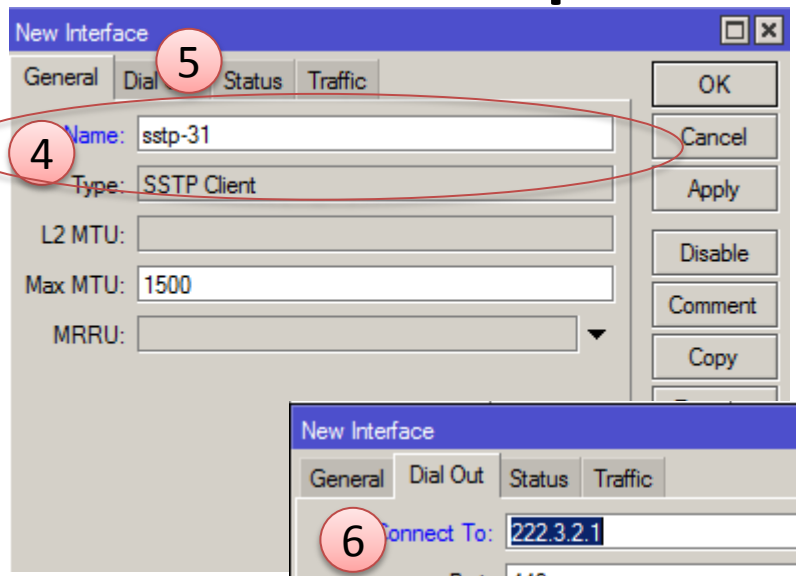


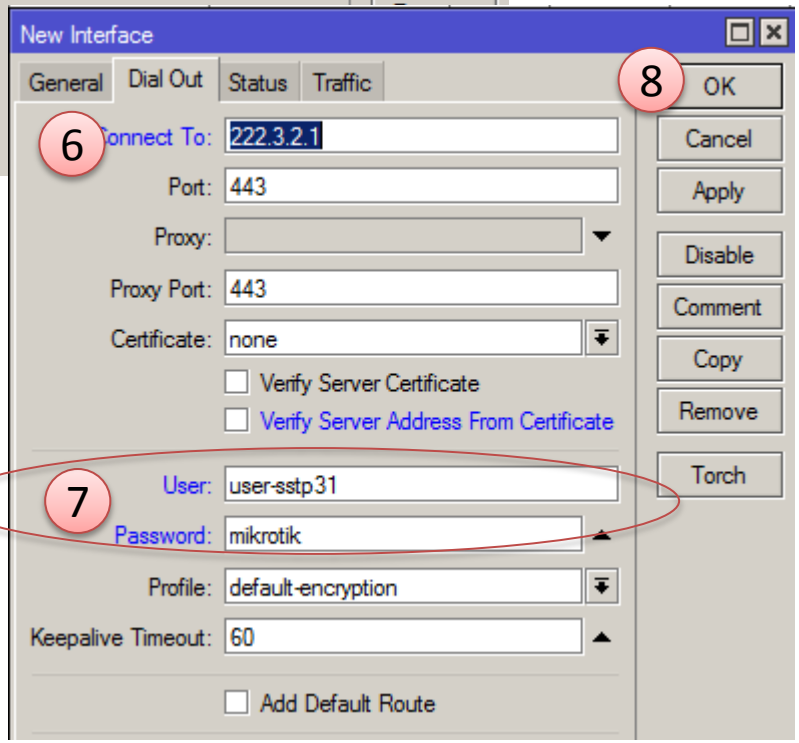


SSTP Client Setup

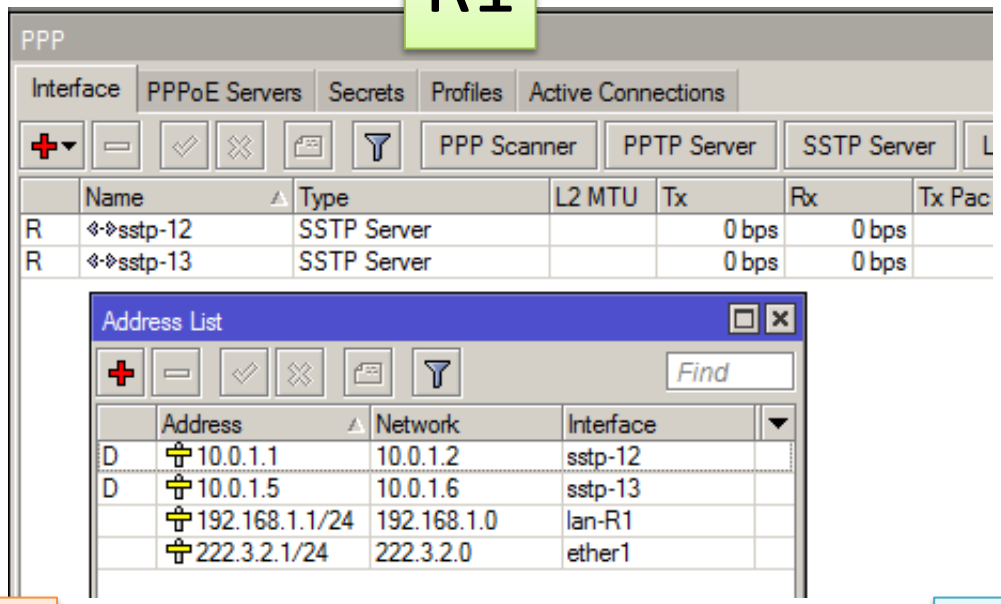
R3



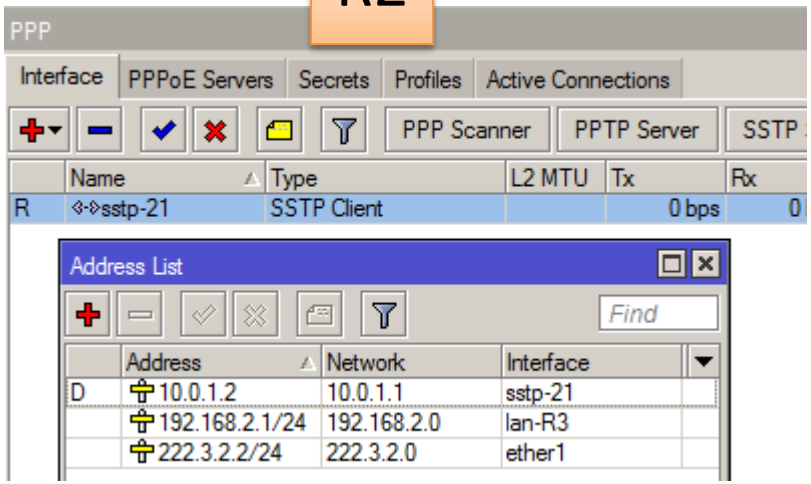




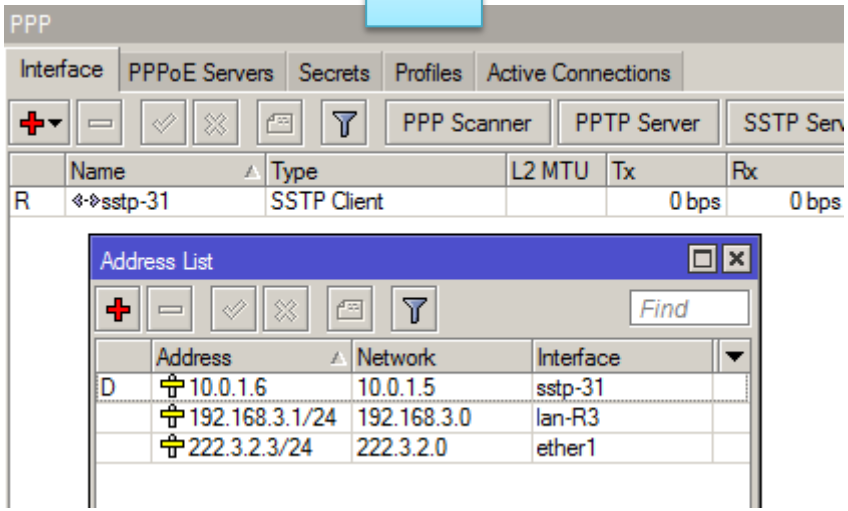
R1


PPP
Interface
PPPoE Servers
Secrets
Profiles
Active Connections
PPP Scanner
PPTP Server
SSTP Server
Name
Type
L2 MTU
Tx
Rx
Tx Pac
R sstp-12 SSTP Server 0 bps 0 bps
R sstp-13 SSTP Server 0 bps 0 bps
Address List
Find
Address
Network
Interface
D 10.0.1.1 10.0.1.2 sstp-12
D 10.0.1.5 10.0.1.6 sstp-13
192.168.1.1/24 192.168.1.0 lan-R1
222.3.2.1/24 222.3.2.0 ether1

R2


PPP
Interface
PPPoE Servers
Secrets
Profiles
Active Connections
PPP Scanner
PPTP Server
Name
Type
L2 MTU
Tx
Rx
R sstp-21 SSTP Client 0 bps 0
Address List
Find
Address
Network
Interface
D 10.0.1.2 10.0.1.1 sstp-21
192.168.2.1/24 192.168.2.0 lan-R3
222.3.2.2/24 222.3.2.0 ether1

R3


PPP
Interface
PPPoE Servers
Secrets
Profiles
Active Connections
PPP Scanner
PPTP Server
Name
Type
L2 MTU
Tx
Rx
R sstp-31 SSTP Client 0 bps 0 bps
Address List
Find
Address
Network
Interface
D 10.0.1.6 10.0.1.5 sstp-31
192.168.3.1/24 192.168.3.0 lan-R3
222.3.2.3/24 222.3.2.0 ether1

Test Ping, Success!

its now Connected without SSL Certificates

R2

```
[admin@R2-LAB] > ping 10.0.1.1
HOST                                     SIZE TTL TIME  STATUS
10.0.1.1                                   56  64 8ms
10.0.1.1                                   56  64 8ms
10.0.1.1                                   56  64 8ms
10.0.1.1                                   56  64 8ms
10.0.1.1                                   56  64 8ms
    sent=5 received=5 packet-loss=0% min-rtt=8ms avg-rtt=8ms max-rtt=8ms

[admin@R2-LAB] >
```

R3

```
[admin@R3-LAB] > ping 10.0.1.5
HOST                                     SIZE TTL TIME  STATUS
10.0.1.5                                   56  64 14ms
10.0.1.5                                   56  64 11ms
10.0.1.5                                   56  64 18ms
10.0.1.5                                   56  64 18ms
10.0.1.5                                   56  64 18ms
    sent=5 received=5 packet-loss=0% min-rtt=11ms avg-rtt=15ms max-rtt=18ms
```

Generates Certificate

- Generates self signed certificate using OpenSSL, **FREE**!
- if not familiar using linux, you can install ubuntu over virtual box, just for generates certificate
- #apt-get install openssl

Steps for generate self-signed SSL Certificate

Step 1 : ca.key

(CA = Certificate Authority) is the company which issues the SSL Certificate in this case, we use self-signed, our private CA

```
root@reza:~# 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: mikrotik
Verifying - Enter pass phrase for ca.key: mikrotik
root@reza:~#
```

Step 2 : ca.crt

```
root@reza:~# openssl req -new -x509 -days 3650 -key ca.key -out ca.crt
Enter pass phrase for ca.key: mikrotik
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) [AU]:ID
State or Province Name (full name) [Some-State]:Jakarta
Locality Name (eg, city) []:Jakarta
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Mikrotik
Organizational Unit Name (eg, section) []:IT
Common Name (eg, YOUR name) []:Mikrotik
Email Address []:admin@test.com
```

Certificate Pair for server

Step 3 : server.key

```
root@reza:~# 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: mikrotik
Verifying - Enter pass phrase for server.key:mikrotik
root@reza:~#
```

Step 4 : server.csr

```
root@reza:~# 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) [AU]:ID
State or Province Name (full name) [Some-State]:Jakarta
Locality Name (eg, city) []:Jakarta
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Mikrotik
Organizational Unit Name (eg, section) []:IT
Common Name (eg, YOUR name) []:Mikrotik
Email Address []:admin@test.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:mikrotik
An optional company name []:Mikrotik
```

Step 5 : server.crt

```
root@reza:~# 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=Jakarta/L=Jakarta/O=Mikrotik/OU=IT/CN=Mikrotik/emailAddress=admin@test.com
Getting CA Private Key
Enter pass phrase for ca.key: mikrotik
root@reza:~#
```

Generate Client Certificate Pair

Step 6 : client.key

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

Step 7 : client.csr

```
root@reza:~# openssl req -new -key client.key -out client.csr
Enter pass phrase for client.key: mikrotik
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) [AU]:ID
State or Province Name (full name) [Some-State]:Jakarta
Locality Name (eg, city) []:Jakarta
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Mikrotik
Organizational Unit Name (eg, section) []:IT
Common Name (eg, YOUR name) []:Mikrotik
Email Address []:admin@test.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:mikrotik
An optional company name []:Mikrotik
```

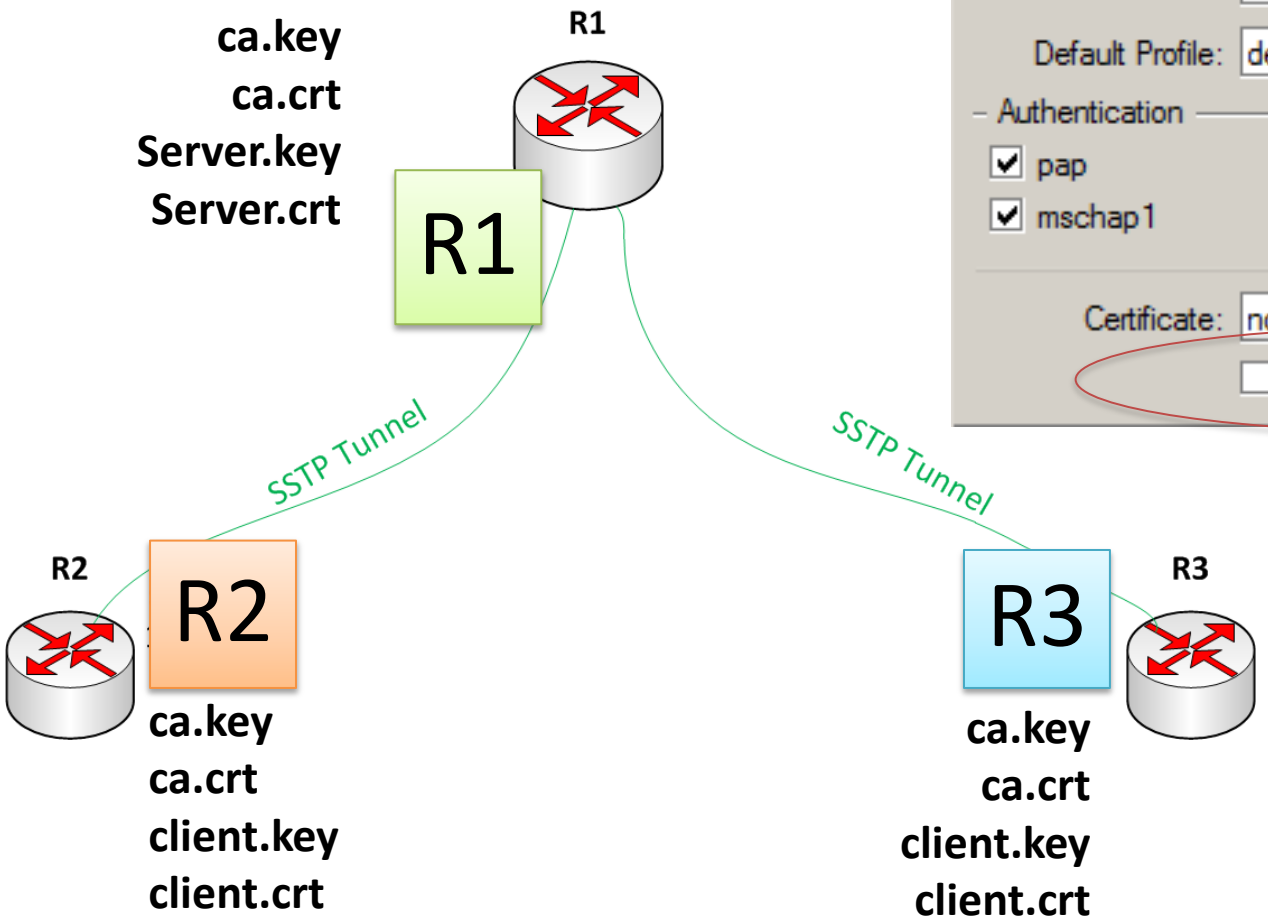
Step 8 : client.crt

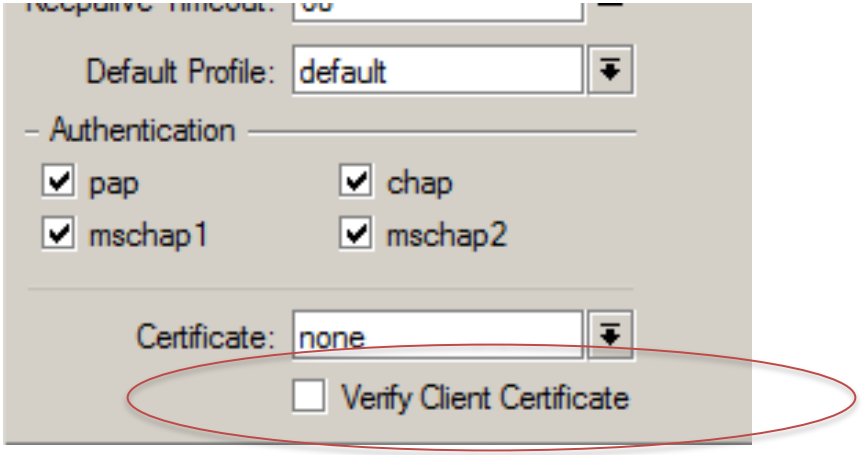
```
root@reza:~# openssl x509 -req -days 3650 -in client.csr -CA ca.crt -CAkey ca.key -
set_serial 01 -out client.crt
Signature ok
subject=/C=ID/ST=Jakarta/L=Jakarta/O=Mikrotik/OU=IT/CN=Mikrotik/emailAddress=ad
min@test.com
Getting CA Private Key
Enter pass phrase for ca.key: mikrotik
```

Now we have 8 files as below, 6 will be used :

```
-rw-r--r-- 1 root root 2297 2012-10-19 16:03 ca.crt          ← used
-rw-r--r-- 1 root root 3311 2012-10-19 15:59 ca.key          ← user
-rw-r--r-- 1 root root 1960 2012-10-19 16:30 client.crt      ← used
-rw-r--r-- 1 root root 1805 2012-10-19 16:28 client.csr
-rw-r--r-- 1 root root 3311 2012-10-19 16:25 client.key      ← user
-rw-r--r-- 1 root root 1960 2012-10-19 16:19 server.crt      ← used
-rw-r--r-- 1 root root 1805 2012-10-19 16:16 server.csr
-rw-r--r-- 1 root root 3311 2012-10-19 16:12 server.key      ←used
```

Certificate Distribution



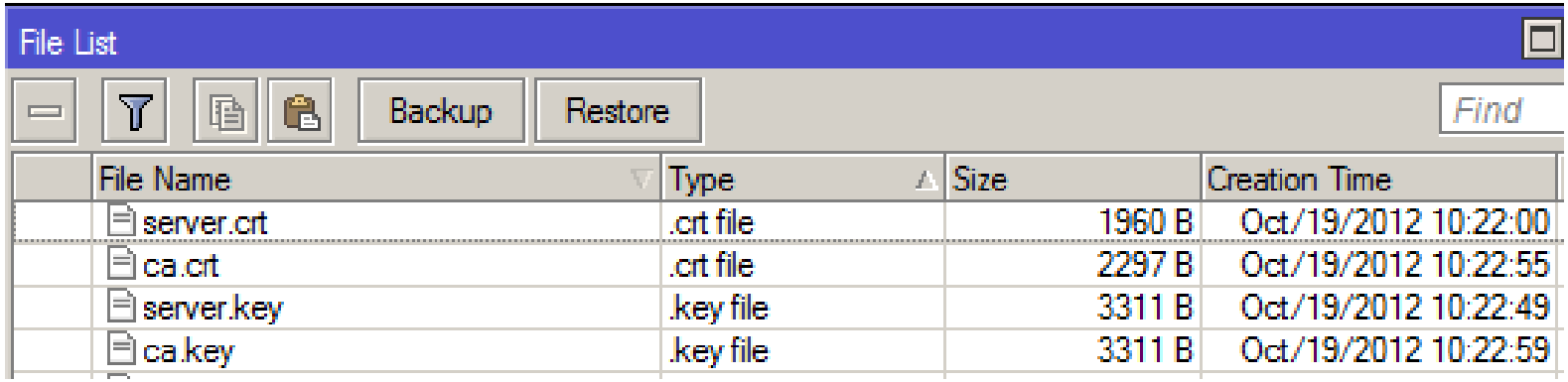


when use **client certificate verification,** CA certificate must be imported too

Upload certificate to each Router

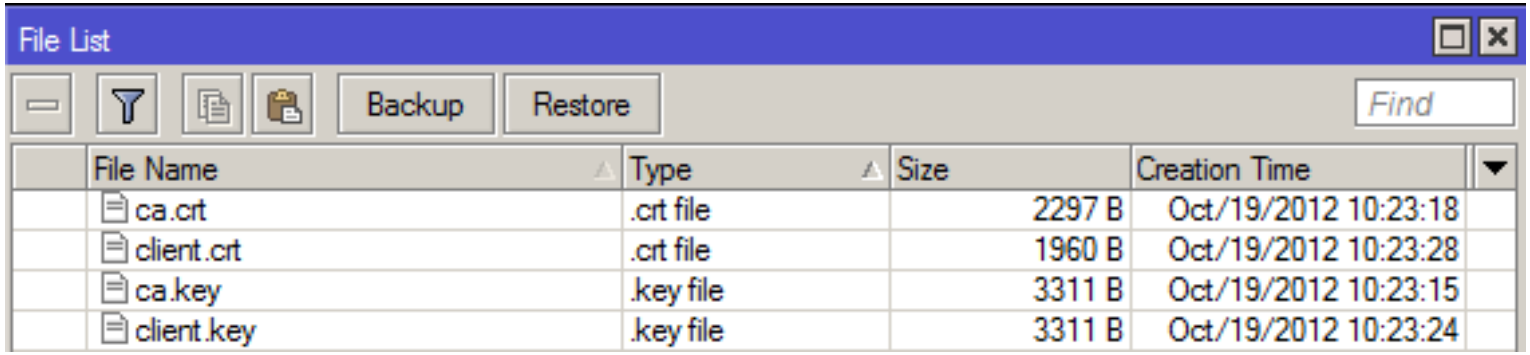
Upload the certificate file according to certificate distribution using ftp, ssh, file copy etc.

R1



R2

R3



IMPORT SERVER CERTIFICATE to MikroTik

R1

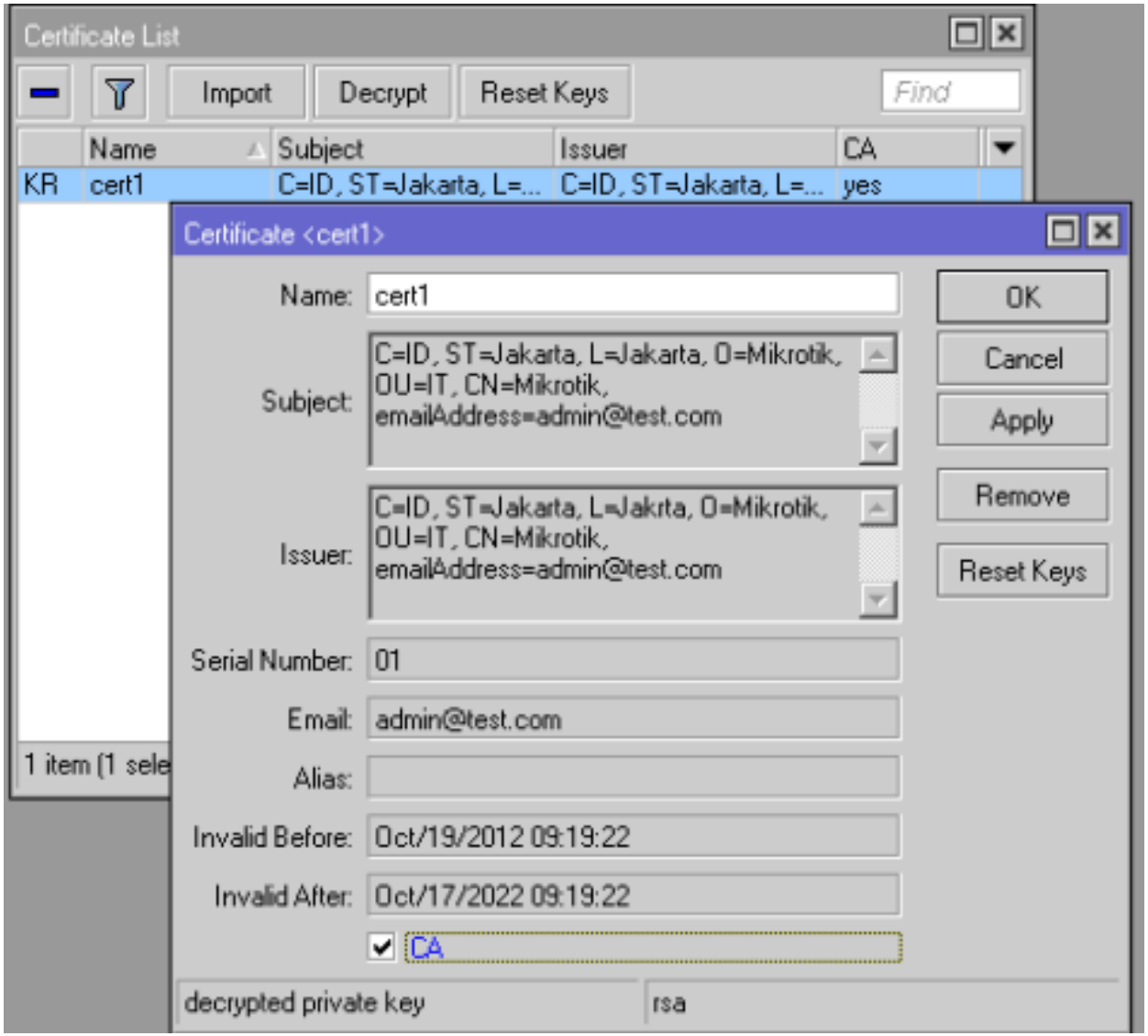
[admin@R1-LAB] /certificate> import file-name=server.crt
passphrase: ***mikrotik***
certificates-imported: 1
private-keys-imported: 0
files-imported: 1
decryption-failures: 0
keys-with-no-certificate: 0

[admin@R1-LAB] /certificate> import file-name=server.key
passphrase: ***mikrotik***
certificates-imported: 0
private-keys-imported: 1
files-imported: 1
decryption-failures: 0
keys-with-no-certificate: 0

If everything is imported properly then certificate should show up with **KR** flag.

```
[admin@R1-LAB] > /certificate print
Flags: K - decrypted-private-key, Q - private-key, R - rsa, D - dsa
 0 KR name="cert1" subject=C=ID,ST=Jakarta,L=Jakarta,O=Mikrotik,OU=IT,CN=Mikrotik,
        emailAddress=admin@test.com
      issuer=C=ID,ST=Jakarta,L=Jakrta,O=Mikrotik,OU=IT,CN=Mikrotik,
       emailAddress=admin@test.com
      serial-number="01" email=admin@test.com
      invalid-before=oct/19/2012 09:19:22 invalid-after=oct/17/2022 09:19:22
      ca=yes
[admin@R1-LAB] >
```

Server Certificate



R1

Import ca.crt

```
[admin@R1-LAB] /certificate> import file-name=ca.crt
passphrase: mikrotik
    certificates-imported: 1
    private-keys-imported: 0
    files-imported: 1
    decryption-failures: 0
    keys-with-no-certificate: 0

[admin@R1-LAB] /certificate> import file-name=ca.key
passphrase: mikrotik
    certificates-imported: 0
    private-keys-imported: 1
    files-imported: 1
    decryption-failures: 0
    keys-with-no-certificate: 0
```

R2

R3

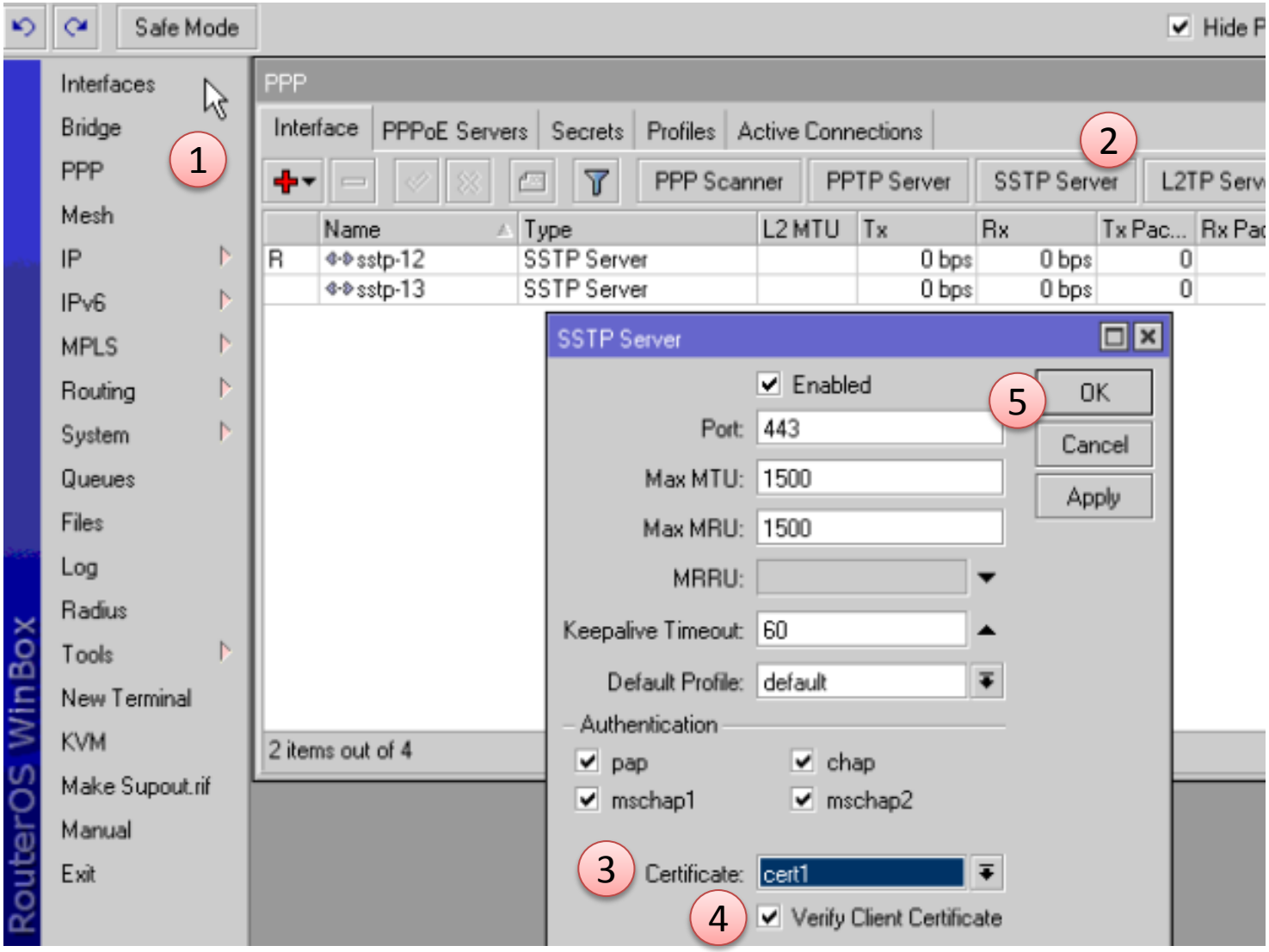
Client Certificate

admin@R2-LAB] /certificate> import file-name=client.crt
passphrase: ***mikrotik***
    certificates-imported: 1
    private-keys-imported: 0
    files-imported: 1
    decryption-failures: 0
     keys-with-no-certificate: 0

[admin@R2-LAB] /certificate> import file-name=client.key
passphrase: ***mikrotik***
    certificates-imported: 0
    private-keys-imported: 1
    files-imported: 1
    decryption-failures: 0
    keys-with-no-certificate: 0

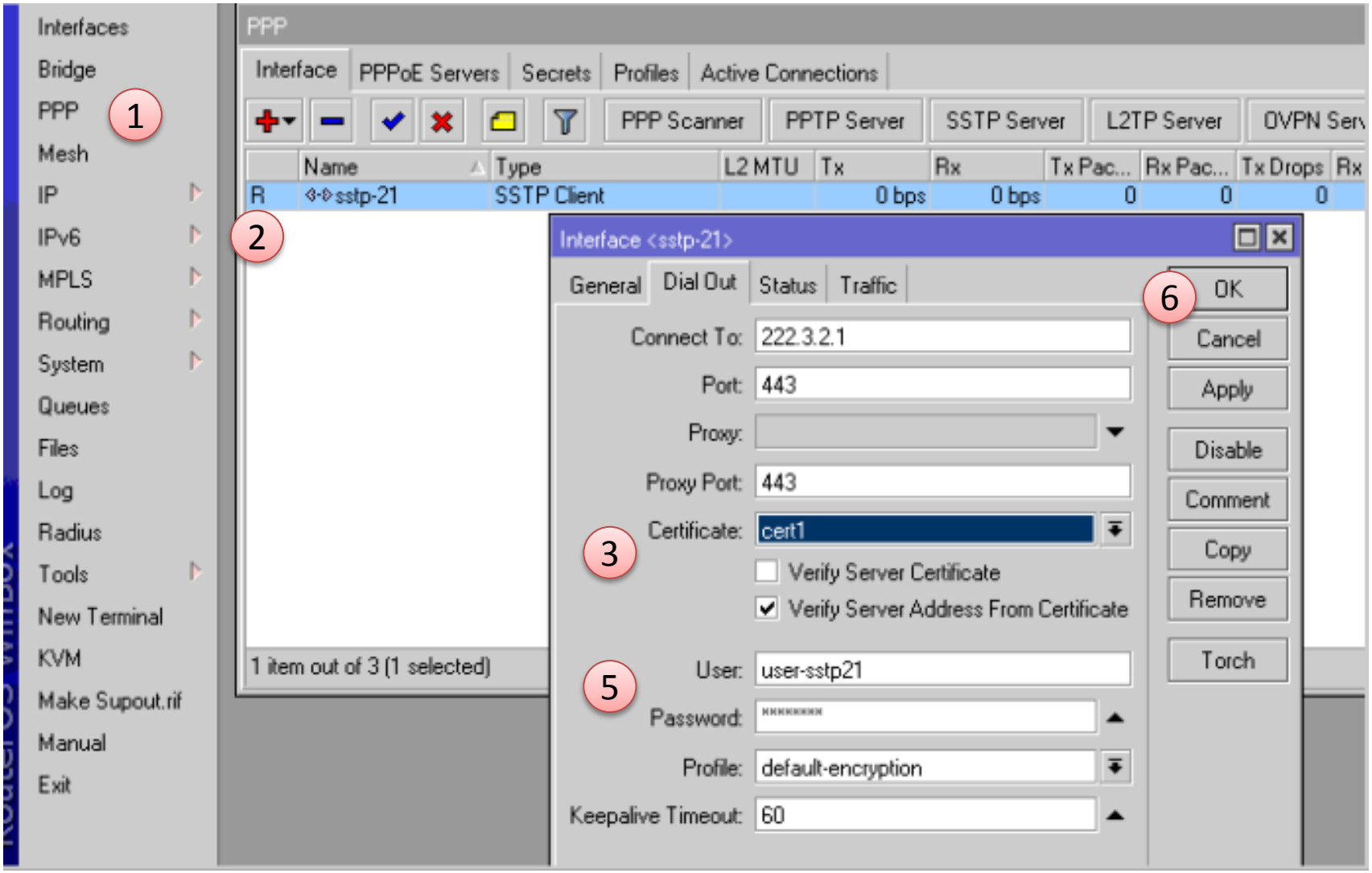
R1

Set SSTP Server Using Certificate

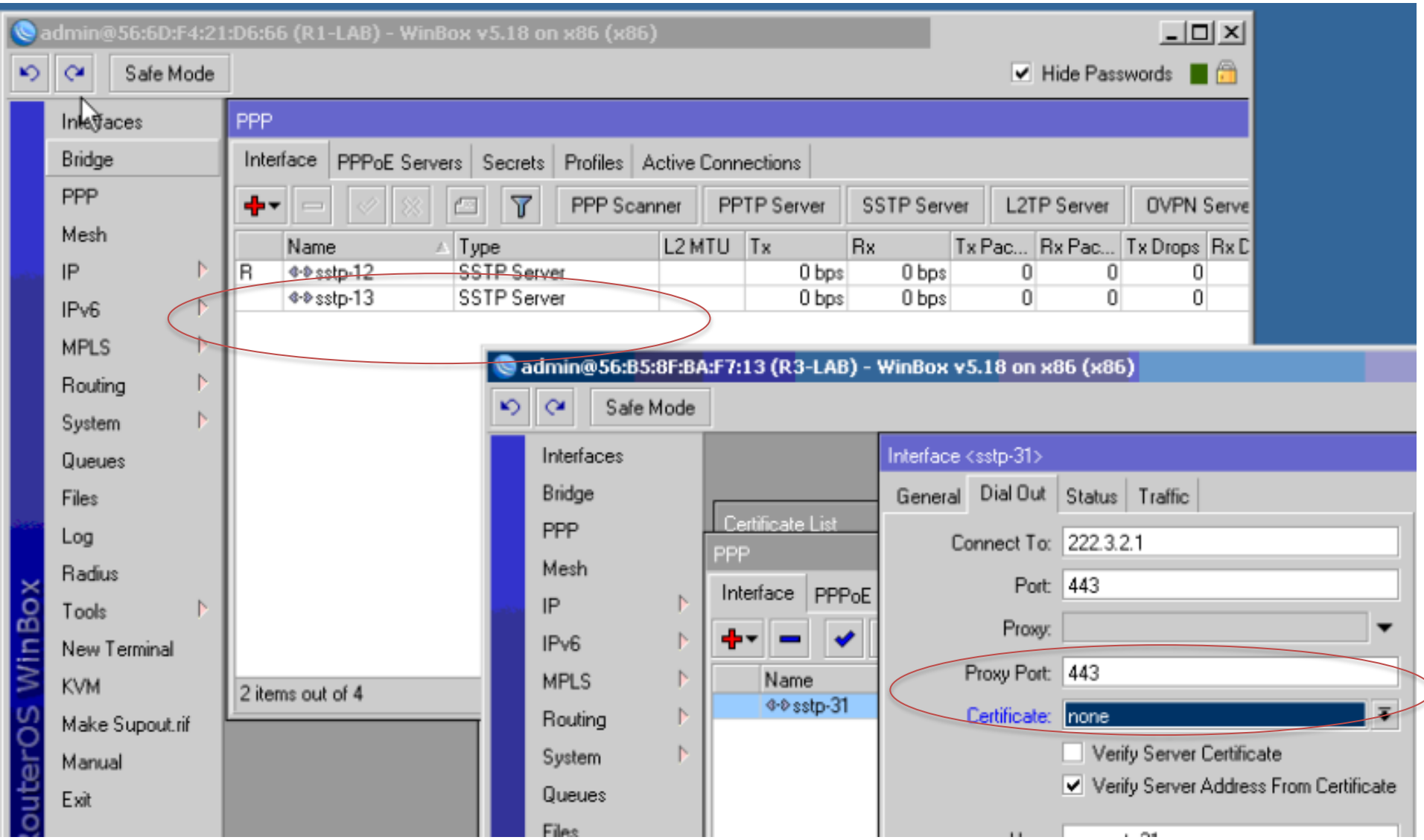


R2

Set SSTP Client Using Certificate

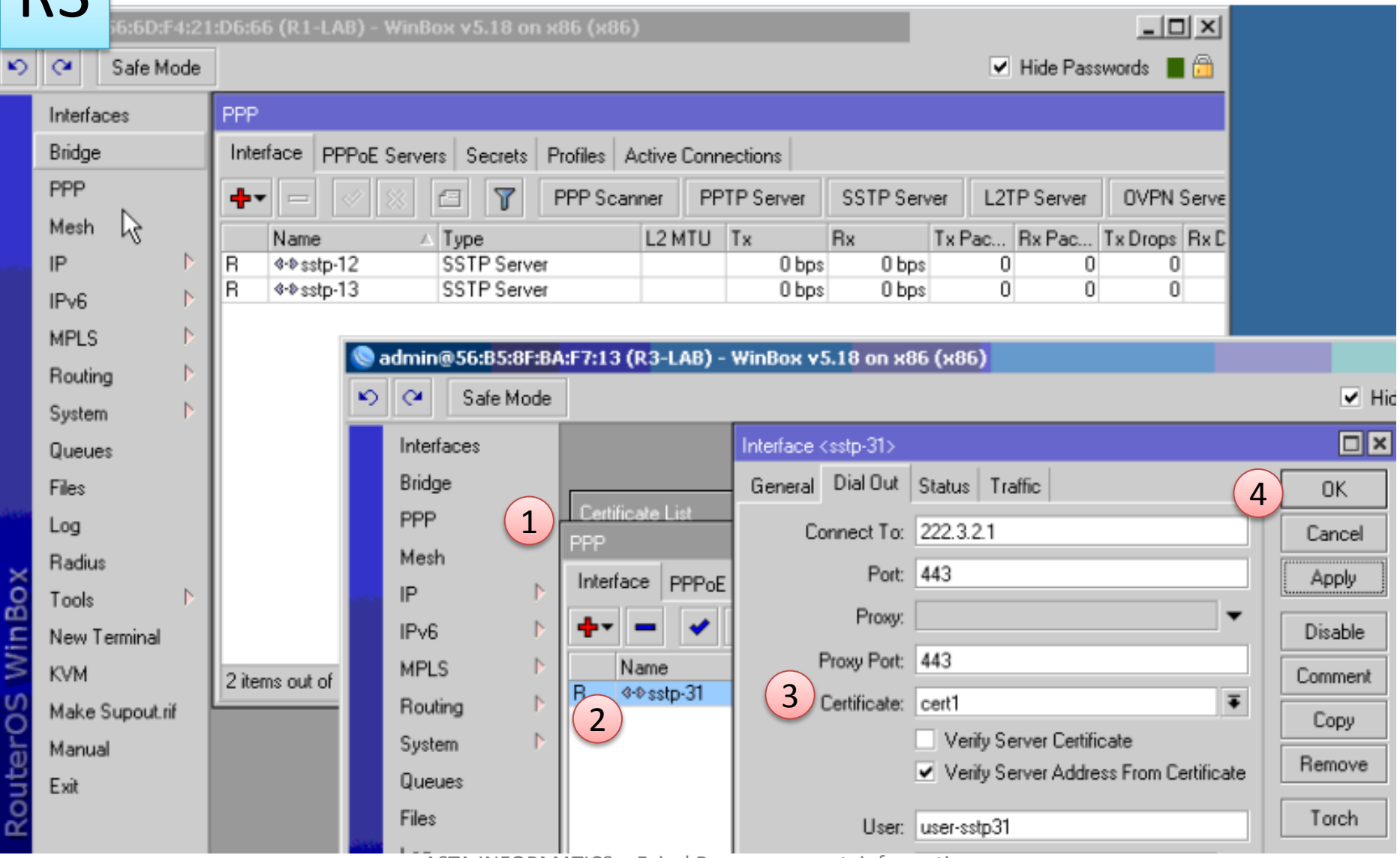


Before Activate the Cert on Client R3



Set certificate, then Secure Connect!

R3



Access to LAN segment in each router

Using static routing using sstp connected ip address as gateway : suitable for small network

R1

```
[admin@R1-LAB] > ip route add dst-address=192.168.2.0/24 gateway=10.0.1.2
[admin@R1-LAB] > ip route add dst-address=192.168.3.0/24 gateway=10.0.1.6
```

R2

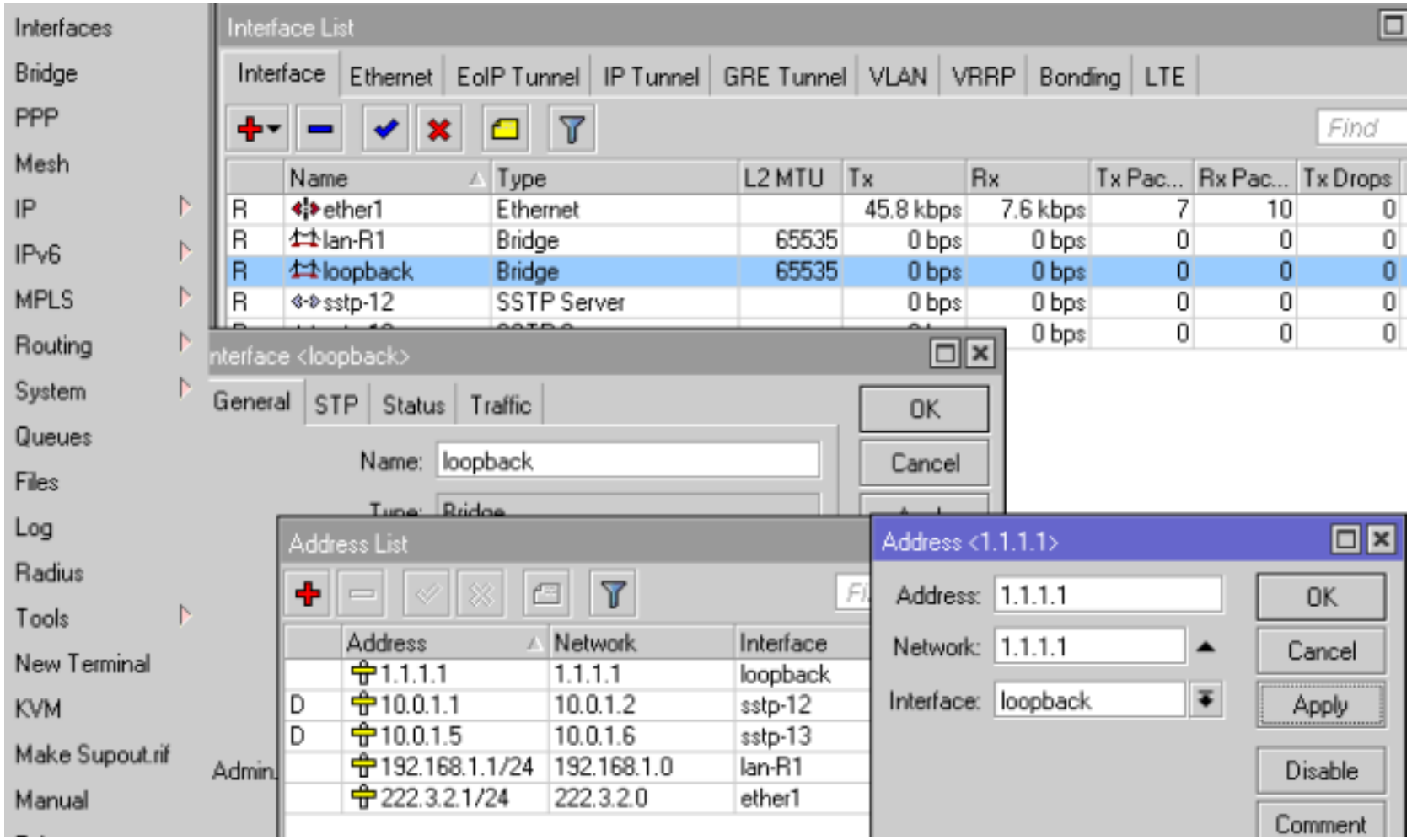
```
[admin@R2-LAB] > ip route add dst-address=192.168.1.0/24 gateway=10.0.1.1
[admin@R2-LAB] > ip route add dst-address=192.168.3.0/24 gateway=10.0.1.1
[admin@R2-LAB] > ip route add dst-address=10.0.1.4/30 gateway=10.0.1.1
```

R3

```
[admin@R3-LAB] > ip route add dst-address=192.168.1.0/24 gateway=10.0.1.1
[admin@R3-LAB] > ip route add dst-address=192.168.2.0/24 gateway=10.0.1.1
[admin@R3-LAB] > ip route add dst-address=10.0.1.0/30 gateway=10.0.1.1
```

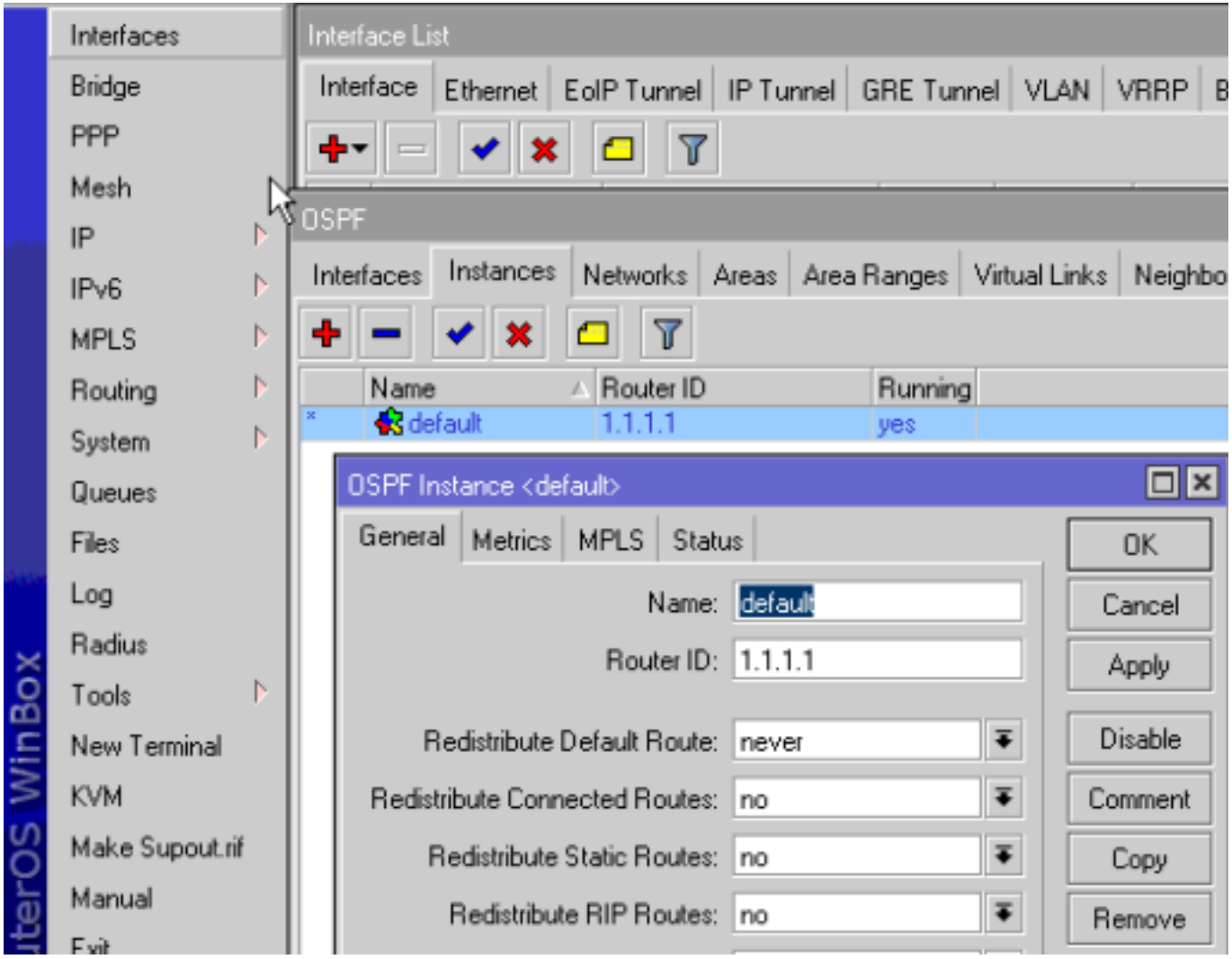
R1

using OSPF ove SSTP for Larger Network



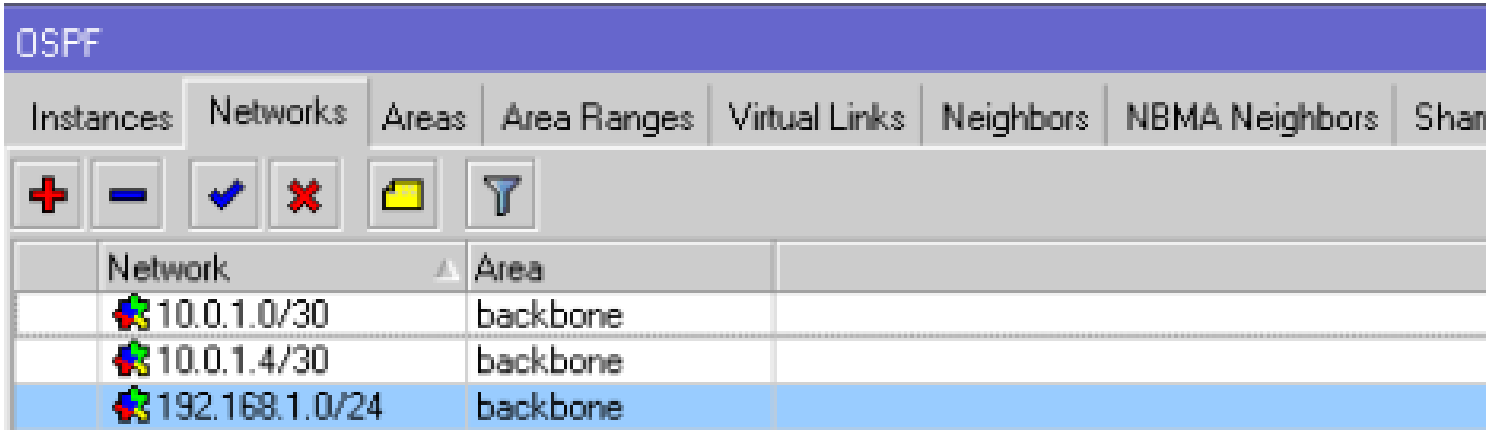
R1

Routing > OSPF > Instances

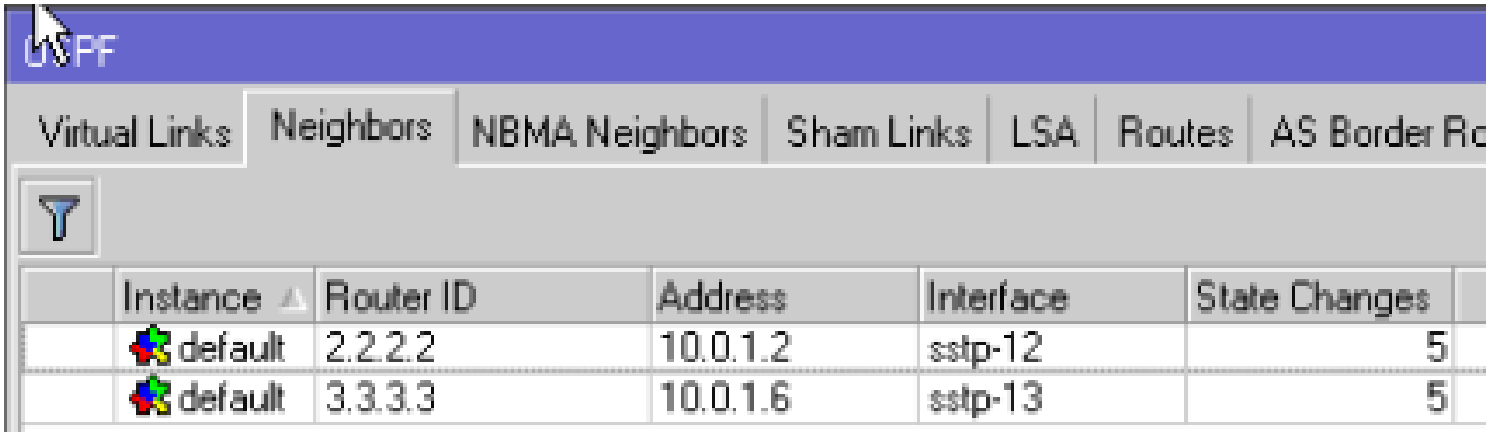


R1

Routing > OSPF > Networks

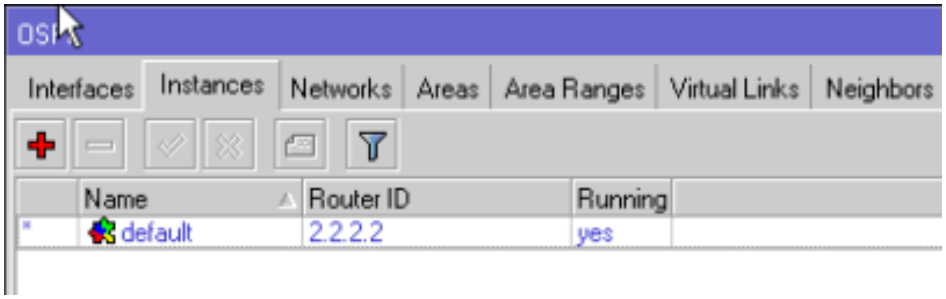


Routing > OSPF > Neighbours

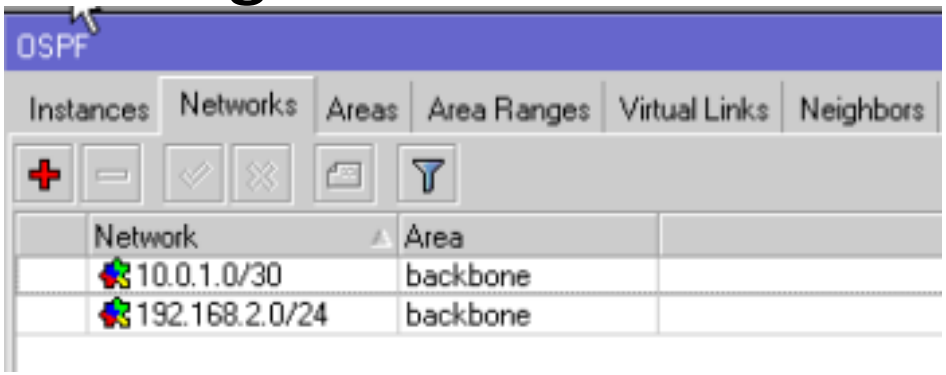


R2

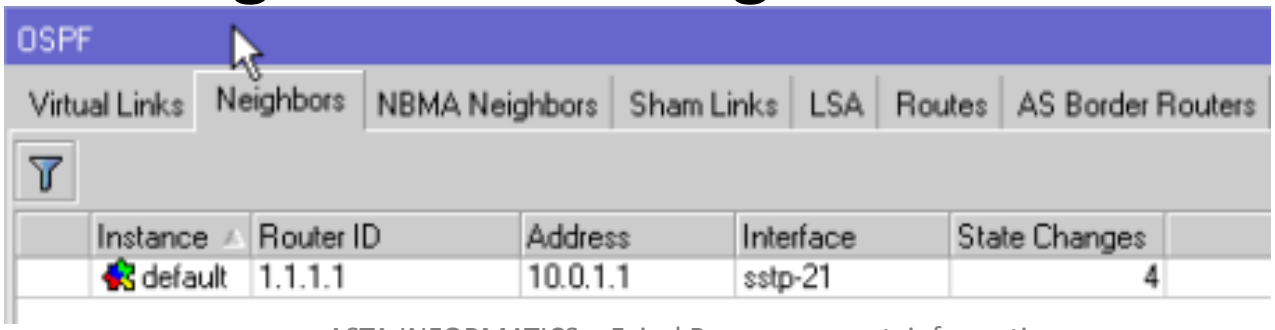
Routing > OSPF > Instances



Routing > OSPF > Networks

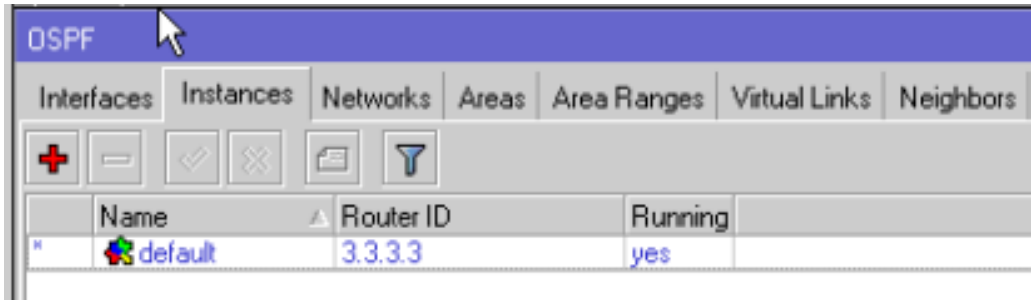


Routing > OSPF > Neighbours

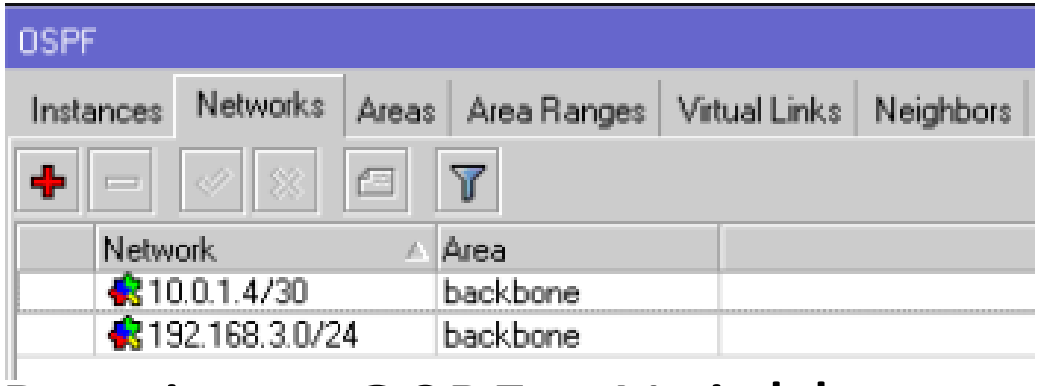


R3

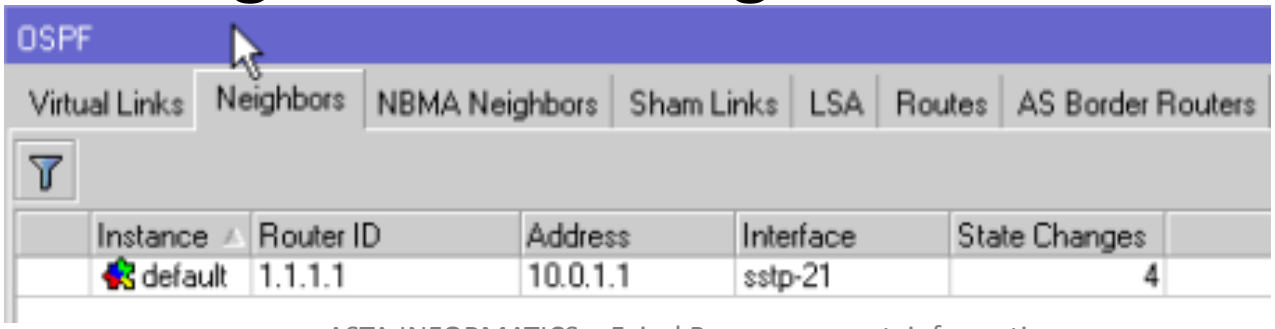
Routing > OSPF > Instances



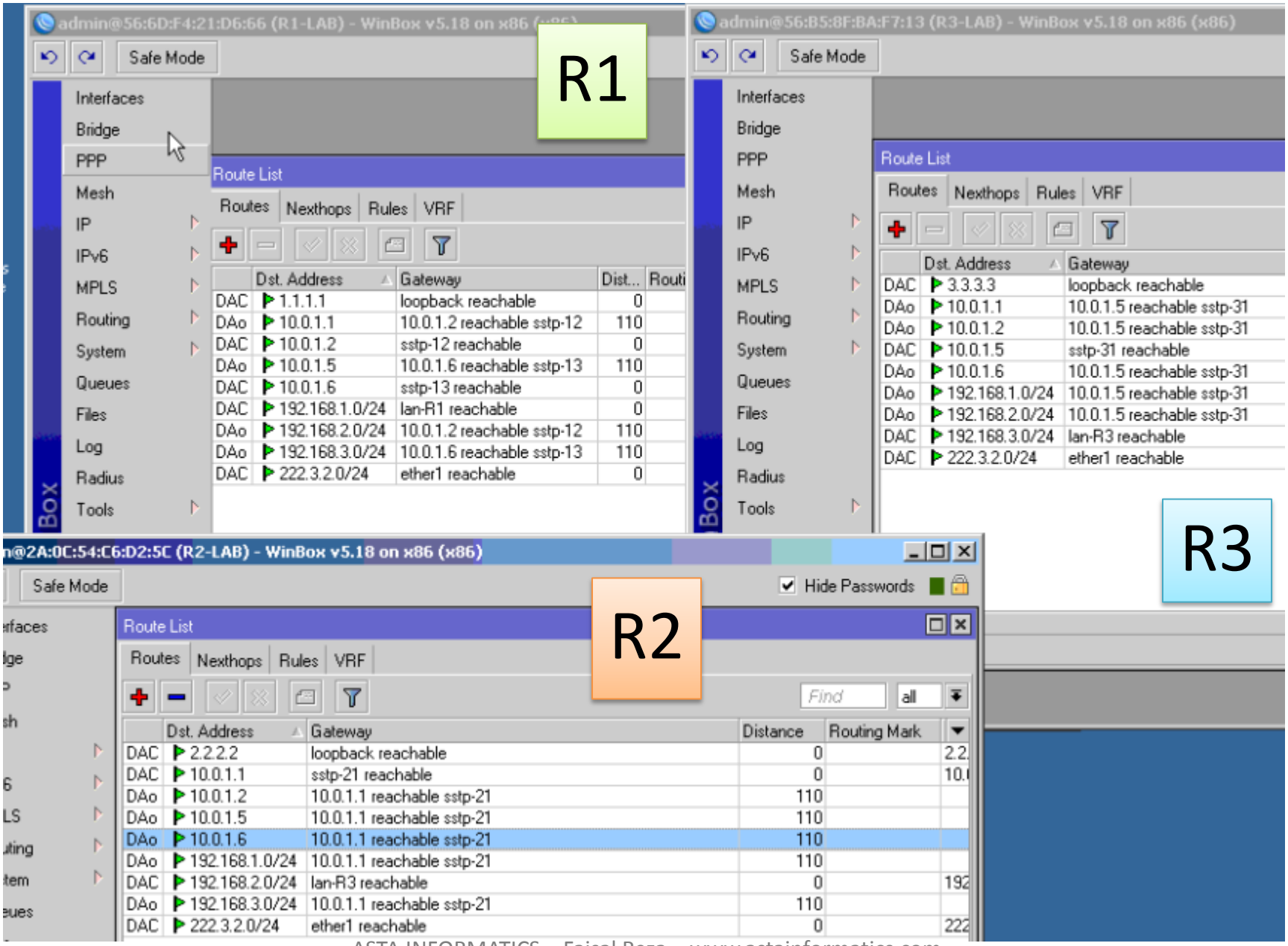
Routing > OSPF > Networks



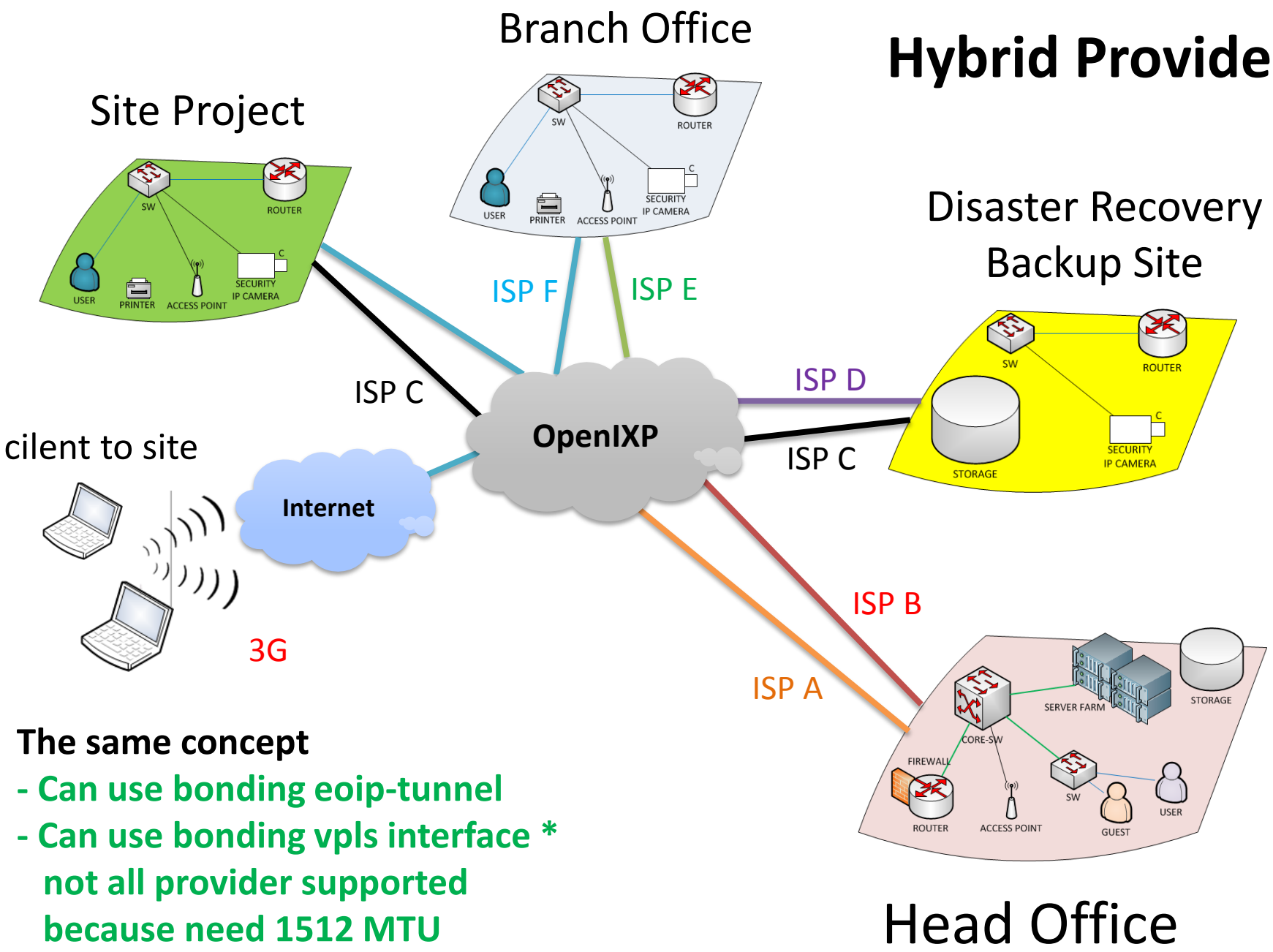
Routing > OSPF > Neighbours



All Routing Table



Hybrid Provider



The same concept

- **Can use bonding eoip-tunnel**
- **Can use bonding vpls interface ***
  not all provider supported
  because need 1512 MTU

Interested with this low cost solution but don’t want to dive deep technical things?

Let me help you!

reza@astainformatics.com

www.astainformatics.com

Any Question?

Please feel free to contact me anytime after the presentation

Thank You
See you in another MUM!

routing the world with Forum MikroTik Indonesia

Special thanks to Forum Mikrotik Indonesia