MTCSE

Chiang Mai , Thailand

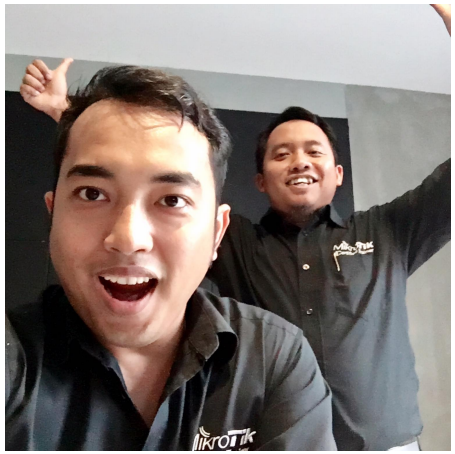
5-6 April , 2022

Schedule

- Training day: 9AM - 5PM
- 30min breaks: 10:30AM and 3PM
- 1h lunch: 12:30PM
- Certification test: last day, 1 hour

INTRODUCE
Trainer

- Mana Kaewcharoen
- MTCNA, MTCTCE, MTCWE
- MTCUME, MTCRE , MTCINE
- MTCIPv6E, MTCSE
- MikroTik Academy Trainer
- MikroTik Trainer

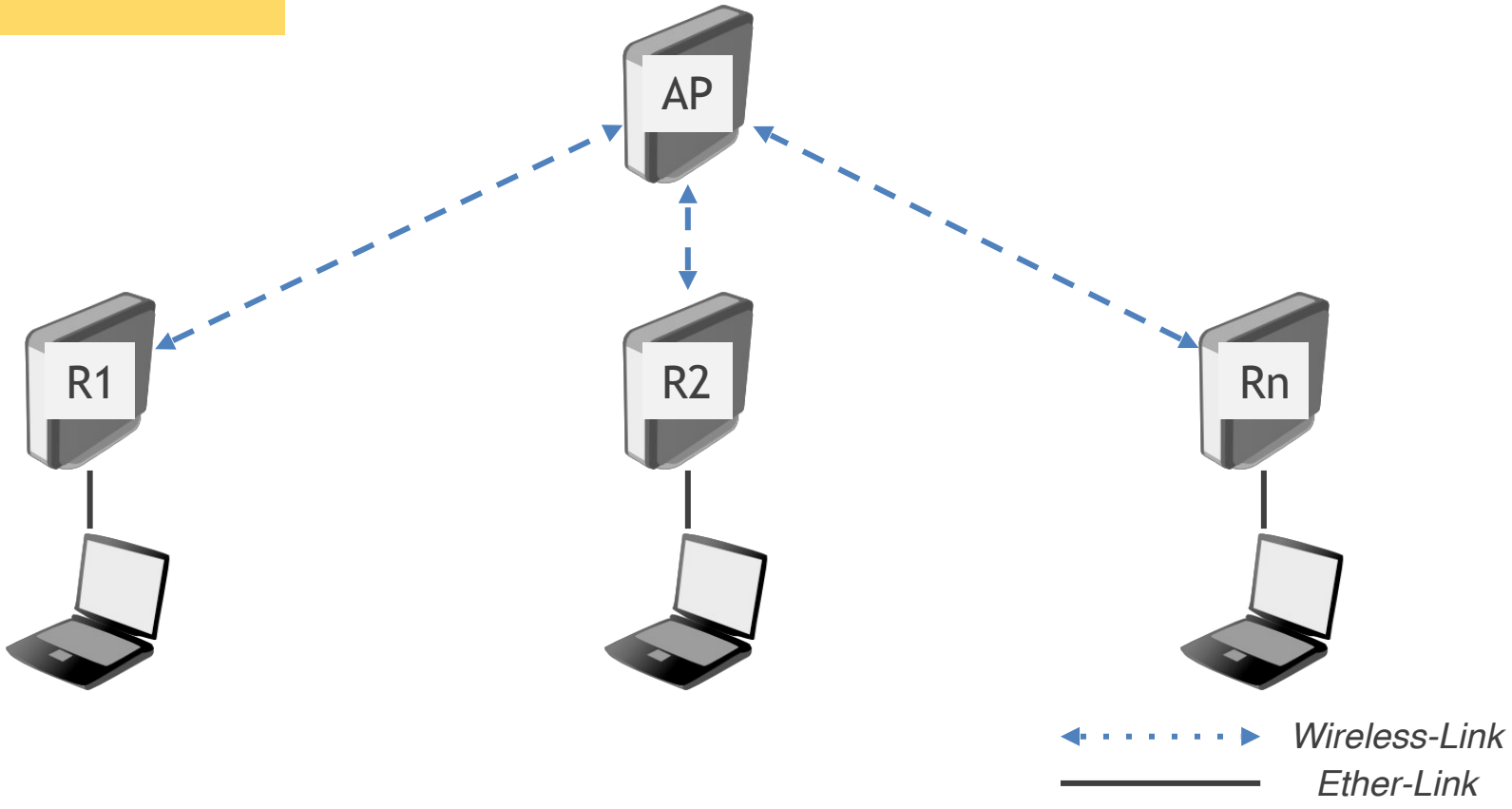


Lab Setup

SSID : .@ **VRPro Training**

BAND : **2.4 / 5 Ghz**

KEY : **vrproservice**



SECURITY INTRO

What Security is all about?

- Security is about protection of assets.
  - *D. Gollmann, Computer Security, Wiley*
- **Confidentiality** : Protecting personal privacy and proprietary information.
- **Integrity** : Ensuring information non-repudiation and authenticity.
- **Availability** : Ensuring timely and reliable access to and use of information

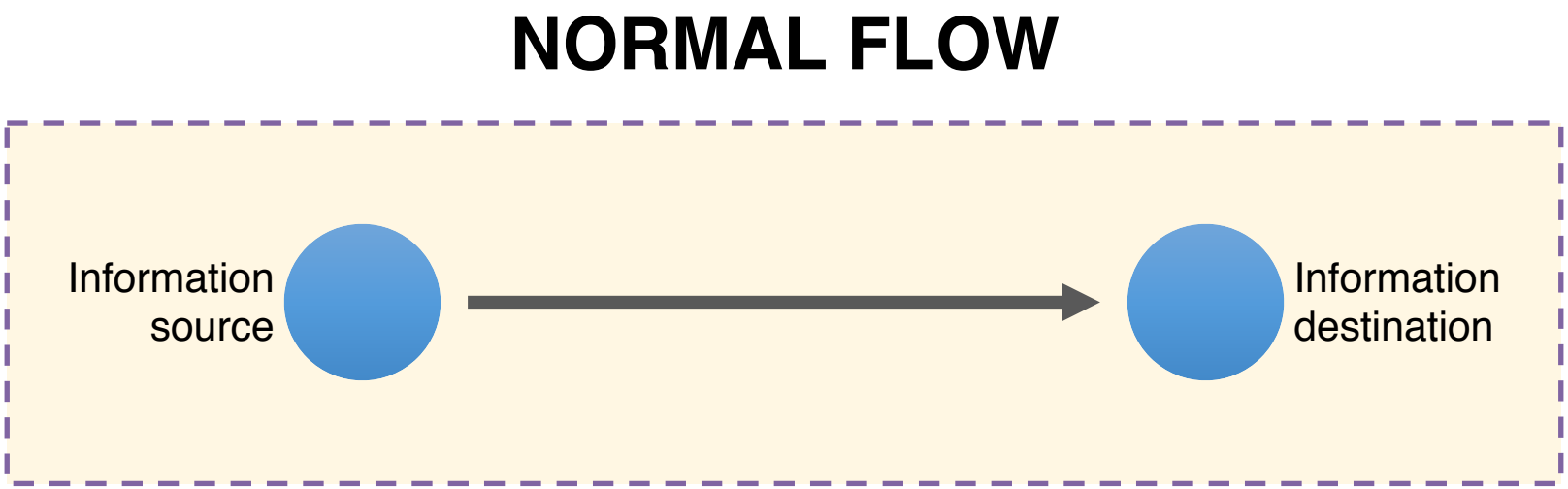
What Security is all about?

- **Prevention** : take measures that prevent your assets from being damaged (or stolen)
- **Detection** : take measures so that you can detect when, how, and by whom an asset has been damaged
- **Reaction** : take measures so that you can recover your assets

Security Attacks, Mechanisms & Services

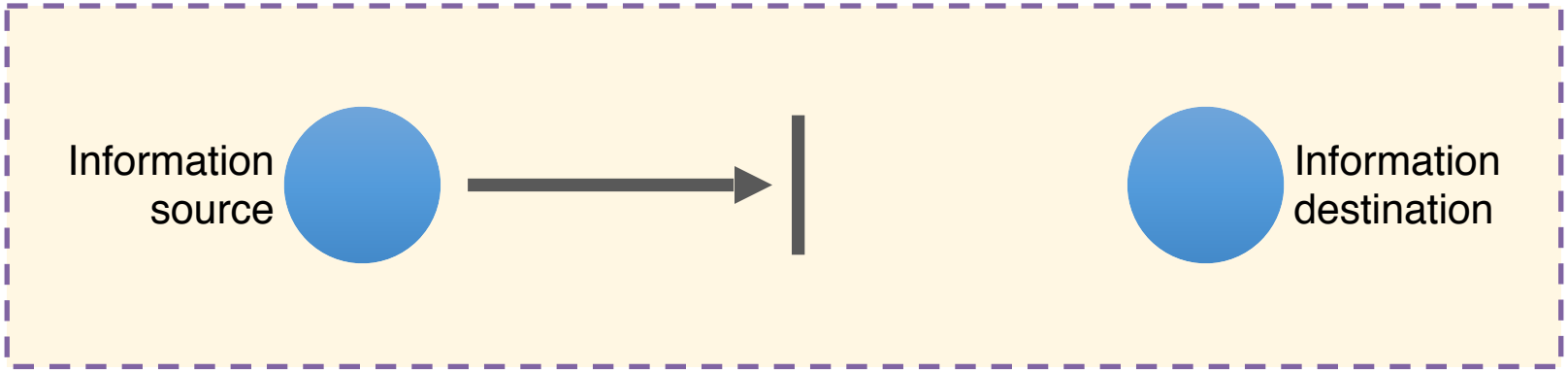
- **Security Attack** : Any action that compromises the security of information
- **Security Mechanism** : a process / device that is designed to detect, prevent or recover from a security attack.
- **Security Service** : a service intended to counter security attacks, typically by implementing one or more mechanisms.

Security Threats / Attacks



Security Threats / Attacks

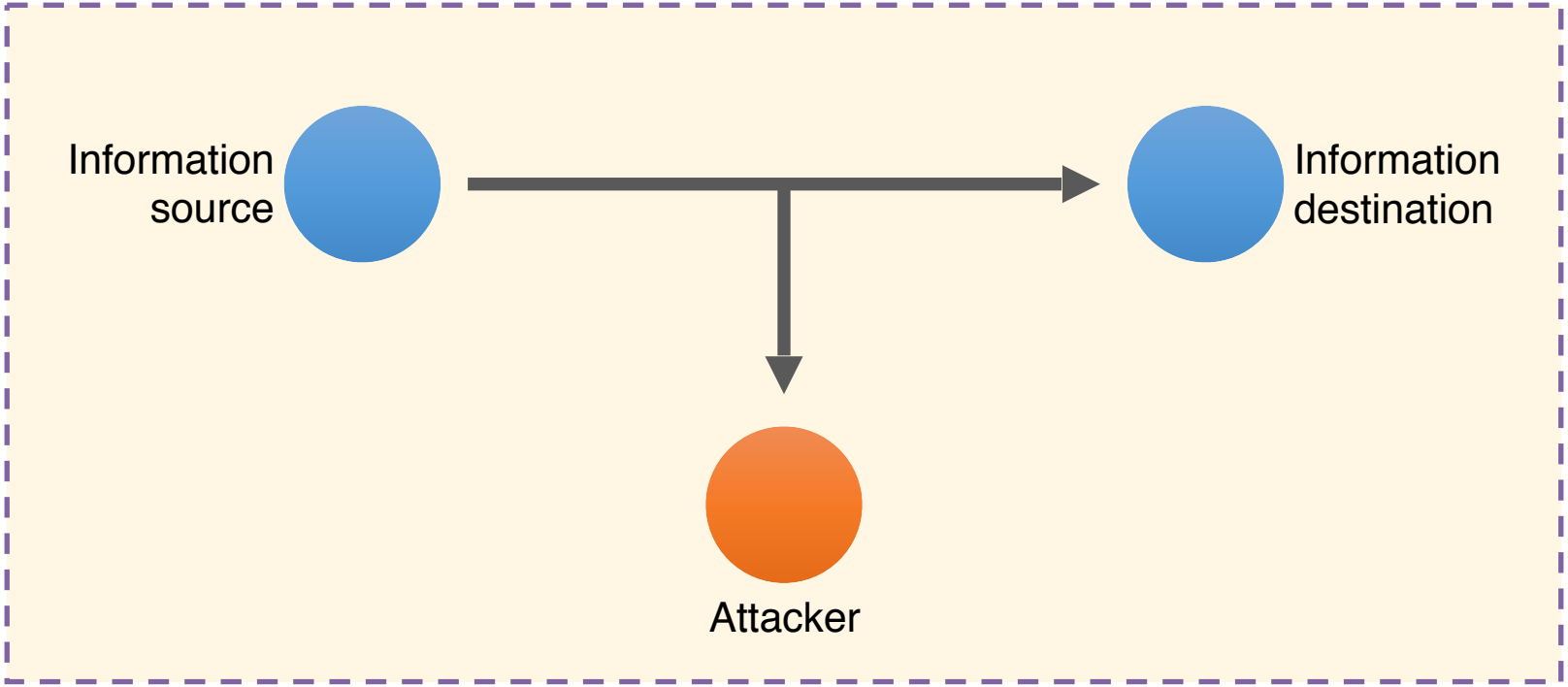
INTERRUPTION



“services or data become unavailable, unusable, destroyed, and so on, such as loss of file, denial of service, etc.”

Security Threats / Attacks

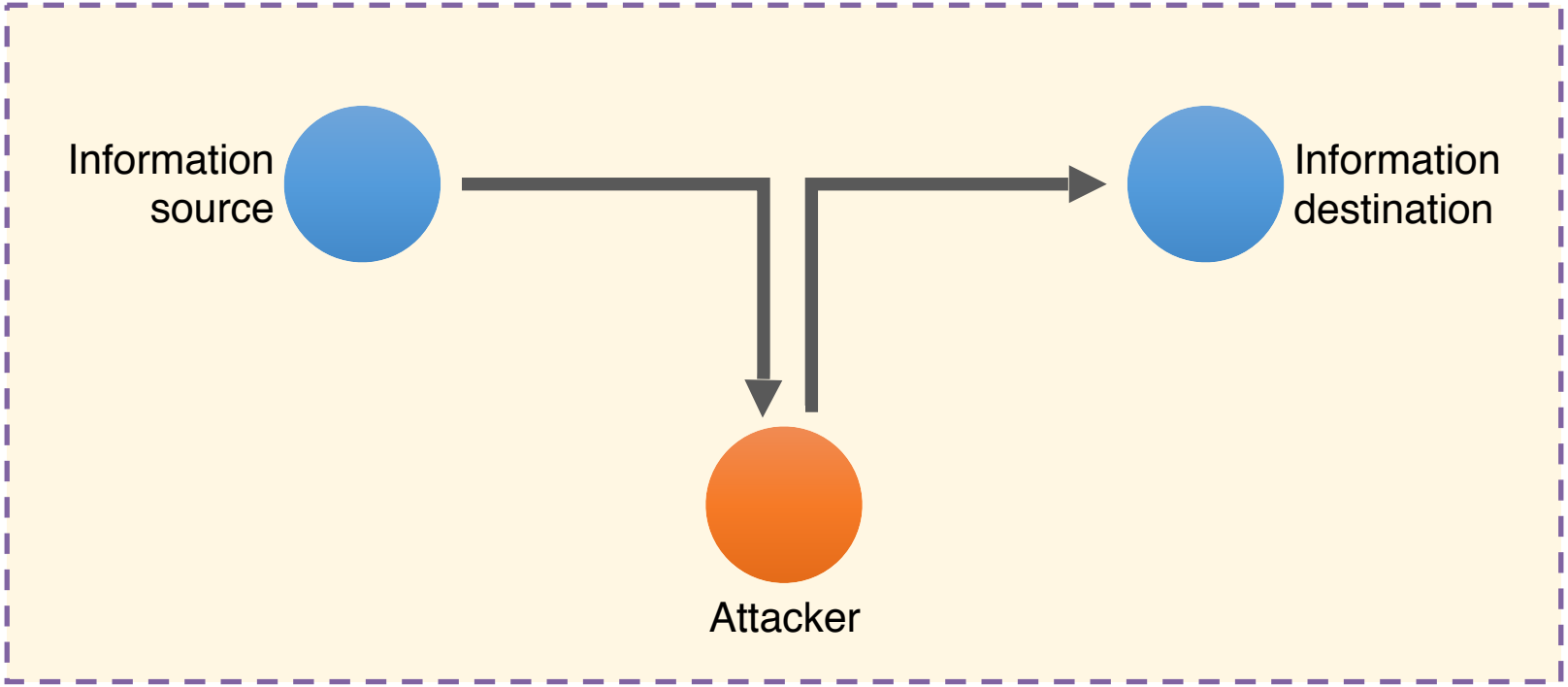
INTERCEPTION



"an unauthorized 3rd party has gained access to an object, such as stealing data, overhearing another's communication, etc."

Security Threats / Attacks

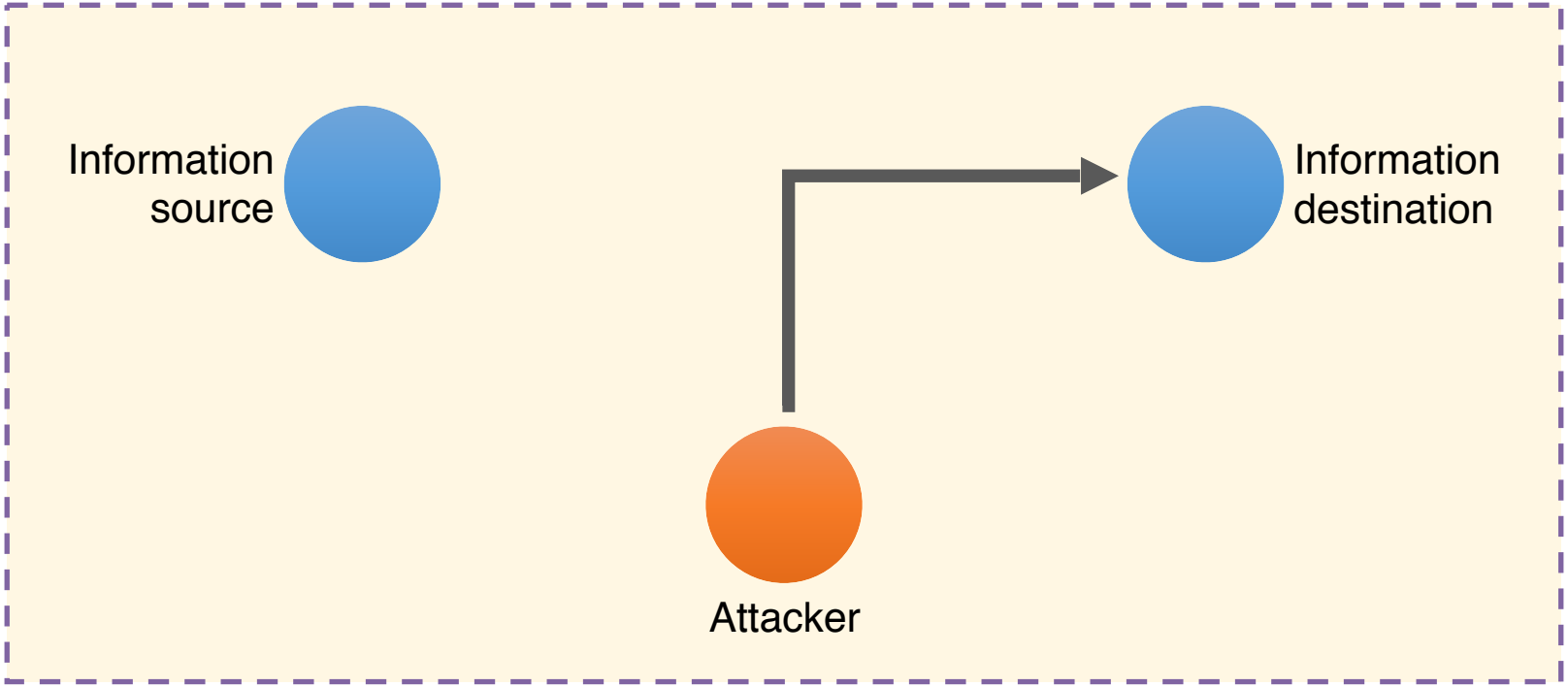
MODIFICATION



unauthorized changing of data or tampering with services, such as alteration of data, modification of messages, etc.

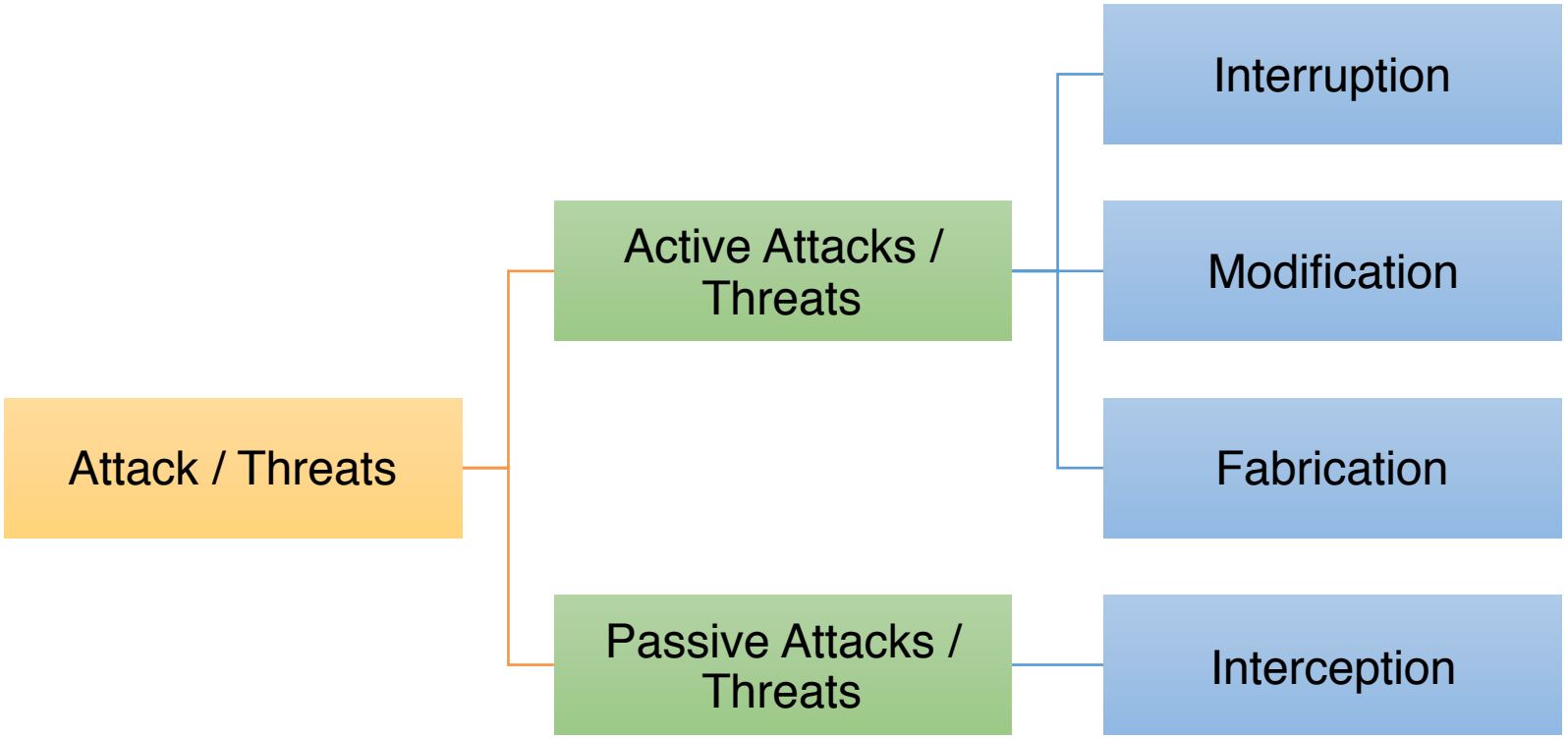
Security Threats / Attacks

FABRICATION



"additional data or activities are generated that would normally not exist, such as adding a password to a system, replaying previously sent messages, etc."

Type of Threats / Attacks



Security Mechanisms

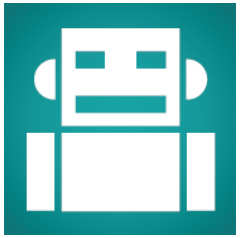
- **Encryption** : transforming data into something an attacker cannot understand, i.e., providing a means to implement confidentiality, as well as allowing the user to check whether data has been modified.
- **Authentication** : verifying the claimed identity of a user, such as user name, password, etc.
- **Authorization** : checking whether the user has the right to perform the action requested.
- **Auditing** : tracing which users accessed what, when, and which way. In general, auditing does not provide protection, but can be a tool for analysis of problems.

COMMON THREATS

Common Security Threats

Botnet

“Collection of software robots, or 'bots', that creates an army of infected computers (known as ‘zombies') that are remotely controlled by the originator”



What it can do :

- Send spam emails with viruses attached.
- Spread all types of malware.
- Can use your computer as part of a denial of service attack against other systems.

Common Security Threats

Distributed denial-of-service *(DDoS)*

"A distributed denial-of-service (DDoS) attack — or DDoS attack — is when a malicious user gets a network of zombie computers to sabotage a specific website or server."

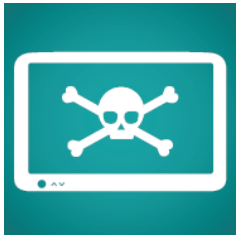
What it can do :

- The most common and obvious type of DDoS attack occurs when an attacker "floods" a network with useless information.
- The flood of incoming messages to the target system essentially forces it to shut down, thereby denying access to legitimate users.

Common Security Threats

Hacking

"Hacking is a term used to describe actions taken by someone to gain unauthorised access to a computer."



What it can do :

- Find weaknesses (or pre-existing bugs) in your security settings and exploit them in order to access your.
- Install a Trojan horse, providing a back door for hackers to enter and search for your information.

Common Security Threats

Malware

"Malware is one of the more common ways to infiltrate or damage your computer, it's software that infects your computer, such as computer viruses, worms, Trojan horses, spyware, and adware."

What it can do :

- Intimidate you with scareware, which is usually a pop-up message that tells you your computer has a security problem or other false information.
- Reformat the hard drive of your computer causing you to lose all your information.
- Alter or delete files.
- Steal sensitive information.
- Send emails on your behalf.
- Take control of your computer and all the software running on it.

Common Security Threats

Spam

"Spam is one of the more common methods of both sending information out and collecting it from unsuspecting people."

What it can do :

- Annoy you with unwanted junk mail.
- Create a burden for communications service providers and businesses to filter electronic messages.
- Phish for your information by tricking you into following links or entering details with too-good-to-be-true offers and promotions.
- Provide a vehicle for malware, scams, fraud and threats to your privacy.

Common Security Threats

Spoofing

"This technique is often used in conjunction with phishing in an attempt to steal your information."

What it can do :

- Sends spam using your email address, or a variation of your email address, to your contact list.
- Recreates websites that closely resemble the authentic site. This could be a financial institution or other site that requires login or other personal information.

Common Security Threats

Spyware & Adware

"This technique is often used by third parties to infiltrate your computer or steal your information without you knowing it."

What it can do :

- Collect information about you without you knowing about it and give it to third parties.
- Send your usernames, passwords, surfing habits, list of applications you've downloaded, settings, and even the version of your operating system to third parties.
- Change the way your computer runs without your knowledge.
- Take you to unwanted sites or inundate you with uncontrollable pop-up ads.

Common Security Threats

Trojan Horses

"A malicious program that is disguised as, or embedded within, legitimate software. It is an executable file that will install itself and run automatically once it's downloaded."

What it can do :

- Delete your files.
- Use your computer to hack other computers.
- Watch you through your web cam.
- Log your keystrokes (such as a credit card number you entered in an online purchase).
- Record usernames, passwords and other personal information.

Common Security Threats

Virus

"Malicious computer programs that are often sent as an email attachment or a download with the intent of infecting your computer."

What it can do :

- Send spam.
- Provide criminals with access to your computer and contact lists.
- Scan and find personal information like passwords on your computer.
- Hijack your web browser.
- Disable your security settings.
- Display unwanted ads.

Common Security Threats

Worm

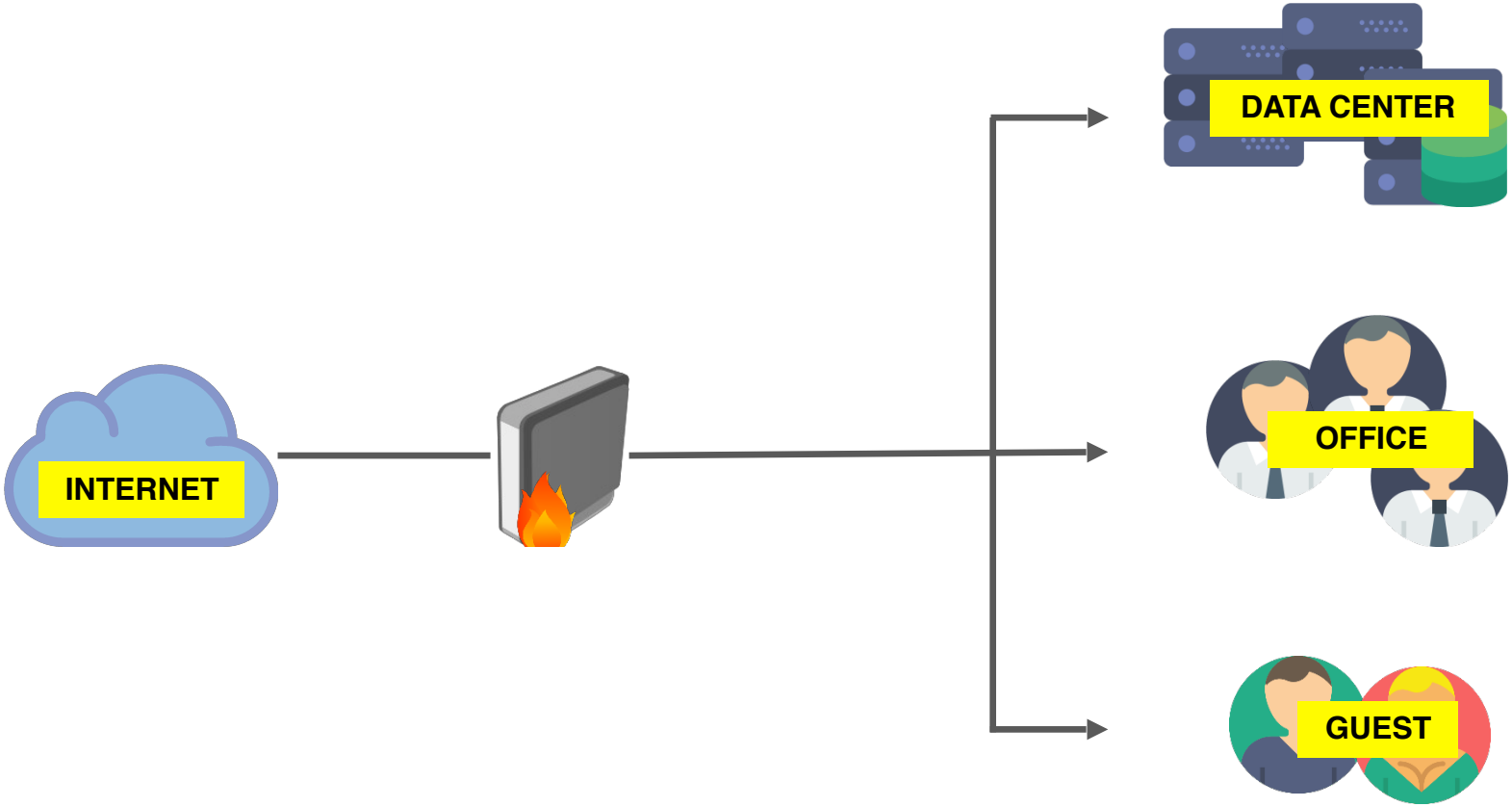
"A worm, unlike a virus, goes to work on its own without attaching itself to files or programs. It lives in your computer memory, doesn't damage or alter the hard drive and propagates by sending itself to other computers in a network."

What it can do :

- Spread to everyone in your contact list.
- Cause a tremendous amount of damage by shutting down parts of the Internet, wreaking havoc on an internal network and costing companies enormous amounts of lost revenue.

MIKROTIK SECURITY DEPLOYMENT

MikroTik as a Global Firewall Router



MikroTik as a Global Firewall Router

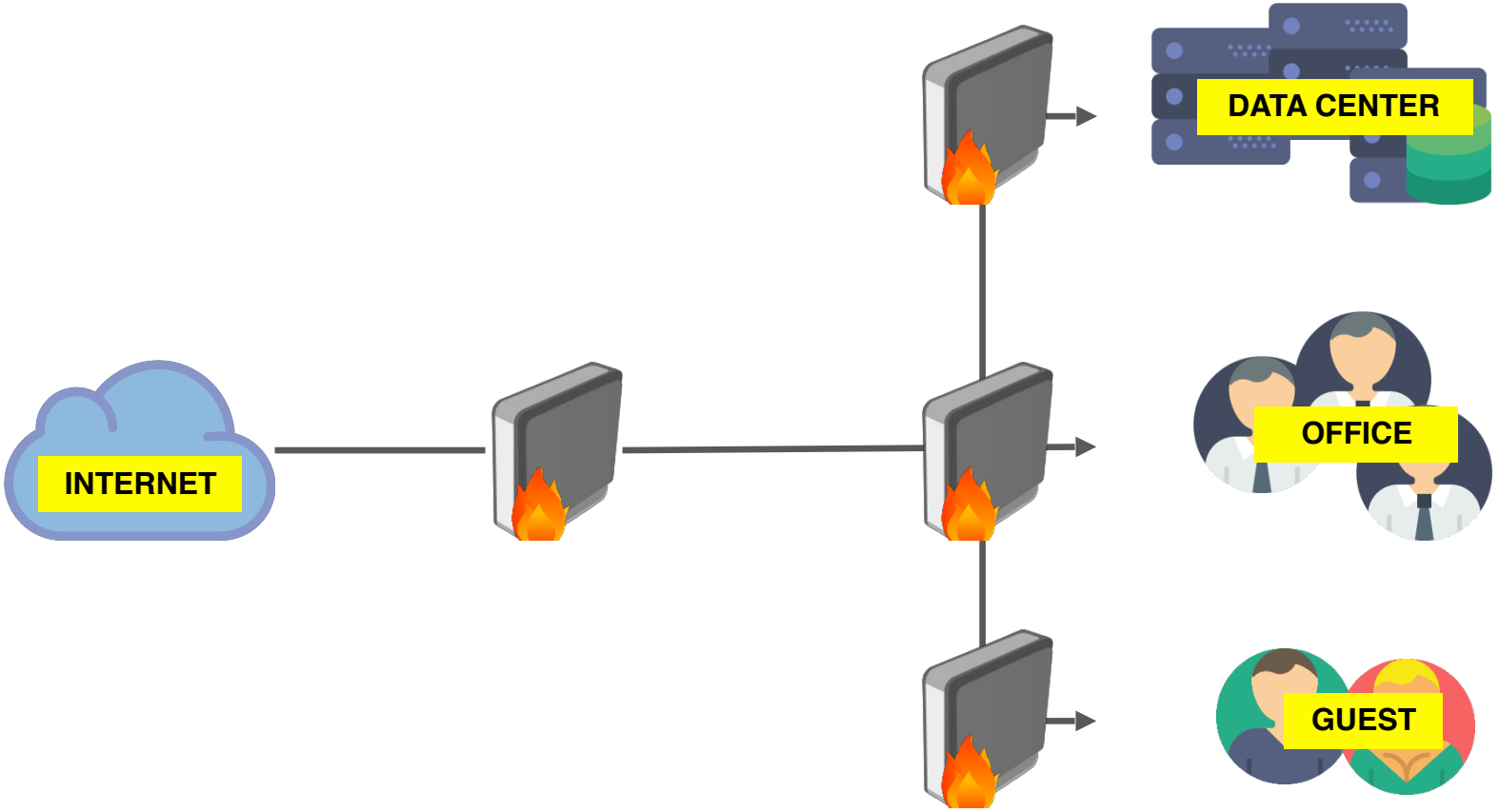
Pro's

- Simple topology
- Easy to manage

Con's

- Concentrate in one single-point-of-failure
- Demands high resources

MikroTik as a Specific Router Firewall



MikroTik as a Specific Router Firewall

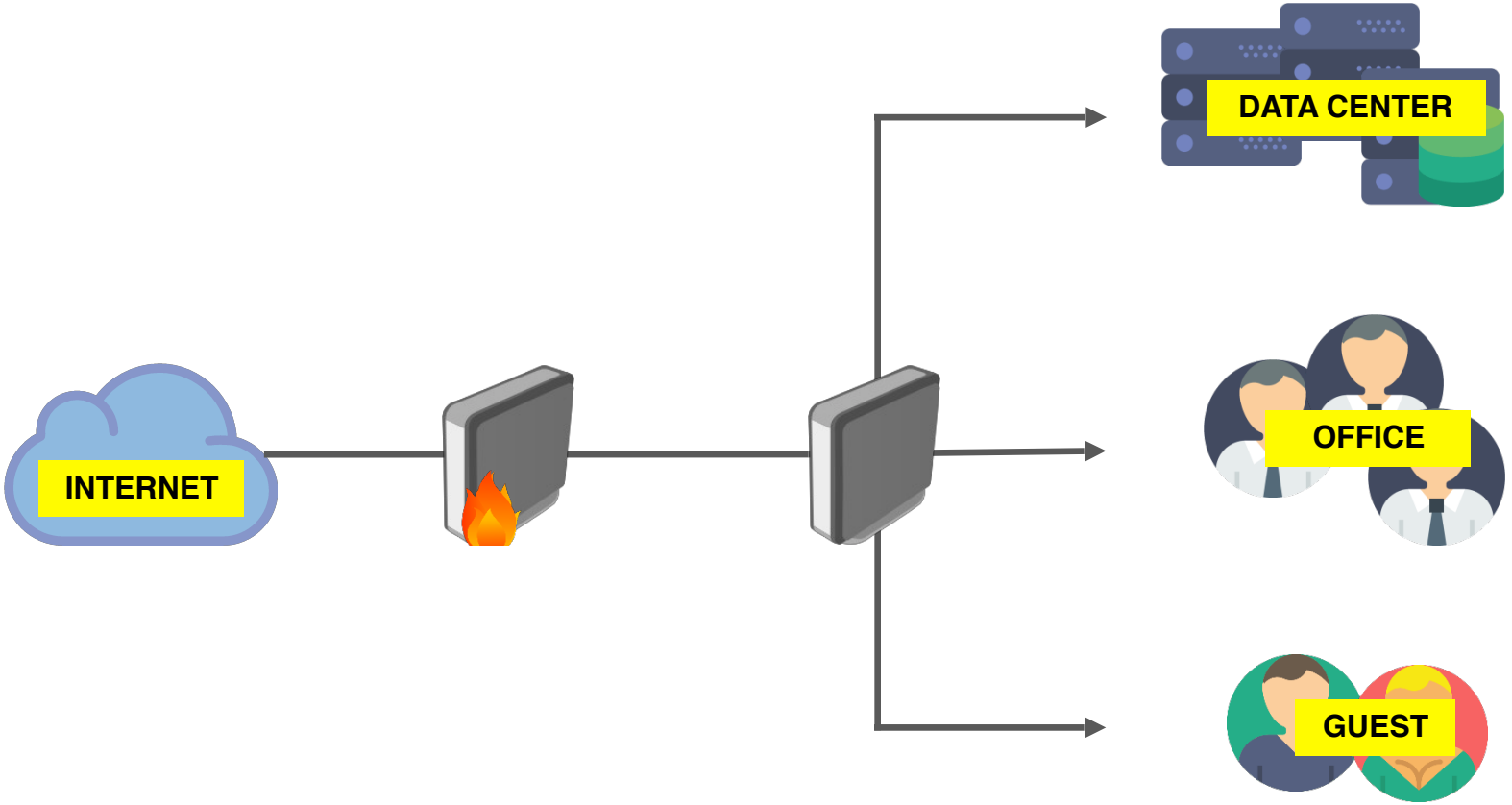
Pro's

- Less resource consumption on each router
- Only focusing security firewall on each network

Con's

- Different network segment, different treatment
- Need to configure firewall differently on each router
- Possible to configure double firewall rules on one another's routers

MikroTik as an IPS



MikroTik as an IPS

Pros

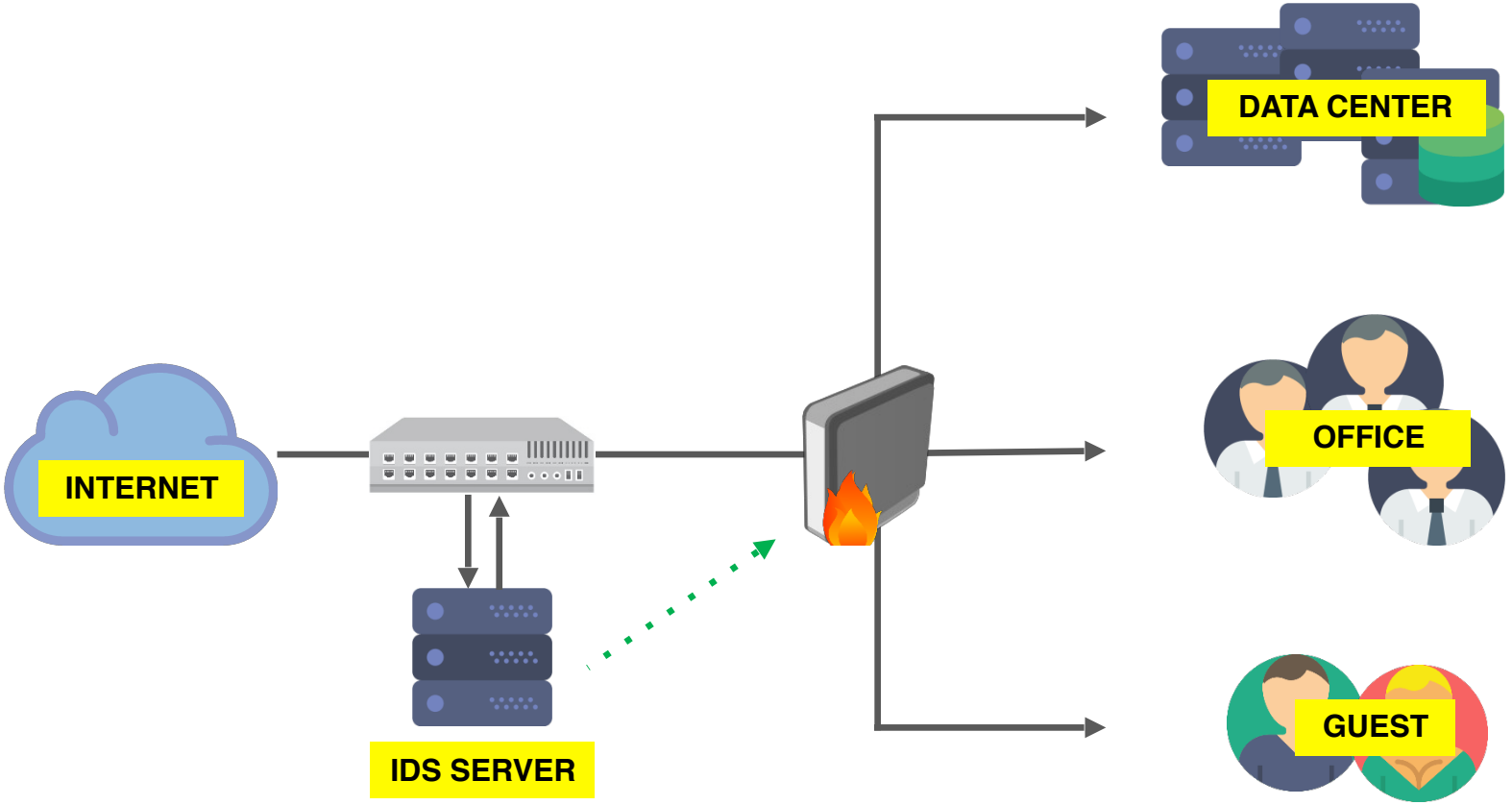
- Clean firewall configuration on router, because all firewall configuration already defined on an IPS** router

Cons

- Need high resource Device on MikroTik as an IPS

**Intrusion Prevention System ??

MikroTik with IDS as a trigger



MikroTik with IDS as a trigger

Pro's

- All firewall rules are made automatically by API from IDS server

Con's

- Need additional device for triggering by bad traffic
- Need powerful device for mirroring all traffic from networks
- Need special scripting for sending information to router
- Expensive

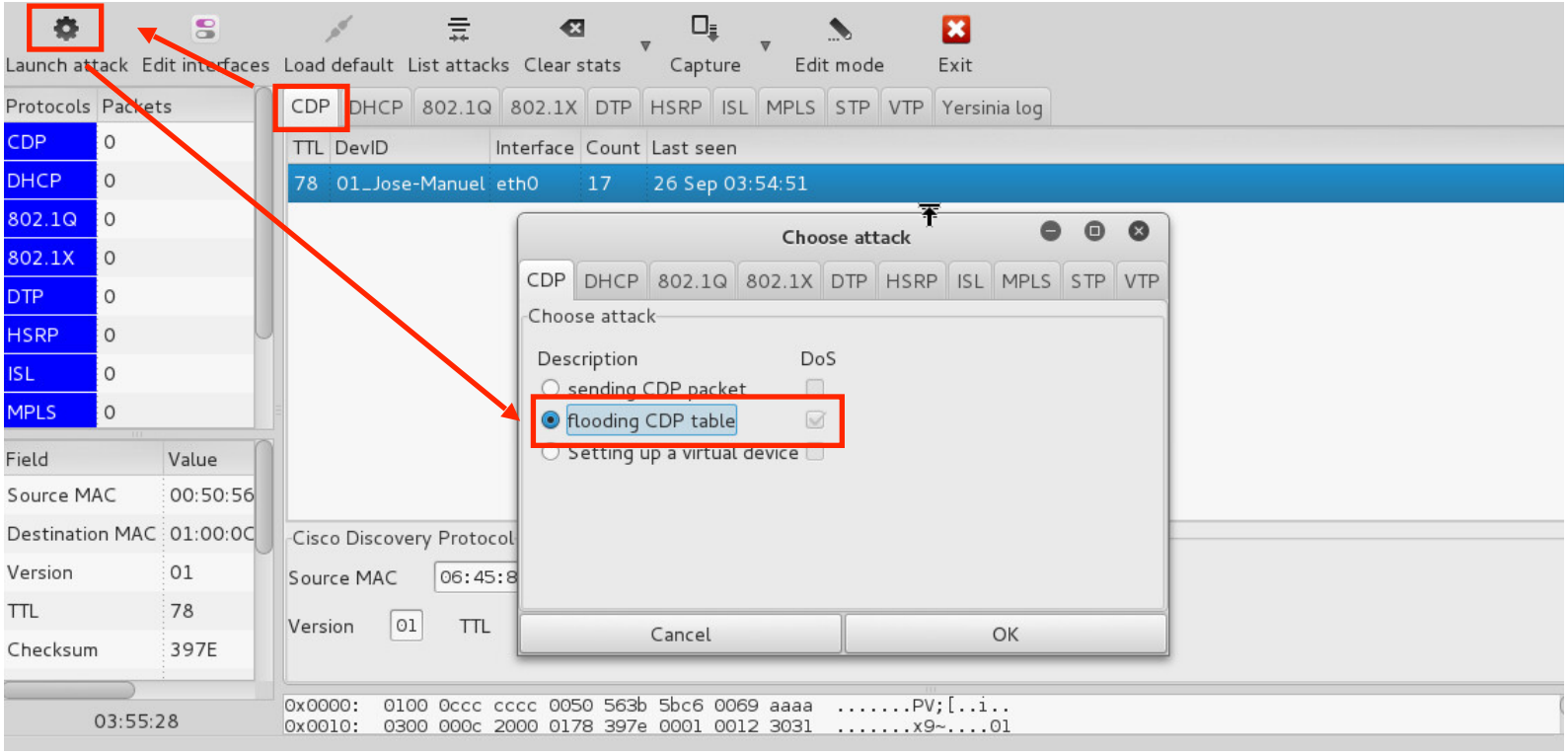
OSI LAYER ATTACKS

MikroTik Neighbor Discovery Protocol

- MikroTik Neighbor Discovery protocol (MNDP) allows to "find" other devices compatible with MNDP or CDP (Cisco Discovery Protocol) or LLDP in Layer2 broadcast domain.
- works on interfaces that support IP protocol and have at least one IP address and on all ethernet-like interfaces even without IP addresses
- is enabled by default for all new Ethernet-like interfaces
- uses UDP protocol port 5678

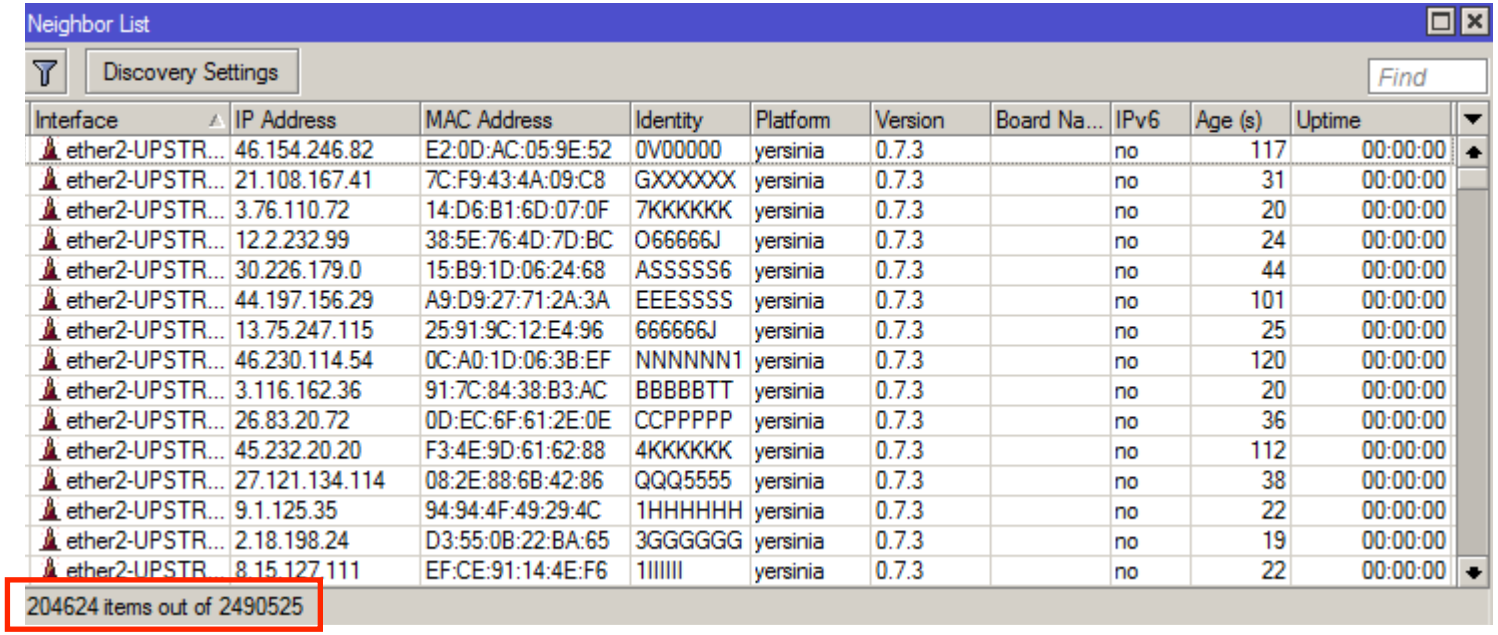
MNDP Attack

- This tool will be sending a lot of “fake” CDP neighbors to the RouterOS device.



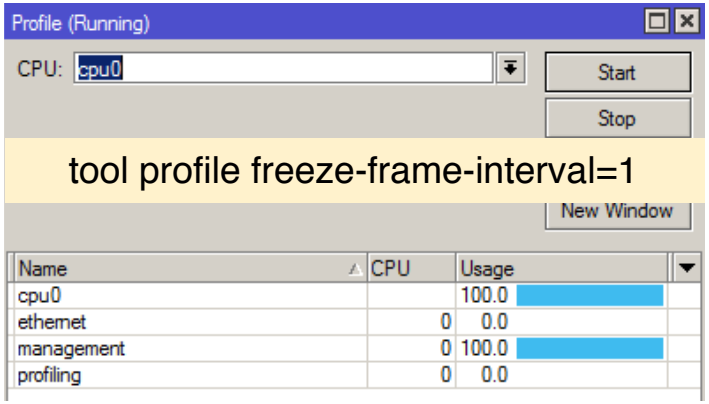
MNDP Attack

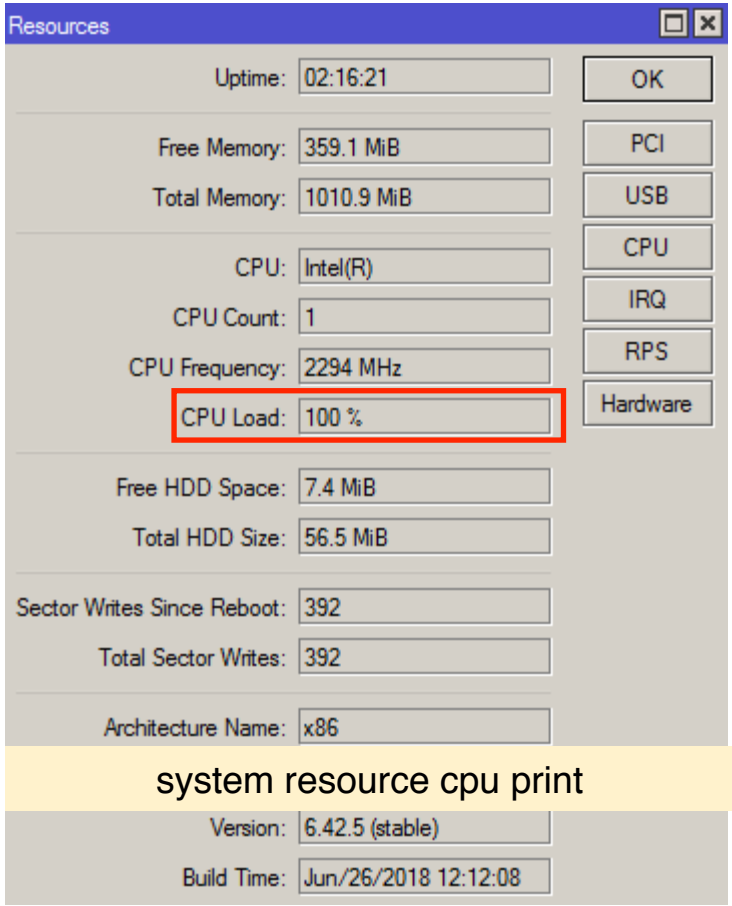
- RouterOS is receiving information about thousands of “fake” neighbor devices.



MNDP Attack

- It's exhausting the resources of the router and impacting the performance



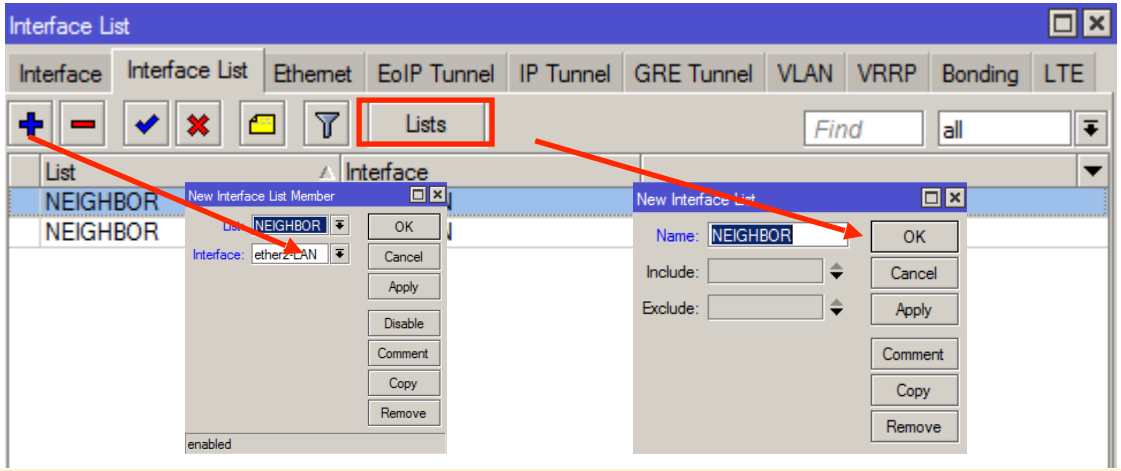


Preventing MNDP Attacks

- To prevent such attacks we must select which interfaces can communicate using MNDP/CDP/LLDP.
- Creating “interface-list” and selecting which interfaces to enable neighbor discovery on (MNDP)

MNDP Attack

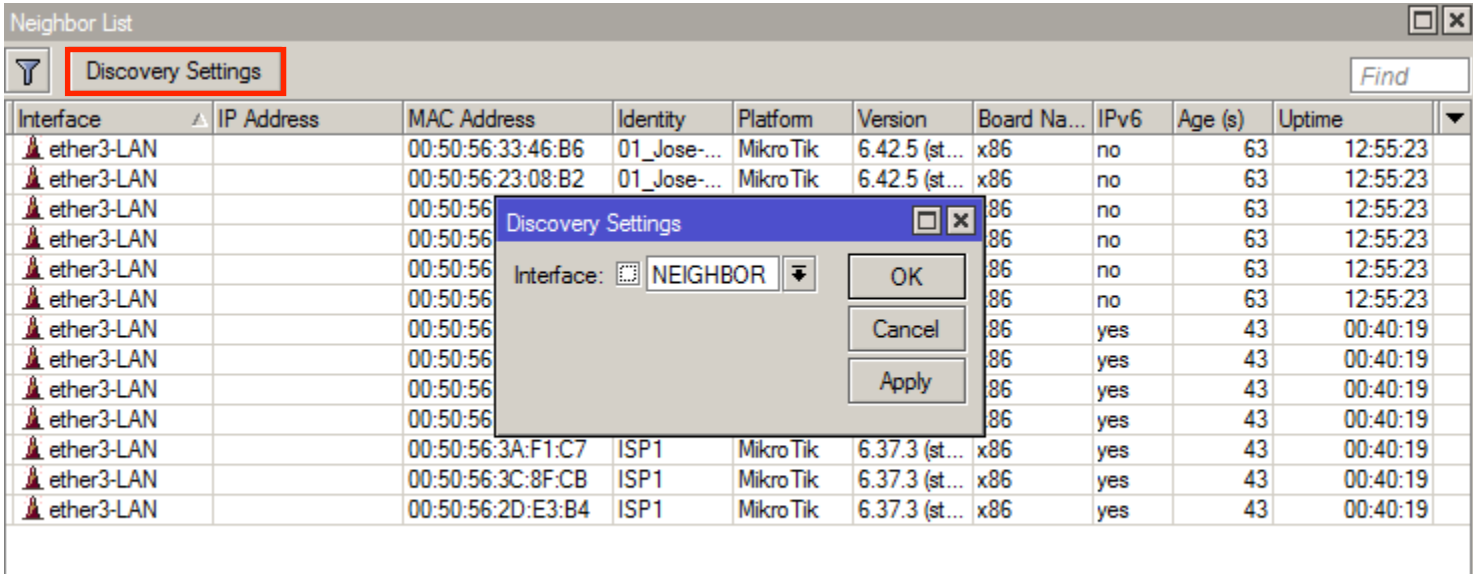
- Creating “interface-list” for accessing MikroTik Neighbor Discovery Protocol




```
/interface list add name=NEIGHBOR
/interface list member
add interface=etherX list=NEIGHBOR
add interface=etherY list=NEIGHBOR
```

MNDP Attack

- IP > Neighbors and set Discovery Settings to previous “interface-list been made.



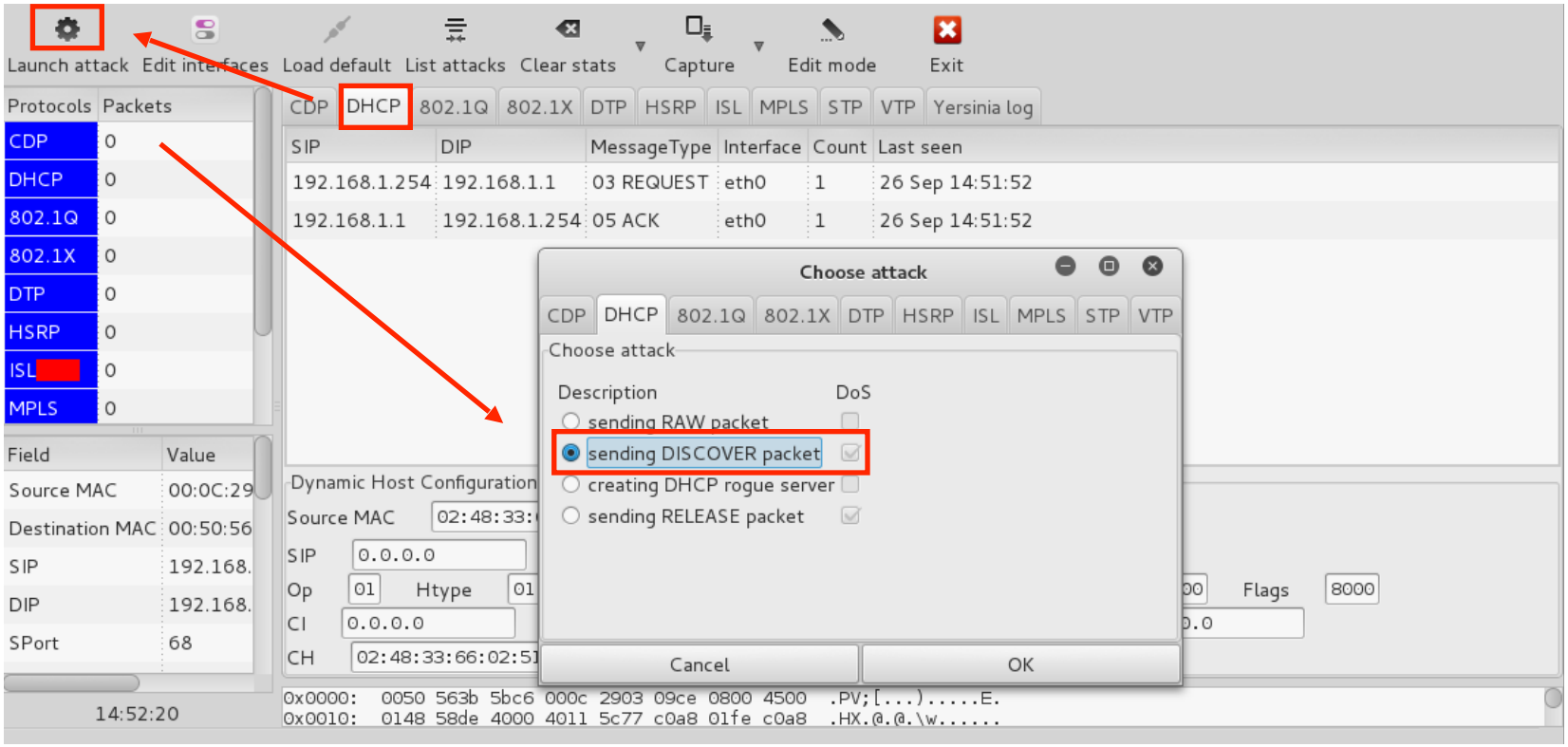
/ip neighbor discovery-settings set discover-interface-list=NEIGHBOR

DHCP Starvation Attack

- An attack that works by broadcasting DHCP requests with spoofed MAC addresses.
- DHCP starvation attack targets DHCP servers whereby forged DHCP requests are crafted by an attacker with the intent of exhausting all available IP addresses that can be allocated by the DHCP server

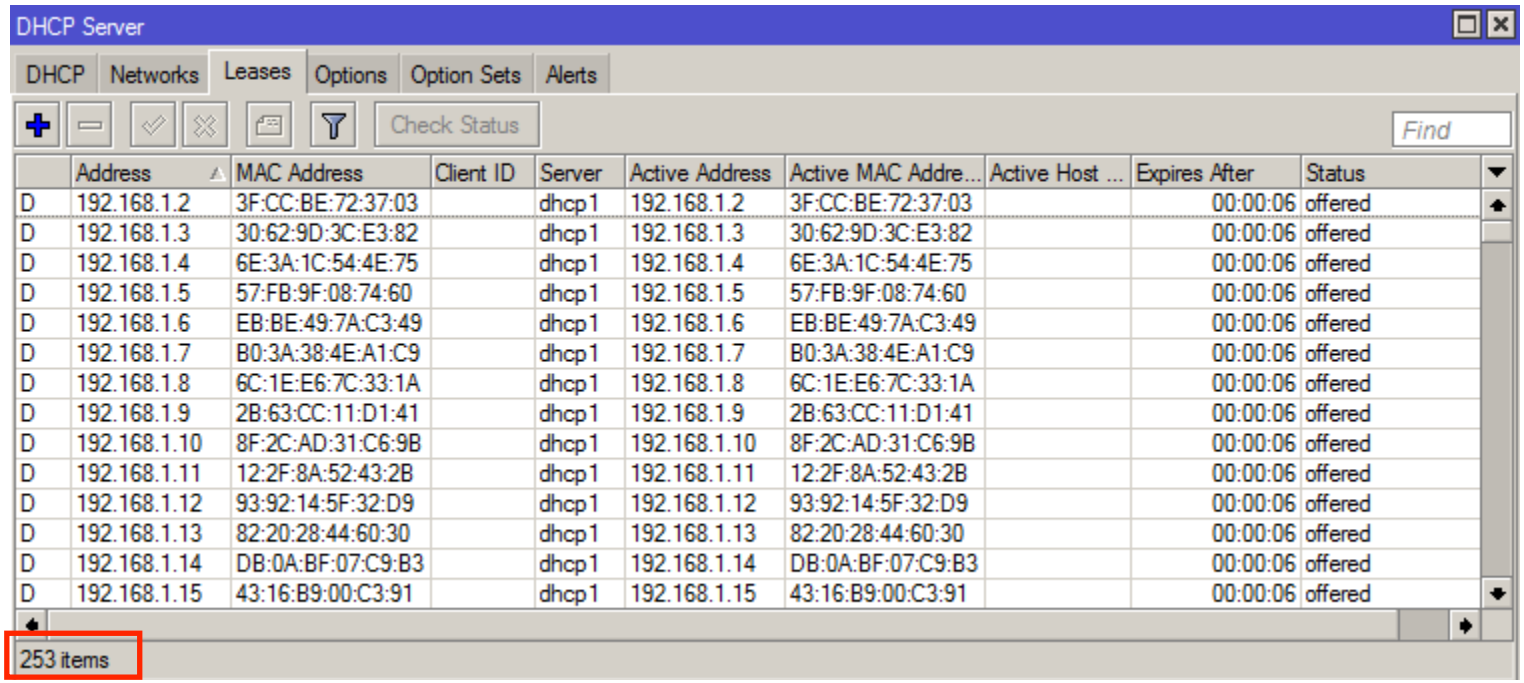
DHCP Starvation Attack

- This tool sends multiple “fake” DHCP requests to the router



DHCP Starvation Attack

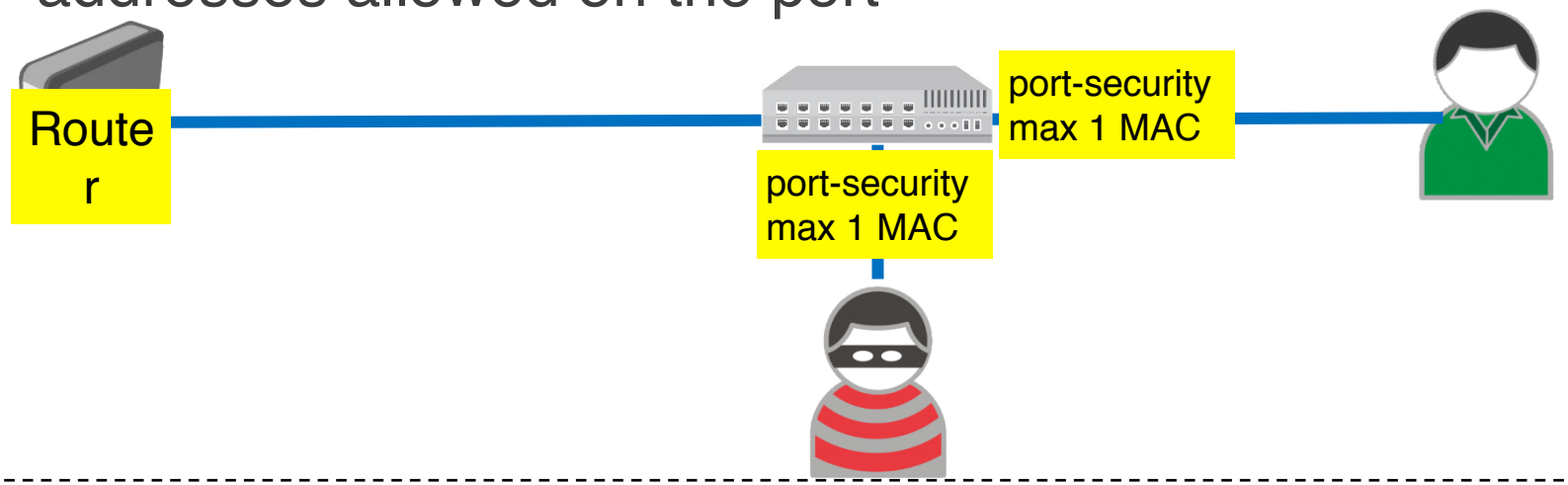
- Attacker exhausts DHCP leases with multiple dhcp-requests to the router.



	Address	MAC Address	Client ID	Server	Active Address	Active MAC Addre...	Active Host ...	Expires After	Status
D	192.168.1.2	3F:CC:BE:72:37:03		dhcp1	192.168.1.2	3F:CC:BE:72:37:03		00:00:06	offered
D	192.168.1.3	30:62:9D:3C:E3:82		dhcp1	192.168.1.3	30:62:9D:3C:E3:82		00:00:06	offered
D	192.168.1.4	6E:3A:1C:54:4E:75		dhcp1	192.168.1.4	6E:3A:1C:54:4E:75		00:00:06	offered
D	192.168.1.5	57:FB:9F:08:74:60		dhcp1	192.168.1.5	57:FB:9F:08:74:60		00:00:06	offered
D	192.168.1.6	EB:BE:49:7A:C3:49		dhcp1	192.168.1.6	EB:BE:49:7A:C3:49		00:00:06	offered
D	192.168.1.7	B0:3A:38:4E:A1:C9		dhcp1	192.168.1.7	B0:3A:38:4E:A1:C9		00:00:06	offered
D	192.168.1.8	6C:1E:E6:7C:33:1A		dhcp1	192.168.1.8	6C:1E:E6:7C:33:1A		00:00:06	offered
D	192.168.1.9	2B:63:CC:11:D1:41		dhcp1	192.168.1.9	2B:63:CC:11:D1:41		00:00:06	offered
D	192.168.1.10	8F:2C:AD:31:C6:9B		dhcp1	192.168.1.10	8F:2C:AD:31:C6:9B		00:00:06	offered
D	192.168.1.11	12:2F:8A:52:43:2B		dhcp1	192.168.1.11	12:2F:8A:52:43:2B		00:00:06	offered
D	192.168.1.12	93:92:14:5F:32:D9		dhcp1	192.168.1.12	93:92:14:5F:32:D9		00:00:06	offered
D	192.168.1.13	82:20:28:44:60:30		dhcp1	192.168.1.13	82:20:28:44:60:30		00:00:06	offered
D	192.168.1.14	DB:0A:BF:07:C9:B3		dhcp1	192.168.1.14	DB:0A:BF:07:C9:B3		00:00:06	offered
D	192.168.1.15	43:16:B9:00:C3:91		dhcp1	192.168.1.15	43:16:B9:00:C3:91		00:00:06	offered

Preventing DHCP Starvation Attacks

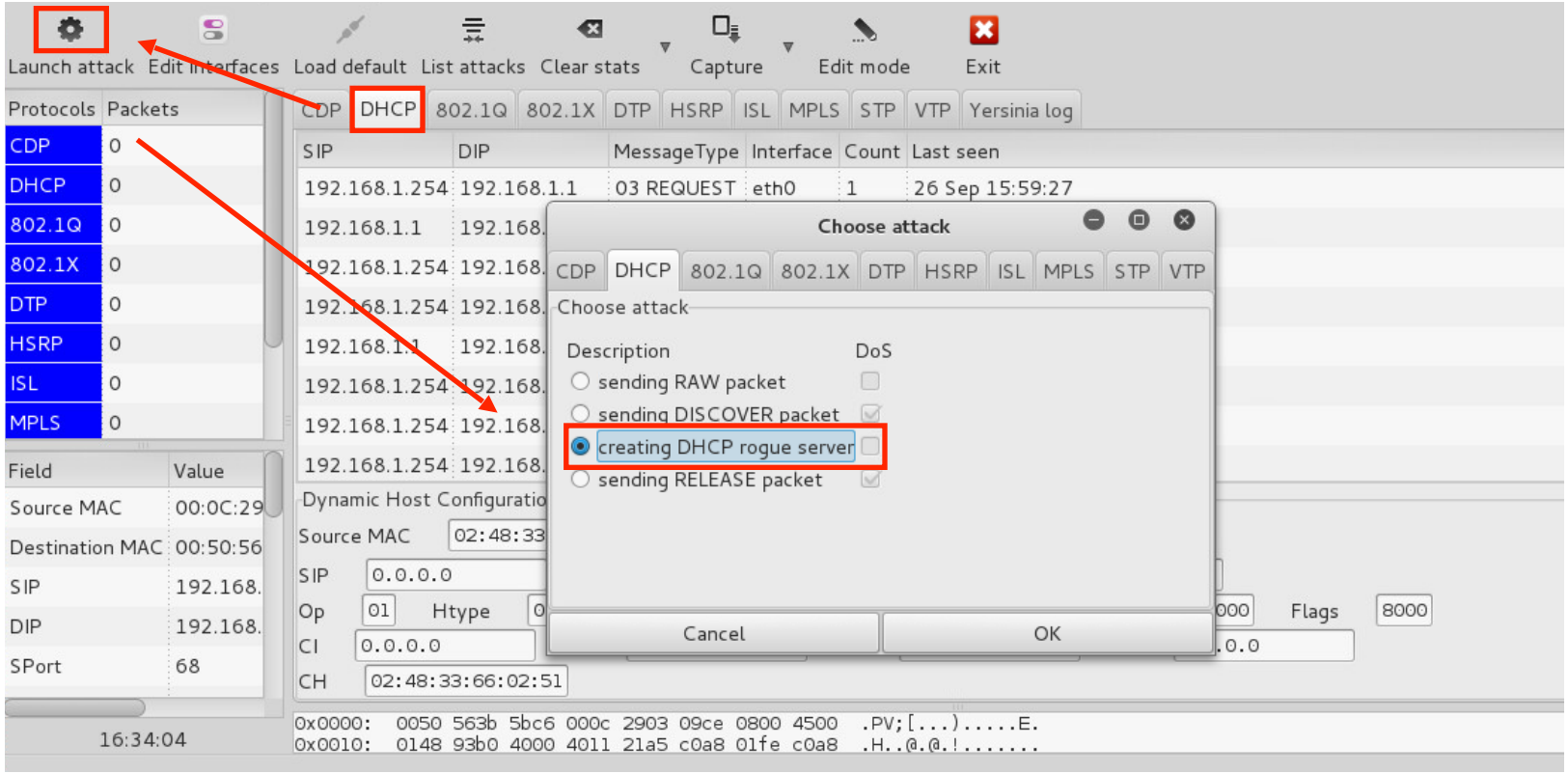
- Attacker uses a new MAC address to request a new DHCP lease
- Restrict the number of MAC addresses on the port of switch.
- Will not be able to lease more IP addresses than MAC addresses allowed on the port



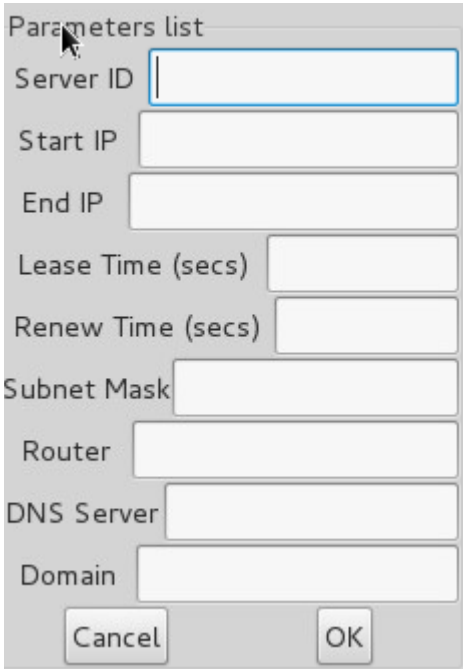
Rogue DHCP server

- A rogue DHCP server is a DHCP server on a network which is not under the administrative control.
- It is set up on a network by an attacker, for taking advantage from clients.

Rogue DHCP server



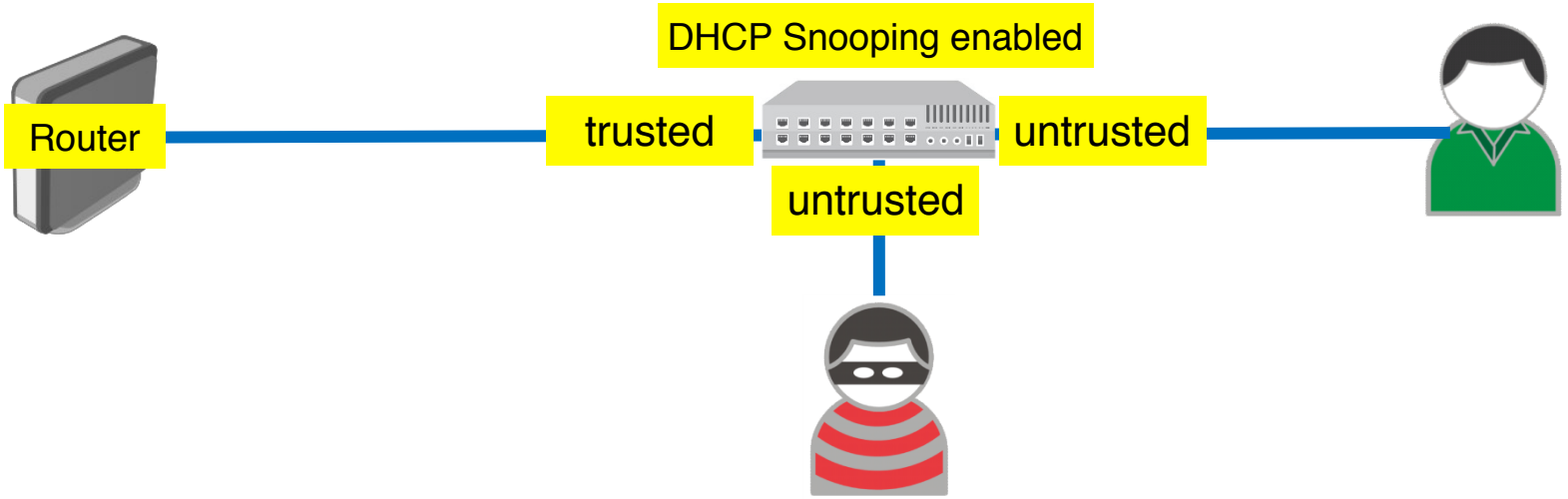
Rogue DHCP server



- **Server IP** – the IP server, the name of which will send the answer the DHCP *(xxx.xxx.xxx.xxx);*
- **Start IP** – initialIP, , issued to customers -address address range *(xxx.xxx.xxx.xxx);*
- **End IP** – IP , issued to customers -address address range *(xxx.xxx.xxx.xxx);*
- **Time The Lease (secs)** – The time in seconds for which the address is given
- **Time The Renew (secs)** – The time in seconds how many clients must renew the address lease
- **Subnet Mask** – Subnet mask for the clients *(xxx.xxx.xxx.xxx);*
- **Router** – router address issued to clients *(xxx.xxx.xxx.xxx ,the address of a fake router);*
- **DNS Server** – DNS server provided to clients *(xxx.xxx.xxx.xxx ,the address of a fake DNS server);*
- **The Domain** – a domain name in the local area network *(abc.def);*

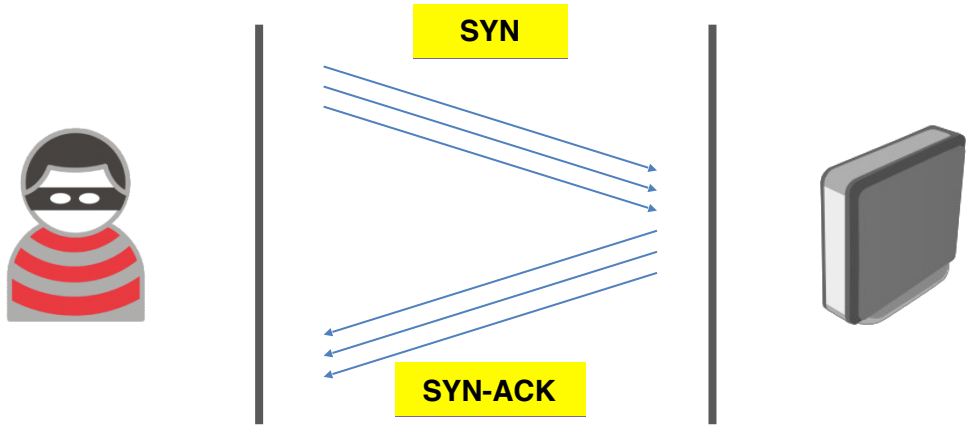
Preventing Rogue DHCP

- Enable DHCP Snooping on the switch
- Make port facing router as DHCP Snooping Trusted
- Binding Address and MAC for known clients
- RouterOS DHCP alert is ONLY sending information, not stopping or preventing an attack.



https://wiki.mikrotik.com/wiki/Manual:Interface/Bridge#DHCP_Snooping_and_DHCP_Option_82

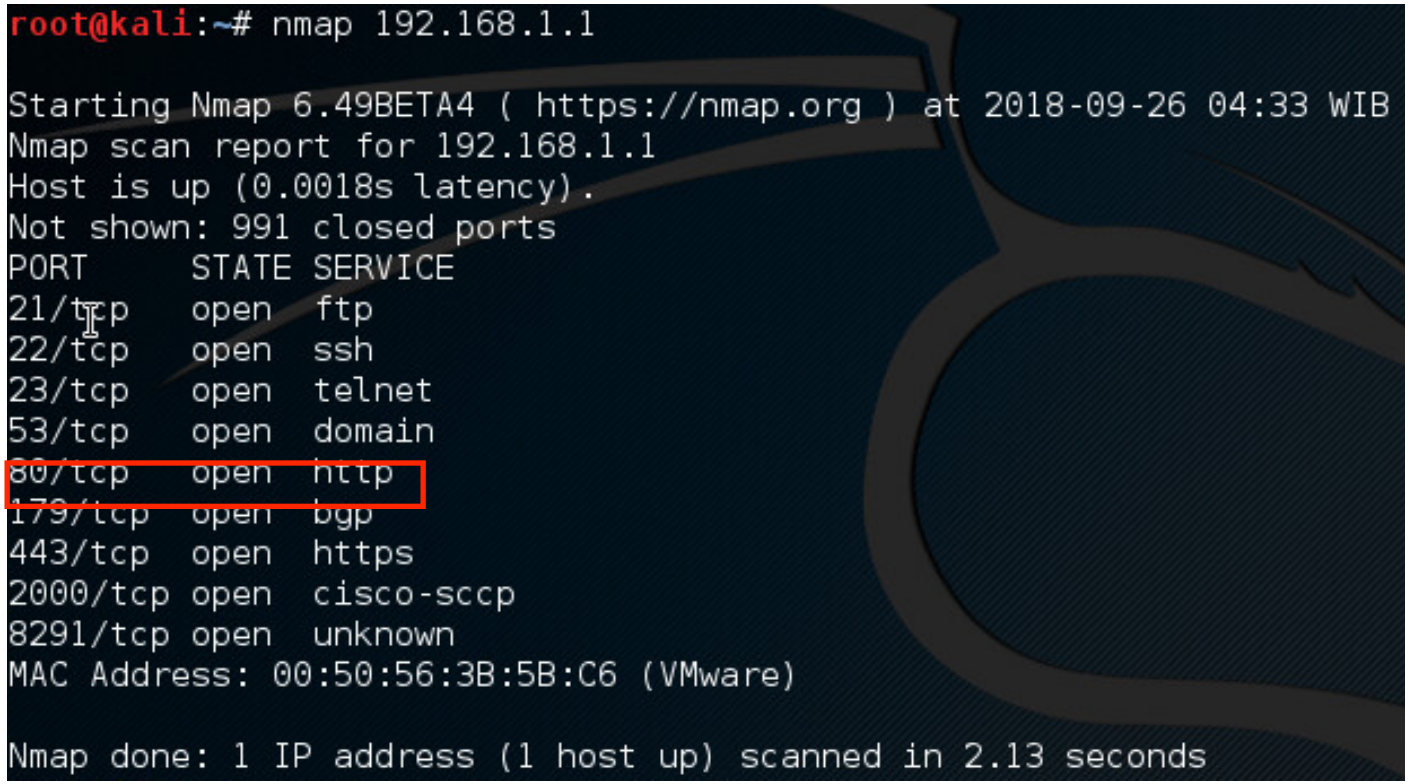
TCP SYN Attack



- This type of attack takes advantage of the three-way handshake to establish communication
- In SYN flooding, the attacker send the target a large number of TCP/SYN packets.
- These packets have a source address, and the target computer replies (TCP/SYN-ACK packet) back to the source IP, trying to establish a TCP connection

TCP SYN Attack

- Scanning available port on target, normal target usually port 80/http service



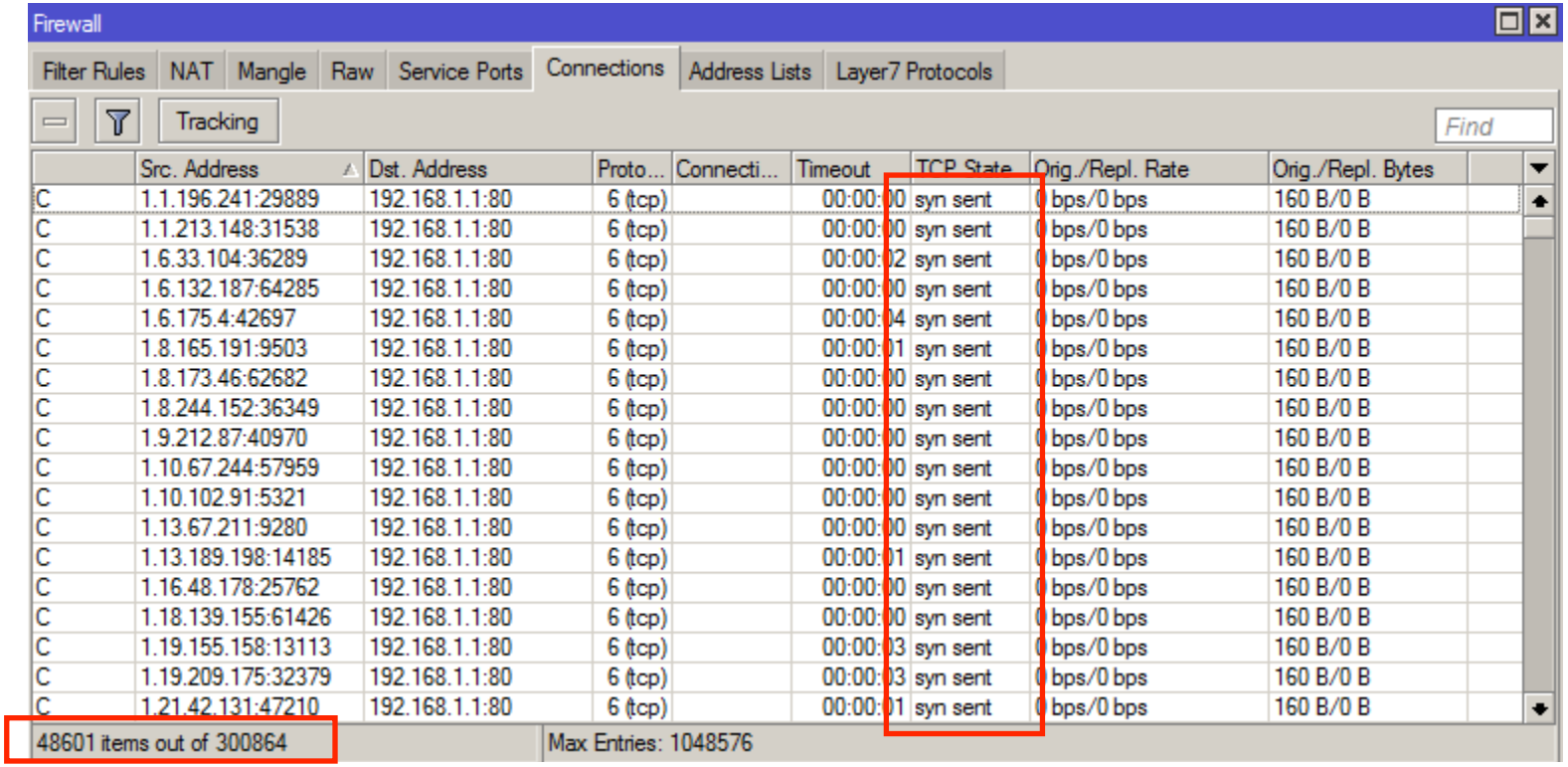
TCP SYN Attack

- Download and install “hping3” and run command bellow

```
root@kali:~# hping3 -c 20000 -d 120 -S -w 64 -p 80 --flood --rand-source 192.168.1.1
HPING 192.168.1.1 (eth0 192.168.1.1): S set, 40 headers + 120 data bytes
hping in flood mode, no replies will be shown
```

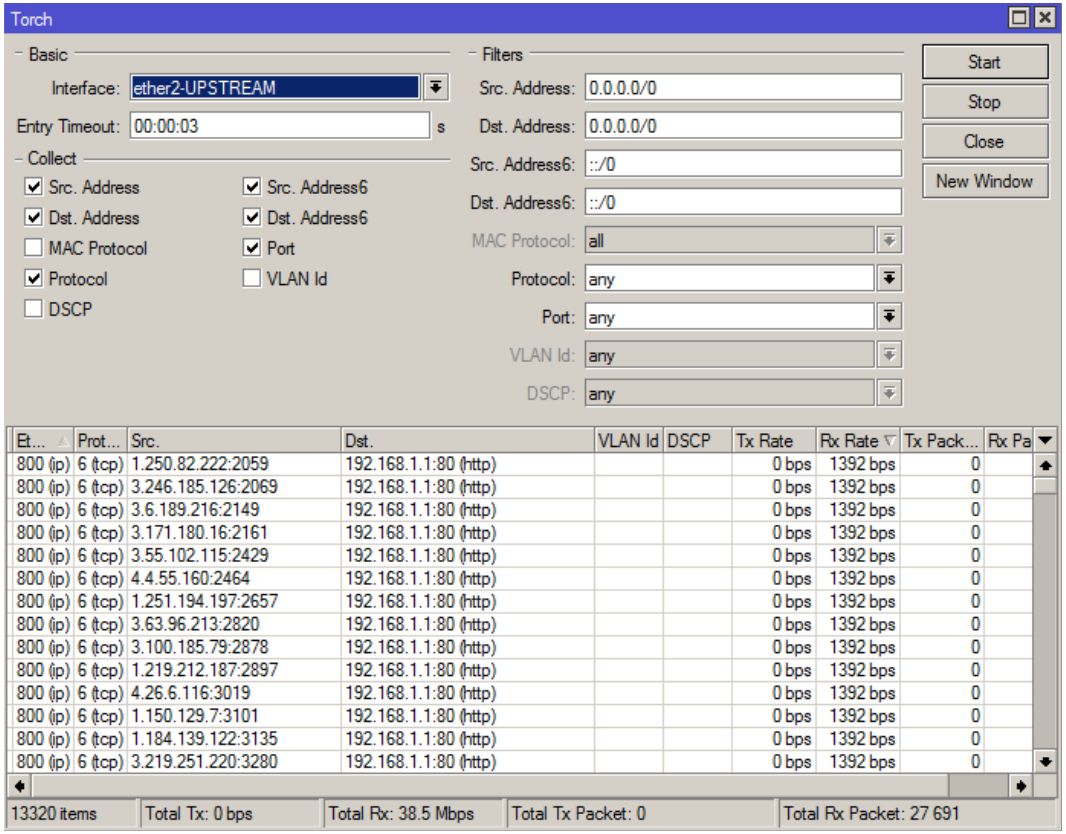
TCP SYN Attack

- “IP > Firewall > Connections” please observe the “syn sent” from random source addresses



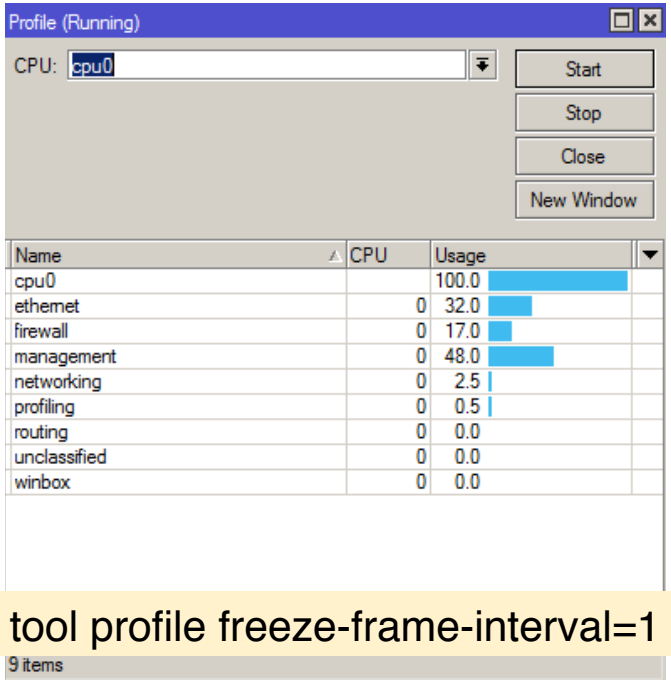
TCP SYN Attack

- Torch interface traffic

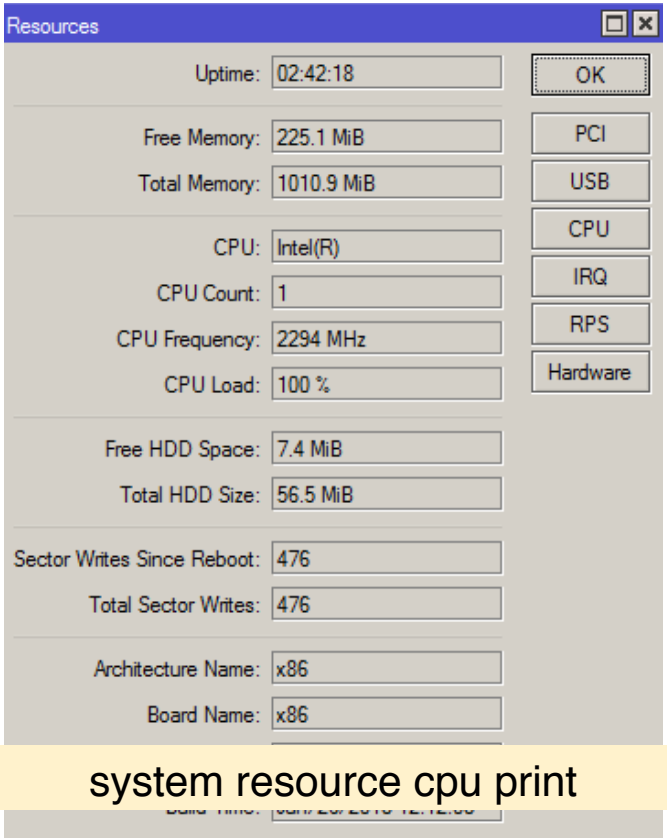


TCP SYN Attack

- The attack is exhausting the resources of the router and impacting the performance



tool profile freeze-frame-interval=1



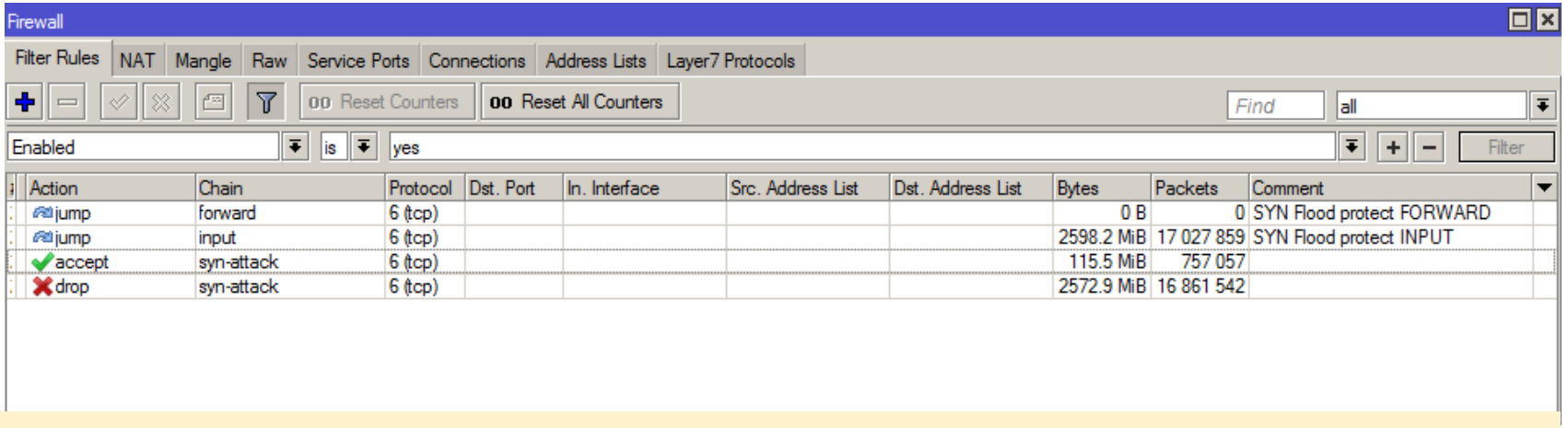
system resource cpu print

Preventing TCP SYN Attack

- Rate-limiting for each new tcp connection
- Reduce syn-received timer
- And setup tcp syn-cookies

Preventing TCP SYN Attack

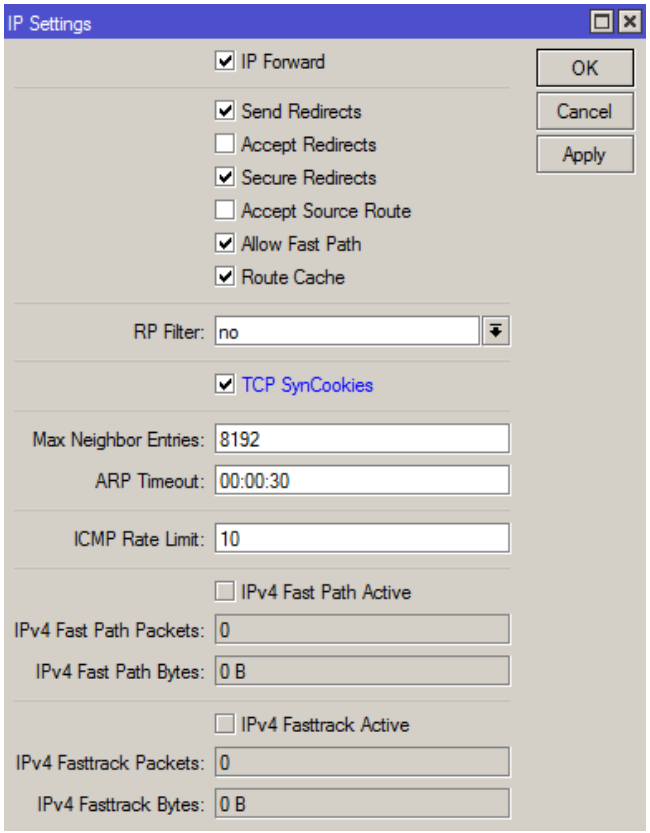
- Creating firewall for preventing tcp SYN flood




```
/ip firewall filter
add action=jump chain=forward comment="SYN Flood protect FORWARD" connection-state=new
jump-target=syn-attack protocol=tcp tcp-flags=syn
add action=jump chain=input comment="SYN Flood protect INPUT" connection-state=new jump-
target=syn-attack protocol=tcp tcp-flags=syn
add action=accept chain=syn-attack connection-state=new limit=400,5:packet protocol=tcp tcp-
flags=syn
add action=drop chain=syn-attack connection-state=new protocol=tcp tcp-flags=syn
```

Preventing TCP SYN Attack

- IP > Settings and enable “TCP SynCookies”




```
/ip settings set tcp-syncookies=yes
```

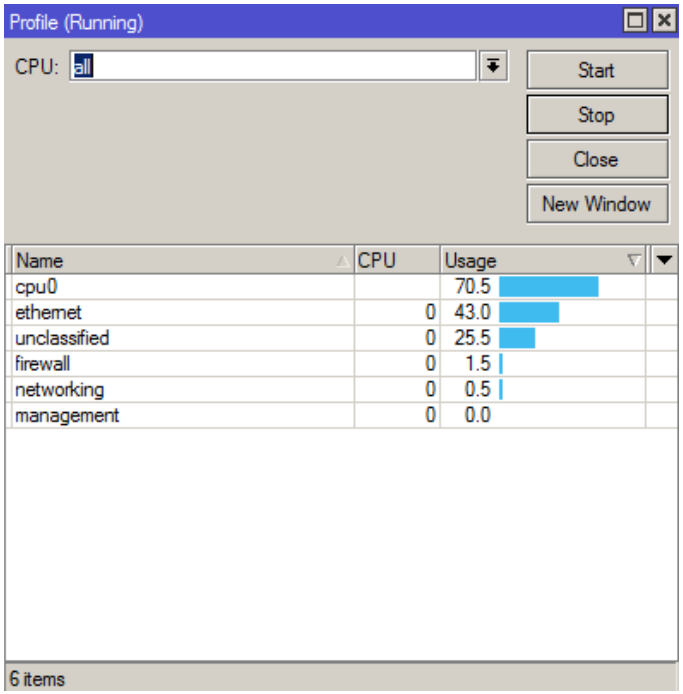
TCP SYN Attack

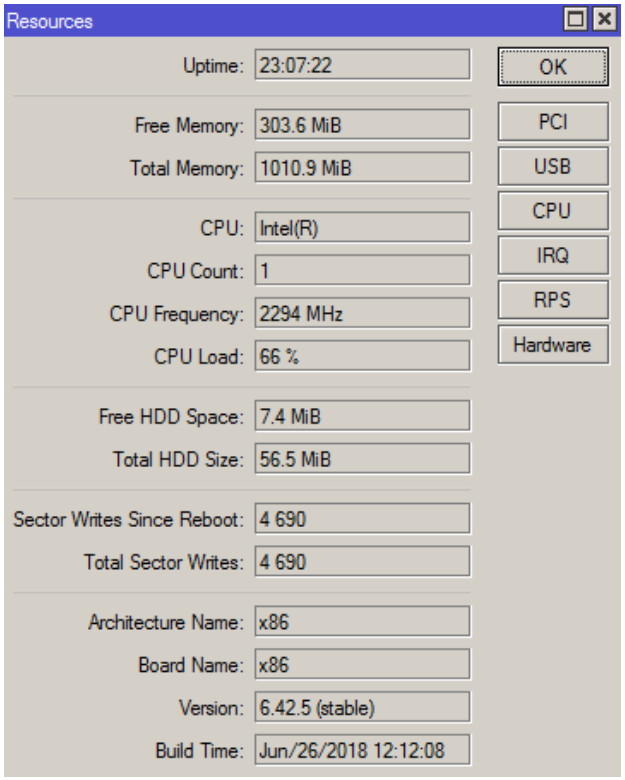
- Run hping3 again

```
root@kali:~# hping3 -c 20000 -d 120 -S -w 64 -p 80 --flood --rand-source 192.168.1.1
HPING 192.168.1.1 (eth0 192.168.1.1): S set, 40 headers + 120 data bytes
hping in flood mode, no replies will be shown
```

Preventing TCP SYN Attack

- These rules are stopping the tcp SYN attack, but still affecting the CPU resources. *(need more powerful router for preventing)*



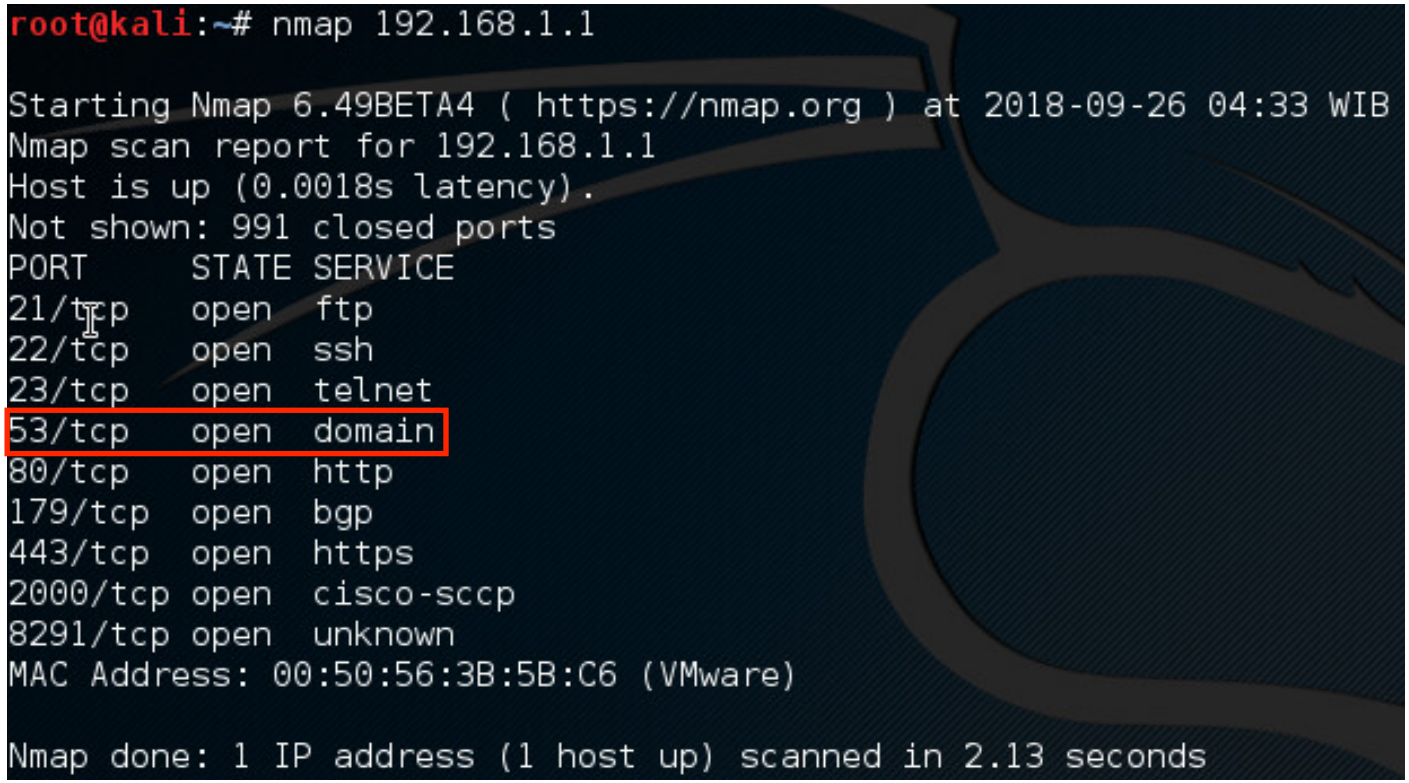


UDP Flood Attack

- An UDP flood does not exploit any vulnerability.
- The aim of UDP floods is creating and sending large amount of UDP datagrams from spoofed IP's to the target server.
- When a server receives this type of traffic, it is unable to process every request and it consumes its bandwidth with sending ICMP "destination unreachable" packets.

UDP Flood Attack

- Scanning available port on target, normal target usually port 53/dns service



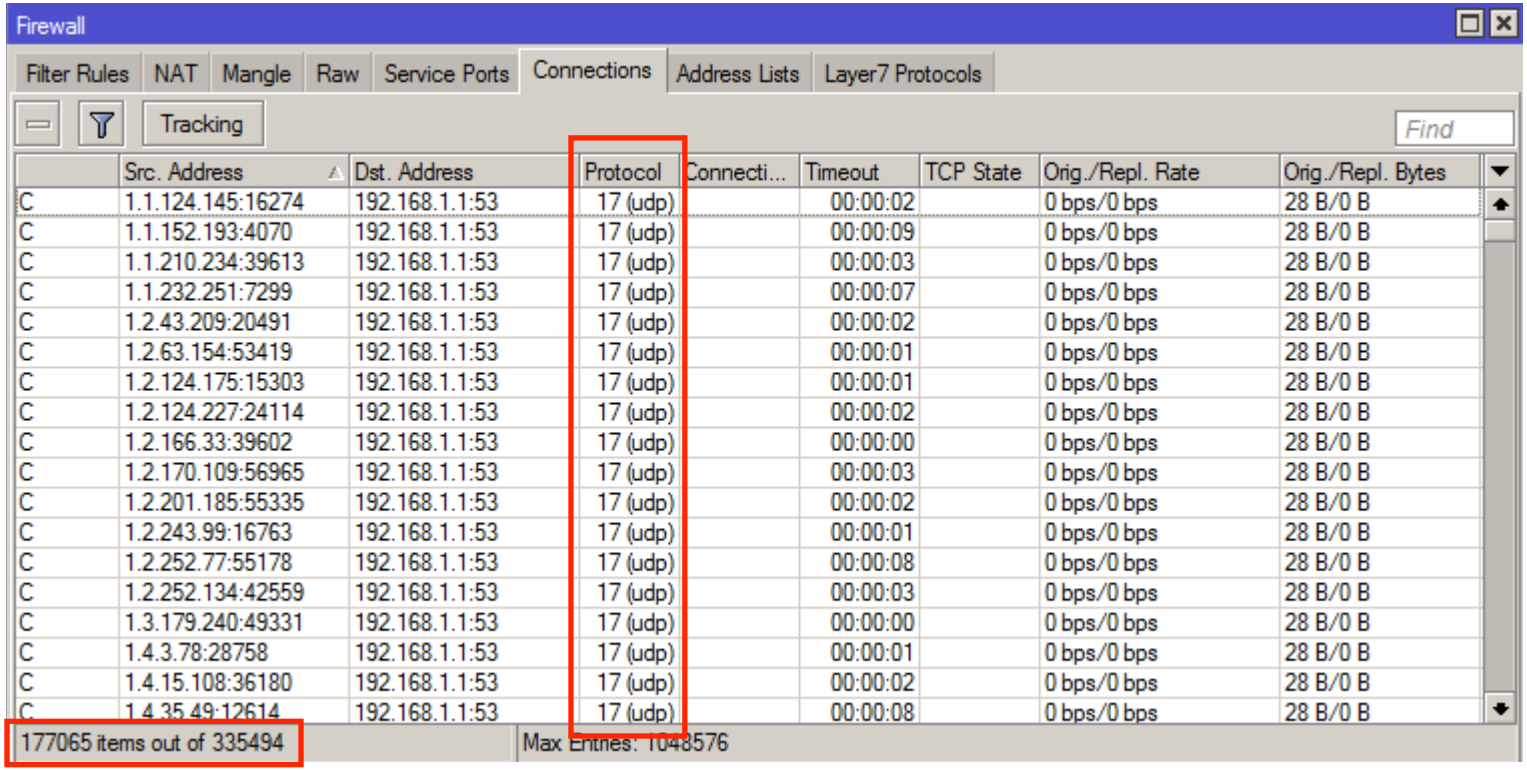
UDP Flood Attack

- Start attacking UDP protocol port 53(dns) with hping3

```
root@kali:~# hping3 --flood --rand-source --udp -p 53 192.168.1.1
HPING 192.168.1.1 (eth0 192.168.1.1): udp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

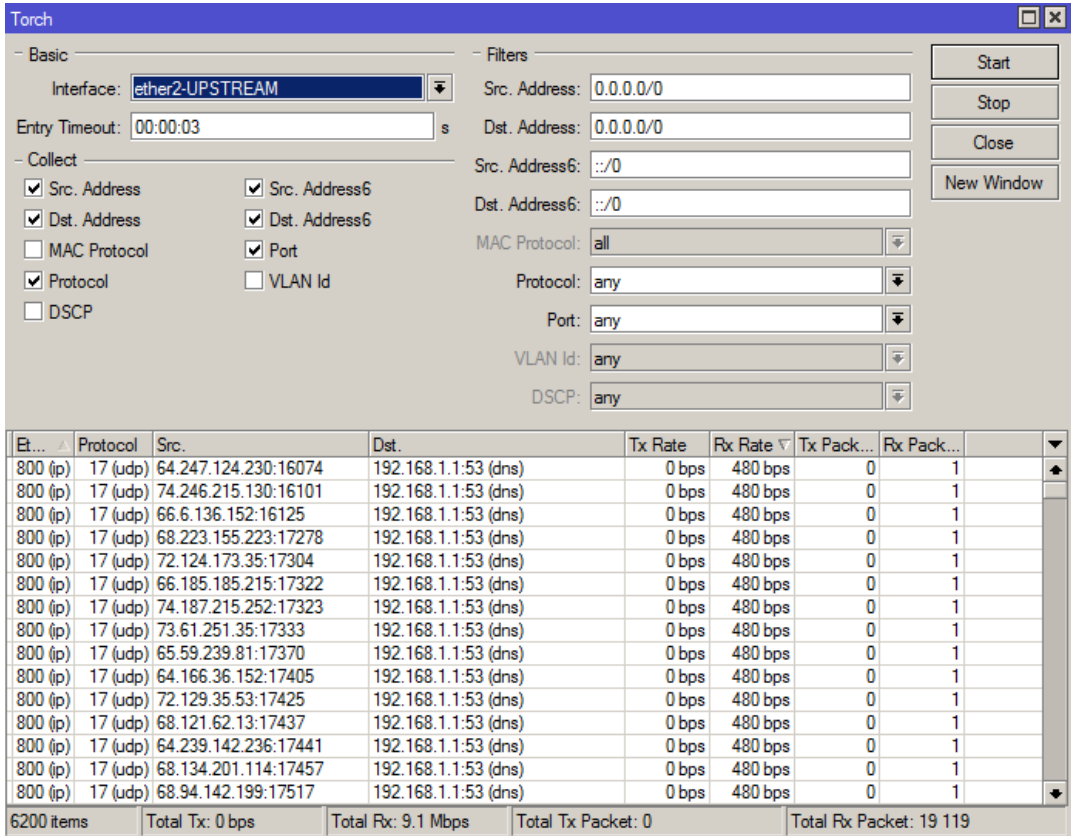
UDP Flood Attack

- “IP > Firewall > Connections” please observe “udp” protocol from random source addresses



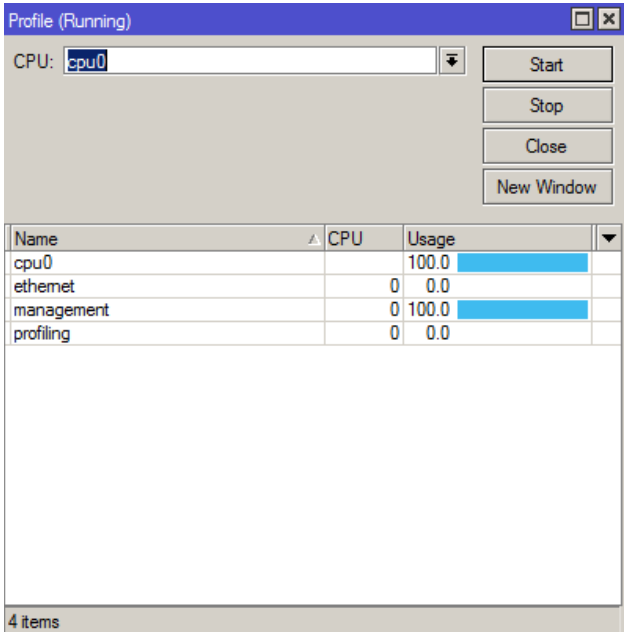
UDP Flood Attack

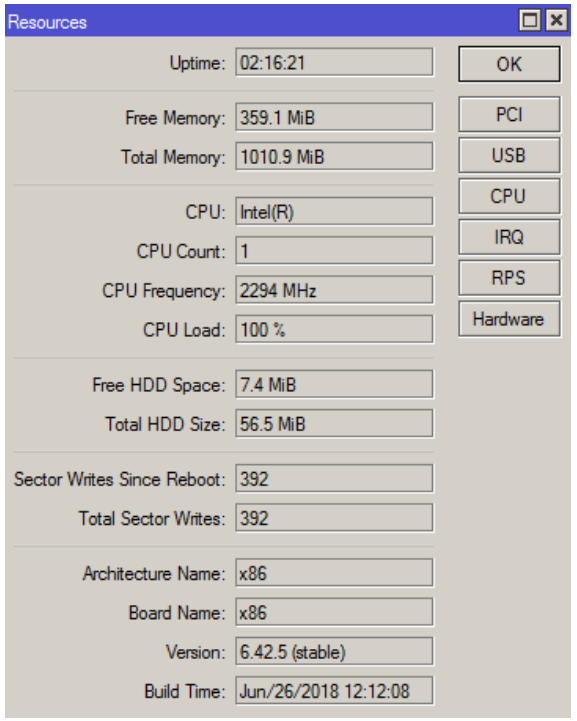
- Torch interface traffic



UDP Flood Attack

- The attack is exhausting the resources of the router and impacting the performance



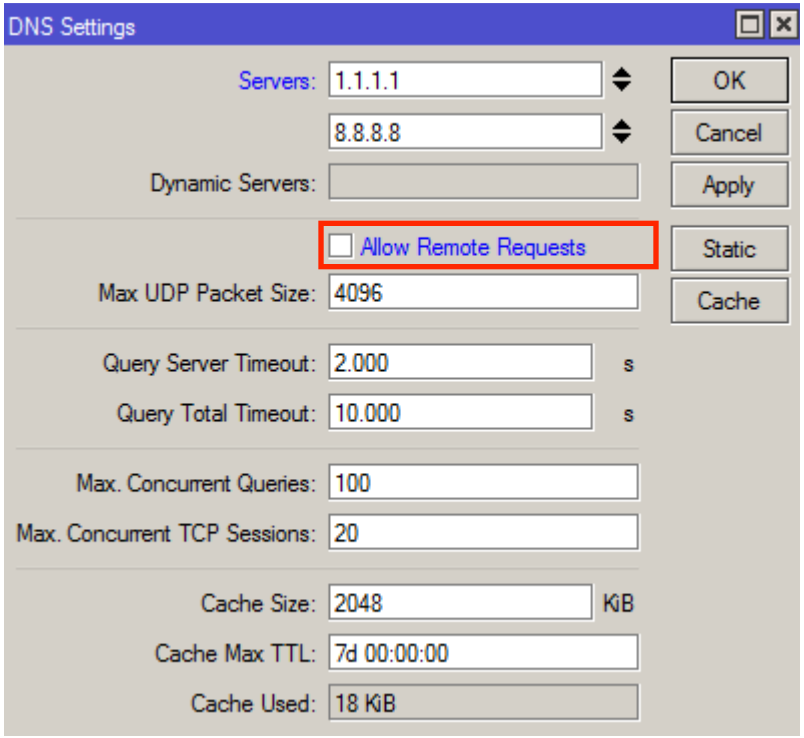


Preventing UDP Flood Attack

- Disable DNS forwarder on MikroTik if not required.
- If “IP -> DNS” – *Allow remote request* is enabled, make sure appropriate filter rule is set to prevent incoming DNS attacks.
- Rate-limiting for each new udp connection.

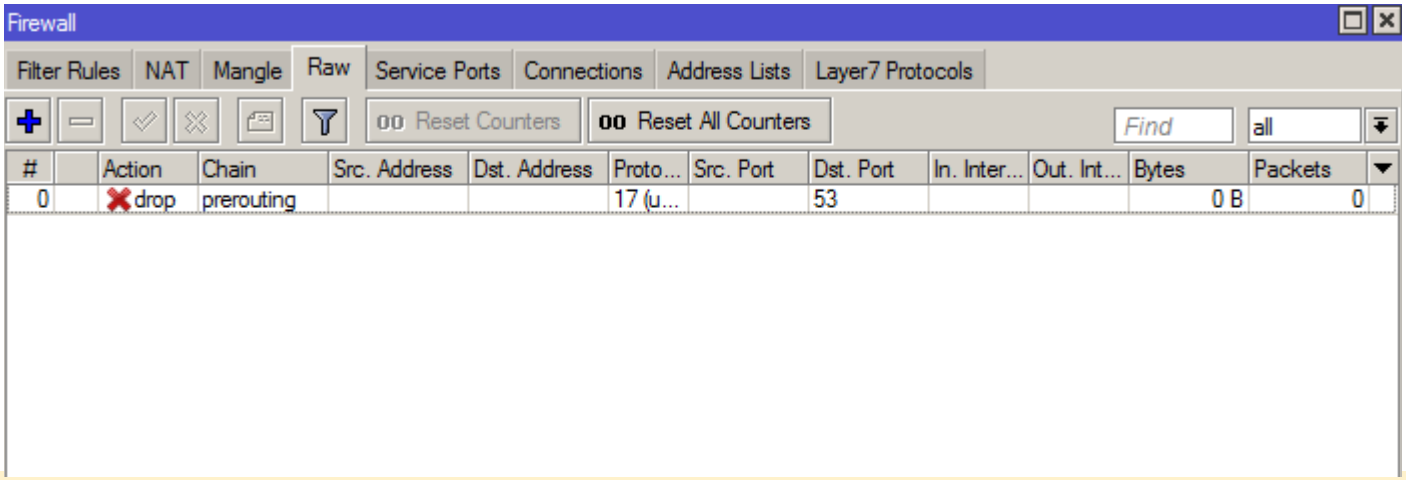
Preventing UDP Flood Attack

- Uncheck *Allow Remote Requests* on router



Preventing UDP Flood Attack

- Block dns request *“udp/53”* traffic from outside

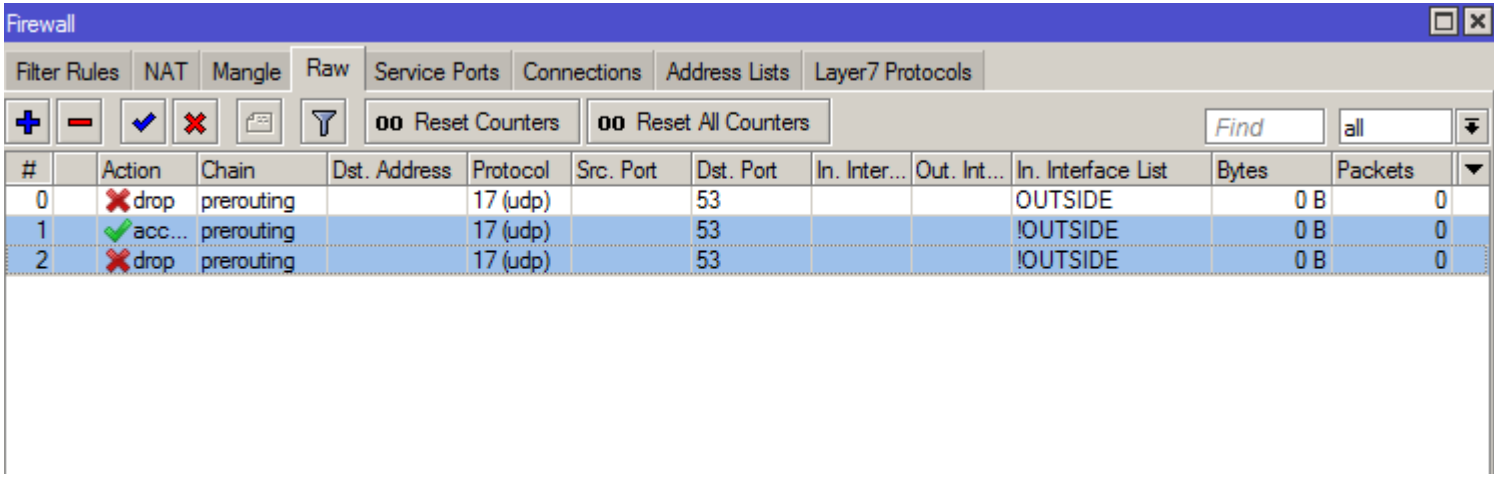



```
/interface list add name=OUTSIDE
/interface list member add interface=ether3-internet list=OUTSIDE
```

```
/ip firewall raw add action=drop chain=prerouting dst-port=53 in-interface-list=OUTSIDE
protocol=udp
```

Preventing UDP Flood Attack

- Rate-limiting every udp/53 packet requests




```
/ip firewall raw
add action=accept chain=prerouting dst-port=53 in-interface-list=!OUTSIDE limit=100,5:packet
protocol=udp
add action=drop chain=prerouting dst-port=53 in-interface-list=!OUTSIDE protocol=udp
```

ICMP Smurf Attack

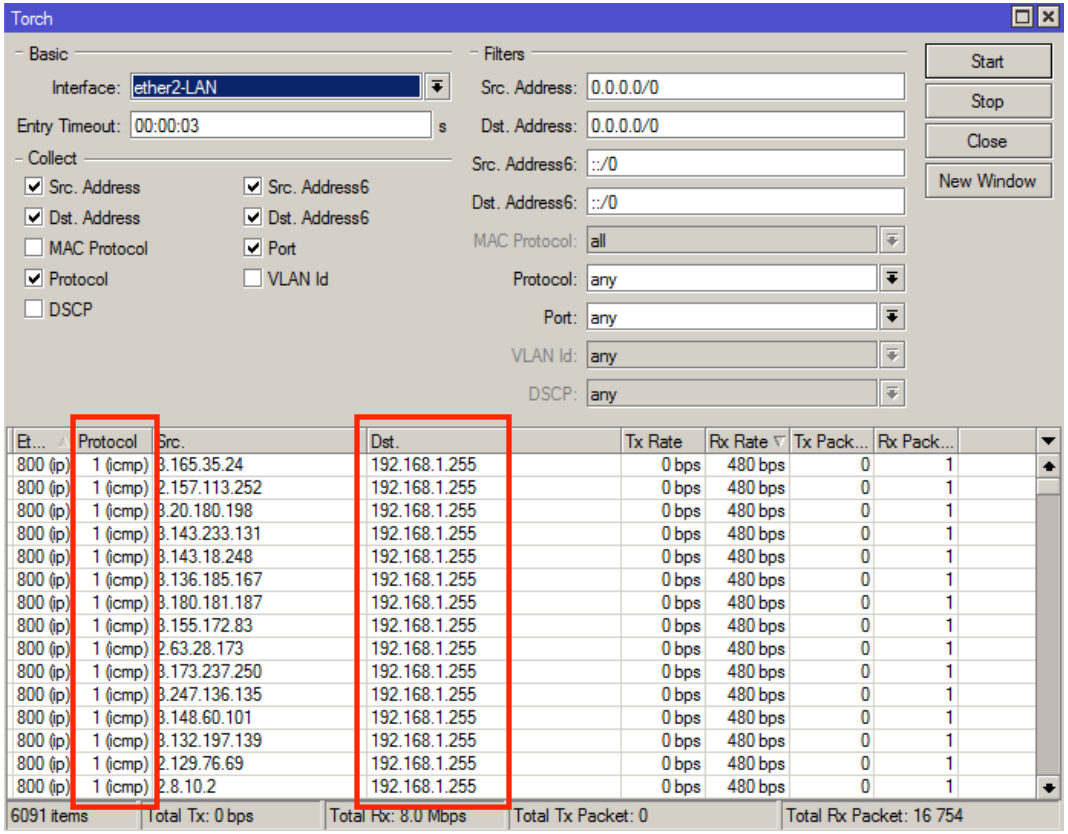
- This type of attack uses large amount of Internet Control Message Protocol (ICMP) ping traffic targeted at an Internet Broadcast Address e.g 192.168.1.255.
- The reply IP address is spoofed to that of the intended victim e.g 1.2.3.4
- All the replies are sent to the victim instead of the IP used for the pings.
- Since a single Internet Broadcast Address can support a maximum of 255 hosts, a smurf attack amplifies a single ping 255 times.

ICMP Smurf Attack

- Start attacking ICMP smurf with random source

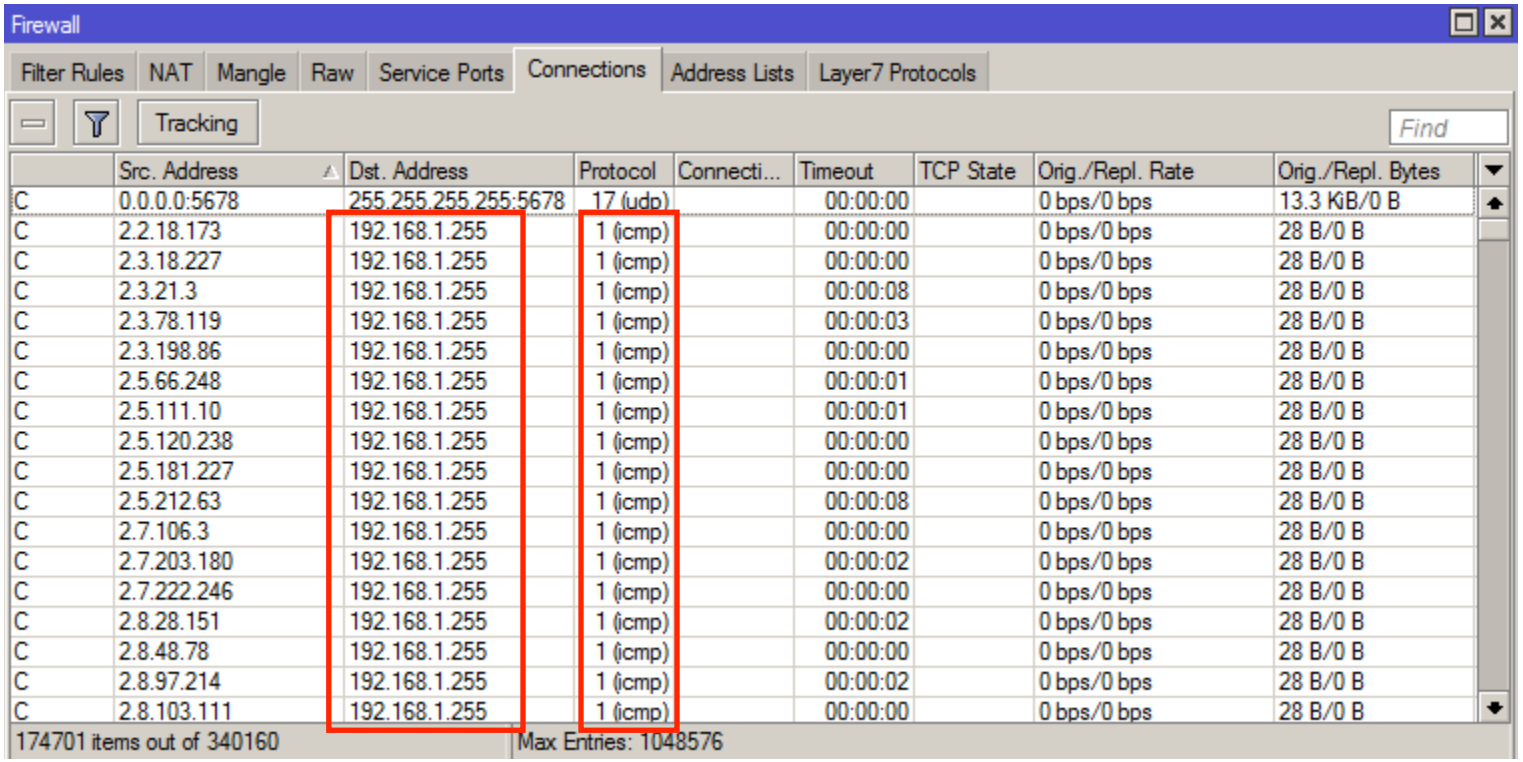
```
root@kali:~# hping3 --icmp --flood --rand-source  -c 20000 --spoof 192.168.1.1 192.168.1.255
HPING 192.168.1.255 (eth0 192.168.1.255): icmp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

ICMP Smurf Attack



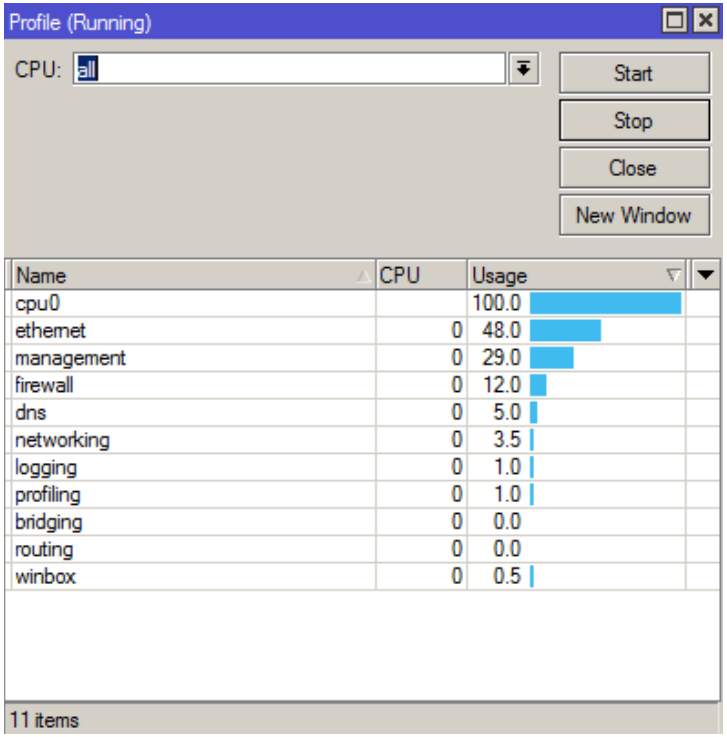
- All of attacker's traffic as a destination address has the broadcast address of the network

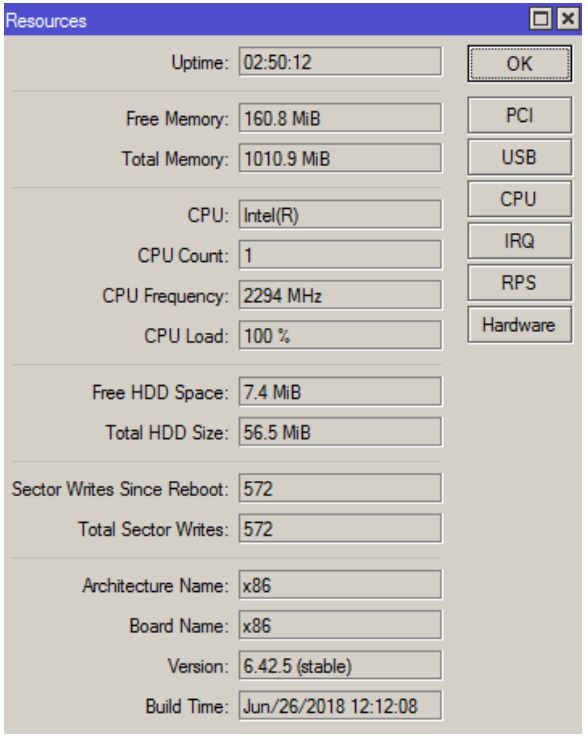
ICMP Smurf Attack



ICMP Smurf Attack

- The attack is exhausting the resources of the router and impacting the performance

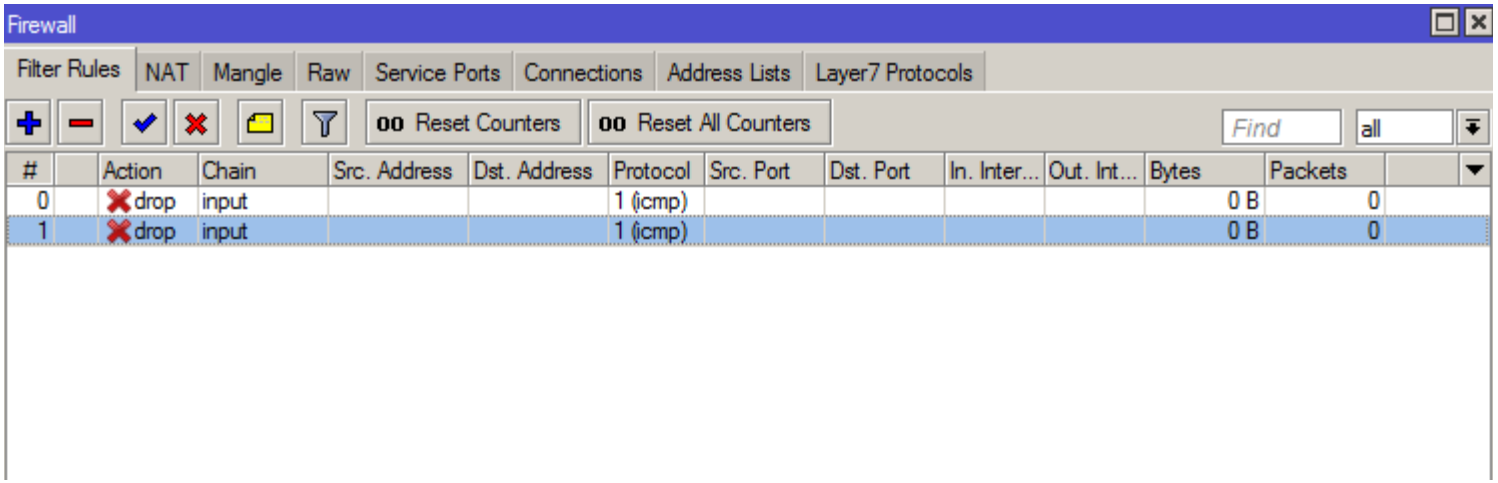




Preventing ICMP Smurf Attack

- Configure routers not to forward or accept packets directed to broadcast addresses.
- Configure individual hosts or routers to not respond to ping requests from outside

Preventing ICMP Smurf Attack



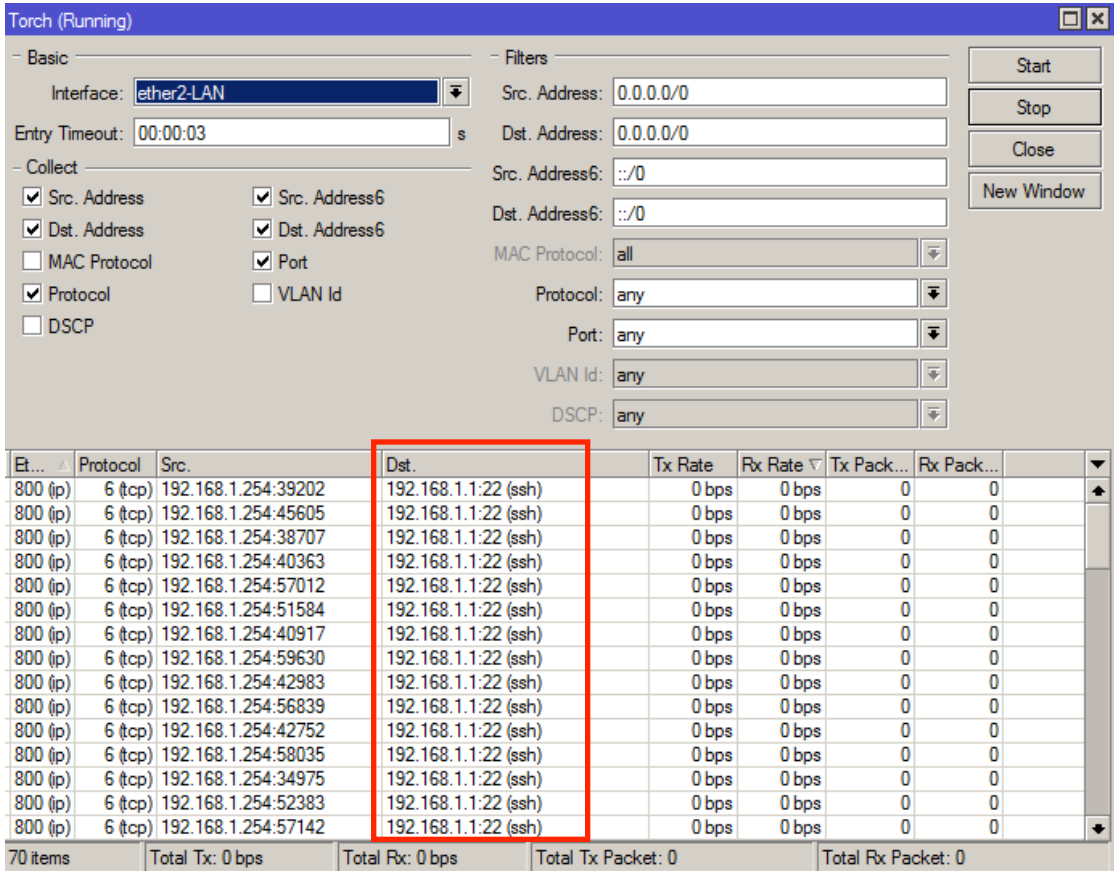

```
/ip firewall filter
add action=drop chain=input dst-address-type=broadcast icmp-options=0:0-255 protocol=icmp
add action=drop chain=input in-interface-list=OUTSIDE protocol=icmp
```

Password Brute Force Attack

- A brute force attack is a trial-and-error method used to obtain information such as a users password or any other credential information.
- In a brute force attack, automated software is used to generate a large number of consecutive guesses as to the value of the desired data.

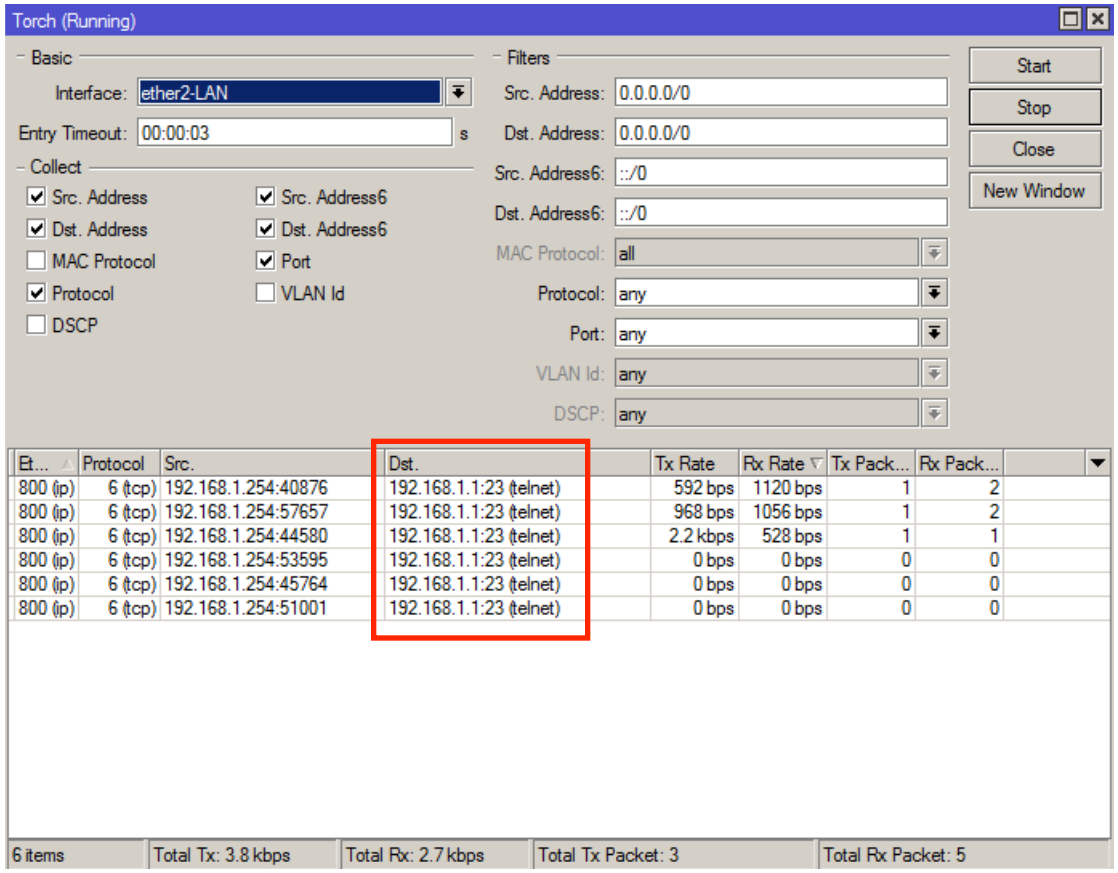
Password Brute Force Attack

- Router under SSH Brute Force Attack



Password Brute Force Attack

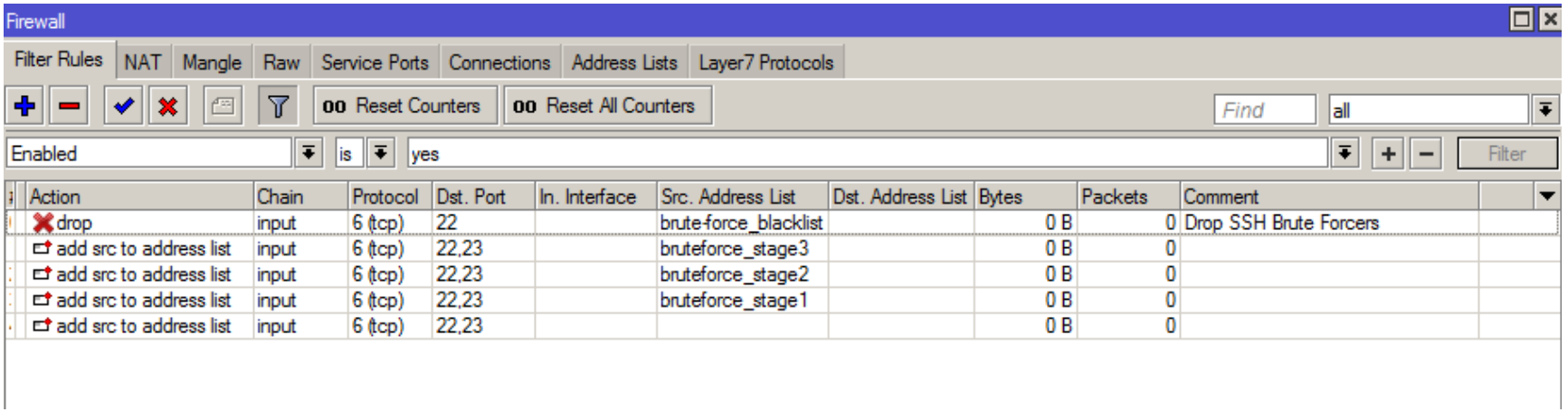
- Router under Telnet Brute Force Attack



Preventing Brute Force Attack

- Limiting the number of times a user can unsuccessfully attempt to log in
- Temporarily locking out users who exceed the specified maximum number of failed login attempts
- Requiring users to create complex passwords
- Periodically changing a password

Preventing Brute Force Attack



Preventing Brute Force Attack

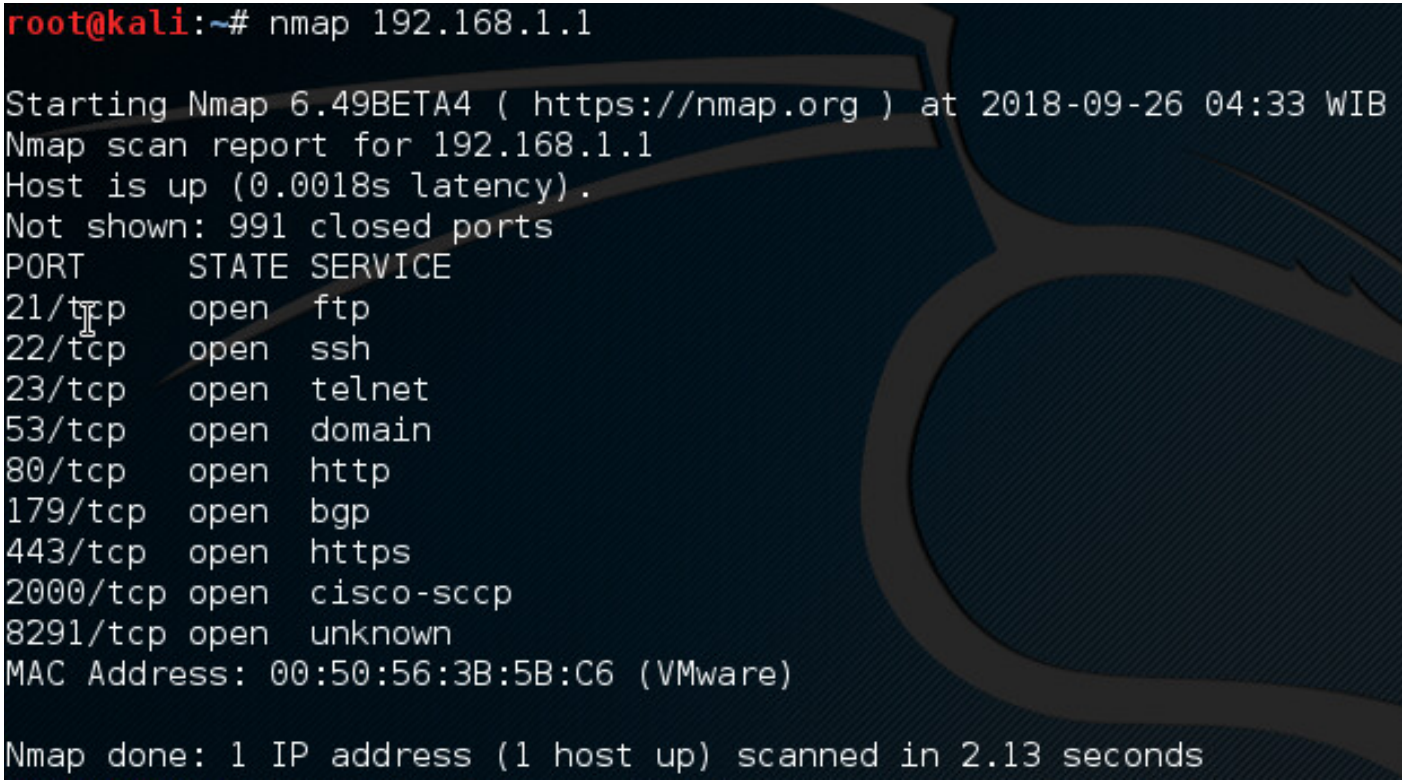
```
/ip firewall filter
add action=drop chain=input comment="Drop SSH Brute Forcers" dst-port=22 protocol=tcp \
    src-address-list=brute-force_blacklist
add action=add-src-to-address-list address-list=brute-force_blacklist address-list-timeout=1d
chain=input \
    connection-state=new dst-port=22,23 protocol=tcp src-address-list=bruteforce_stage3
add action=add-src-to-address-list address-list=bruteforce_stage3 address-list-timeout=30s
chain=input  \
    connection-state=new dst-port=22,23 protocol=tcp src-address-list=bruteforce_stage2
add action=add-src-to-address-list address-list=bruteforce_stage2 address-list-timeout=30s
chain=input \
    connection-state=new dst-port=22,23 protocol=tcp src-address-list=bruteforce_stage1
add action=add-src-to-address-list address-list=bruteforce_stage1 address-list-timeout=1m
chain=input \
    connection-state=new dst-port=22,23 protocol=tcp
```

Port Scanner Detection

- A port scan is a method for determining which ports on a network are open or available.
- Running a port scan on a network or server reveals which ports are open and listening *(receiving information)*
- Port Scan tools (like NMAP) can detect what version of an application is running on a port
- Port scanning is the “gate” for starting an attack or penetration to your networks

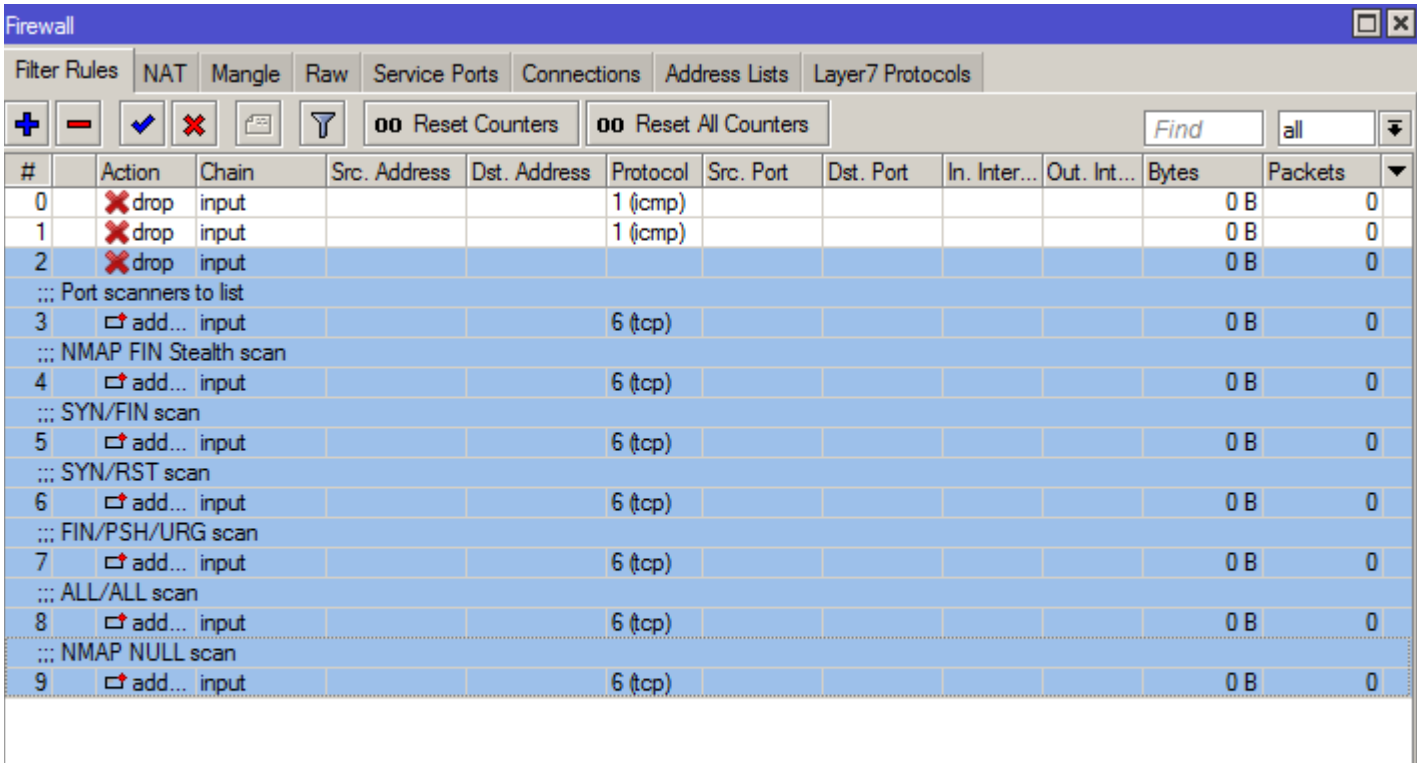
Port Scanner Detection

- Scanning available ports on the target



Preventing Port Scanner

- Create Port Scanner Detection on router and block the address



#	Action	Chain	Src. Address	Dst. Address	Protocol	Src. Port	Dst. Port	In. Inter...	Out. Int...	Bytes	Packets
0	drop	input			1 (icmp)					0 B	0
1	drop	input			1 (icmp)					0 B	0
2	drop	input								0 B	0
;;; Port scanners to list											
3	add...	input			6 (tcp)					0 B	0
;;; NMAP FIN Stealth scan											
4	add...	input			6 (tcp)					0 B	0
;;; SYN/FIN scan											
5	add...	input			6 (tcp)					0 B	0
;;; SYN/RST scan											
6	add...	input			6 (tcp)					0 B	0
;;; FIN/PSH/URG scan											
7	add...	input			6 (tcp)					0 B	0
;;; ALL/ALL scan											
8	add...	input			6 (tcp)					0 B	0
;;; NMAP NULL scan											
9	add...	input			6 (tcp)					0 B	0

Preventing Port Scanner

```
/ip firewall filter
add action=drop chain=input src-address-list="port scanners"
add action=add-src-to-address-list address-list="port scanners" address-list-timeout=2w chain=input
comment="Port scanners to list " protocol=tcp psd=21,3s,3,1
add action=add-src-to-address-list address-list="port scanners" address-list-timeout=2w chain=input
comment="NMAP FIN Stealth scan" protocol=tcp tcp-flags=\
    fin,!syn,!rst,!psh,!ack,!urg
add action=add-src-to-address-list address-list="port scanners" address-list-timeout=2w chain=input
comment="SYN/FIN scan" protocol=tcp tcp-flags=fin,syn
add action=add-src-to-address-list address-list="port scanners" address-list-timeout=2w chain=input
comment="SYN/RST scan" protocol=tcp tcp-flags=syn,rst
add action=add-src-to-address-list address-list="port scanners" address-list-timeout=2w chain=input
comment="FIN/PSH/URG scan" protocol=tcp tcp-flags=\
    fin,psh,urg,!syn,!rst,!ack
add action=add-src-to-address-list address-list="port scanners" address-list-timeout=2w chain=input
comment="ALL/ALL scan" protocol=tcp tcp-flags=\
    fin,syn,rst,psh,ack,urg
add action=add-src-to-address-list address-list="port scanners" address-list-timeout=2w chain=input
comment="NMAP NULL scan" protocol=tcp tcp-flags=\
    !fin,!syn,!rst,!psh,!ack,!urg
```

SECURING THE ROUTER

PORT KNOCKING

What is Port Knocking

- Port knocking is a method that enables access to the router only after receiving a sequenced connection attempts on a set of “pre-specified” open ports.
- Once the correct sequence of the connection attempts is received, the RouterOS dynamically adds a host source IP to the allowed address list and you will be able to connect to your router.
- You can use some online available port-knock clients, or manually connect router IP address with defined ports.
- The port "knock" itself is similar to a secret handshake and can consist of any number of TCP, UDP, or ICMP or other protocol packets to numbered ports on the destination machine

How the Port Knocking works

Host trying to make a connection to first “knocking-port”

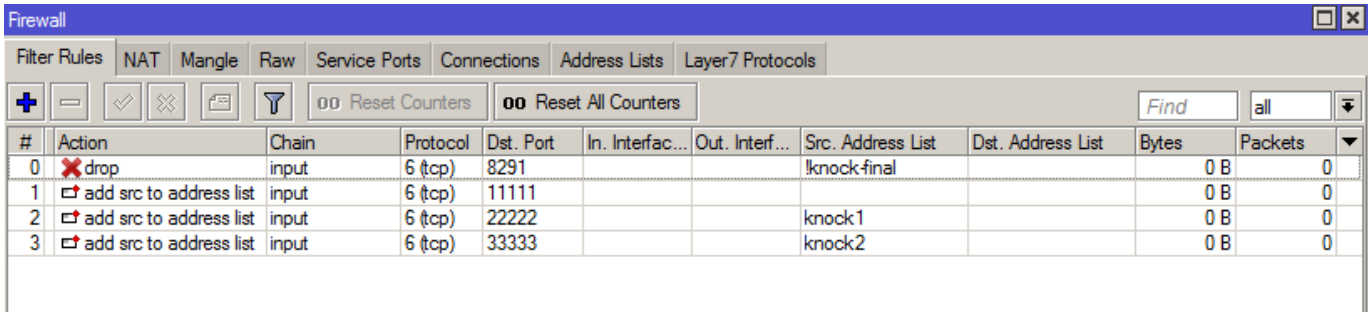
RouterOS dynamically adds a host source IP to the allowed address-list

Host trying to make a second attempt “knocking-port”

RouterOS will check if IP coming from the same first connection on allowed address-list

If the IP is the same and the time between first attempt and seconds within a specified time then the host IP will be allowed to access the router

How the Port Knocking works



#	Action	Chain	Protocol	Dst. Port	In. Interfac...	Out. Interf...	Src. Address List	Dst. Address List	Bytes	Packets
0	drop	input	6 (tcp)	8291			!knock-final		0 B	0
1	add src to address list	input	6 (tcp)	11111					0 B	0
2	add src to address list	input	6 (tcp)	22222			knock1		0 B	0
3	add src to address list	input	6 (tcp)	33333			knock2		0 B	0

```
/ip firewall filter
add action=drop chain=input dst-port=8291 protocol=tcp src-address-list=!knock-final
add action=add-src-to-address-list address-list=knock1 address-list-timeout=10s chain=input dst-port=11111 \
    protocol=tcp
add action=add-src-to-address-list address-list=knock2 address-list-timeout=10s chain=input dst-port=22222 \
    protocol=tcp src-address-list=knock1
add action=add-src-to-address-list address-list=knock-final address-list-timeout=1d chain=input \
    dst-port=33333 protocol=tcp src-address-list=knock2
```

How the Port Knocking works

```
D:\Saya\Apps\knock>dir
 Volume in drive D has no label.
 Volume Serial Number is F258-BA8D

 Directory of D:\Saya\Apps\knock

09/09/2018  12:40 PM    <DIR>          .
09/09/2018  12:40 PM    <DIR>          ..
07/03/2005  02:30 AM         1,295,582 cygwin1.dll
08/10/2005  02:52 PM            15,238 knock.exe
               2 File(s)      1,310,820 bytes
               2 Dir(s)  127,842,557,952 bytes free

D:\Saya\Apps\knock>knock.exe
usage: knock [options] <host> <port[:proto]> [port[:proto]] ...
options:
  -u, --udp             make all ports hits use UDP (default is TCP)
  -v, --verbose         be verbose
  -V, --version         display version
  -h, --help            this help

example:  knock myserver.example.com 123:tcp 456:udp 789:tcp


D:\Saya\Apps\knock>knock your.mikrotik.ip-or-domain 12345:tcp 54321:udp
```

Port Knocking for Windows

Port Knocking for Linux

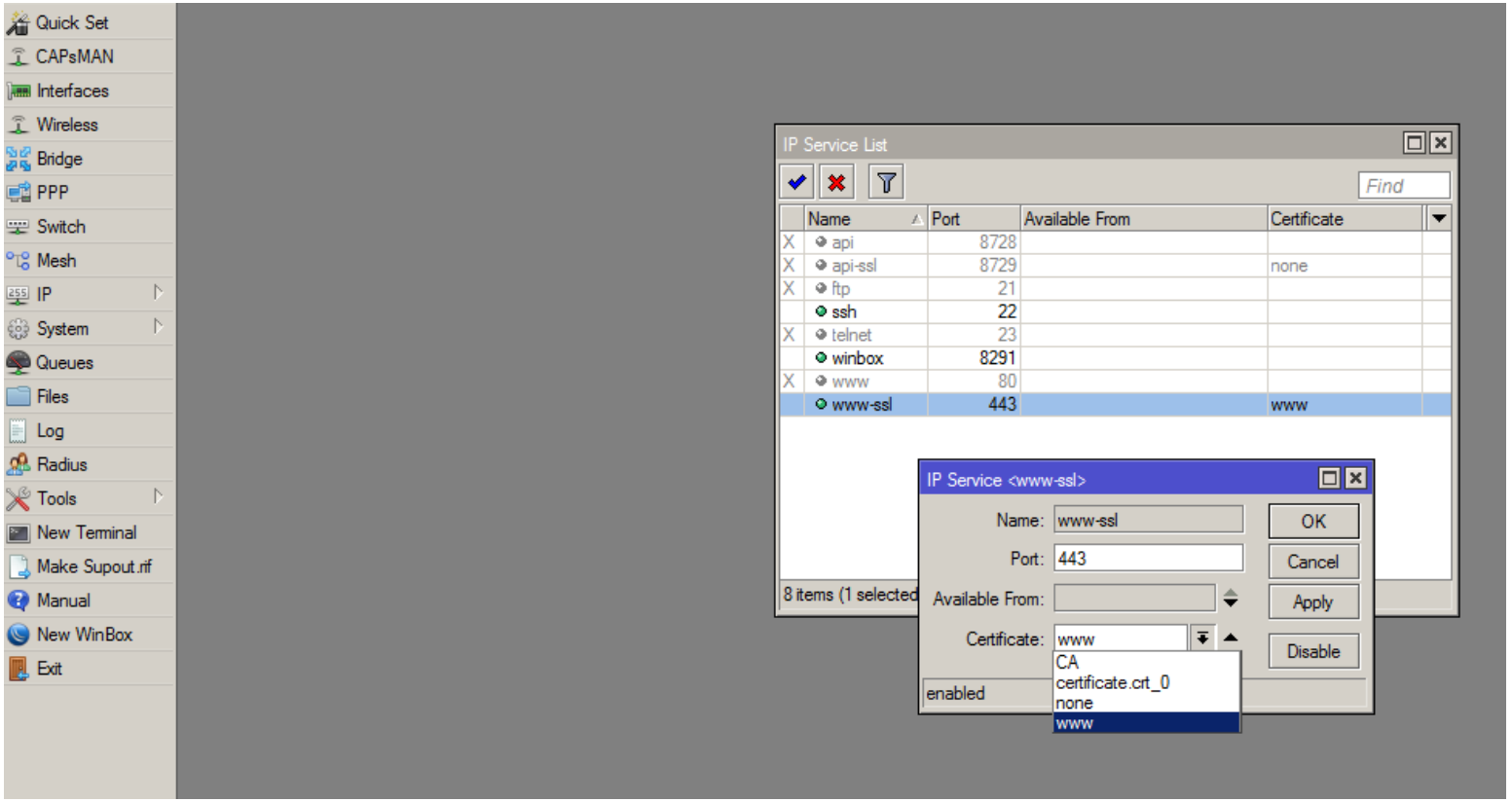
apt-get install knockd or yum install knockd
knock your.mikrotik.ip-address-or-domain 12345:tcp 54321:udp

SECURE CONNECTIONS

What is a Secure Connection

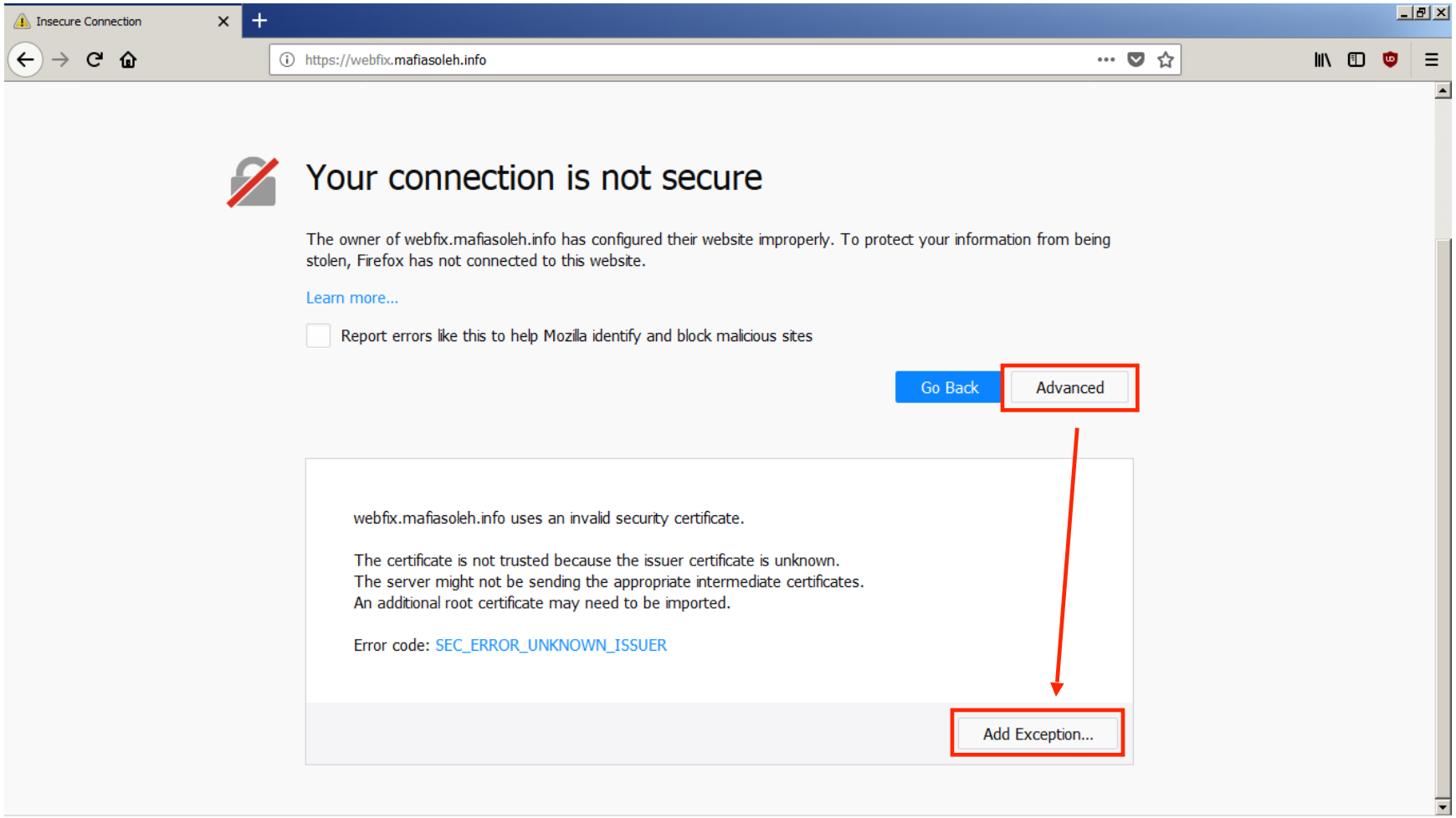
- A connection that is encrypted by one or more security protocols to ensure the security of data flowing between two or more nodes.
- When a connection is not encrypted, it can be easily listened to by anyone with the knowledge on how to do it.
- Protect the data being transferred from one computer to another

Self-signed Certificate

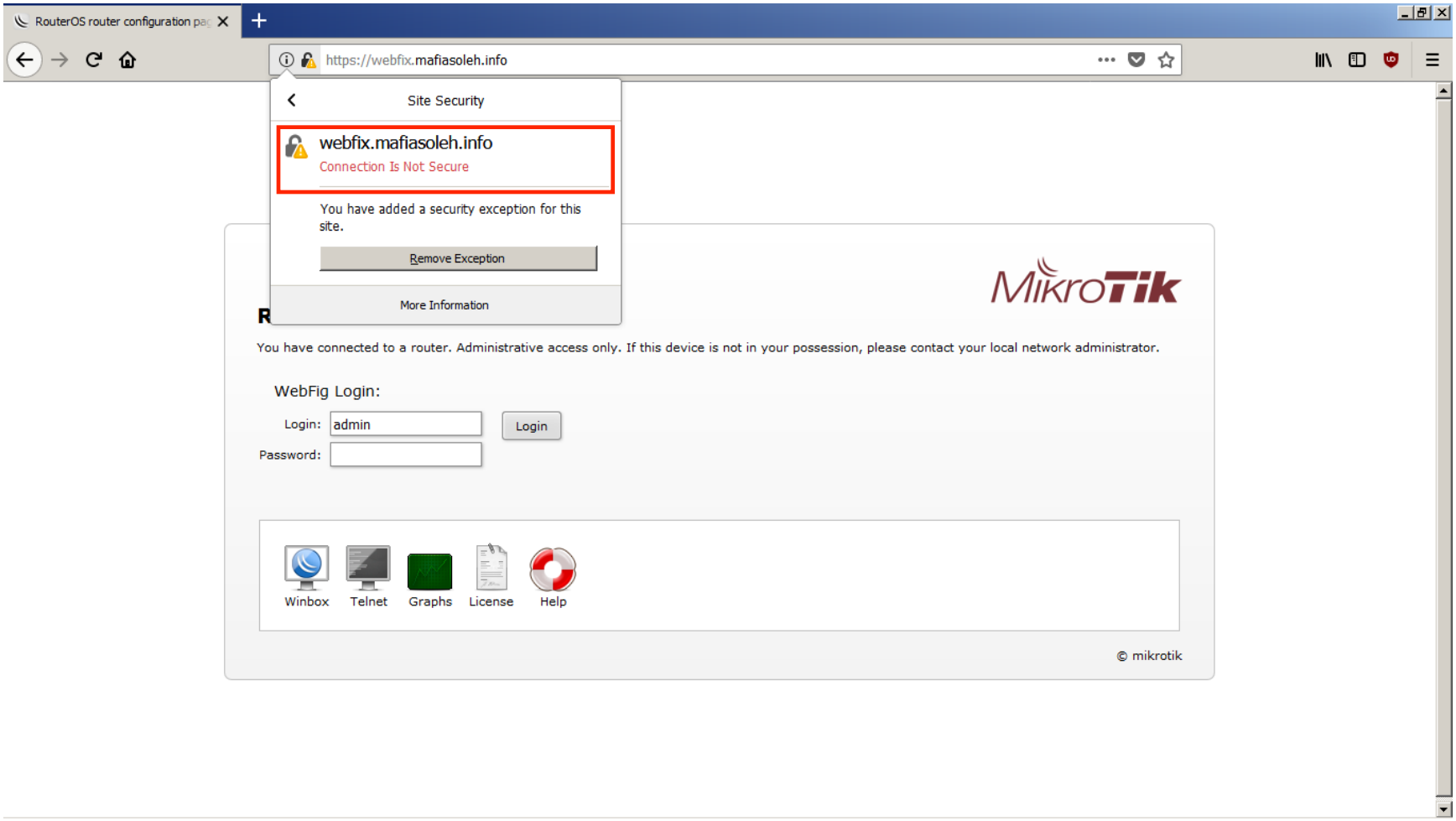


ip service set www-ssl certificate=www

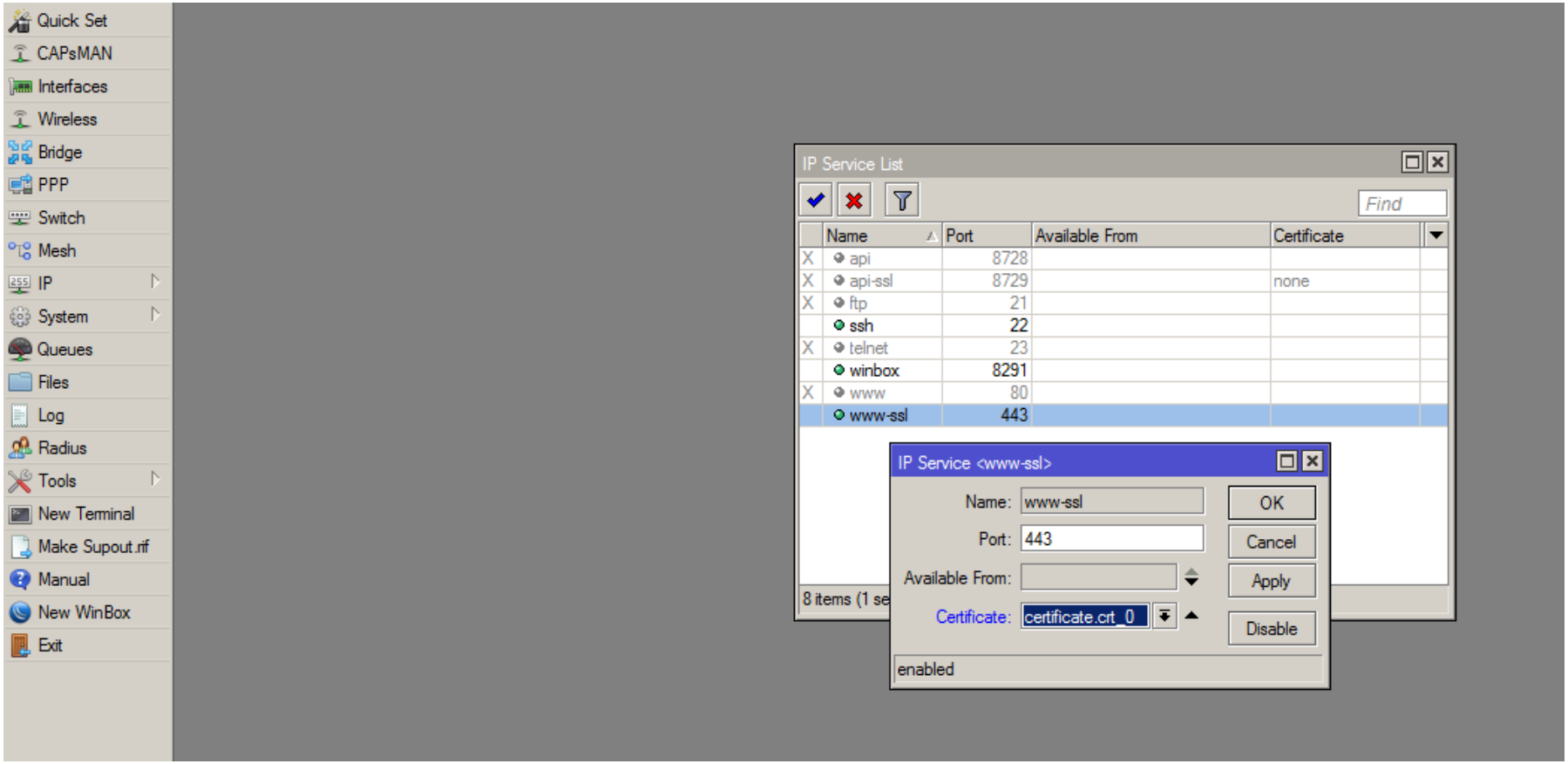
Self-signed Certificate



Self-signed Certificate

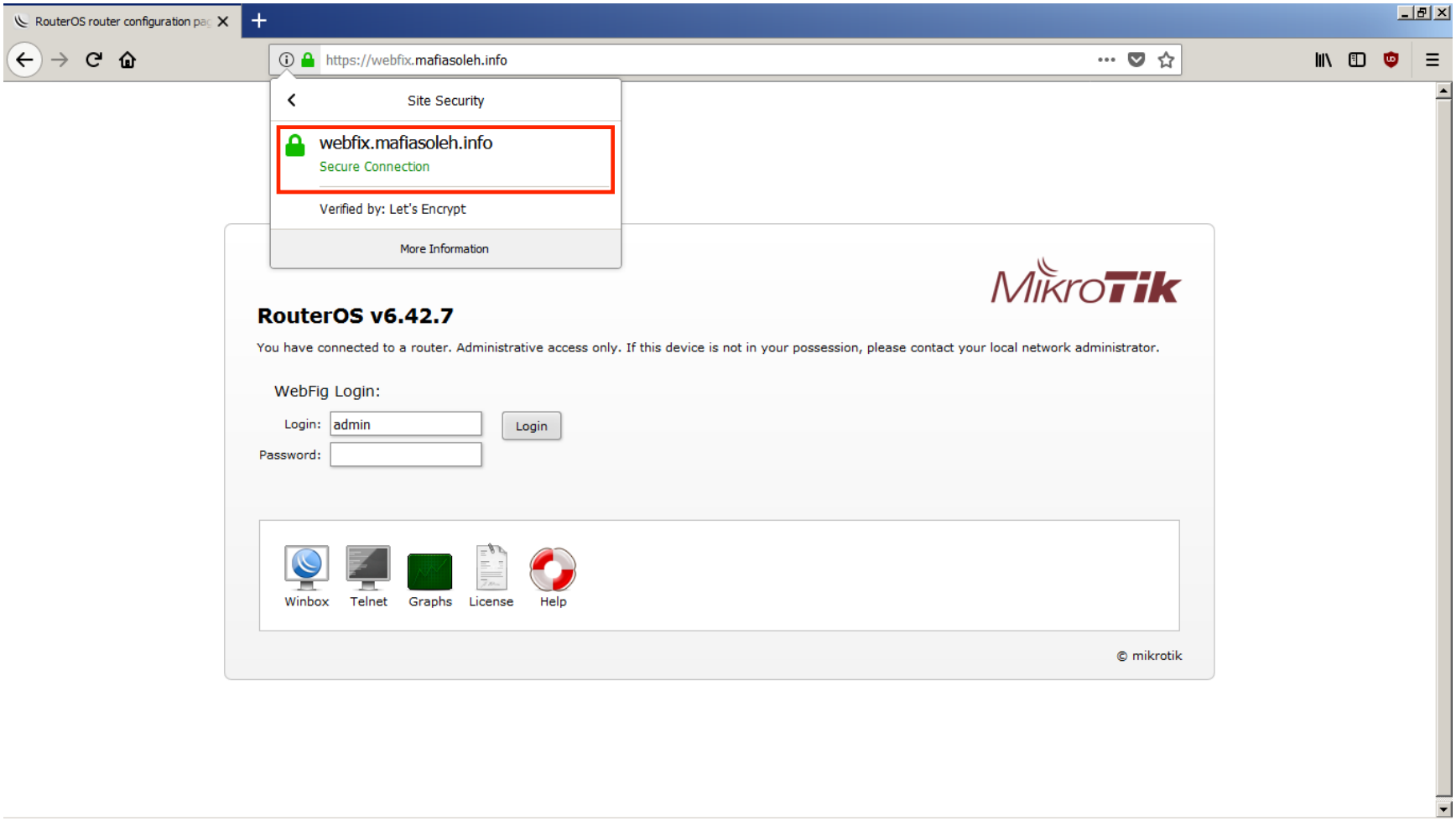


Free of Charge Valid Certificate




```
ip service set www-ssl certificate=certificate.crt_0
```

Free of Charge Valid Certificate

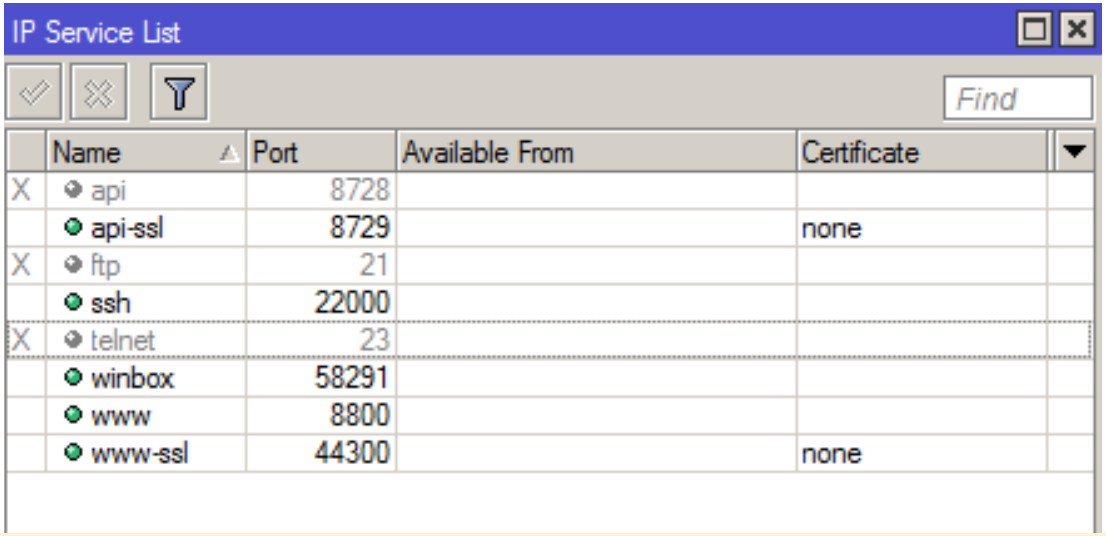


DEFAULT PORTS FOR THE SERVICES

Default Ports for the Services

- In TCP/IP and UDP networks, a port is an endpoint to a logical connection and the way a client program specifies a specific server program on a computer in a network.
- The port number identifies what type of port it is, and what kind of service those port is serving
- Some ports have numbers that are assigned to them by the IANA, and these are called the "well-known ports" which are specified in RFC1700.
- Port numbers range from 0 to 65535, but only port numbers 0 to 1023 are reserved for privileged services and designated as well-known ports.

Default Ports for the Services




```
/ip service set telnet disabled=yes
/ip service set ftp disabled=yes
/ip service set www port=8800
/ip service set ssh port=22000
/ip service set www-ssl disabled=no port=44300
/ip service set api disabled=yes
/ip service set winbox port=58291
```

NB: Obscurity is not security - you should also use firewall rules

TUNNELING THROUGH SSH

What is an SSH Tunnel

- An SSH tunnel consists of an encrypted tunnel created using the SSH protocol connection
- The SSH tunnel can be used to encapsulate unencrypted traffic and transmit it via an encrypted channel.

How SSH Works

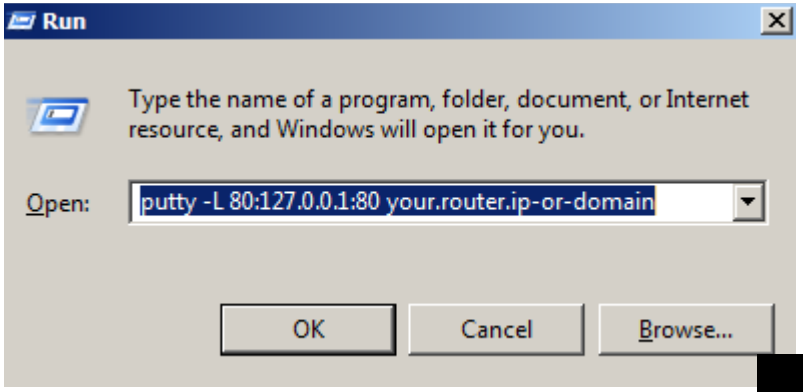
Host connects to RouterOS using ssh
with local-port forwarding parameter

RouterOS accepted ssh connections from host

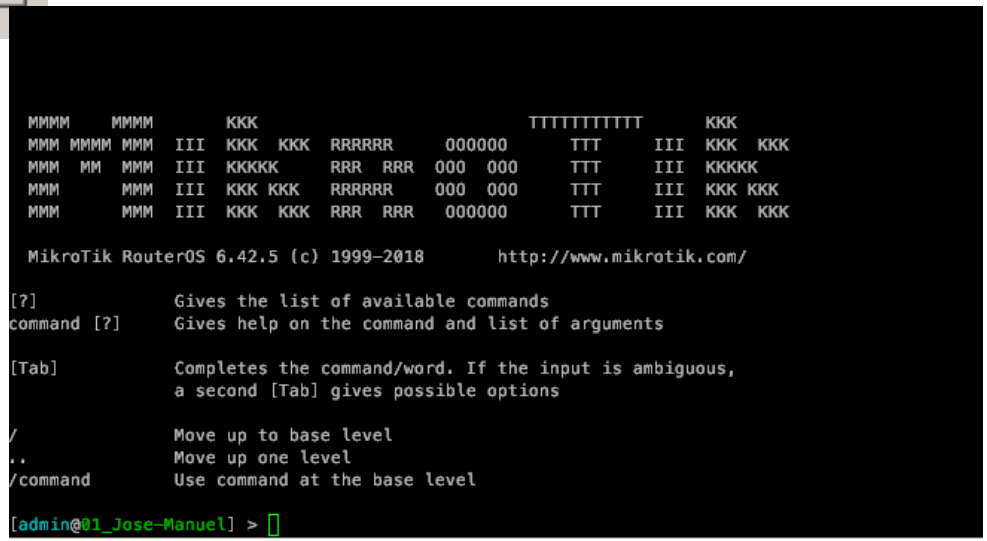
Host trying to open unencrypted port (80)
from ssh tunnel via local-port forwarding ip

RouterOS sending http request from host
via ssh tunnel

Configuring the SSH tunnel



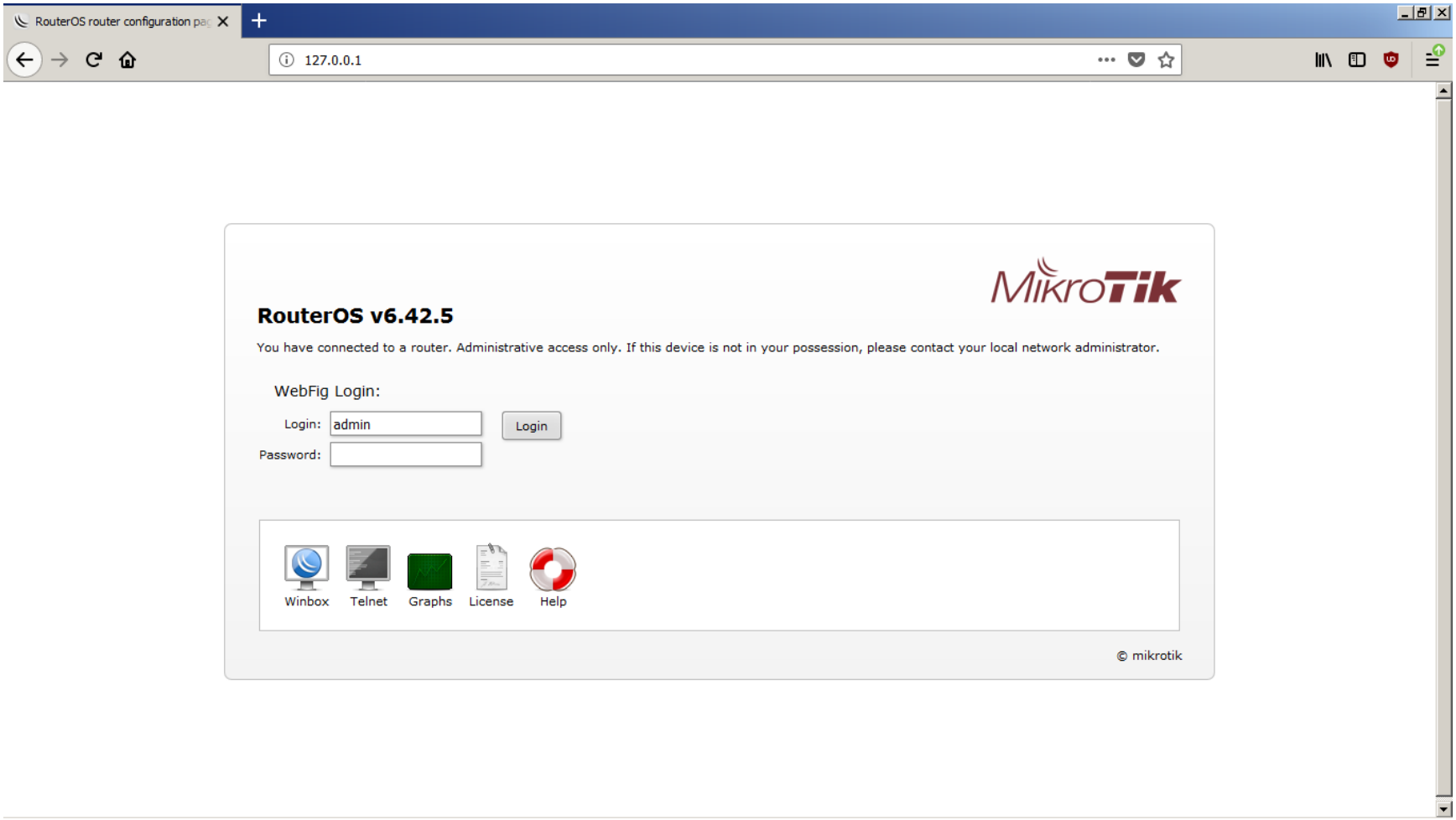
SSH Local-Forwarding for Windows



SSH Local-Forwarding for Linux

```
ssh –L 80:127.0.0.1:80 your.router.ip-or-domain
```

Configuring the SSH tunnel



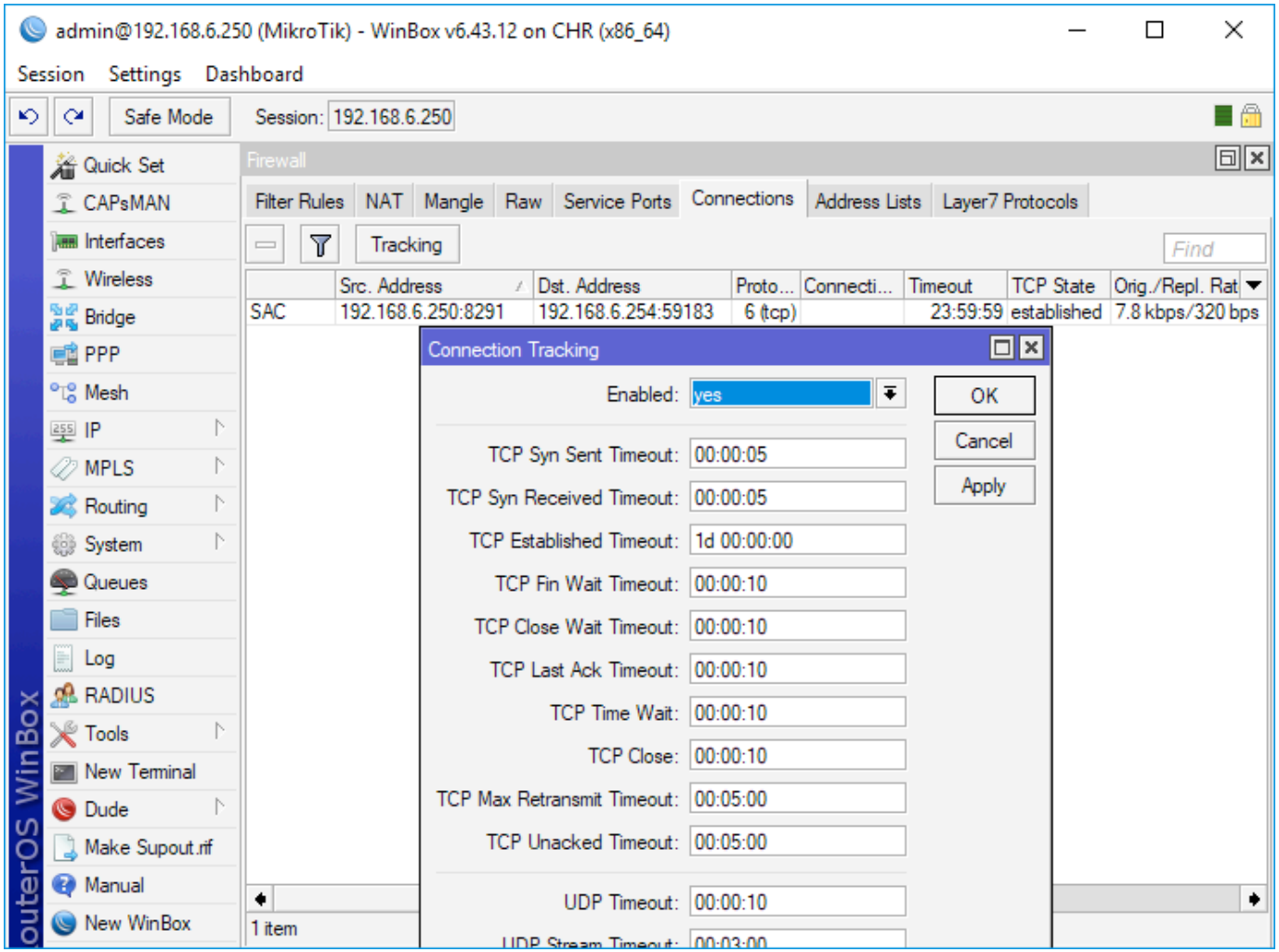
STATEFUL FIREWAL

RouterOS implements a stateful firewall. A stateful-firewall is a firewall capable of tracking ICMP, UDP, and TCP connections.

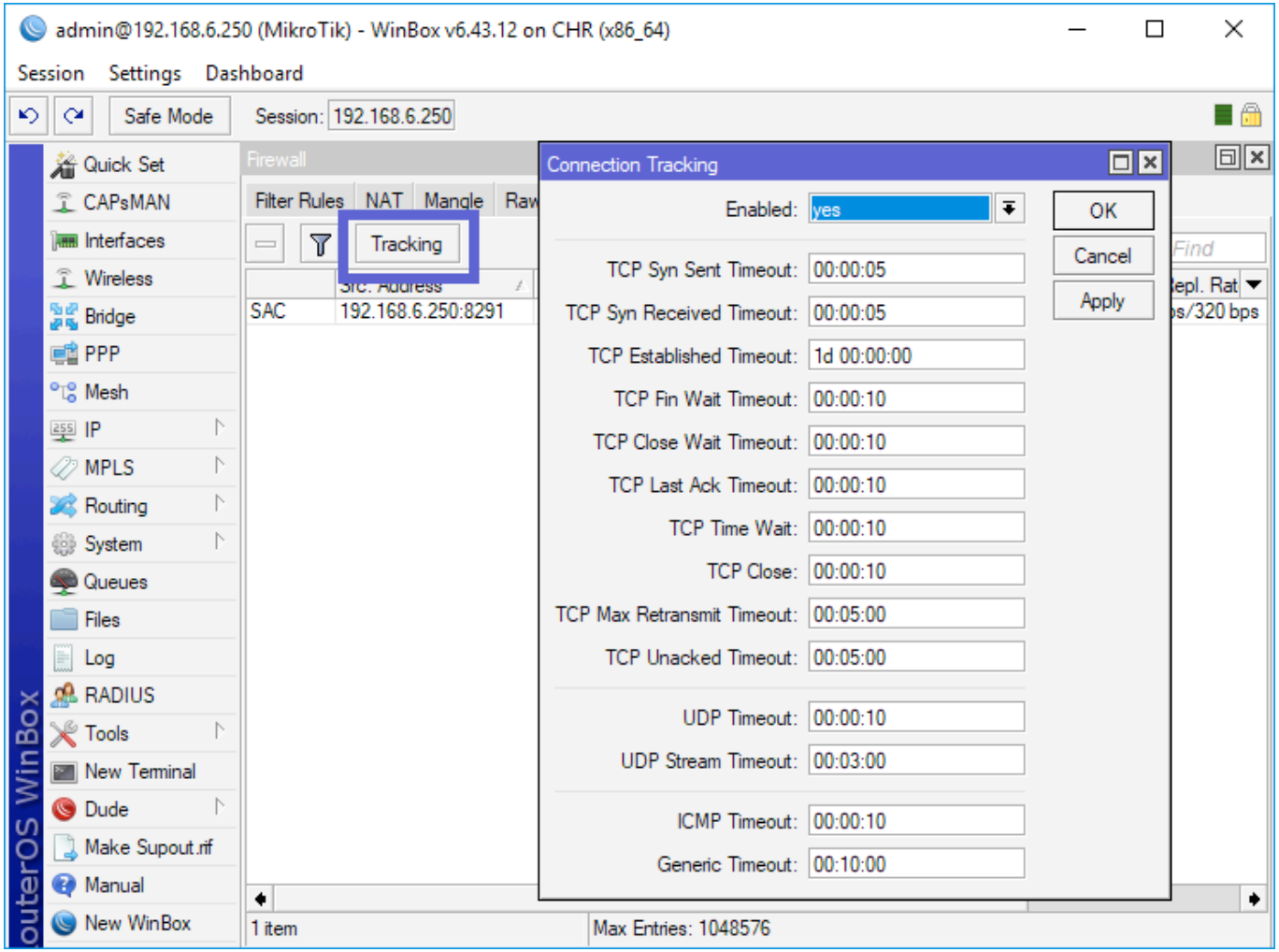
This means that the firewall is able to identify if a packet is related to previous packet.

Firewall can track operating state.

Connection tracking



Connection tracking



Connection tracking

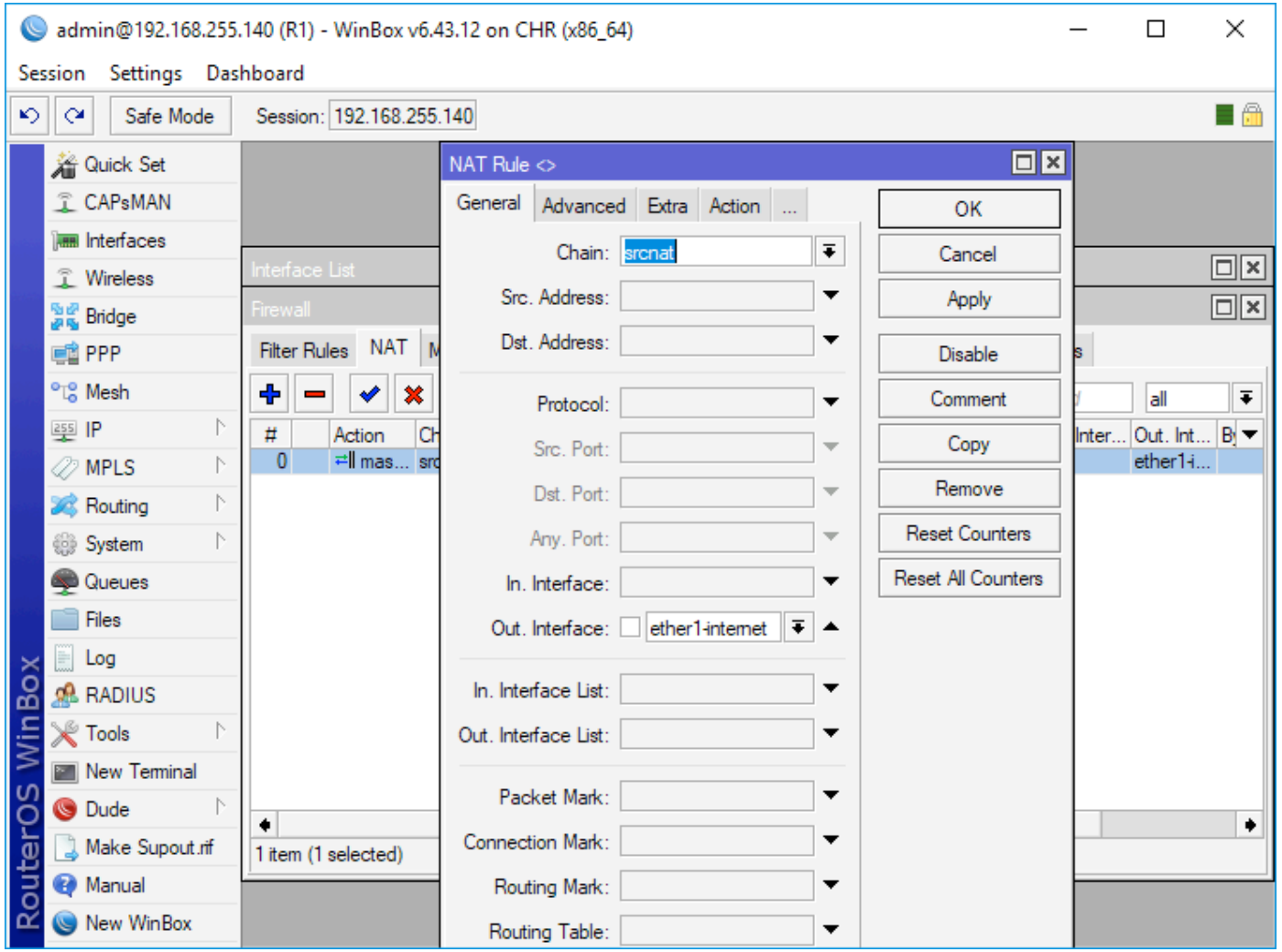
Lab. ICMP tracking

```
/interface ethernet
set [ find default-name=ether1 ] comment="To Internet" name=ether1-
internet
set [ find default-name=ether2 ] comment="To Lan" name=ether2-Lan

/ip pool
add name=dhcp_pool0 ranges=192.168.11.2-192.168.11.254

/ip dhcp-server
add address-pool=dhcp_pool0 disabled=no interface=ether2-Lan
name=dhcp1
```

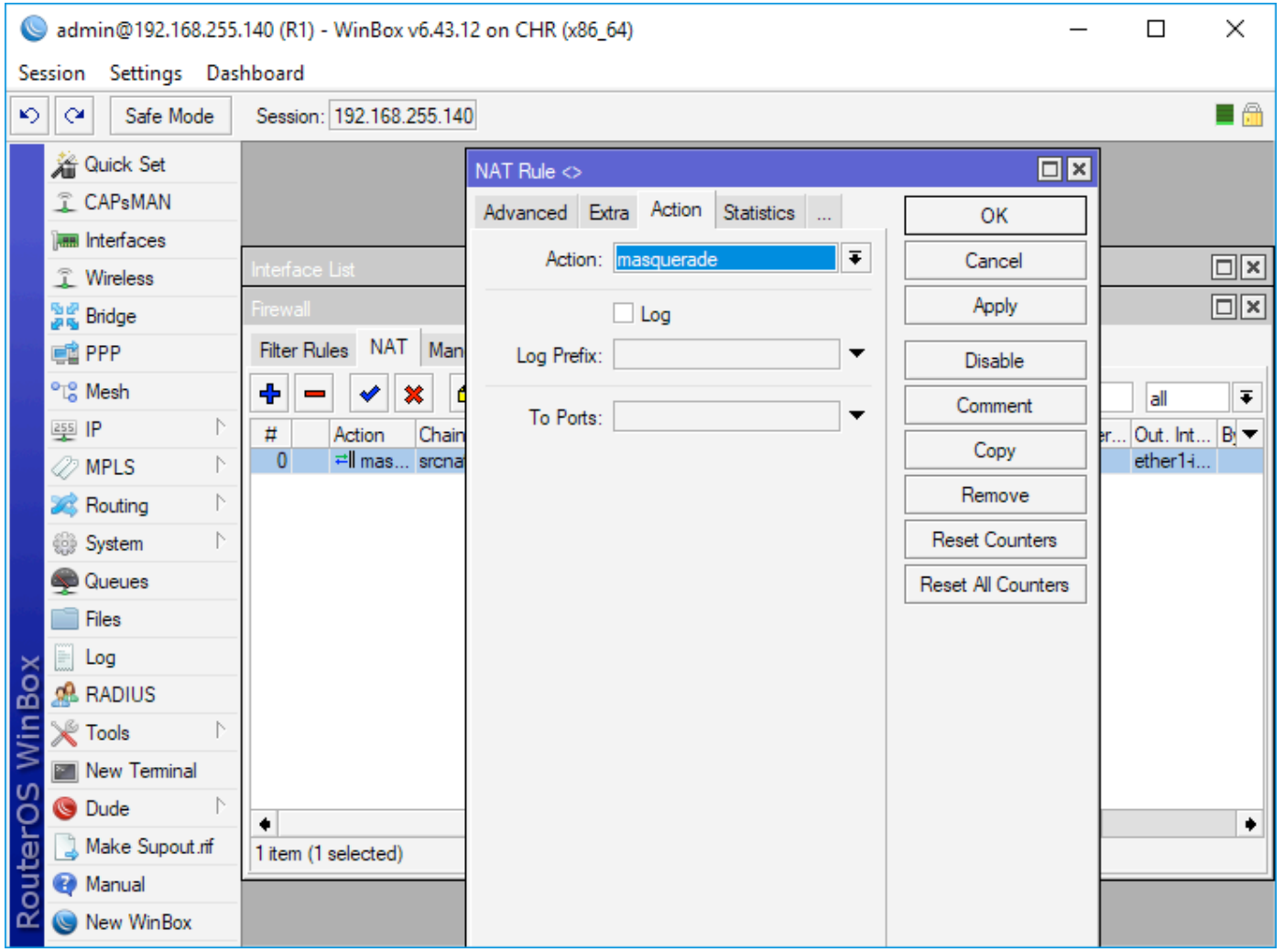
Lab. ICMP tracking

```
/ip address
add address=192.168.11.1/24 interface=ether2-Lan network=192.168.11.0

/ip dhcp-client
add dhcp-options=hostname,clientid disabled=no interface=ether1-
internet

/ip dhcp-server network
add address=192.168.11.0/24 gateway=192.168.11.1

/ip firewall nat
add action=masquerade chain=srcnat out-interface=ether1-internet

/system identity
set name=R1
```

Lab. ICMP tracking



Lab. ICMP tracking



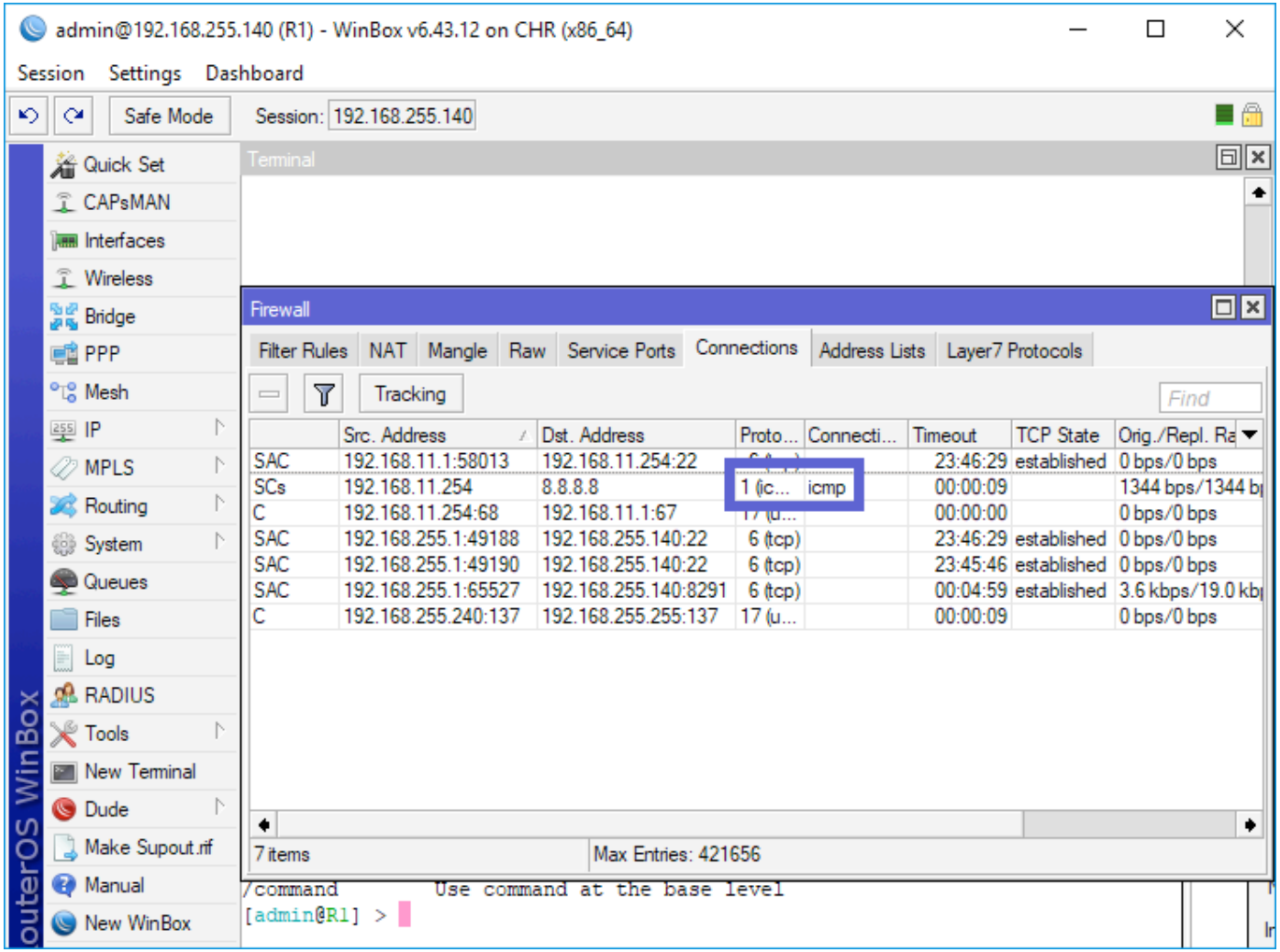
Lab. ICMP tracking

```
/ip firewall mangle
add action=mark-connection chain=forward dst-address=8.8.8.8 new-
connection-mark=icmp passthrough=yes protocol=icmp
add action=mark-packet chain=forward connection-mark=icmp new-packet-
mark=icmpout out-interface=ether1-internet passthrough=yes
add action=mark-packet chain=forward connection-mark=icmp new-packet-
mark=icmpin out-interface=ether2-Lan passthrough=yes
```

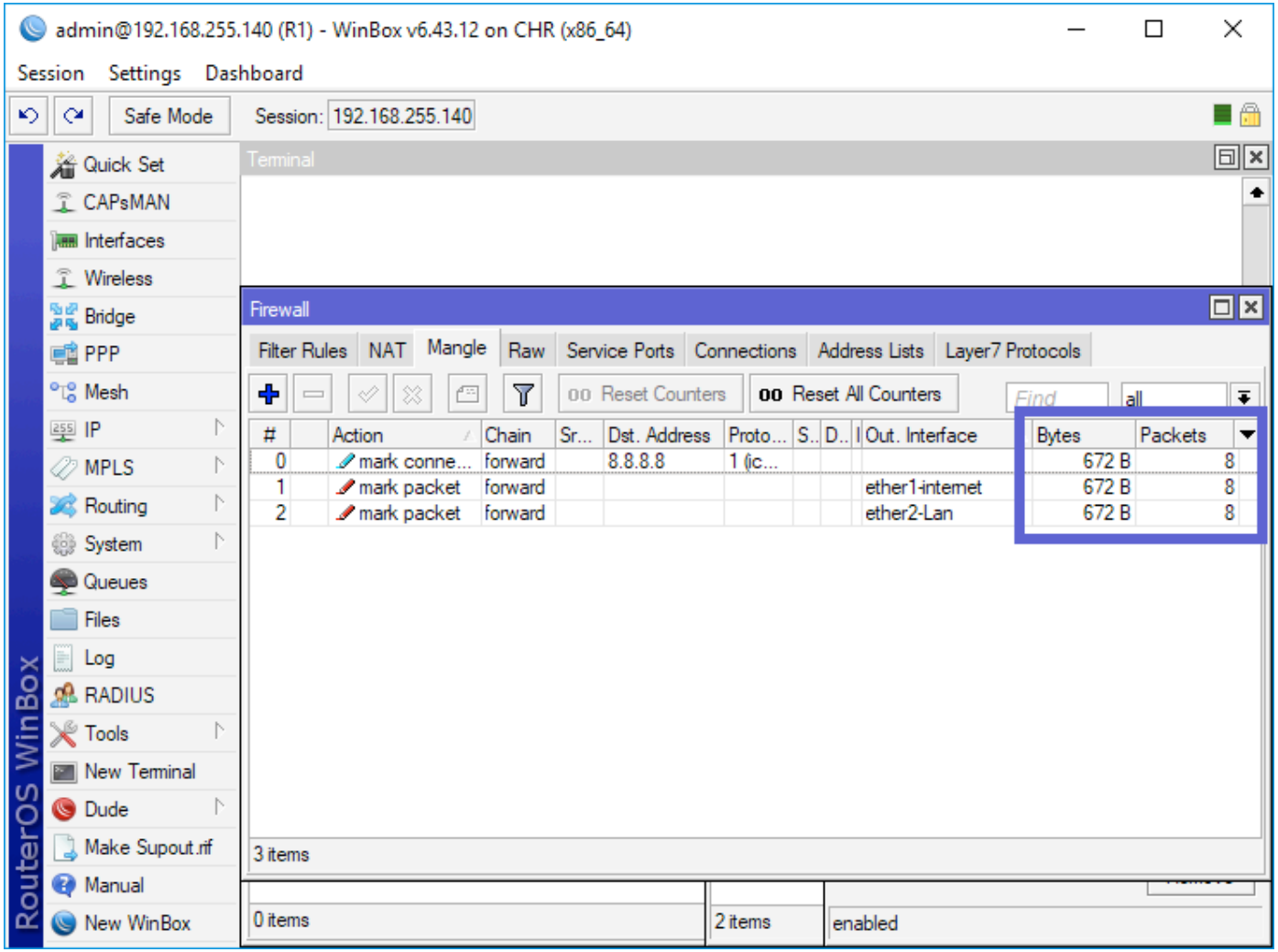
Lab. ICMP tracking

```
/ip firewall mangle
add action=mark-connection chain=forward dst-address=8.8.8.8
new-connection-mark=icmp passthrough=yes protocol=icmp
```

Lab. ICMP tracking



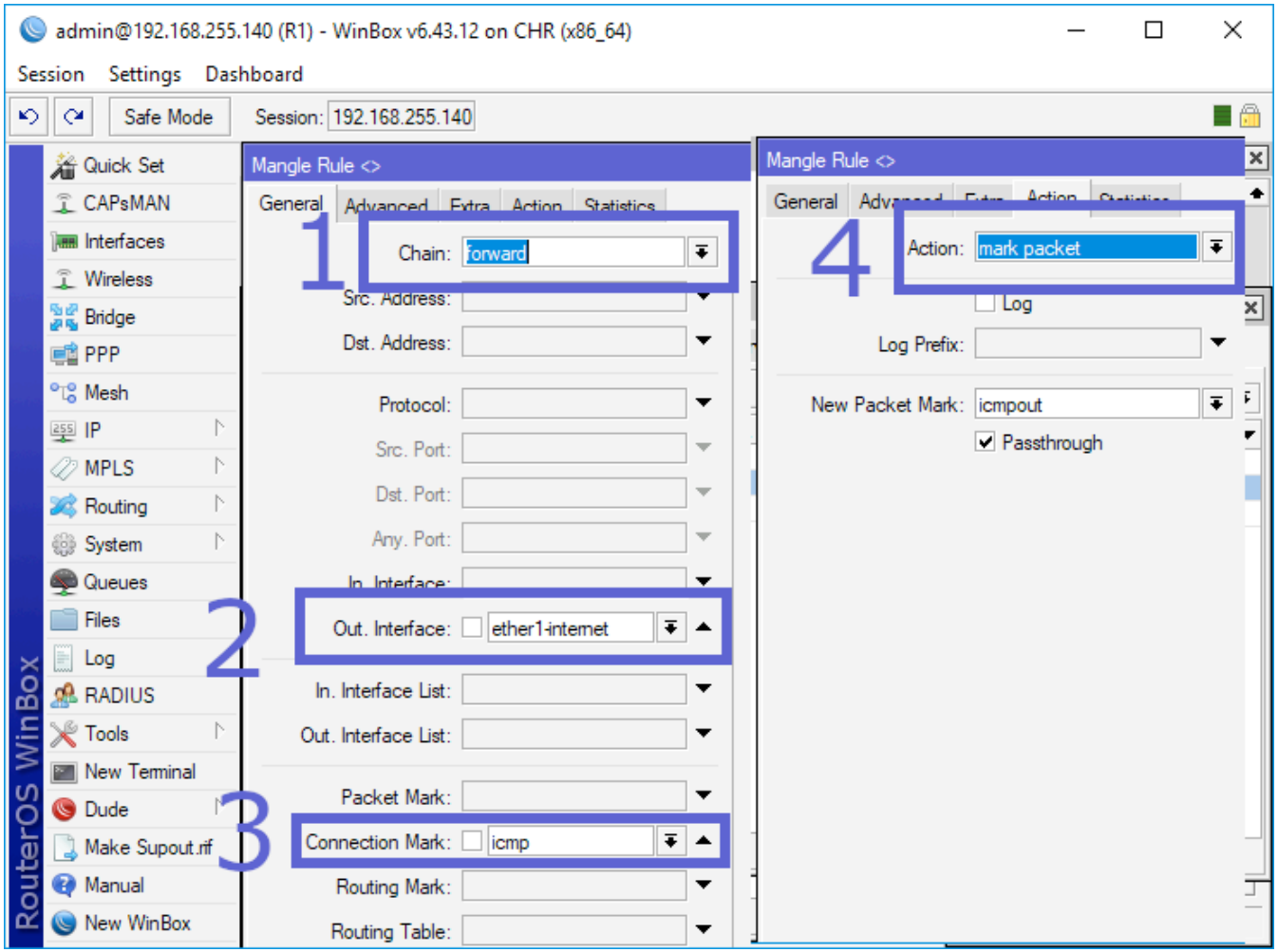
Lab. ICMP tracking



Lab. ICMP tracking

```
/ip firewall mangle
add action=mark-packet chain=forward connection-mark=icmp new-packet-
mark=icmpout out-interface=ether1-internet passthrough=yes
```

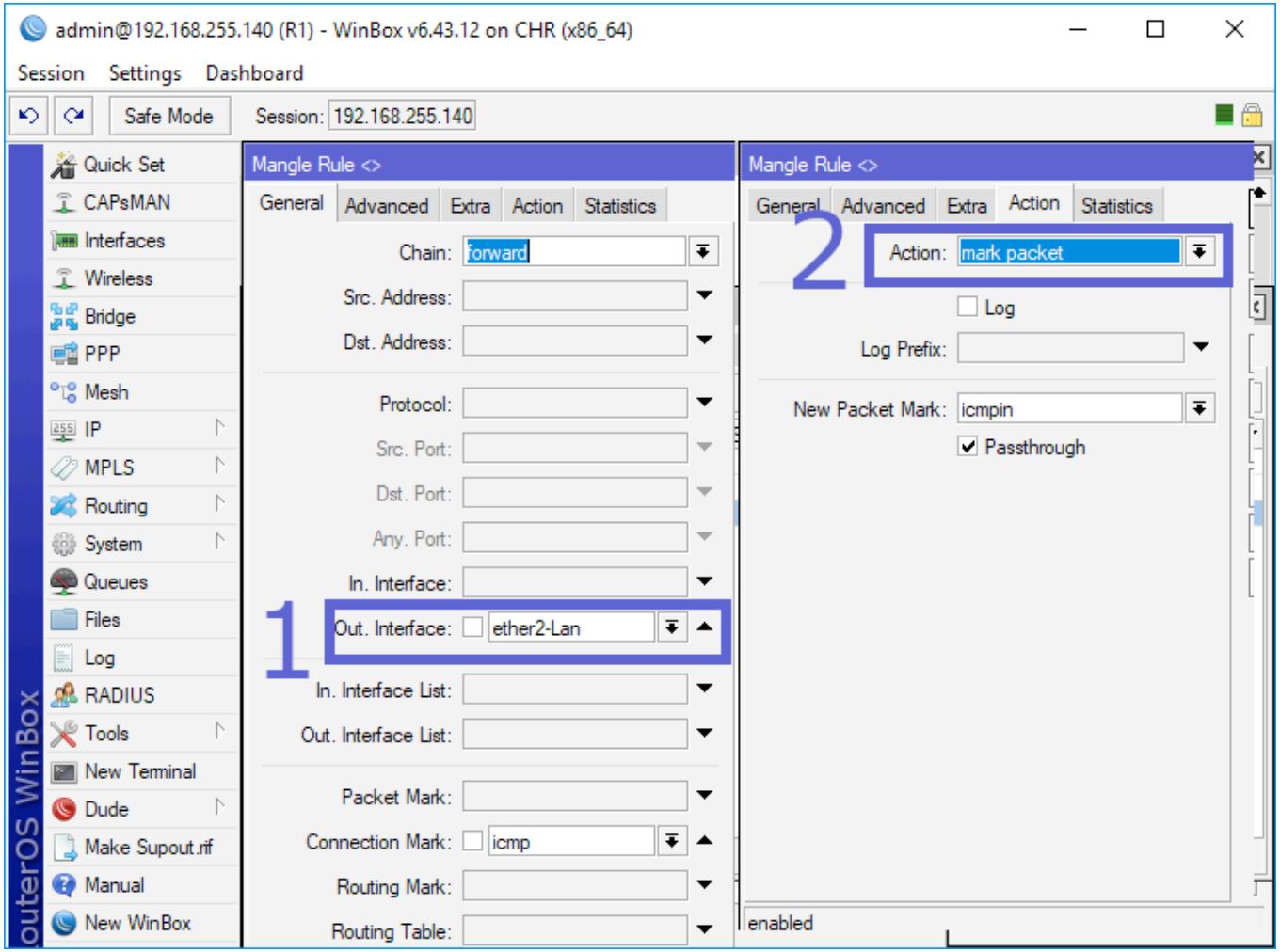
Lab. ICMP tracking



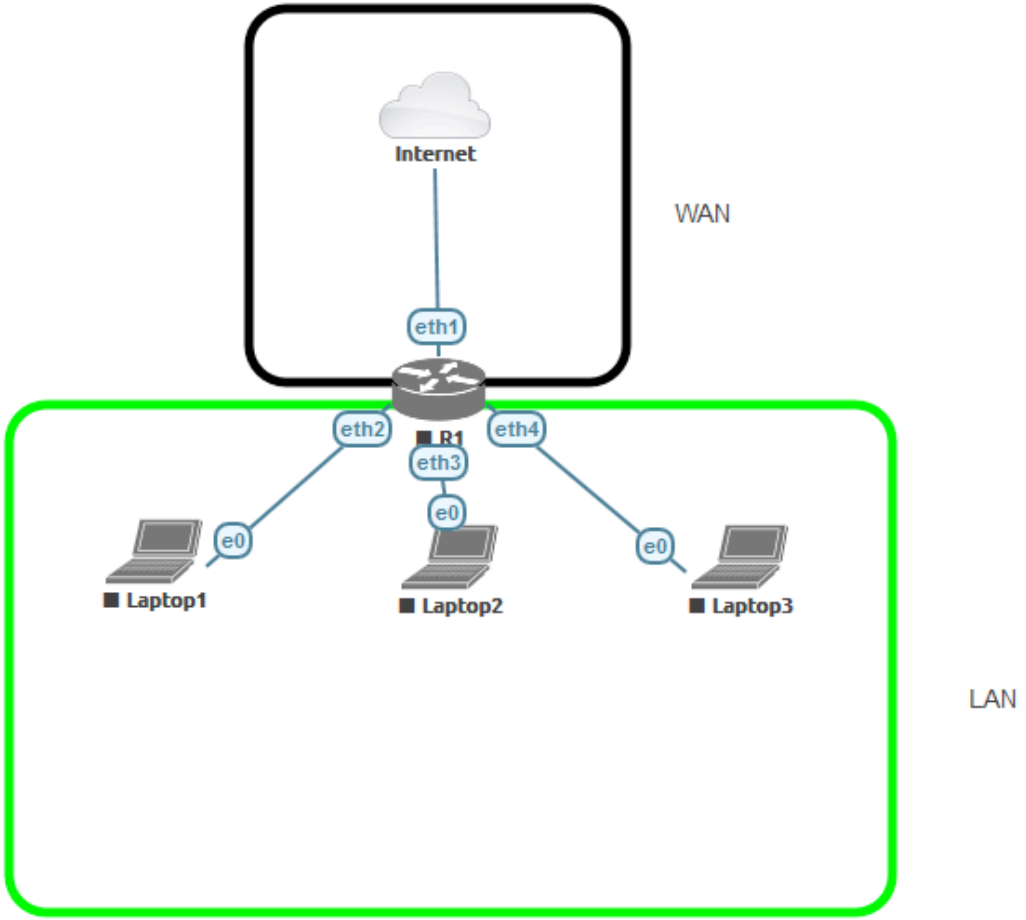
Lab. ICMP tracking

```
/ip firewall mangle
add action=mark-packet chain=forward connection-mark=icmp new-packet-
mark=icmpin out-interface=ether2-Lan passthrough=yes
```

Lab. ICMP tracking



Lab. Securing areas



Lab. Securing areas

```
/interface bridge
add fast-forward=no name=Lan

/interface ethernet
set [ find default-name=ether1 ] name=E1-ToInternet

/interface list
add name=WAN
add name=LAN
```

Lab. Securing areas

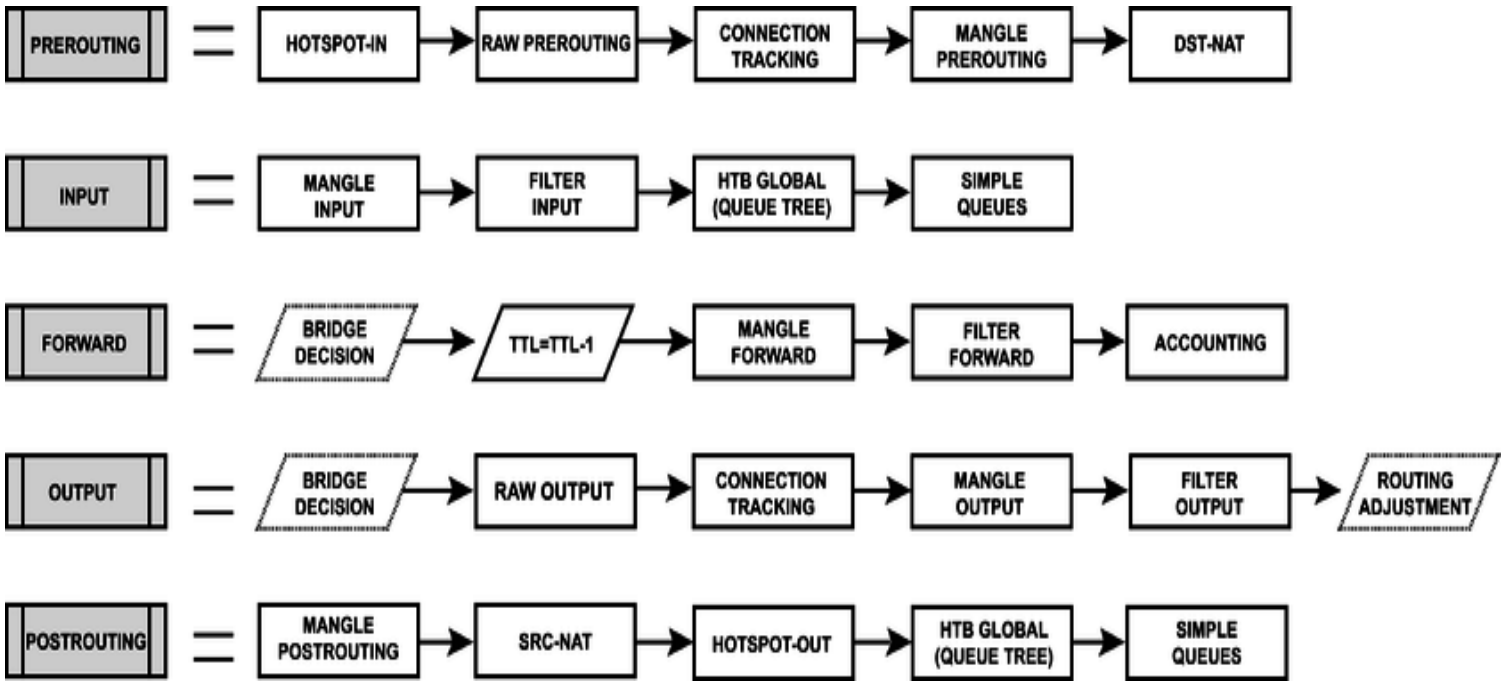
```
/ip pool
add name=dhcp_pool0 ranges=192.168.188.2-192.168.188.254

/ip dhcp-server
add address-pool=dhcp_pool0 disabled=no interface=Lan name=dhcp1

/interface bridge port
add bridge=Lan interface=ether2
add bridge=Lan interface=ether3
add bridge=Lan interface=ether4

/interface list member
add interface=E1-ToInternet list=WAN
add interface=Lan list=LAN
```

Lab. Securing areas

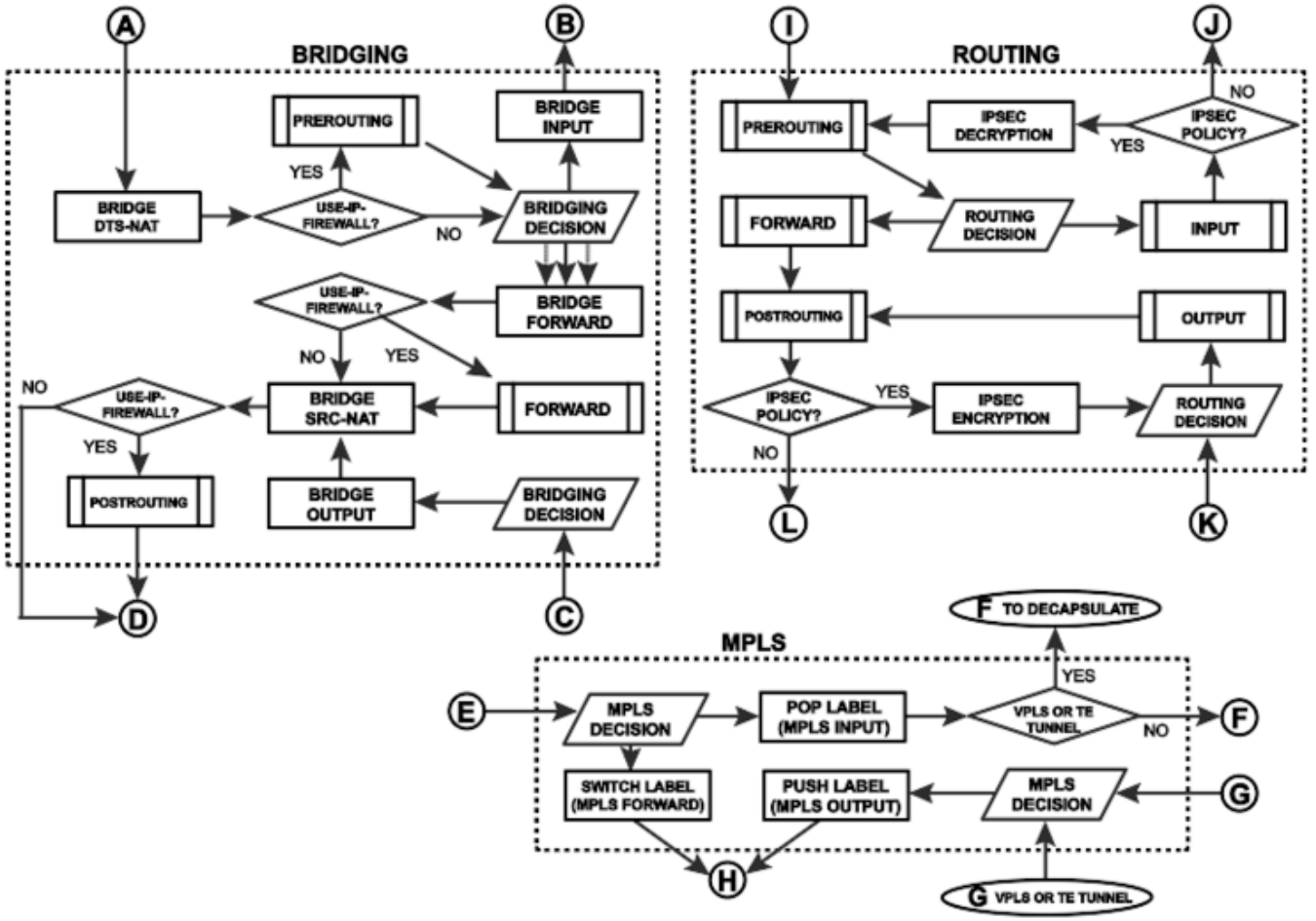
```
/ip address
add address=192.168.188.1/24 interface=Lan network=192.168.188.0

/ip dhcp-client
add dhcp-options=hostname,clientid disabled=no interface=E1-
ToInternet

/ip dhcp-server network
add address=192.168.188.0/24 gateway=192.168.188.1

/ip firewall filter
add action=drop chain=forward comment="Drop external traffic"
connection-state=new in-interface-list=WAN

/ip firewall nat
add action=masquerade chain=srcnat out-interface-list=WAN

/system identity
set name=R1
```

PACKET FLOW

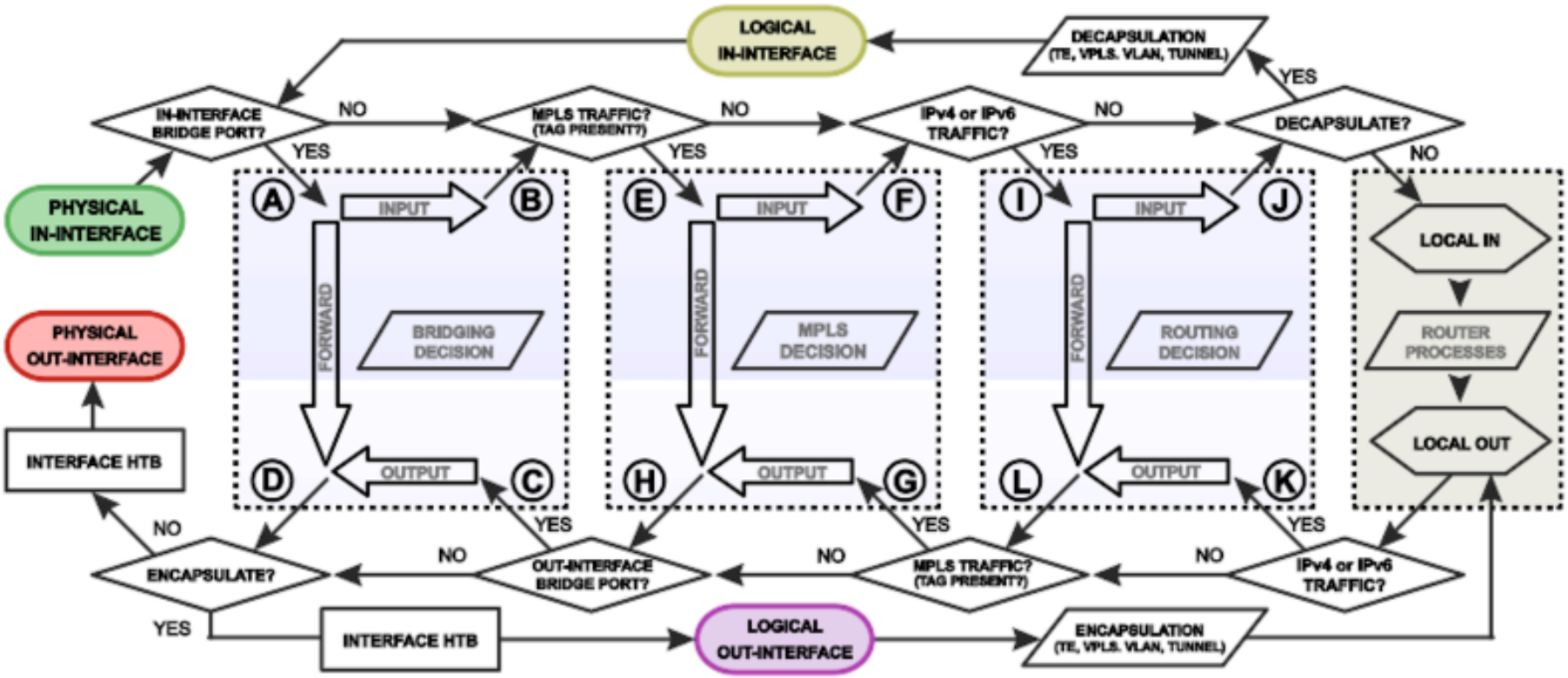
Packet flow



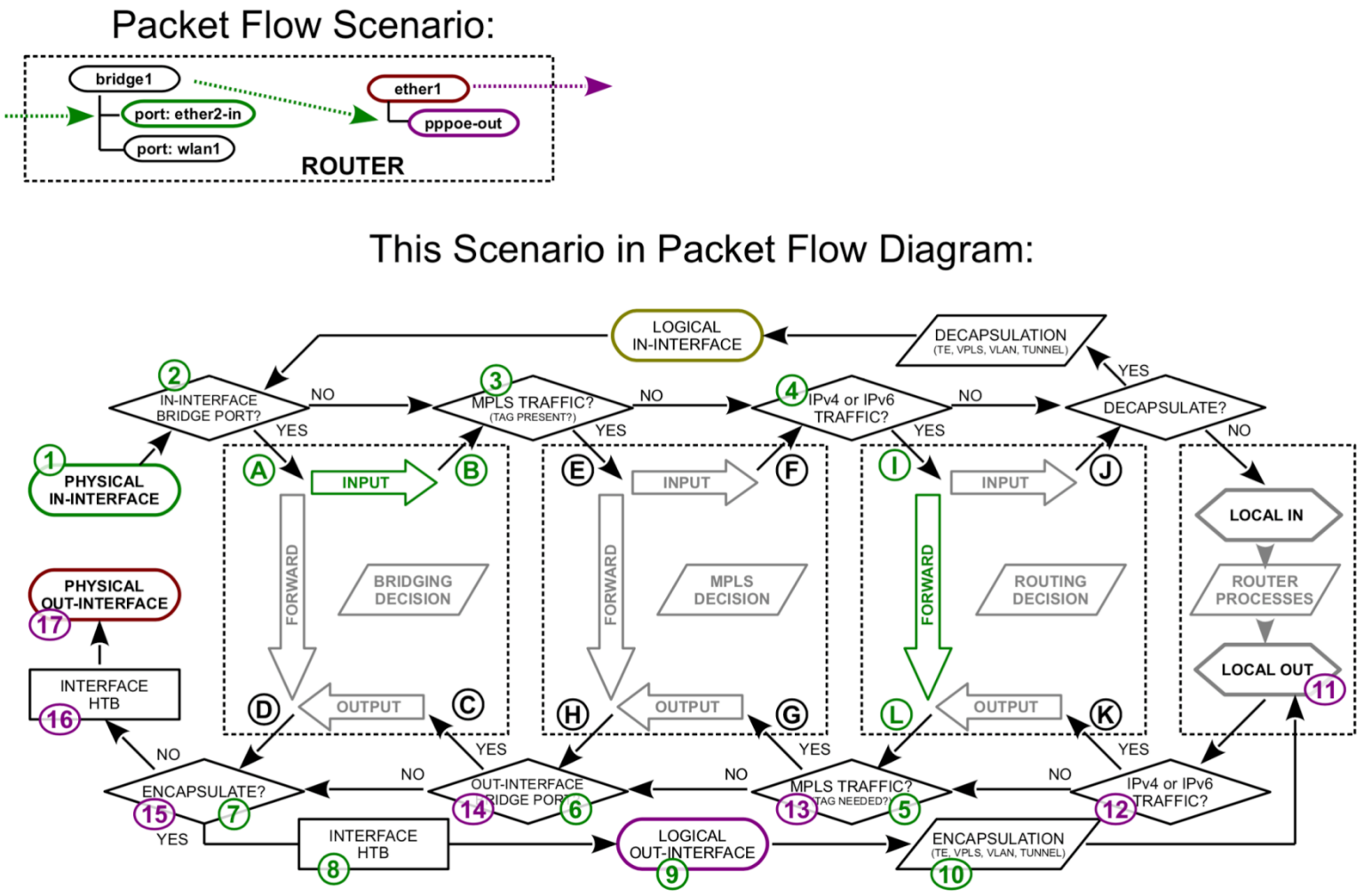
Packet flow



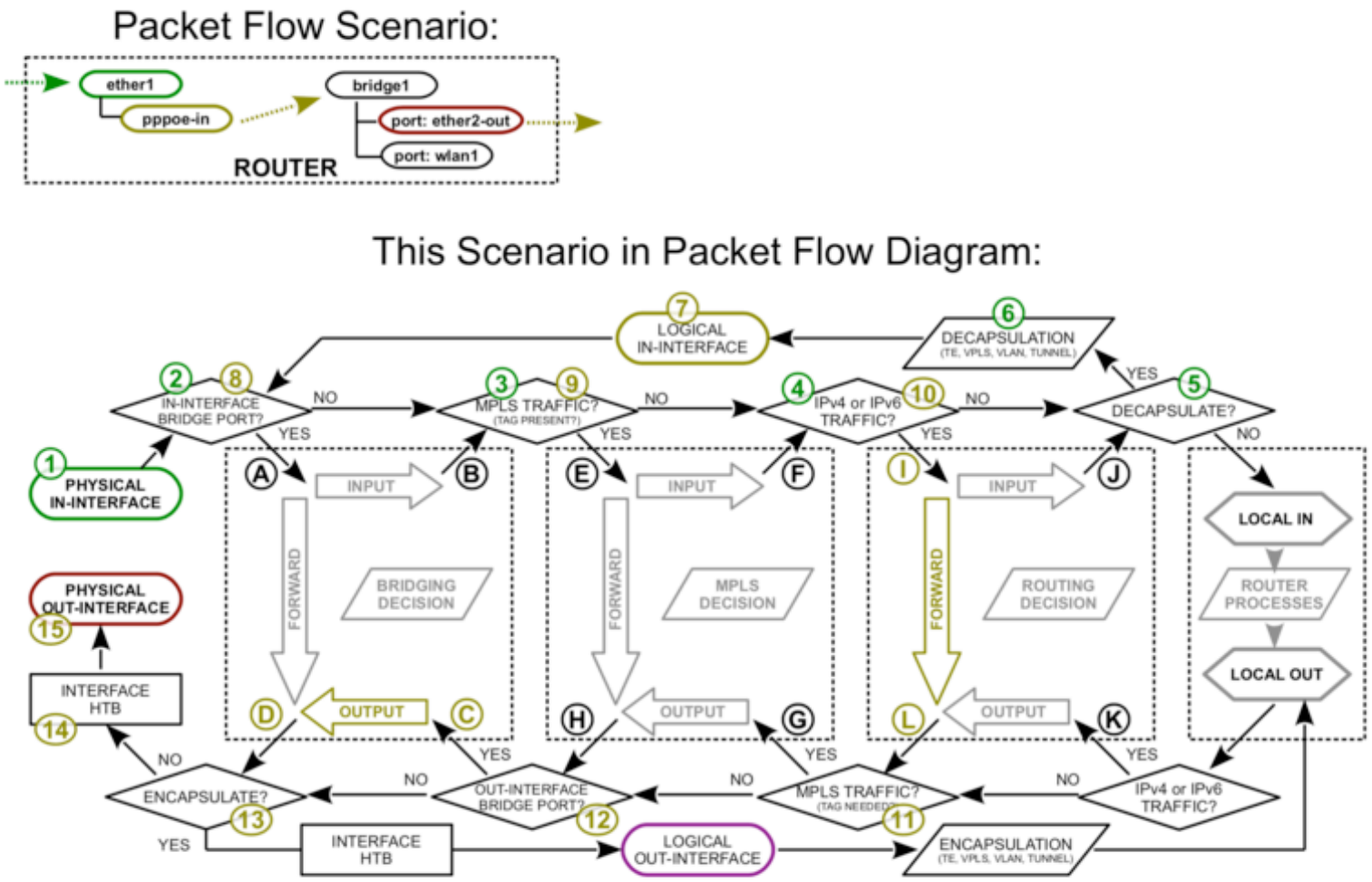
Packet flow



Packet flow



Packet flow

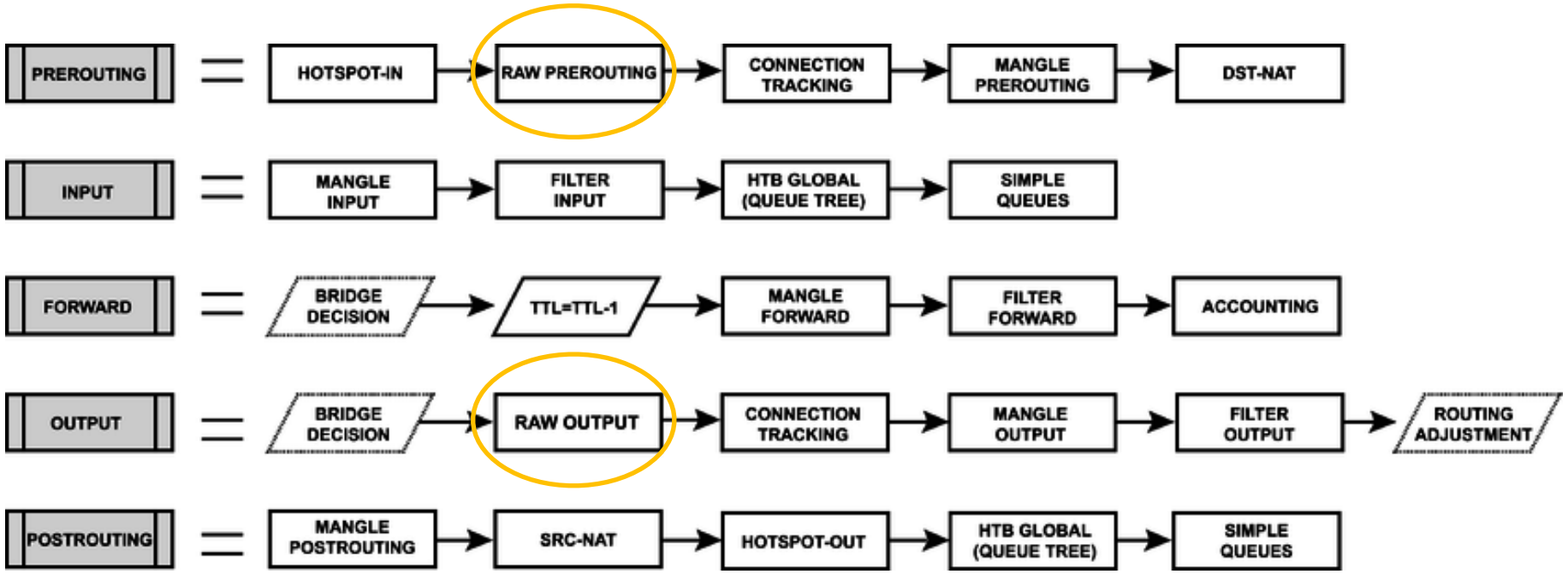


RAW TABLE

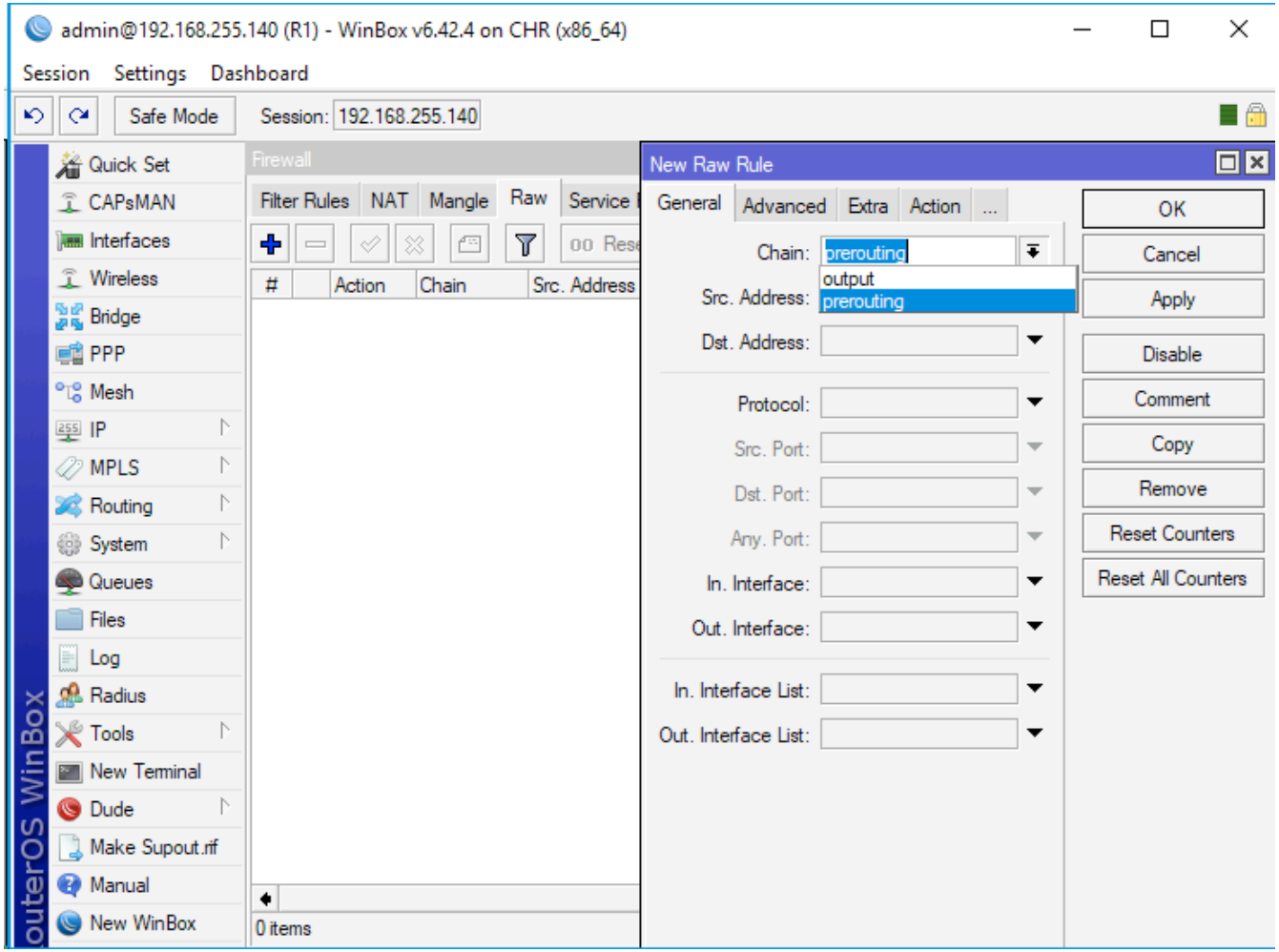
Raw table

- Raw table offer two chains - prerouting and output.
- The function of the raw table is to process the packets before the conntrack process.
- This is much more efficient.

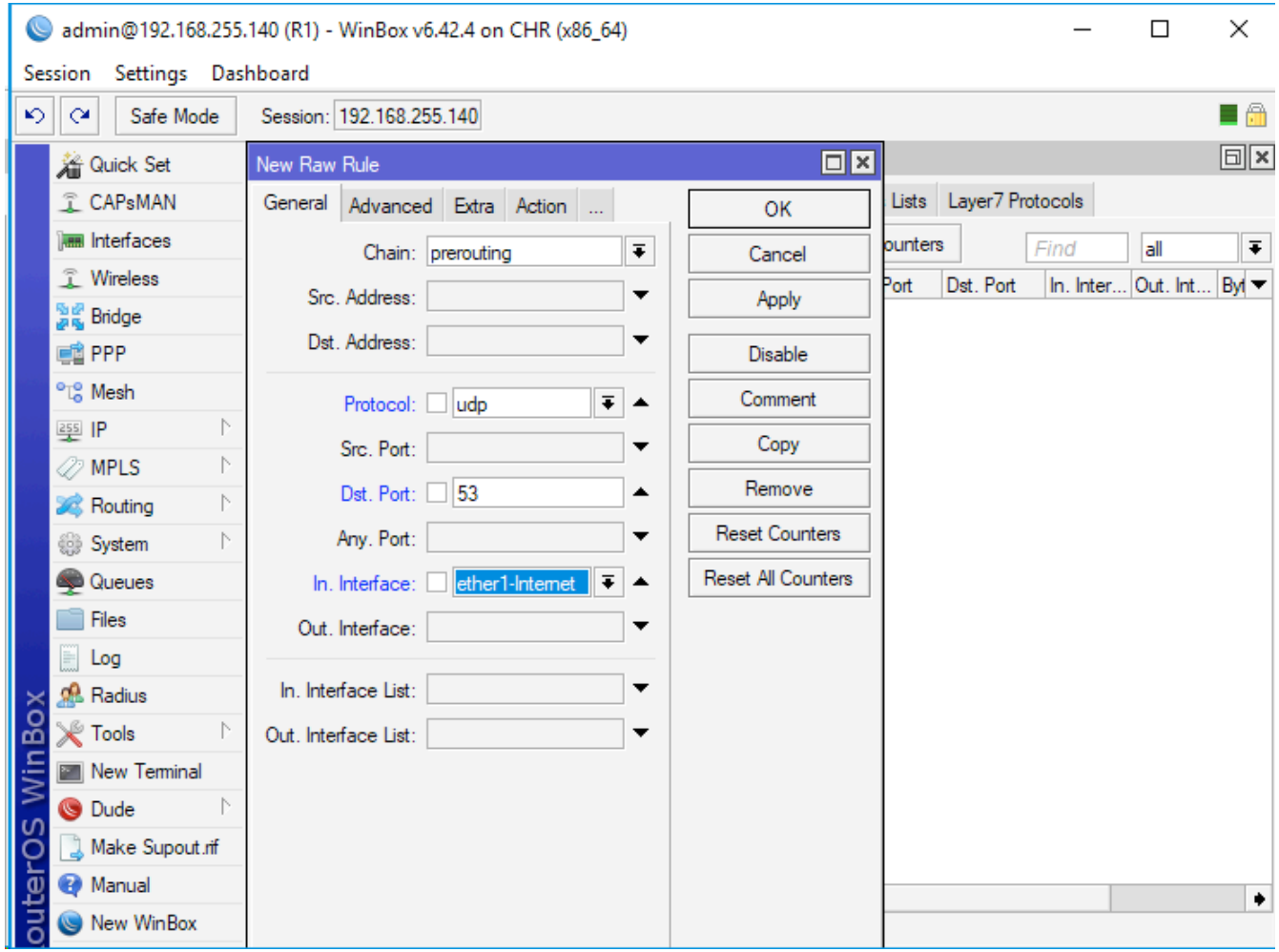
Raw table chains



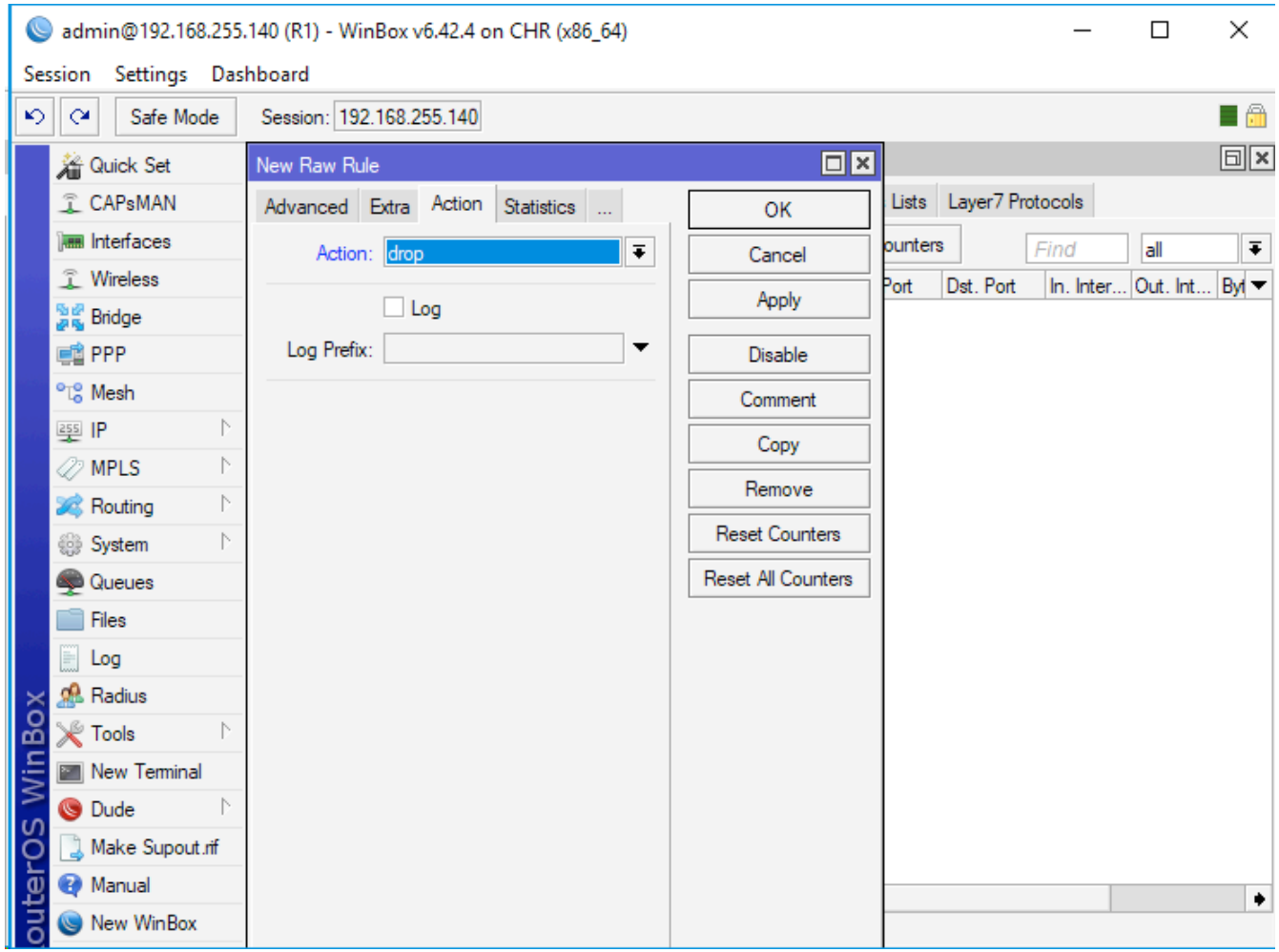
Raw table



Raw table. Drop packets



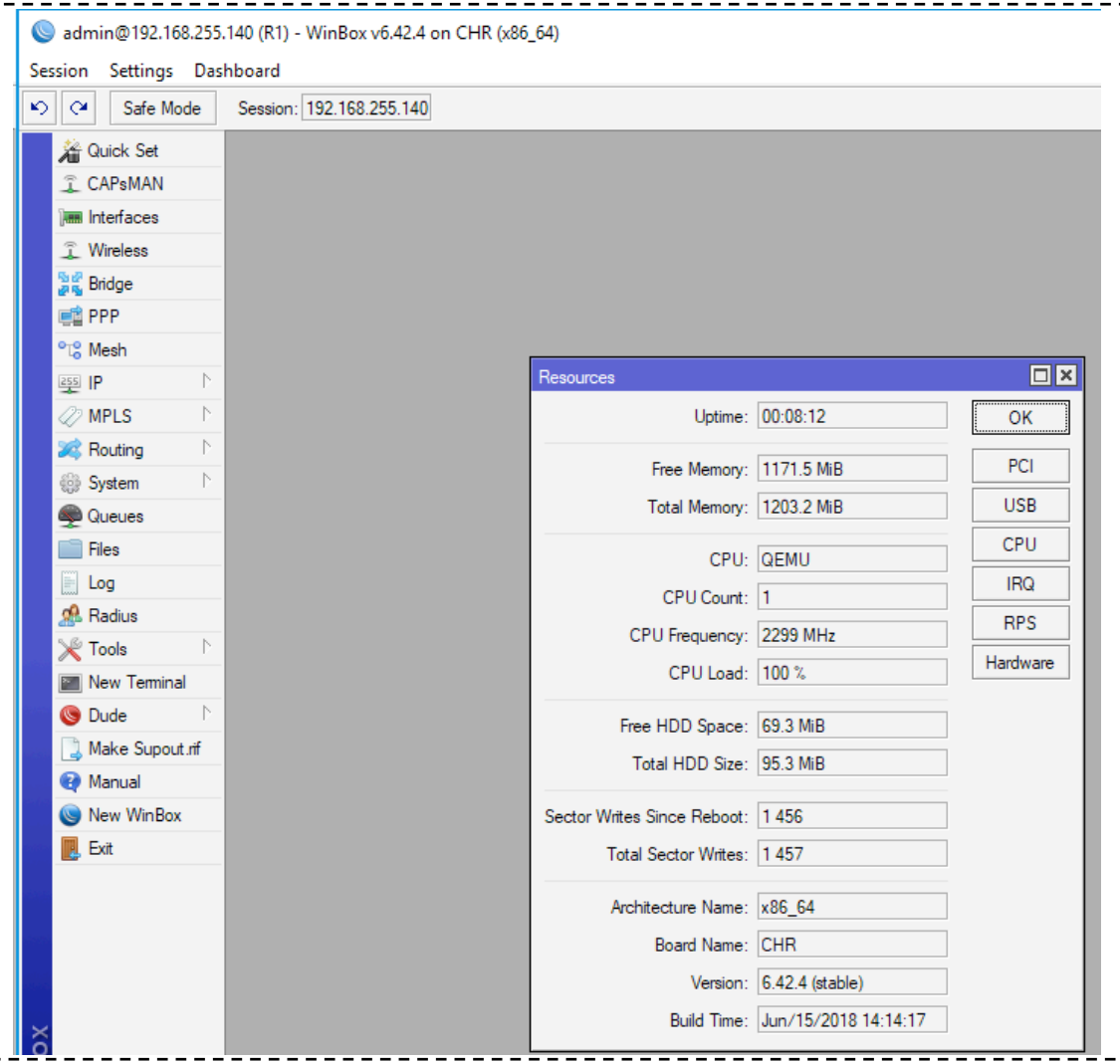
Raw table. Drop packets



Raw table. Synflood attack

```
/ip firewall filter
add action=drop chain=input protocol=tcp tcp-flags=syn in-
interface=E1-ToInternet
```

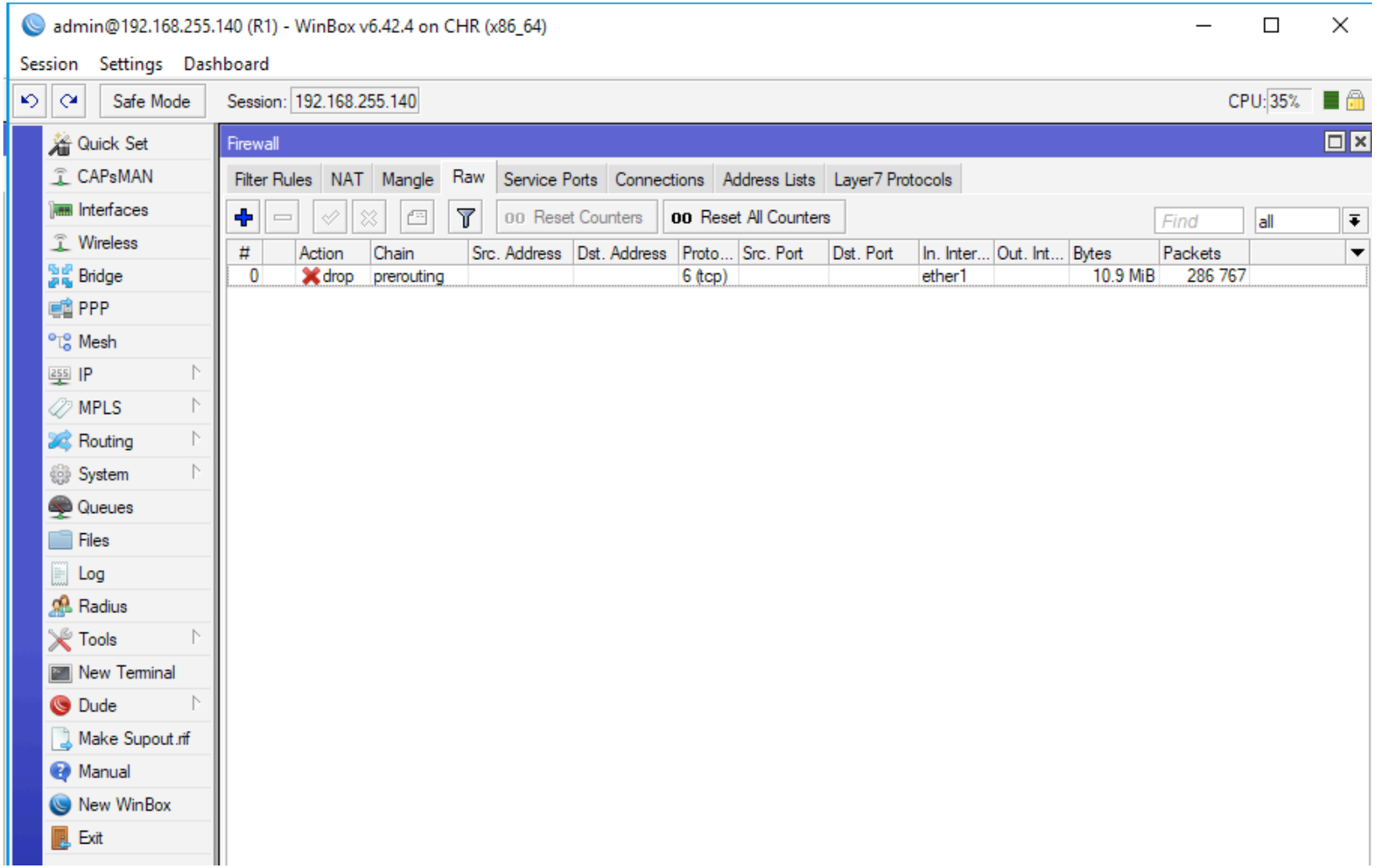
Raw table. Synflood attack



Raw table. Synflood attack

```
/ip firewall raw
chain=input action=drop tcp-flags=syn protocol=tcp in-interface=E1-
ToInternet
```

Raw table. Synflood attack



Raw table. Synflood attack

Test it on your router!

DEFAULT CONFIGURATION

MikroTik Default Configuration

- All RouterBOARDs from factory come with a default configuration. There are several different configurations depending on the board type:
  - CPE router
  - LTE CPE AP router
  - AP router (single or dual band)
  - PTP Bridge (AP or CPE)
  - WISP Bridge (AP in ap_bridge mode)
  - Switch
  - IP only
  - CAP (Controlled Access Point)
- When should you remove the default-configuration and set up the router from scratch?

CPE Router

- In this type of configurations router is configured as wireless client device.
- WAN interface is Wireless interface.
- WAN port has configured DHCP client, is protected by IP firewall and MAC discovery/connection is disabled.

CPE Router

- List of routers using this type of configuration:
  - RB711, 911, 912, 921, 922 - with Level3 (CPE) license
  - SXT
  - QRT
  - SEXTANT
  - LHG
  - LDF
  - DISC
  - Groove
  - Metal

LTE CPE AP router

- This configuration type is applied to routers that have both an LTE and a wireless interface.
- The LTE interface is considered as a WAN port protected by the firewall and MAC discovery/connection disabled.
- IP address on the WAN port is acquired automatically. Wireless is configured as an access point and bridged with all available Ethernet ports.
- List of routers using this type of configuration:
  - wAP LTE kit
  - LtAP mini kit

AP Router (single or dual band)

- This type of configuration is applied to home access point routers to be used straight out of the box without additional configuration (except router and wireless passwords)
- First Ethernet port is configured as a WAN port (protected by firewall, with a DHCP client and disabled MAC connection/ discovery)
- Other Ethernet ports and wireless interfaces are added to local LAN bridge with an IP 192.168.88.1/24 and a DHCP server
- In case of dual band routers, one wireless is configured as 5 GHz access point and the other as 2.4 GHz access point.
- List of routers using this type of configuration:
  - RB450, 751, 850, 951, 953, 2011, 3011, 4011
  - mAP, wAP, hAP, OmniTIK

PTP Bridge (AP or CPE)

- Bridged ethernet with wireless interface
- Default IP address 192.168.88.1/24 is set on the bridge interface
- There are two possible options - as CPE and as AP
  - For CPE wireless interface is set in "station-bridge" mode.
  - For AP "bridge" mode is used.
- List of routers using this type of configuration:
  - DynaDish - as CPE

WISP Bridge

- Configuration is the same as PTP Bridge in AP mode, except that wireless mode is set to ap_bridge for PTMP setups.
- Router can be accessed directly using MAC address.
- If device is connected to the network with enabled DHCP server, configured DHCP client configured on the bridge interface will get the IP address, that can be used to access the router.
- List of routers using this type of configuration:
  - RB 911,912,921,922 - with Level4 license
  - cAP, Groove A, Metal A, RB711 A
  - BaseBox, NetBox
  - mANTBox, NetMetal

Switch

- This configuration utilises switch chip features to configure dumb switch.

- All ethernet ports are added to switch group and default IP address 192.168.88.1/24 is set on master port.
- RoS 6.41 onwards uses Hardware Offload and places all ports into a Bridge instead.
- List of routers using this type of configuration:
  - FiberBox
  - CRS without wireless interface

IP Only

- When no specific configuration is found, IP address 192.168.88.1/24 is set on ether1, or combo1, or sfp1.
- List of routers using this type of configuration:
  - RB 411,433,435,493,800,M11,M33,1100
  - CCR

CAP

- This type of configuration is used when device is to be used as a wireless access point which is controlled by the CAPsMAN
- When CAP default configuration is loaded, ether1 is considered as a management port with a DHCP client
- All other Ethernet interfaces are bridged and all wireless interfaces are set to be managed by the CAPsMAN
- None of the current boards come with the CAP mode enabled from the factory. The above mentioned configuration is applied to all boards with at least one wireless interfaces when set to the CAP mode

IPv6

- Note. The IPv6 package by default is disabled on RouterOS v6. When enabled, after the first reboot, default configuration will be applied to the IPv6 firewall as well.

Print the factory default-configuration

- /system default-configuration print

IP firewall to a router

- Work with new connections to decrease load on a router;
- Create address-list for IP addresses that are allowed to access your router;
- Enable ICMP access (optionally);
- Drop everything else, log=yes might be added to log packets that hit the specific rule;

IP firewall for clients

- Established/related packets are added to fasttrack** for faster data throughput
  - firewall will work with new connections only;
- Drop invalid connection and log them with prefix invalid;
- Drop attempts to reach non public addresses from your local network (rfc1918) (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16)
  - drop forward dst-address-list=not_in_internet
  - bridge1 is local network interface
  - log attempts with prefix="!public_from_LAN";

  ** note Fasttrack limitations for Queues etc

IP firewall for clients

- Drop incoming packets that are not NATed,
  - ether1 is public interface, log attempts with !NAT prefix;
- Drop incoming packets from Internet, which are not public IP addresses (rfc1918),
  - ether1 is public interface,
  - log attempts with prefix="!public";
- Drop packets from LAN that does not have LAN IP,
  - 192.168.88.0/24 is local network used subnet;

MANAGEMENT ACCESS

RouterOS services

- /ip service disable telnet,ftp,www,api,api-ssl

Change default ports

- /ip service set ssh port=2200

Restrict access by ip

- /ip service set winbox address=192.168.88.0/24

Mac-server

RouterOS has built-in options for easy management access to network devices even without IP configuration. On production networks the particular services should be set to restricted access (e.g. only internal interfaces) or disable entirely!

```
/tool mac-server set allowed-interface-list=none
/tool mac-server mac-winbox set allowed-interface-list=none
/tool mac-server ping set enabled=no
```

Bandwidth Test

Bandwidth test server is used to test throughput between two MikroTik routers. It is recommended to disable it on a production environment.

```
/tool bandwidth-server set enabled=no
```

DNS Cache

DNS cache facility can be used to provide domain name resolution for the router itself as well as for the clients connected to it.

In case the DNS cache is not required on your router or if another router is used for such purposes, DNS cache should be disabled:

/ip dns set allow-remote-requests=no

If DNS cache is left enabled be sure to protect UDP/53 on the input chain with firewall rules

Other Client Services

```
/ip proxy set enabled=no
/ip socks set enabled=no
/ip upnp set enabled=no
/ip cloud set ddns-enabled=no update-time=no
```

More Secure SSH - Strong-Crypto=Yes

Introduces following changes in the SSH configuration:

- Prefer 256 and 192 bit encryption instead of 128 bits
- Disable null encryption
- Prefer sha256 for hashing instead of sha1
- Disable md5
- Use 2048bit prime for Diffie Hellman exchange instead of 1024bit

/ip ssh set strong-crypto=yes

Unused interfaces

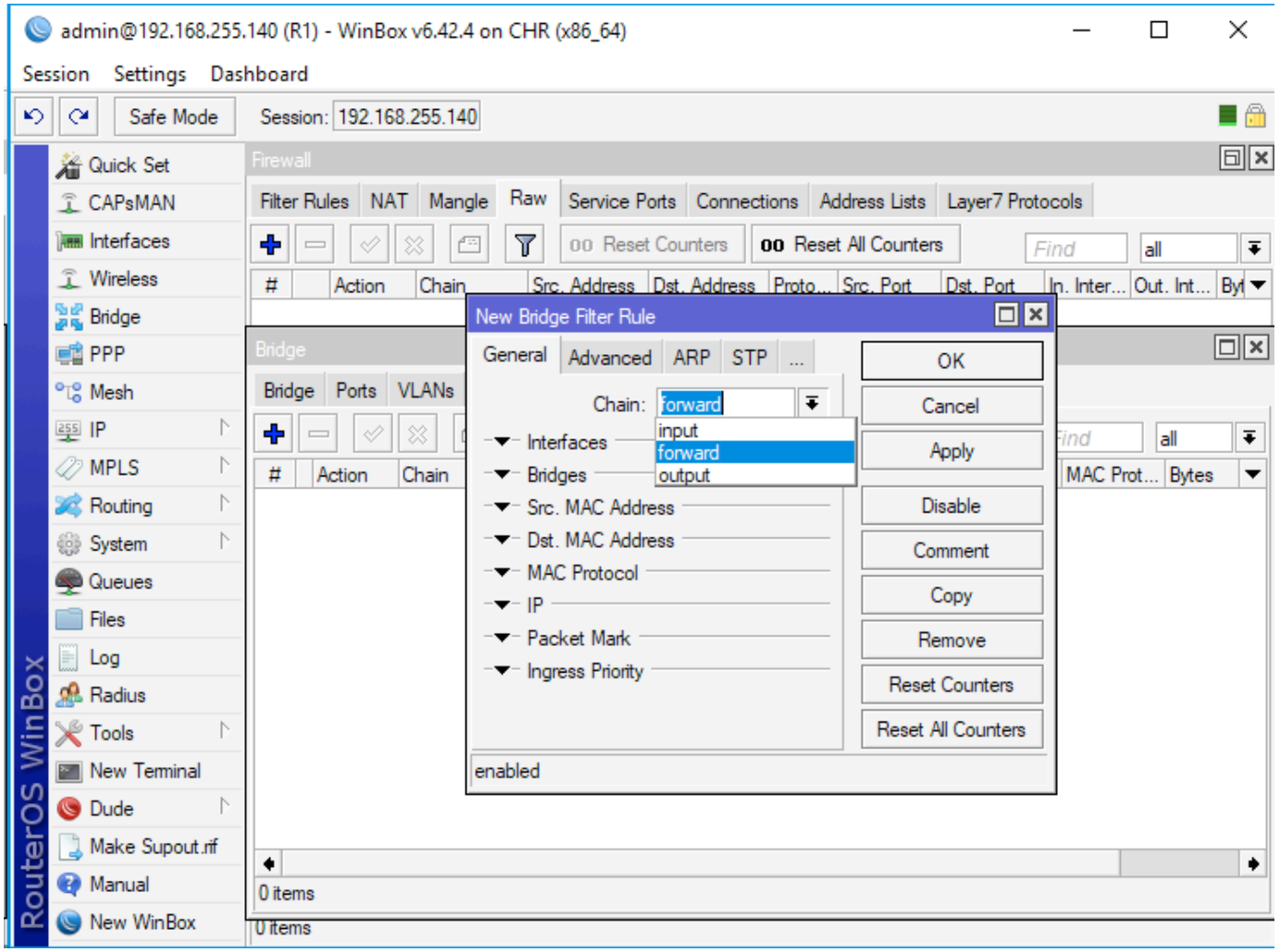
In order to protect from unauthorised access, it is considered good practice to disable all unused interfaces on the router

BRIDGE FIREWALL

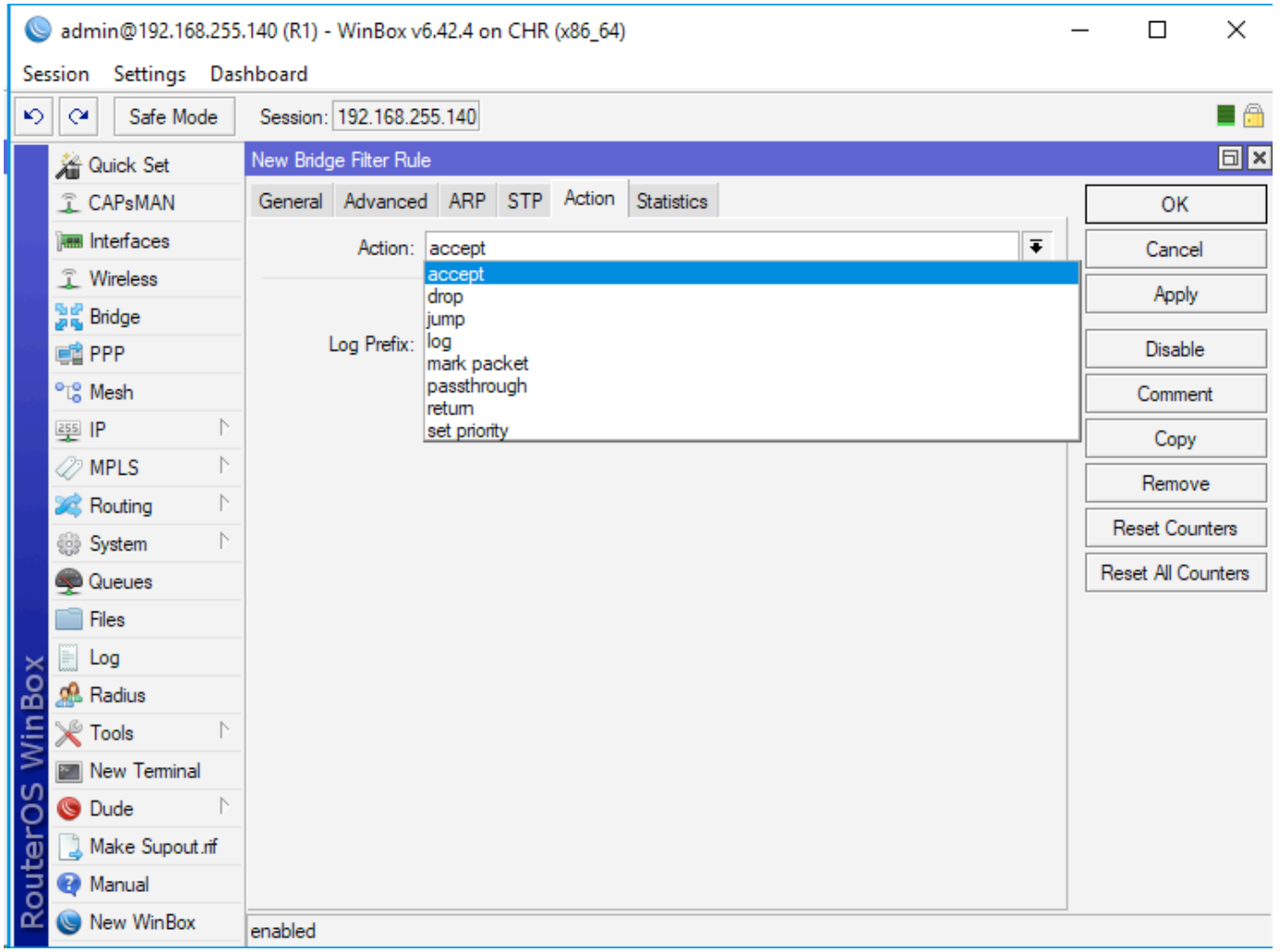
Bridge Firewall

The bridge firewall implements packet filtering and thereby provides security functions that are used to manage data flow to, from and through bridge.

Bridge Firewall



Bridge Firewall



Lab. Only PPPoE Traffic

Lab. Only PPPoE Traffic

R1 Setup (PPPoE Server)

```
/interface ethernet
set [ find default-name=ether1 ] name=E1-ToBridge

/ip address
add address=192.168.100.1/30 interface=E1-ToBridge
network=192.168.100.0
```

Lab. Only PPPoE Traffic

```
/interface pppoe-server server
add disabled=no interface=E1-ToBridge

/ppp secret
add local-address=10.100.100.1 name=test password=test \
remote-address=10.200.200.2 service=pppoe

/system identity
set name=R1
```

Lab. Only PPPoE Traffic

R3 Setup (PPPoE Client)

```
/interface ethernet
set [ find default-name=ether1 ] name=E1-ToBridge

/interface pppoe-client
add disabled=no interface=E1-ToBridge name=test password=test \
user=test

/ip address
add address=192.168.100.2/30 interface=E1-ToBridge \
network=192.168.100.0

/system identity set name=R3
```

Lab. Only PPPoE Traffic

Bridge Setup

```
/interface bridge
add name=bridge1

/interface ethernet
set [ find default-name=ether2 ] name=E2-ToR1
set [ find default-name=ether3 ] name=E3-ToR3

/interface bridge filter
add action=accept chain=forward mac-protocol=pppoe
add action=accept chain=forward mac-protocol=pppoe-discovery
add action=drop chain=forward
```

Lab. Only PPPoE Traffic

```
/interface bridge port
add bridge=bridge1 interface=E2-ToR1
add bridge=bridge1 interface=E3-ToR3

/system identity
set name=Bridge
```

ICMP FILTERING

What is ICMP Filtering

- ICMP helps networks to cope with communication problems
- No authentication method; can be used by hackers to crash computers on the network
- Firewall/packet filter must be able to determine, based on its message type, whether an ICMP packet should be allowed to pass

ICMPv4 FILTERING

Table Filtering Recommendations

ICMPv4 Message	Sourced from Device	Through Device	Destined to Device
ICMPv4-unreach-net	Rate-Limit	Rate-Limit	Rate-Limit
ICMPv4-unreach-host	Rate-Limit	Rate-Limit	Rate-Limit
ICMPv4-unreach-proto	Rate-Limit	Deny	Rate-Limit
ICMPv4-unreach-port	Rate-Limit	Deny	Rate-Limit
ICMPv4-unreach-frag-needed	Send	Permit	Rate-Limit
ICMPv4-unreach-src-route	Rate-Limit	Deny	Rate-Limit
ICMPv4-unreach-net-unknown *(Depr)*	Deny	Deny	Deny
ICMPv4-unreach-host-unknown	Rate-Limit	Deny	Ignore
ICMPv4-unreach-host-isolated *(Depr)*	Deny	Deny	Deny
ICMPv4-unreach-net-tos	Rate-Limit	Deny	Rate-Limit

Recommendations for ICMPv4

Table Filtering Recommendations

ICMPv4 Message	Sourced from Device	Through Device	Destined to Device
ICMPv4-unreach-host-tos	Rate-Limit	Deny	Rate-Limit
ICMPv4-unreach-admin	Rate-Limit	Rate-Limit	Rate-Limit
ICMPv4-unreach-prec-violation	Rate-Limit	Deny	Rate-Limit
ICMPv4-unreach-prec-cutoff	Rate-Limit	Deny	Rate-Limit
ICMPv4-quench	Deny	Deny	Deny
ICMPv4-redirect-net	Rate-Limit	Deny	Rate-Limit
ICMPv4-redirect-host	Rate-Limit	Deny	Rate-Limit
ICMPv4-redirect-tos-net	Rate-Limit	Deny	Rate-Limit
ICMPv4-redirect-tos-host	Rate-Limit	Permit	Rate-Limit
ICMPv4-timed-ttl	Rate-Limit	Permit	Rate-Limit

Recommendations for ICMPv4

Table Filtering Recommendations

ICMPv4 Message	Sourced from Device	Through Device	Destined to Device
ICMPv4-timed-reass	Rate-Limit	Permit	Rate-Limit
ICMPv4-parameter-pointer	Rate-Limit	Deny	Rate-Limit
ICMPv4-option-missing	Rate-Limit	Deny	Rate-Limit
ICMPv4-req-echo-message	Rate-Limit	Permit	Rate-Limit
ICMPv4-req-echo-reply	Rate-Limit	Permit	Rate-Limit
ICMPv4-req-router-sol	Rate-Limit	Deny	Rate-Limit
ICMPv4-req-router-adv	Rate-Limit	Deny	Rate-Limit
ICMPv4-req-timestamp-message	Rate-Limit	Deny	Rate-Limit
ICMPv4-req-timestamp-reply	Rate-Limit	Deny	Rate-Limit
ICMPv4-info-message *(Depr)*	Deny	Deny	Deny

Recommendations for ICMPv4

Table Filtering Recommendations

ICMPv4 Message	Sourced from Device	Through Device	Destined to Device
ICMPv4-info-reply (Depr)	Deny	Deny	Deny
ICMPv4-mask-request	Rate-Limit	Deny	Rate-Limit
ICMPv4-mask-reply	Rate-Limit	Deny	Rate-Limit

Recommendations for ICMPv4

ICMPv4 Error Messages

- Echo Reply (Type 0, Code 0)
- Destination Unreachable (Type 3)
  - Net Unreachable (Code 0)
  - Host Unreachable (Code 1)
  - Protocol Unreachable (Code 2)
  - Port Unreachable (Code 3)
  - Fragmentation Needed and DF Set (Code 4)
  - Source Route Failed (Code 5)
  - Destination Network Unknown (Code 6) (Deprecated)
  - Destination Host Unknown (Code 7)
  - Source Host Isolated (Code 8) (Deprecated)
  - Communication with Destination Network Administratively Prohibited (Code 9) (Deprecated)

ICMPv4 Error Messages

- Destination Unreachable (Type 3)
  - Communication with Destination Host Administratively Prohibited (Code 10) (Deprecated)
  - Network Unreachable for Type of Service (Code 11)
  - Host Unreachable for Type of Service (Code 12)
  - Communication Administratively Prohibited (Code 13)
  - Host Precedence Violation (Code 14)
  - Precedence Cutoff in Effect (Code 15)

ICMPv4 Error Messages

- Source Quench (Type 4, Code 0)
- Redirect (Type 5)
  - Redirect Datagrams for the Network (Code 0)
  - Redirect Datagrams for the Host (Code 1)
  - Redirect datagrams for the Type of Service and Network (Code 2)
  - Redirect Datagrams for the Type of Service and Host (Code 3)
- Time Exceeded (Type 11)
  - Time to Live Exceeded in Transit (Code 0)
  - Fragment Reassembly Time Exceeded (Code 1)

ICMPv4 Error Messages

- Parameter Problem (Type 12)
  - Pointer Indicates the Error (Code 0)
  - Required Option is Missing (Code 1)

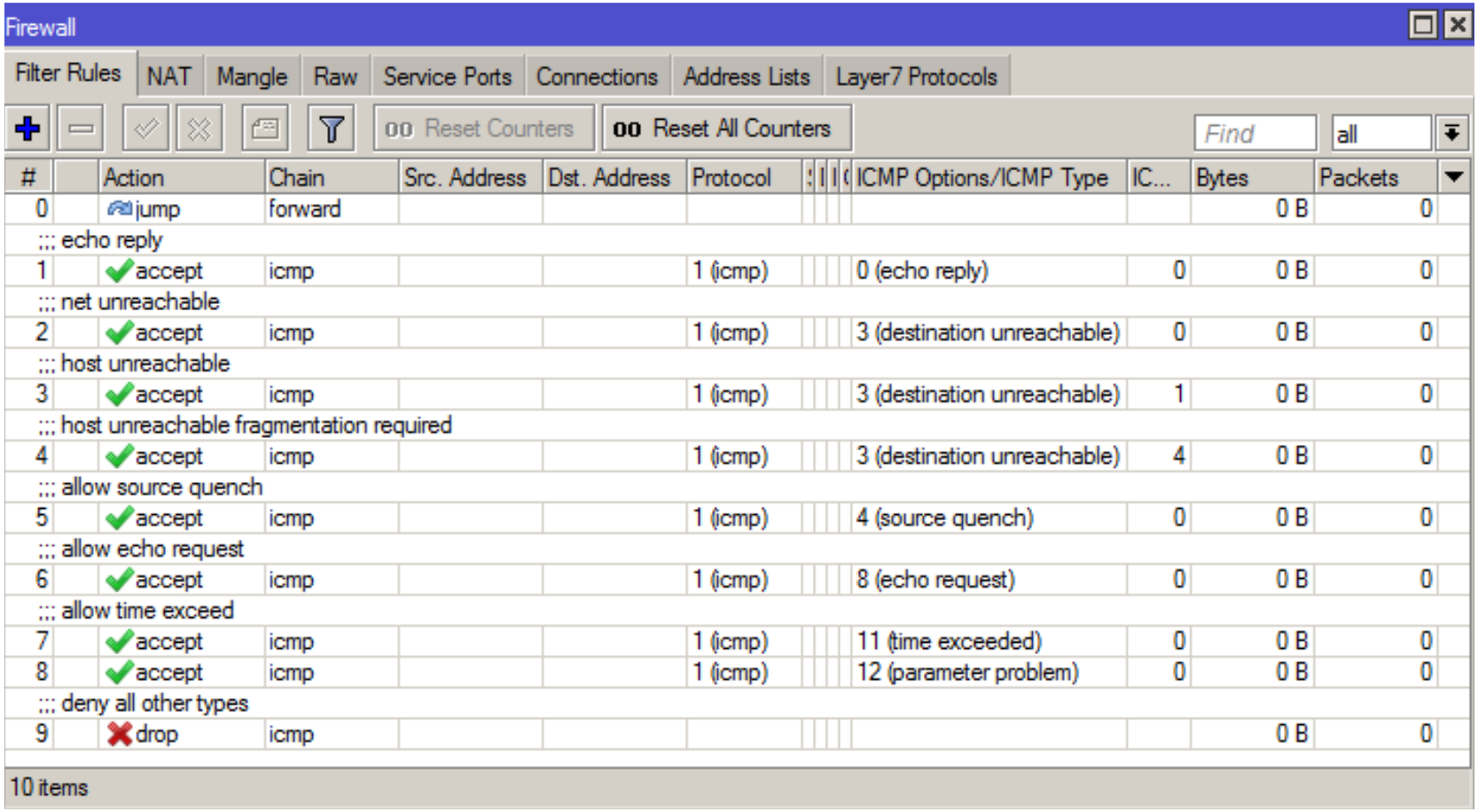
ICMPv4 Informational Messages

- Echo or Echo Reply Message
  - Echo Message (Type 8, Code 0)
  - Echo Reply Message (Type 0, Code 0)
- Router Solicitation or Router Advertisement message
  - Router Solicitation Message (Type 10, Code 0)
  - Router Advertisement Message (Type 9, Code 0)
- Timestamp or Timestamp Reply Message
  - Timestamp Message (Type 13, Code 0)
  - Timestamp Reply Message (Type 14, Code 0)

ICMPv4 Informational Messages

- Information Request or Information Reply Message (Deprecated)
  - Information Request Message (Type 15, Code 0)
  - Information Reply Message (Type 16, Code 0)
- Address Mask Request or Address Mask Reply
  - Address Mask Request (Type 17, Code 0)
  - Address Mask Reply (Type 18, Code 0)

How the ICMP Filtering Works



How the ICMP Filtering Works

```
/ip firewall filter
add action=jump chain=forward jump-target=icmp
add action=accept chain=icmp comment="echo reply" icmp-options=0:0 protocol=icmp
add action=accept chain=icmp comment="net unreachable" icmp-options=3:0 protocol=icmp
add action=accept chain=icmp comment="host unreachable" icmp-options=3:1 protocol=icmp
add action=accept chain=icmp comment="host unreachable fragmentation required" \
    icmp-options=3:4 protocol=icmp
add action=accept chain=icmp comment="allow source quench" icmp-options=4:0 protocol=icmp
add action=accept chain=icmp comment="allow echo request" icmp-options=8:0 protocol=icmp
add action=accept chain=icmp comment="allow time exceed" icmp-options=11:0 protocol=icmp
add action=accept chain=icmp comment="allow parameter bad" icmp-options=12:0 protocol=icmp
add action=drop chain=icmp comment="deny all other types"
```

ENCRYPTED TUNNELS ON ROUTEROS

L2TP/IPsec

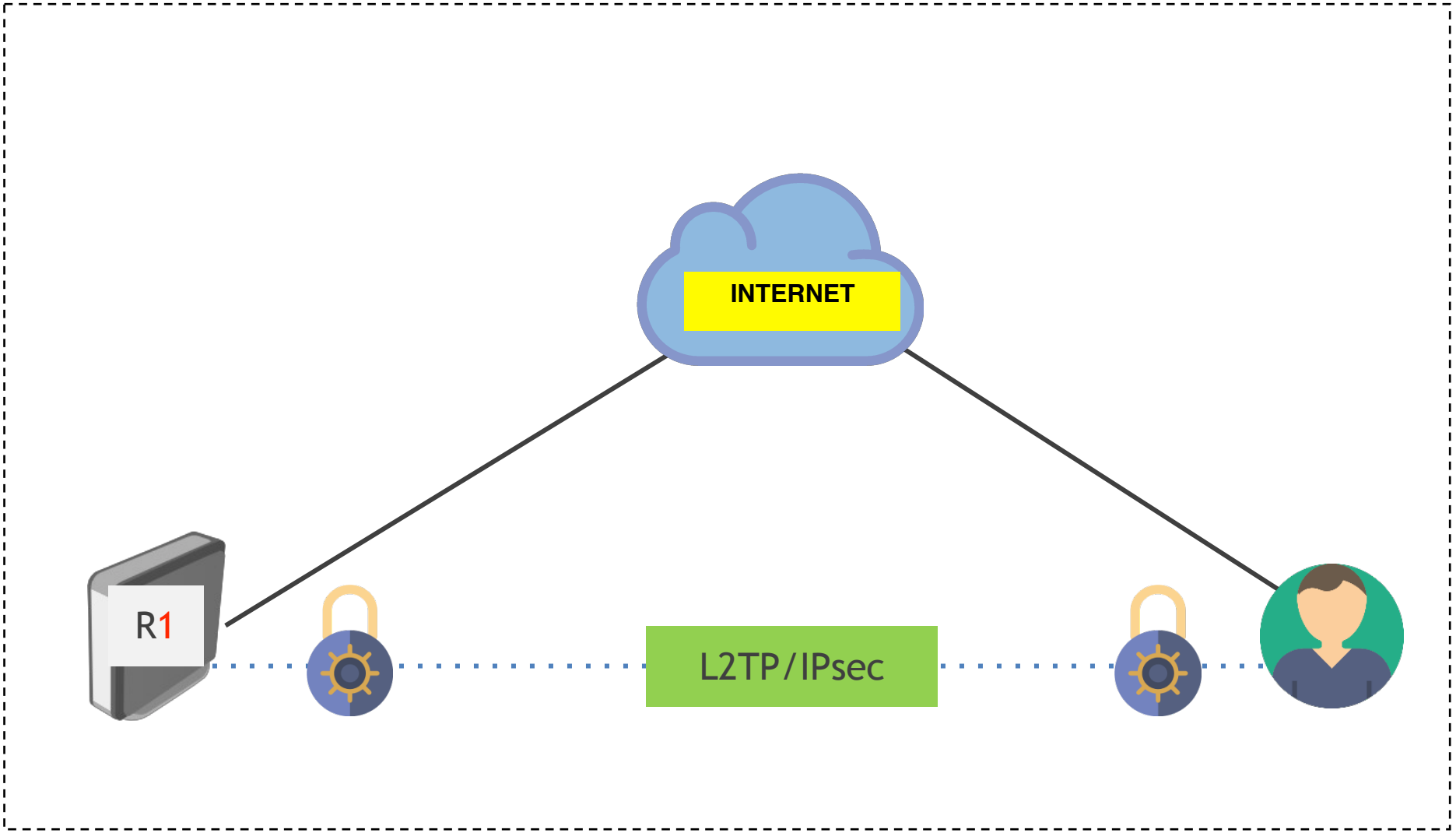
What is L2TP/IPsec

- L2TP stands for Layer 2 Tunnelling Protocol. L2TP was first proposed in 1999 as an upgrade to both L2F (Layer 2 Forwarding Protocol) and PPTP (Point-to-Point Tunnelling Protocol)
- Because L2TP does not provide strong encryption or authentication by itself, another protocol called IPsec is most often used in conjunction with L2TP
- Used together, L2TP and IPsec is much more secure than PPTP (Point-to-Point Tunnelling Protocol), but also slightly slower

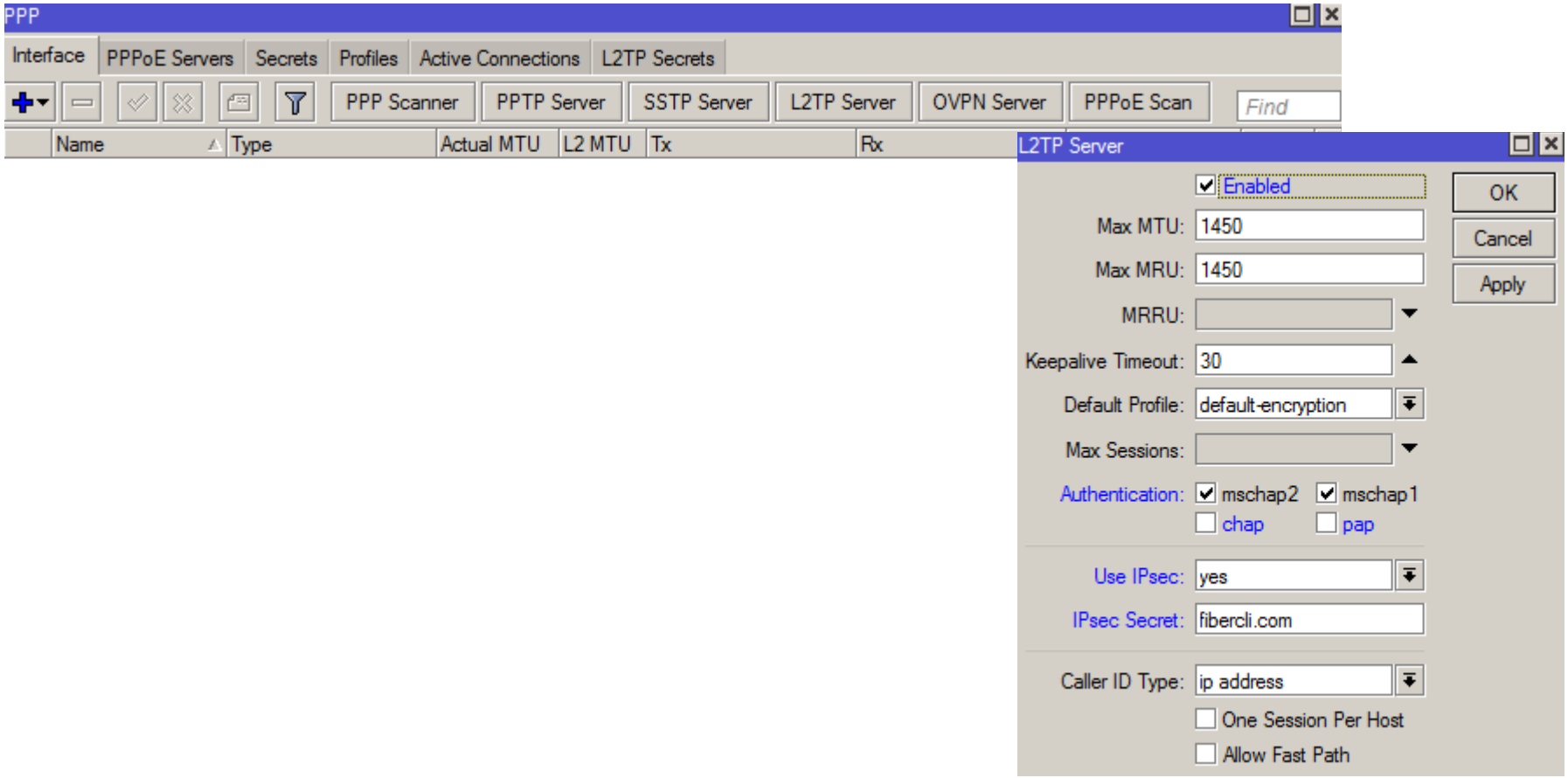
What is L2TP/IPsec

- L2TP/IPSec offers high speeds, and high levels of security for transmitting data
- It generally makes use of AES ciphers for encryption
- L2TP sometimes has problems traversing firewalls due to its use of UDP port 500 which some firewalls have been known to block by default

Lab Setup

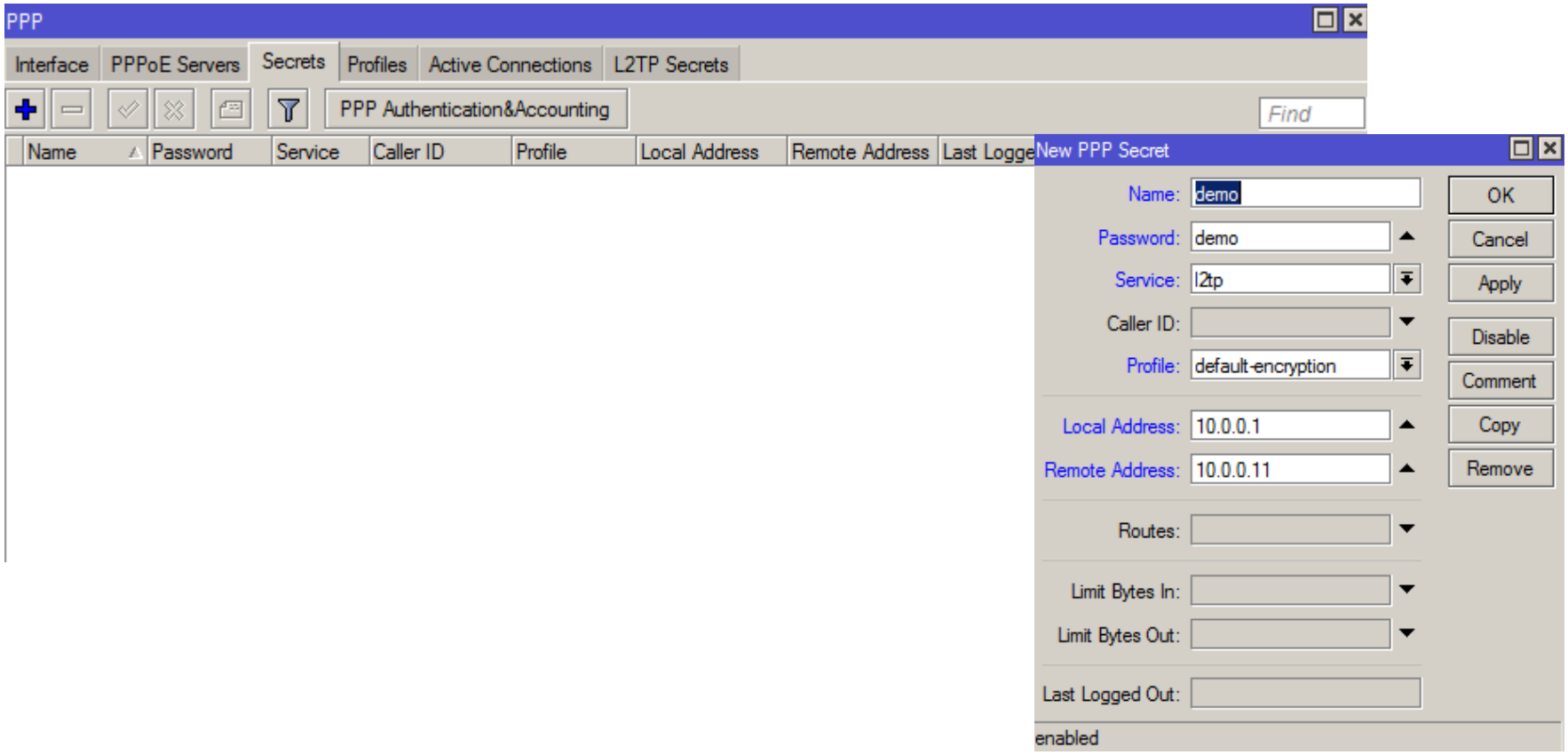


Setup L2TP/IPsec Server



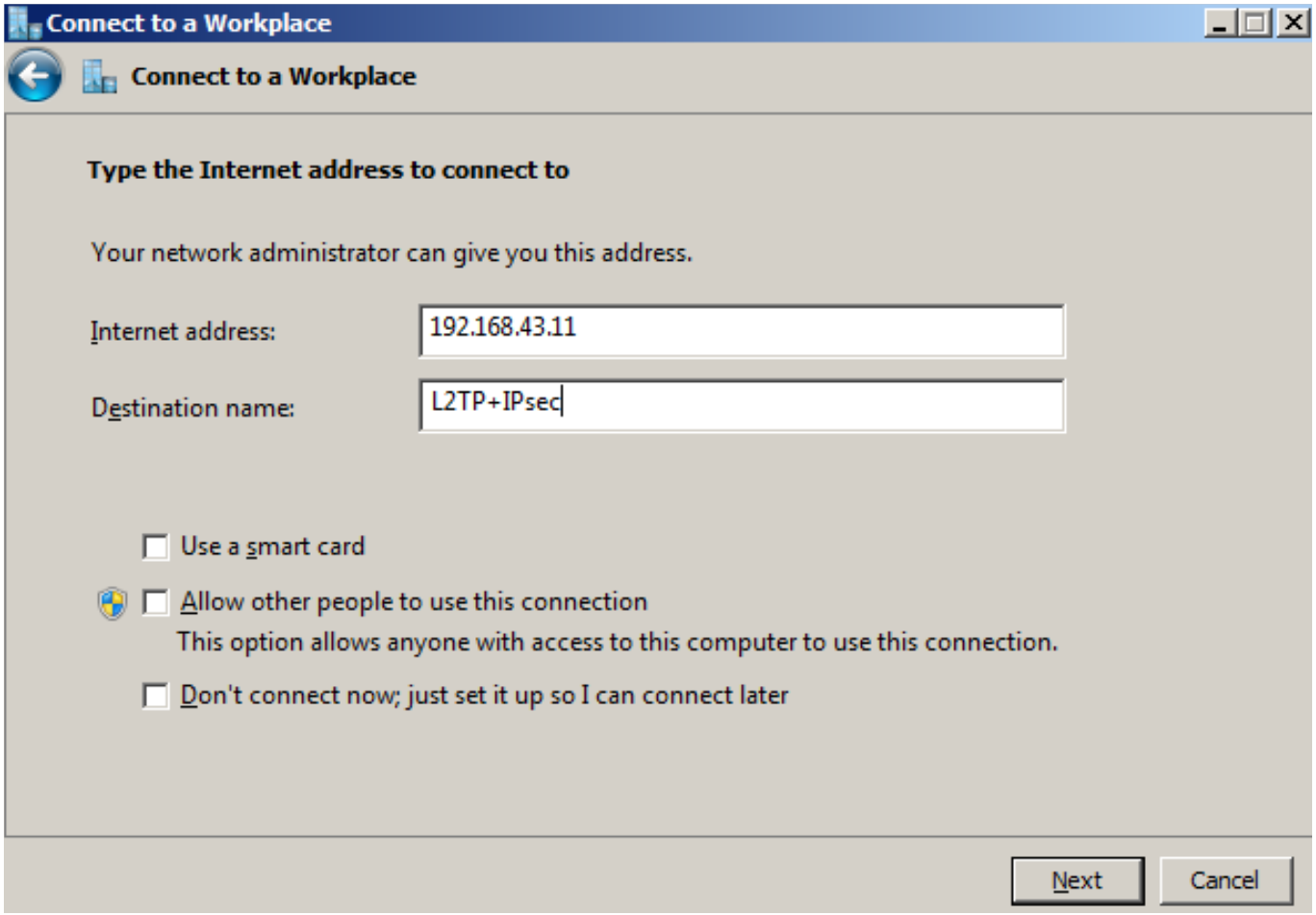

```
/interface l2tp-server server set authentication=mschap1,mschap2 \
enabled=yes ipsec-secret=84GsvZAtUQnE use-ipsec=yes
```

Setup L2TP/IPsec Server

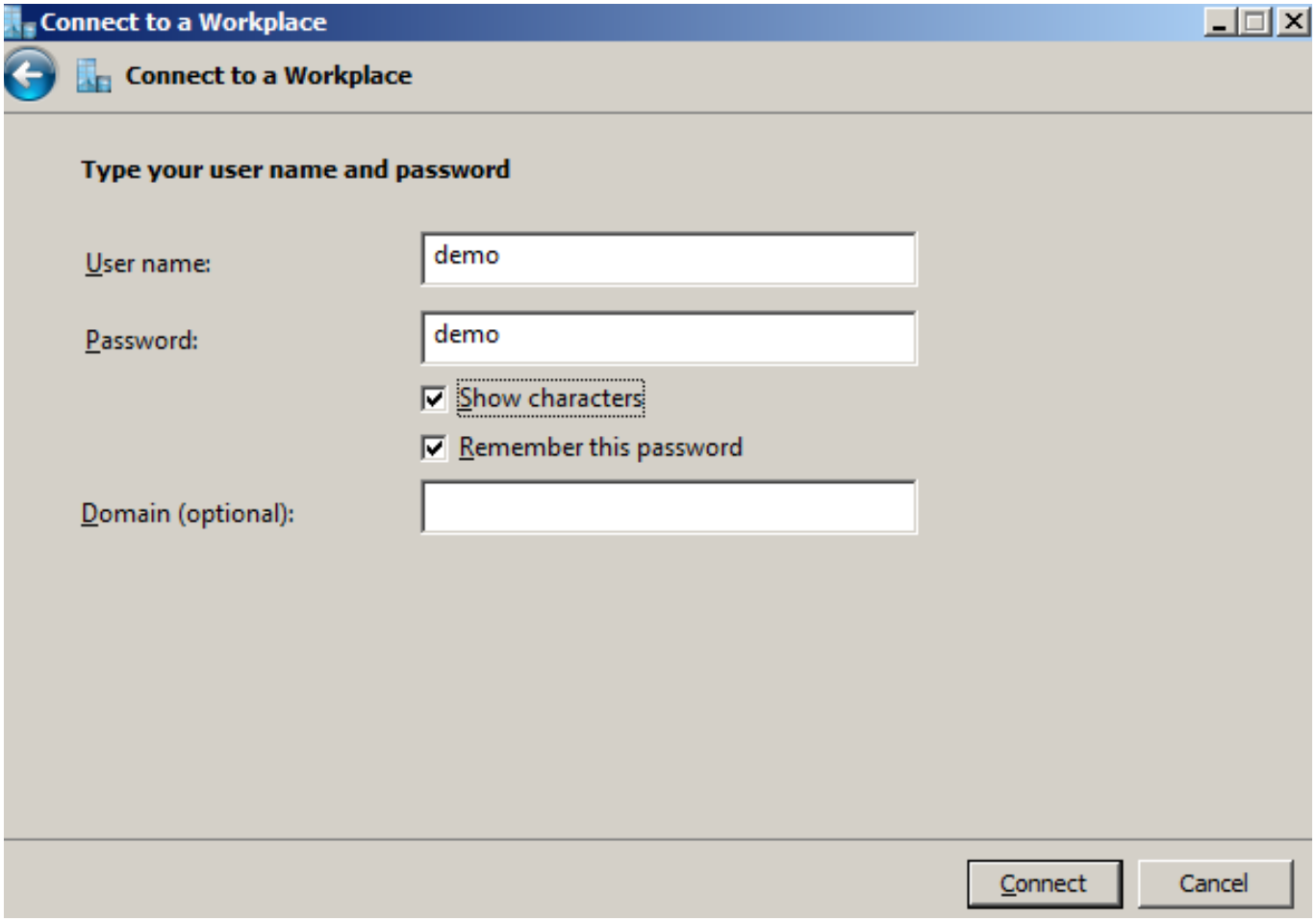



```
/ppp secret add name=demo password=demo local-address=10.0.0.1 \
remote-address=10.0.0.11 profile=default-encryption service=l2tp
```

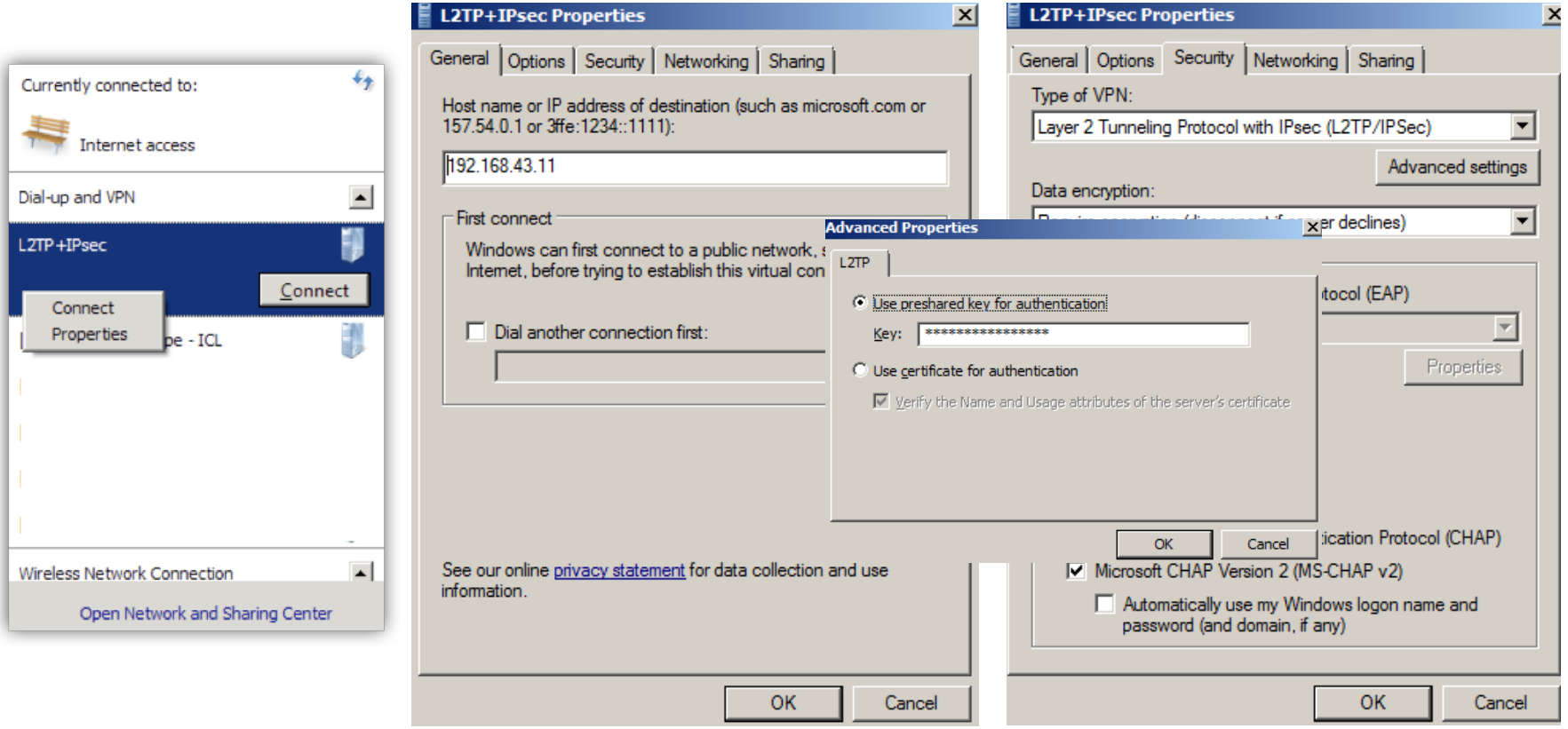
Setup L2TP/IPsec Client



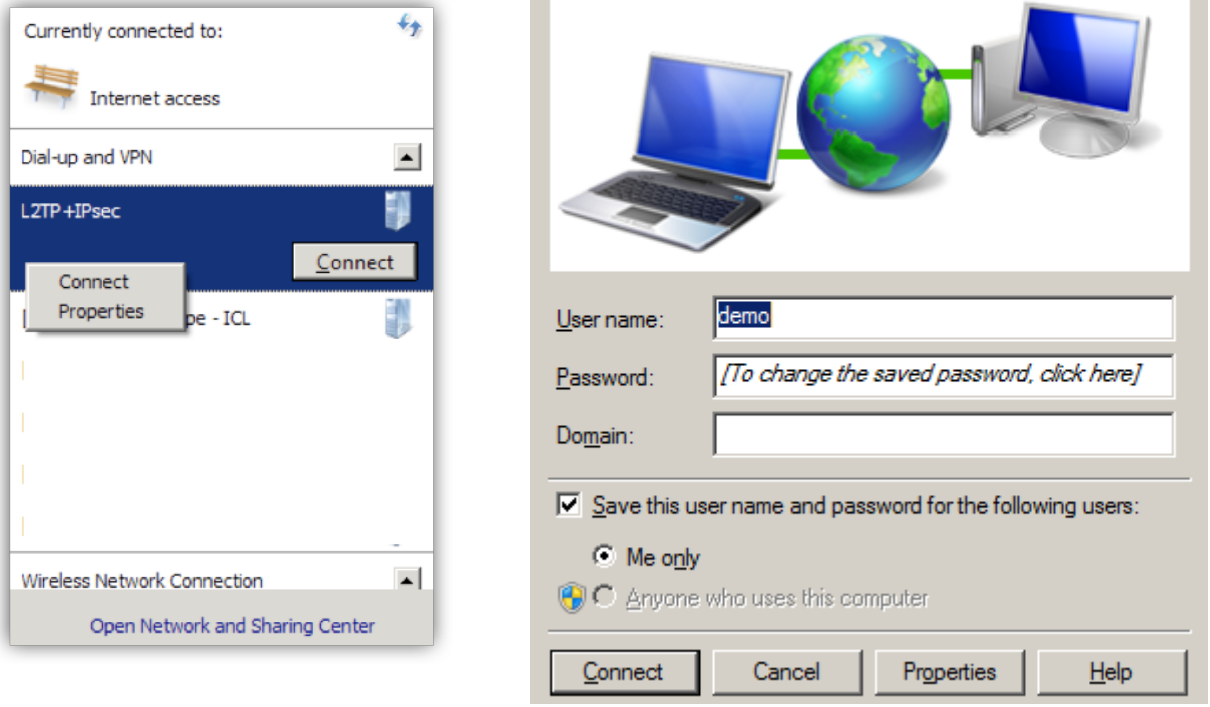
Setup L2TP/IPsec Client



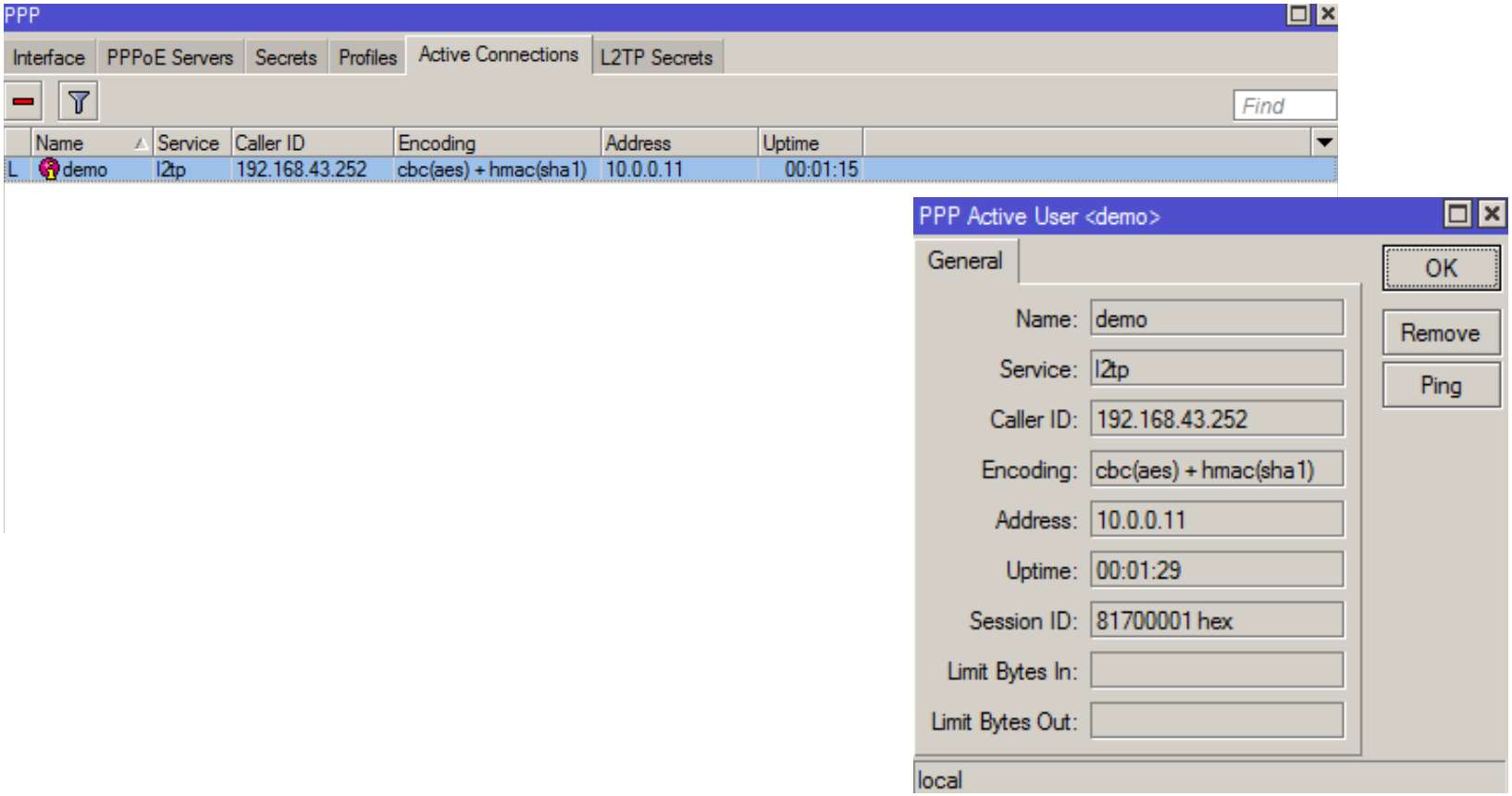
Setup L2TP/IPsec Client



Setup L2TP/IPsec Client



Setup L2TP/IPsec Client

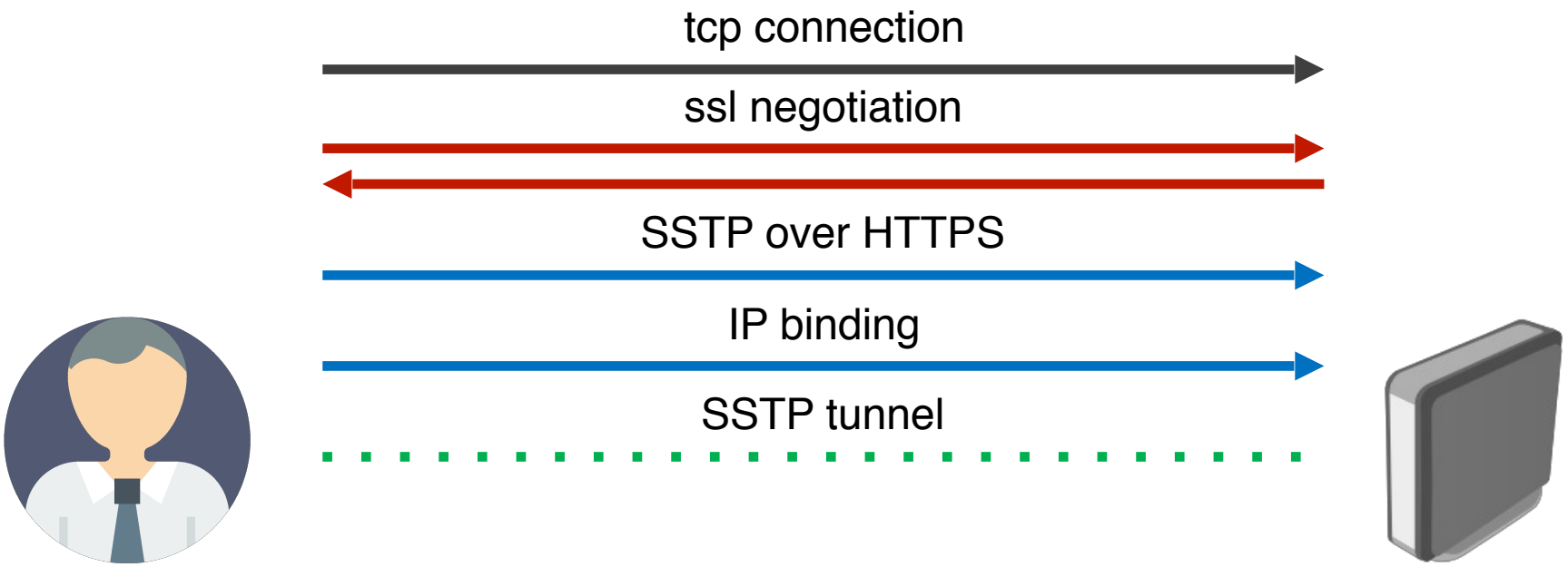


SSTP

What is SSTP

- Microsoft introduced Secure Socket Tunnelling Protocol (SSTP) in Windows Vista and it still considered to be a Windows-only platform even though it is available on a number of other operating systems.
- It has very similar advantages as OpenVPN as SSTP uses SSLv3 and it has greater stability as it is included with Windows which also makes it simpler to use.
- It uses the same port used by SSL connections; port 443.
- It uses 2048 bit encryption and authentication certificates.
- SSTP uses SSL transmissions instead of IPsec because SSL supports roaming instead of just site-to-site transmissions.
- RouterOS has both the SSTP server and client implementation

How the SSTP works



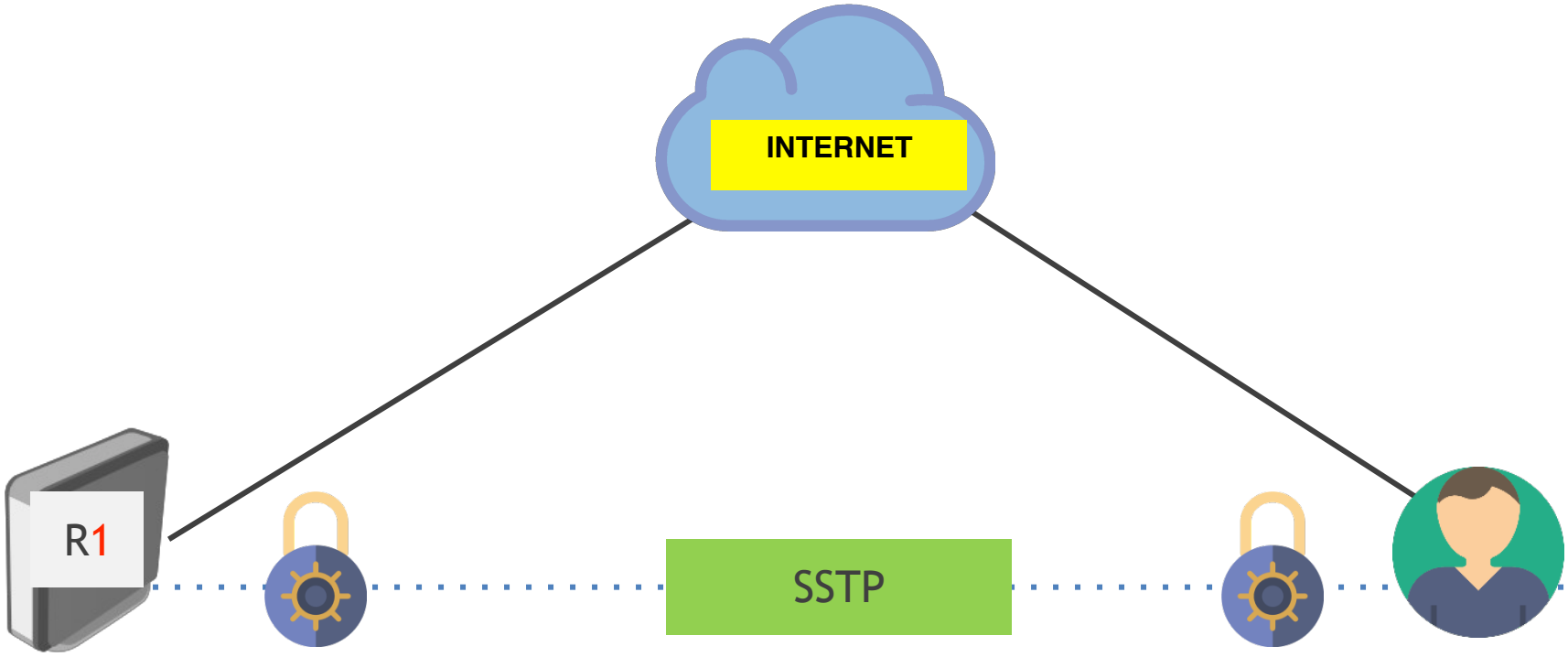
How the SSTP works

- TCP connection is established from client to server (by default on port 443)
- SSL validates server certificate. If certificate is valid connection is established otherwise connection is torn down. (But see note below)
- The client sends SSTP control packets within the HTTPS session which establishes the SSTP state machine on both sides

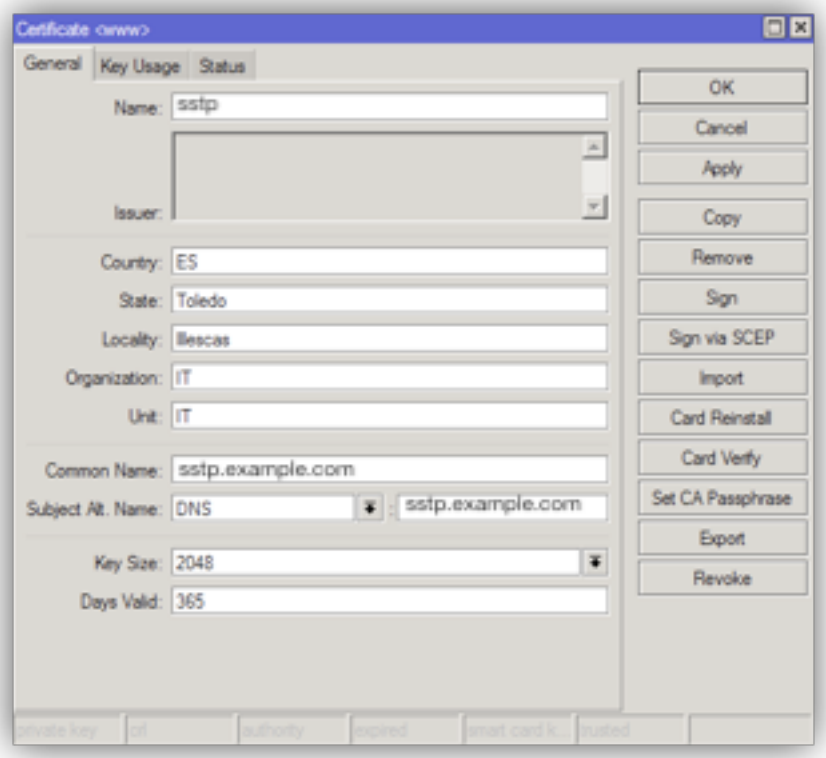
How the SSTP works

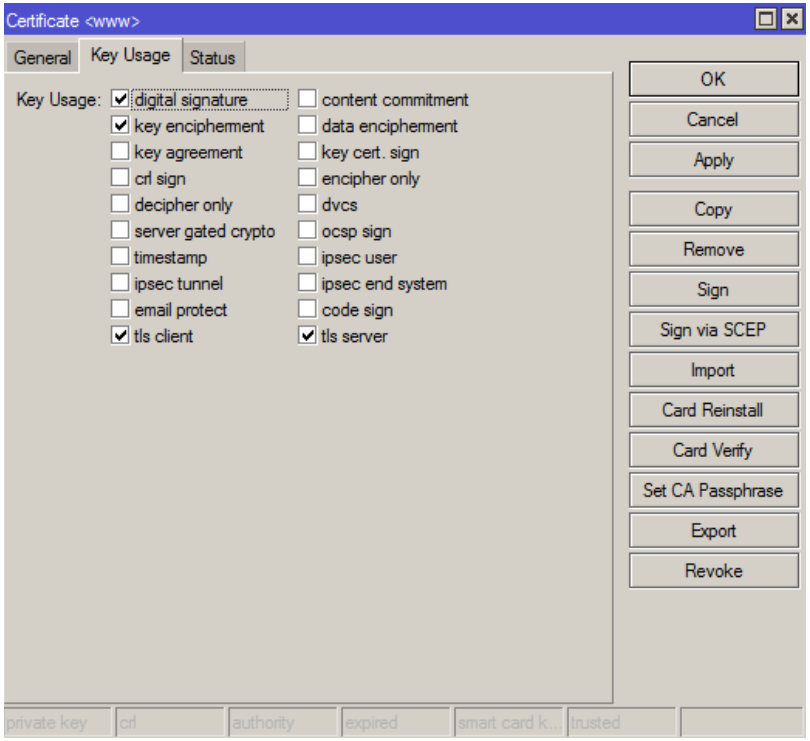
- PPP negotiation over SSTP. Client authenticates to the server and binds IP addresses to SSTP interface
- SSTP tunnel is now established and packet encapsulation can begin.
- Note: Two RouterOS devices can establish an SSTP tunnel even without the use of certificates (not in accordance with Microsoft standard)
- It is recommended to use the certificates at all times!

Lab Setup



Self-signed Certificate

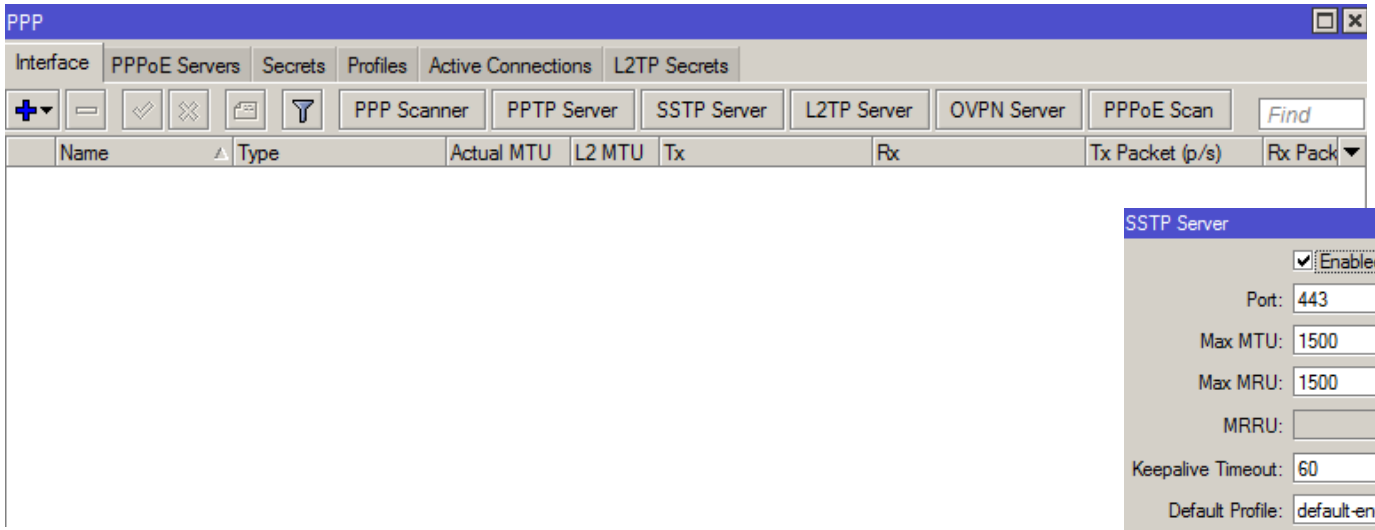


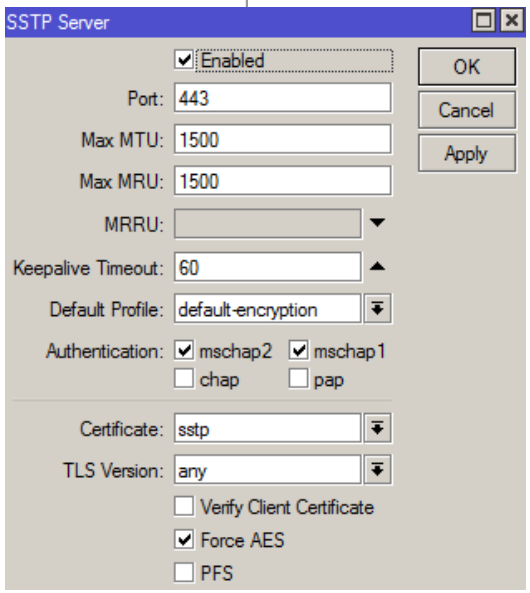



```
certificate add name=sstp country=ES state=Toledo locality=Illescas organization=IT unit=IT \
    common-name=sstp.example.com subject-alt-name=DNS:sstp.example.com key-size=2048 days-valid=365 \
    key-usage=digital-signature,key-encipherment,tls-client,tls-server
```

```
/ certificate sign sstp name=sstp ca=CA
/ certificate set sstp trusted=yes
```

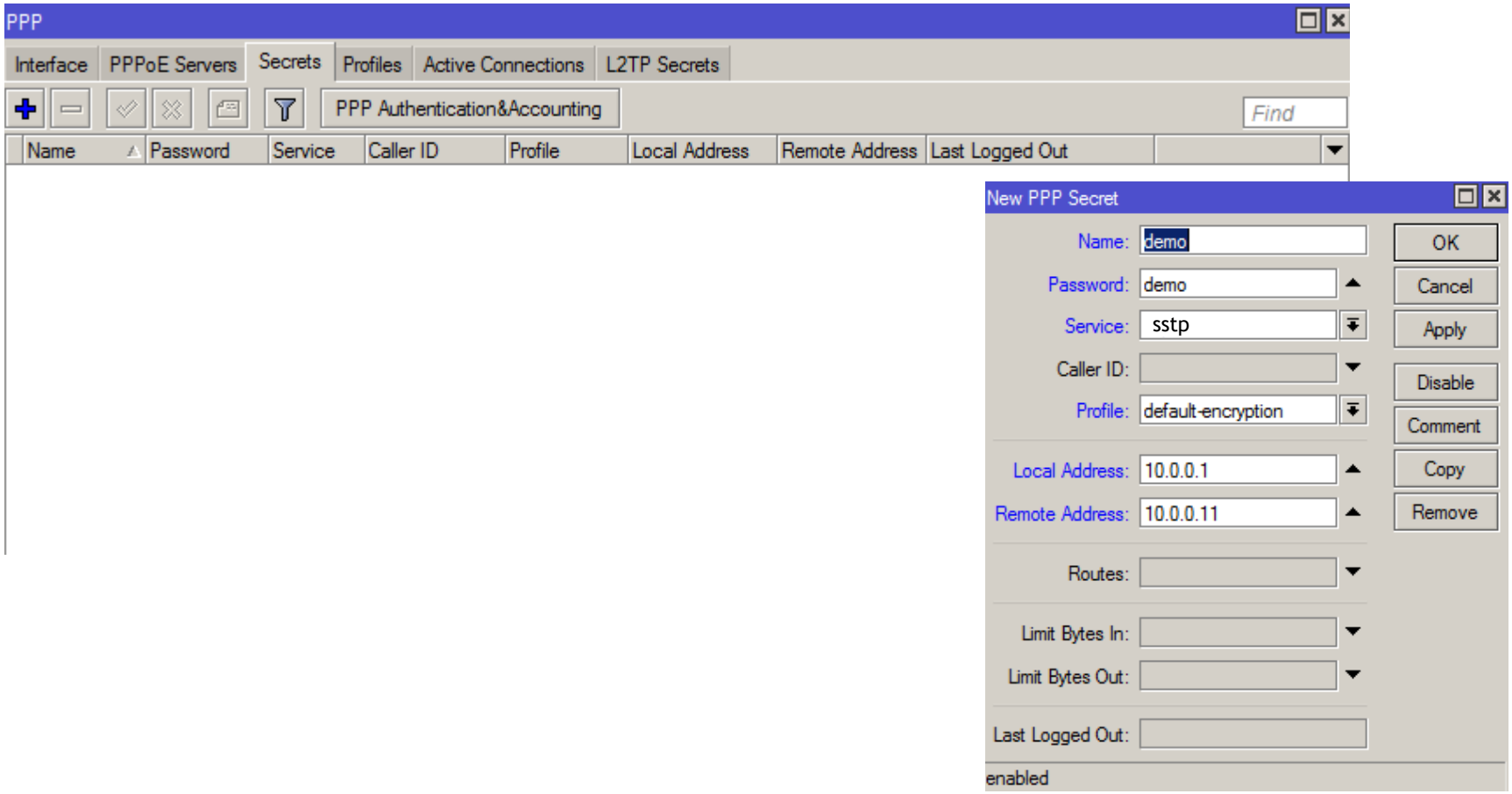
Lab Setup





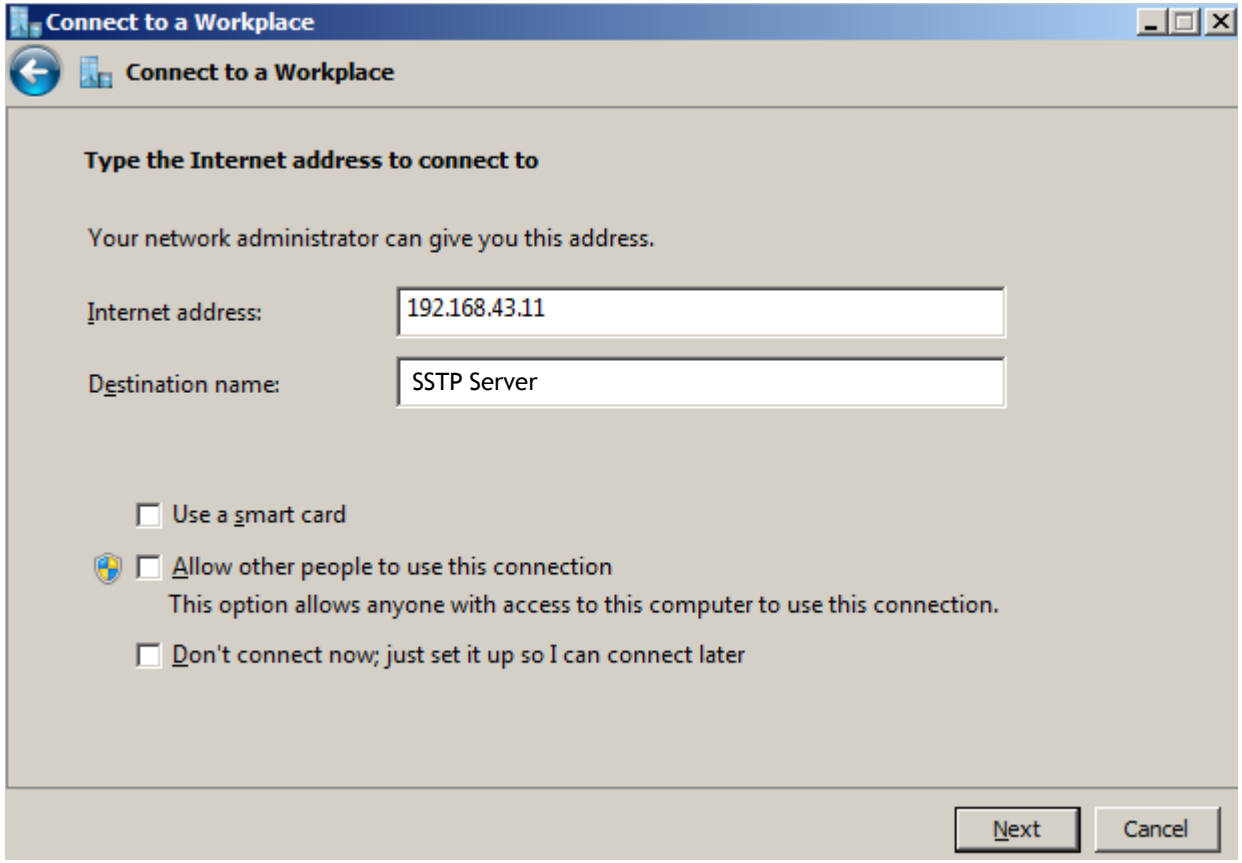

```
/interface sstp-server server set authentication=mschap1,mschap2 certificate=sstp default-profile=default-encryption \
    enabled=yes force-aes=yes
```

Setup SSTP Server

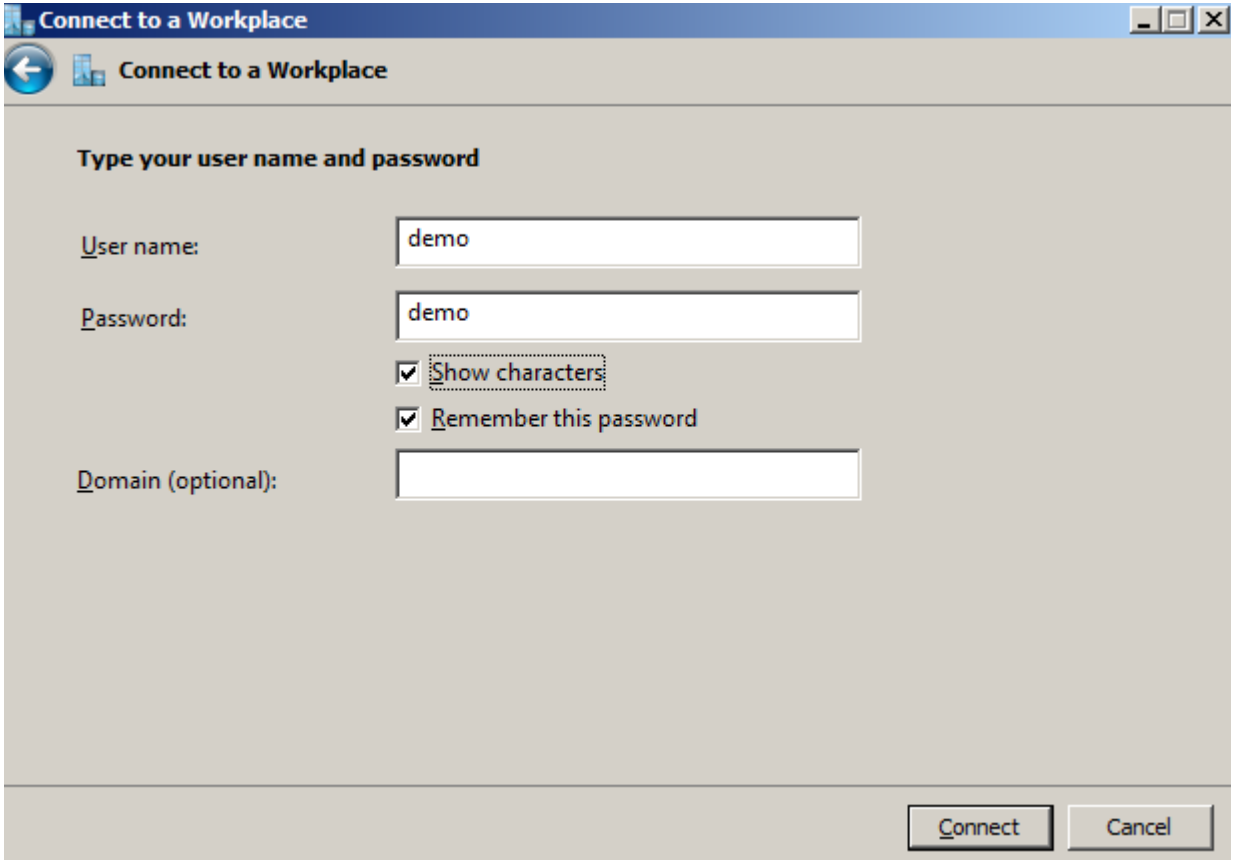



```
/ppp secret add name=demo password=demo local-address=10.0.0.1 remote-address=10.0.0.11 \
    profile=default-encryption service=sstp
```

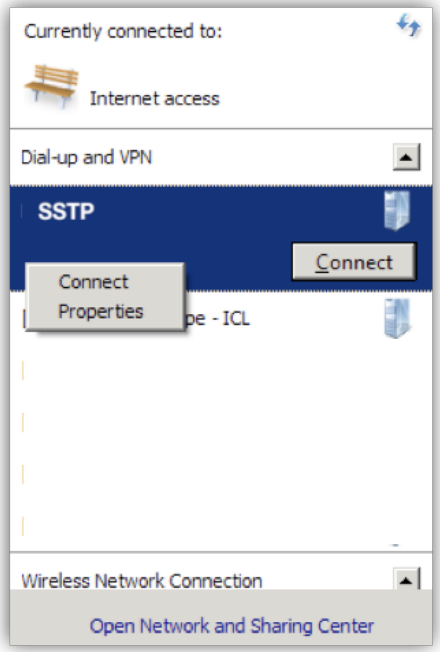
Setup SSTP Server

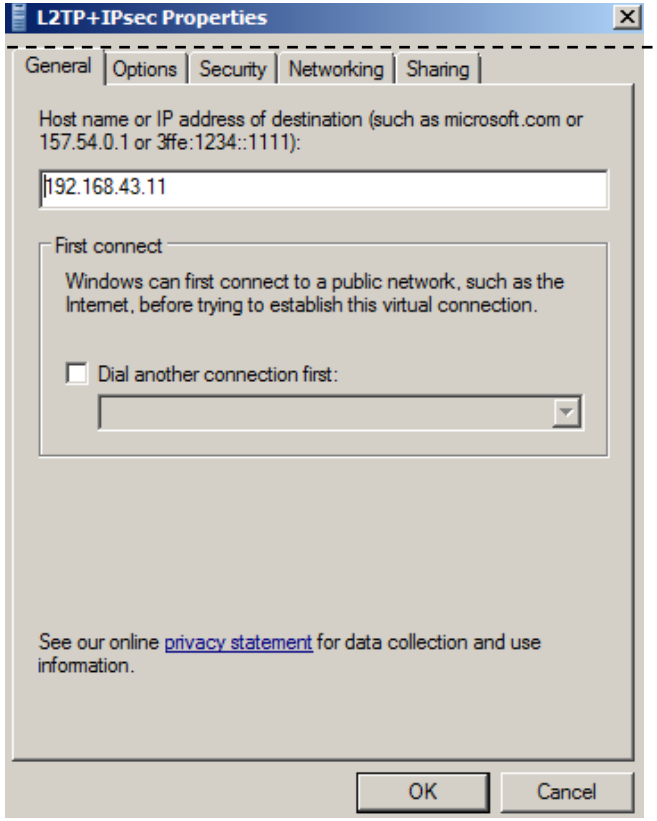


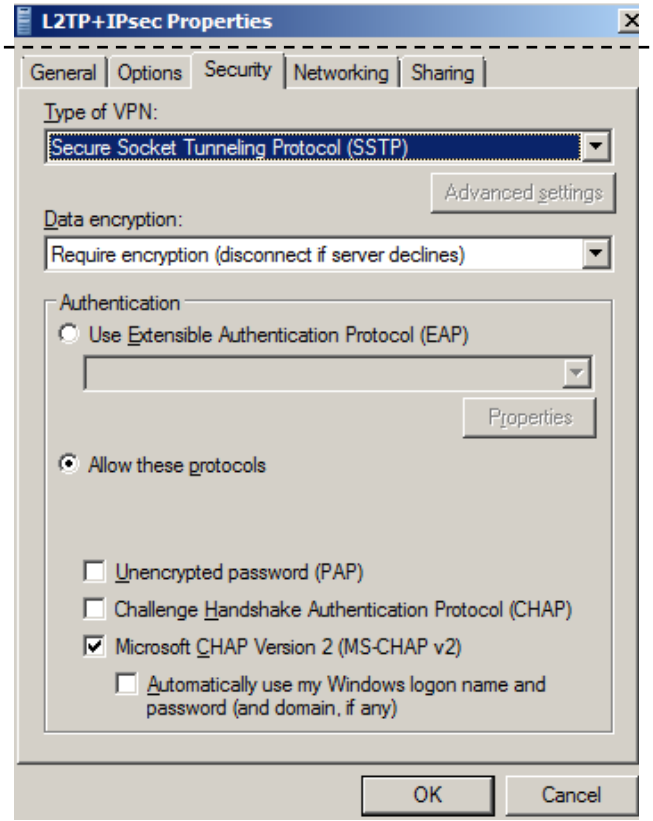
Setup SSTP Client



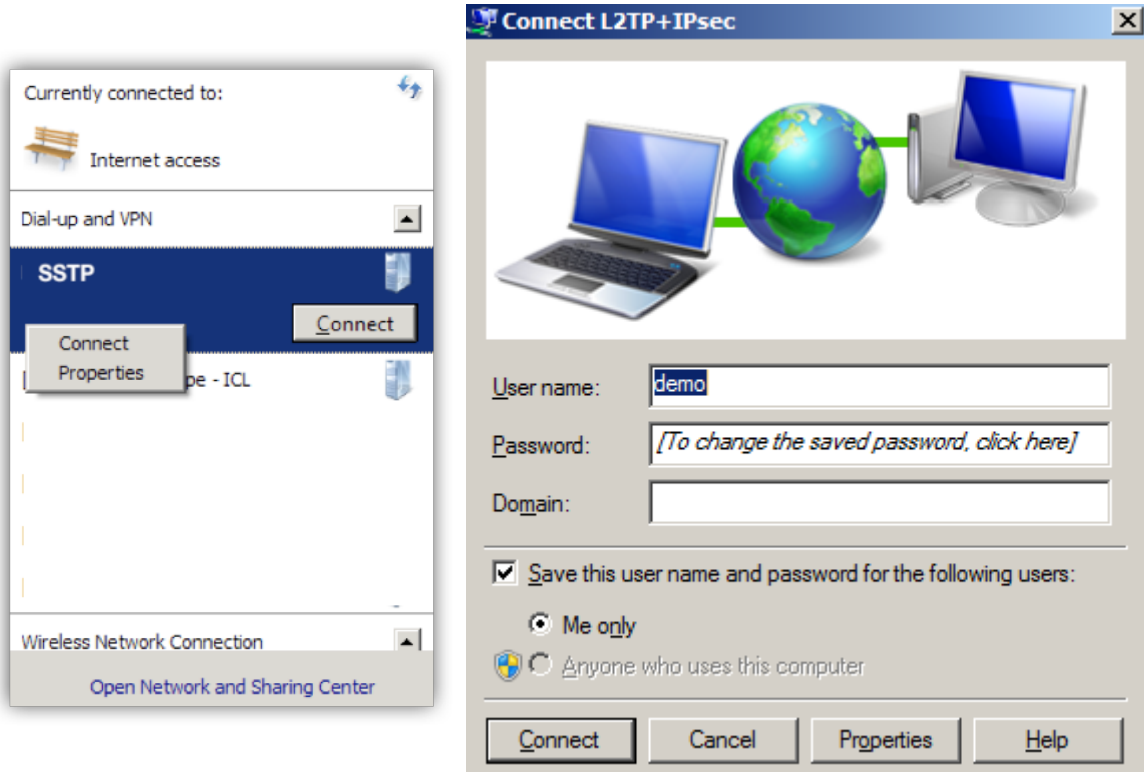
Setup SSTP Client







Setup SSTP Client



IPsec

What is IPsec

Internet Protocol Security (IPsec) is a set of protocols defined by the Internet Engineering Task Force (IETF) to secure packet exchange over unprotected IPv4 or IPv6 networks such as Internet. Provides Layer 3 security (RFC 2401)

IPsec Combines different components :

- Security associations (SA)
- Authentication headers (AH)
- Encapsulating security payload (ESP)
- Internet Key Exchange (IKE)

What is IPsec

IPsec standardisation defined in :

- RFC 4301 Defines the original IPsec architecture and elements common to both AH and ESP
- RFC 4302 Defines authentication headers (AH)
- RFC 4303 Defines the Encapsulating Security Payload (ESP)
- RFC 2408 ISAKMP
- RFC 5996 IKE v2 (Sept 2010)
- RFC 4835 Cryptographic algorithm implementation for ESP and AH

The Benefits of IPsec

- Confidentiality
  - By encrypting data
- Integrity
  - Routers at each end of a tunnel calculate the checksum or hash value of the data
- Authentication
  - Signatures and certificates
  - All these while still maintaining the ability to route through existing IP Networks

The Benefits of IPsec

Data integrity and source authentication

- Data “signed” by sender and “signature” is verified by the recipient
- Modification of data can be detected by signature “verification”
- Because “signature” is based on a shared secret, it gives source authentication

Anti-replay protection

- Optional; the sender must provide it but the recipient may ignore

The Benefits of IPsec

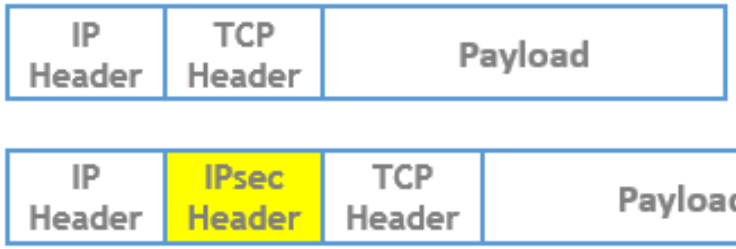
Key management

- IKE – session negotiation and establishment
- Sessions are rekeyed or deleted automatically
- Secret keys are securely established and authenticated
- Remote peer is authenticated through varying options

IPsec Modes

Transport Mode

- IPsec header is inserted into the IP packet
- No new packet is created
- Works well in networks where increasing a packet's size could cause an issue
- Frequently used for remote-access VPNs



normal traffic without IPsec

traffic with transport mode IPsec

IPsec Modes

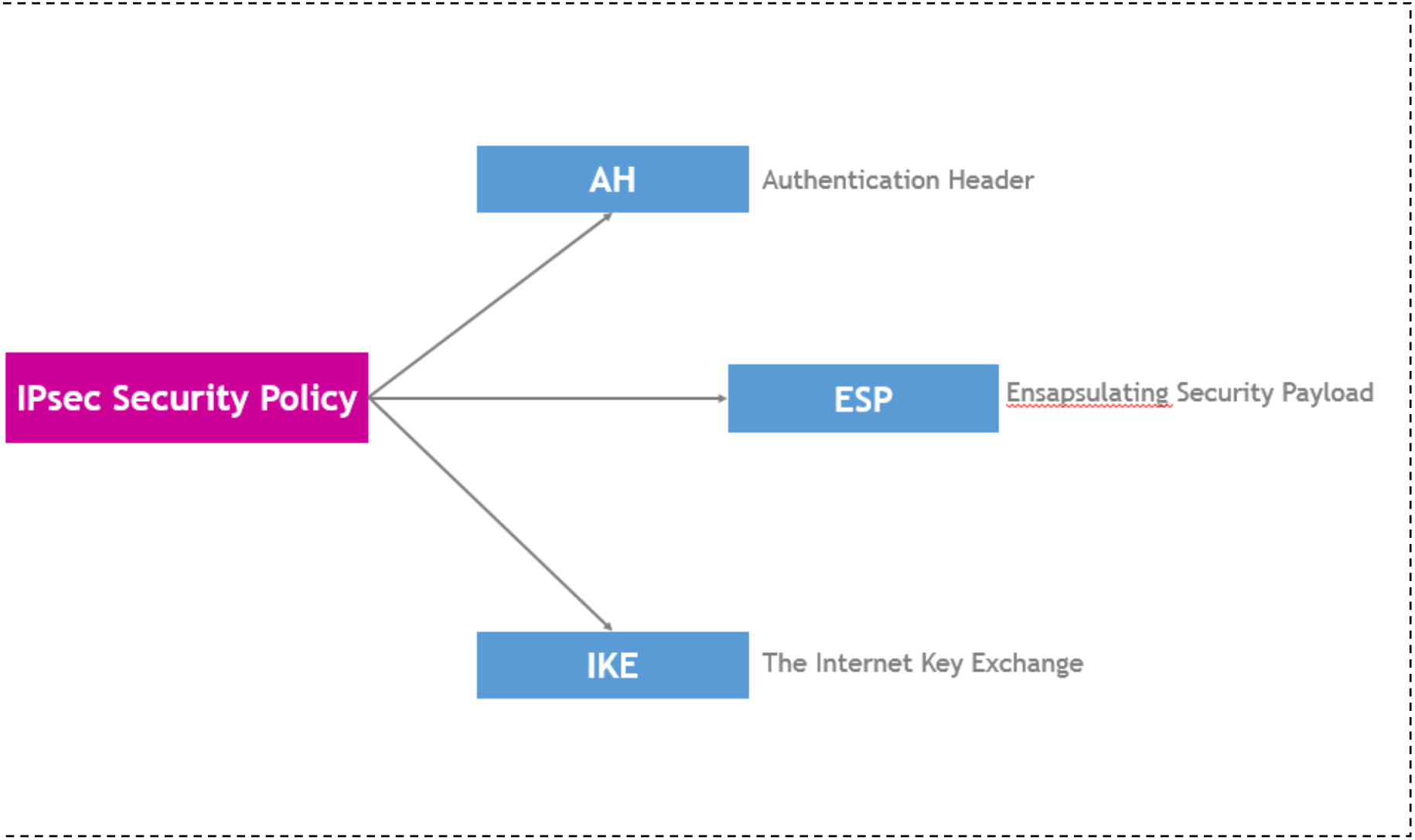
Tunnel Mode

- Entire IP packet is encrypted and becomes the data component of a new (and larger) IP packet.
- Frequently used in an IPsec site-to-site VPN

normal traffic without IPsec

traffic with tunnel mode IPsec

IPsec Architecture



Authentication Header (AH)

AH is a protocol that provides authentication of either all or part of the contents of a datagram through the addition of a header that is calculated based on the values in the datagram.

What parts of the datagram are used for the calculation, and the placement of the header, depends whether tunnel or transport mode is used.

- Provides source authentication and data integrity
  - Protection against source spoofing and replay attacks
- Authentication is applied to the entire packet, with the mutable fields in the IP header zeroed out

Authentication Header (AH)

- Operates on top of IP using protocol 51
- In IPv4, AH protects the payload and all header fields except mutable fields and IP options (such as IPsec option)

MikroTik RouterOS supports the following authentication algorithms for AH:

- SHA1
- MD5

Encapsulating Security Payload (ESP)

Encapsulating Security Payload (ESP) uses shared key encryption to provide data privacy. ESP also supports its own authentication scheme like that used in AH, or can be used in conjunction with AH.

ESP packages its fields in a very different way than AH. Instead of having just a header, it divides its fields into three components:

ESP Header : Comes before the encrypted data and its placement depends on : whether ESP is used in transport mode or tunnel mode.

ESP Trailer : This section is placed after the encrypted data. It : contains padding that is used to align the encrypted data.

ESP Auth Data : This field contains an Integrity Check Value (ICV), computed : in a manner similar to how the AH protocol works, for : when ESP's optional authentication feature is used.

Encapsulating Security Payload (ESP)

- Uses IP protocol 50
- Provides all that is offered by AH, plus data confidentiality
  - It uses symmetric key encryption
- Must encrypt and/or authenticate in each packet
  - Encryption occurs before authentication
- Authentication is applied to data in the IPsec header as well as the data contained as payload

Encapsulating Security Payload (ESP)

RouterOS ESP supports various encryption and authentication algorithms.

Authentication : SHA1, MD5

Encryption :

DES : 56-bit DES-CBC encryption algorithm;
3DES : 168-bit DES encryption algorithm;
AES : 128, 192 and 256-bit key AES-CBC encryption algorithm;
Blowfish : added since v4.5
Twofish : added since v4.5
Camellia : 128, 192 and 256-bit key Camellia encryption algorithm
: added since v4.5

Internet Key Exchanger (IKE)

The Internet Key Exchange (IKE) is a protocol that provides authenticated keying material for Internet Security Association and Key Management Protocol (ISAKMP) framework. There are other key exchange schemes that work with ISAKMP, but IKE is the most widely used one. Together they provide means for authentication of hosts and automatic management of security associations (SA).

- "An IPsec component used for performing mutual authentication and establishing and maintaining Security Associations." (RFC 5996)
- Typically used for establishing IPSec sessions
- A key exchange mechanism
- Five variations of an IKE negotiation:
  - Two modes (aggressive and main modes)
  - Three authentication methods (pre-shared, public key encryption, and public key signature)
- Uses UDP port 500

IKE Mode

Mode	Descriptions
Main Mode	• Three exchanges of information between IPsec peers. Initiator sends one or more proposals to the other peer (responder) • Responder selects a proposal
Aggressive Mode	• Achieves same result as main mode using only 3 packets First packet sent by initiator containing all info to establish SA • Second packet by responder with all security parameters selected Third packet finalizes authentication of the ISAKMP session
Quick Mode	• Negotiates the parameters for the IPsec session. • Entire negotiation occurs within the protection of ISAKMP session

Internet Key Exchanger (IKE)

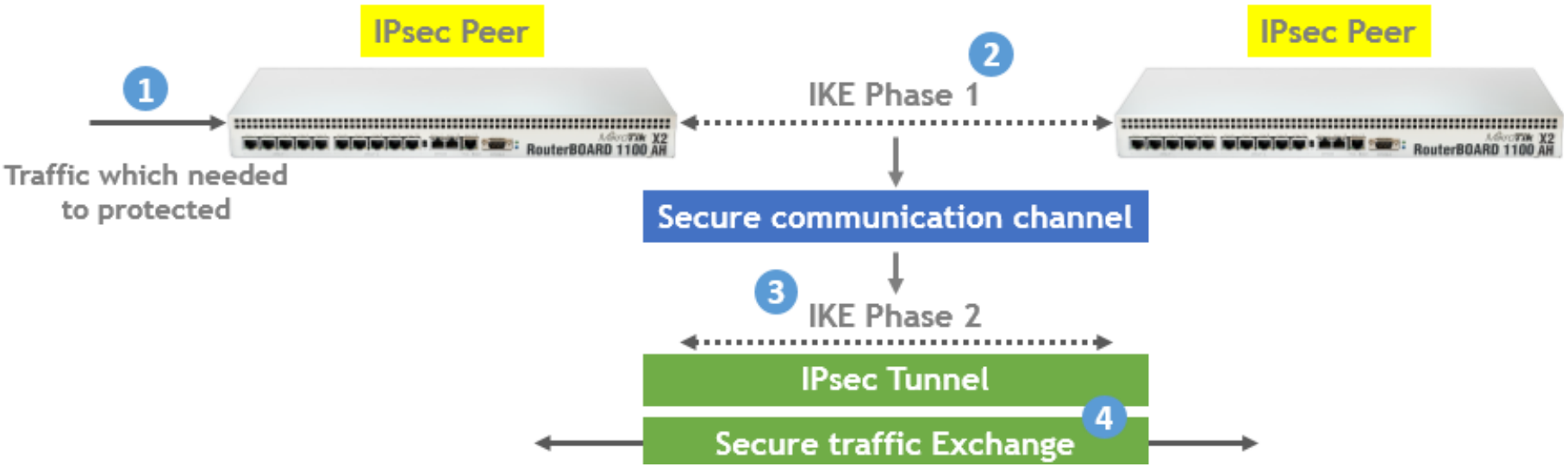
Phase I

- Establish a secure channel (ISAKMP SA)
- Using either main mode or aggressive mode
- Authenticate computer identity using certificates or pre-shared secret

Phase II

- Establishes a secure channel between computers intended for the transmission of data (IPsec SA)
- Using quick mode

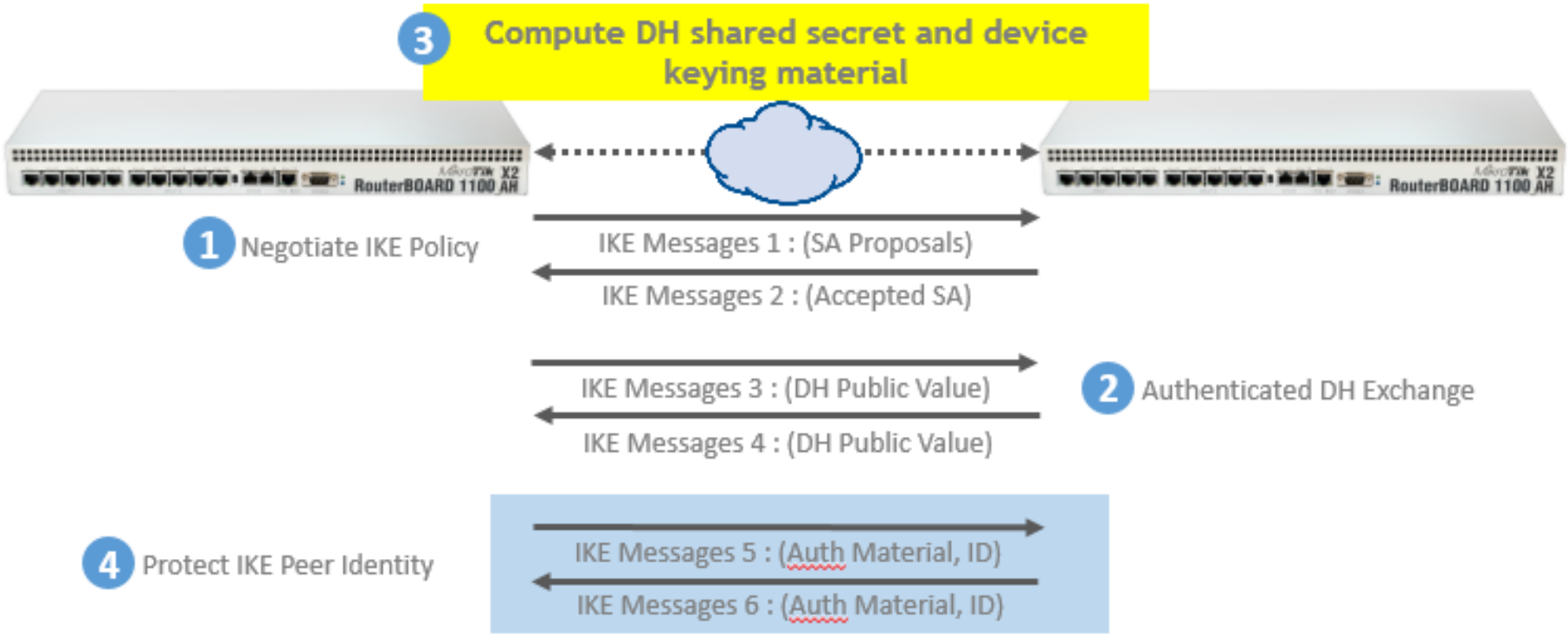
Internet Key Exchanger (IKE)



IKE Phase 1 (Main Mode)

- Main mode negotiates an ISAKMP SA which will be used to create IPsec SAs.
- Three steps
  - SA negotiation (encryption algorithm, hash algorithm, authentication method, which DF group to use)
  - Do a Diffie-Hellman exchange
  - Provide authentication information
  - Authenticate the peer

IKE Phase 1 (Main Mode)



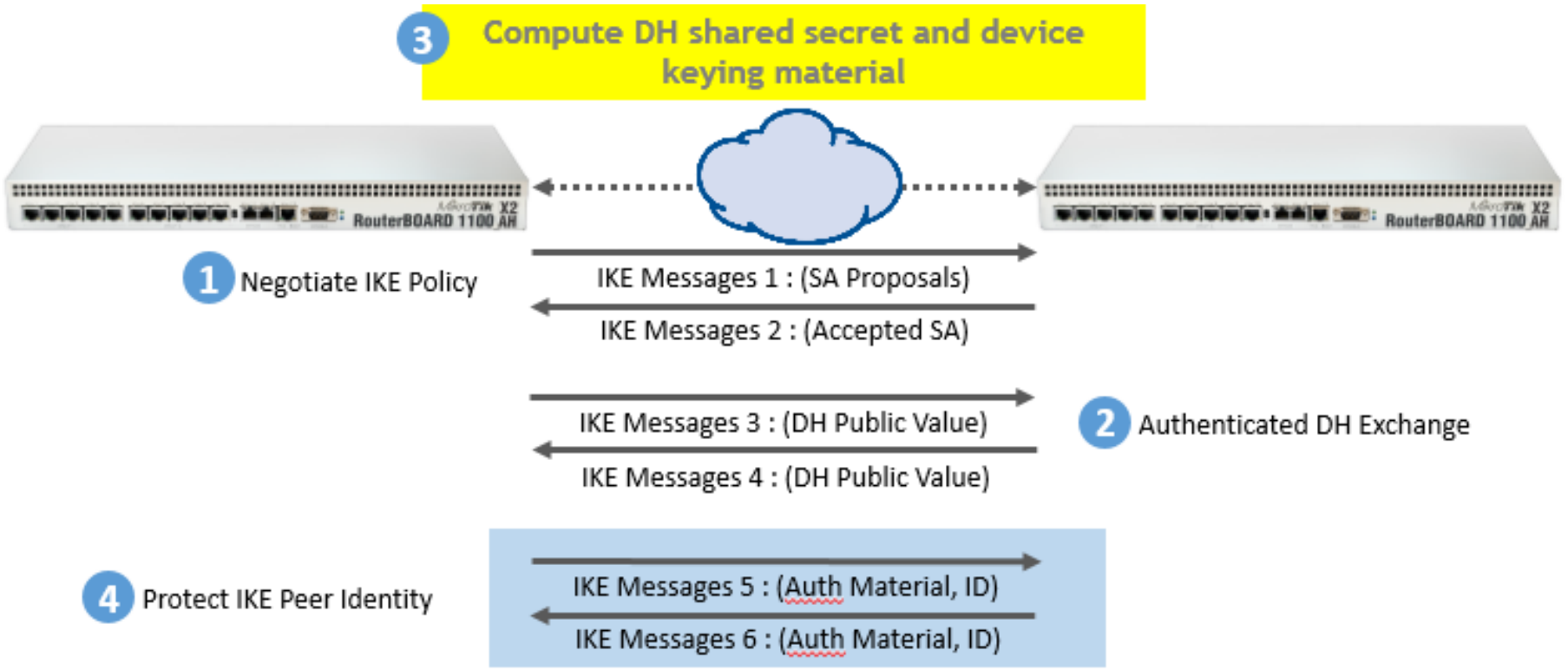
IKE Phase 1 (Aggressive Mode)

- Uses 3 (vs 6) messages to establish IKE SA
- No denial of service protection
- Does not have identity protection
- Optional exchange and not widely implemented

IKE Phase 2 (Quick Mode)

- All traffic is encrypted using the ISAKMP Security Association
- Creates/refreshes keys
- Each quick mode negotiation results in two IPsec Security Associations (one inbound, one outbound)

IKE Phase 2 (Quick Mode)



IKEv2

- Internet Key Exchange Version 2 (IKEv2) is the second-generation standard for a secure key exchange between connected devices.
- IKEv2 works by using an IPsec-based tunnelling protocol to establish a secure connection.
- One of the single most important benefits of IKEv2 is its ability to reconnect very quickly in the event that your VPN connection gets disrupted.
- Quick reconnections and strong encryption IKEv2 makes an excellent candidate to use

Lab Setup

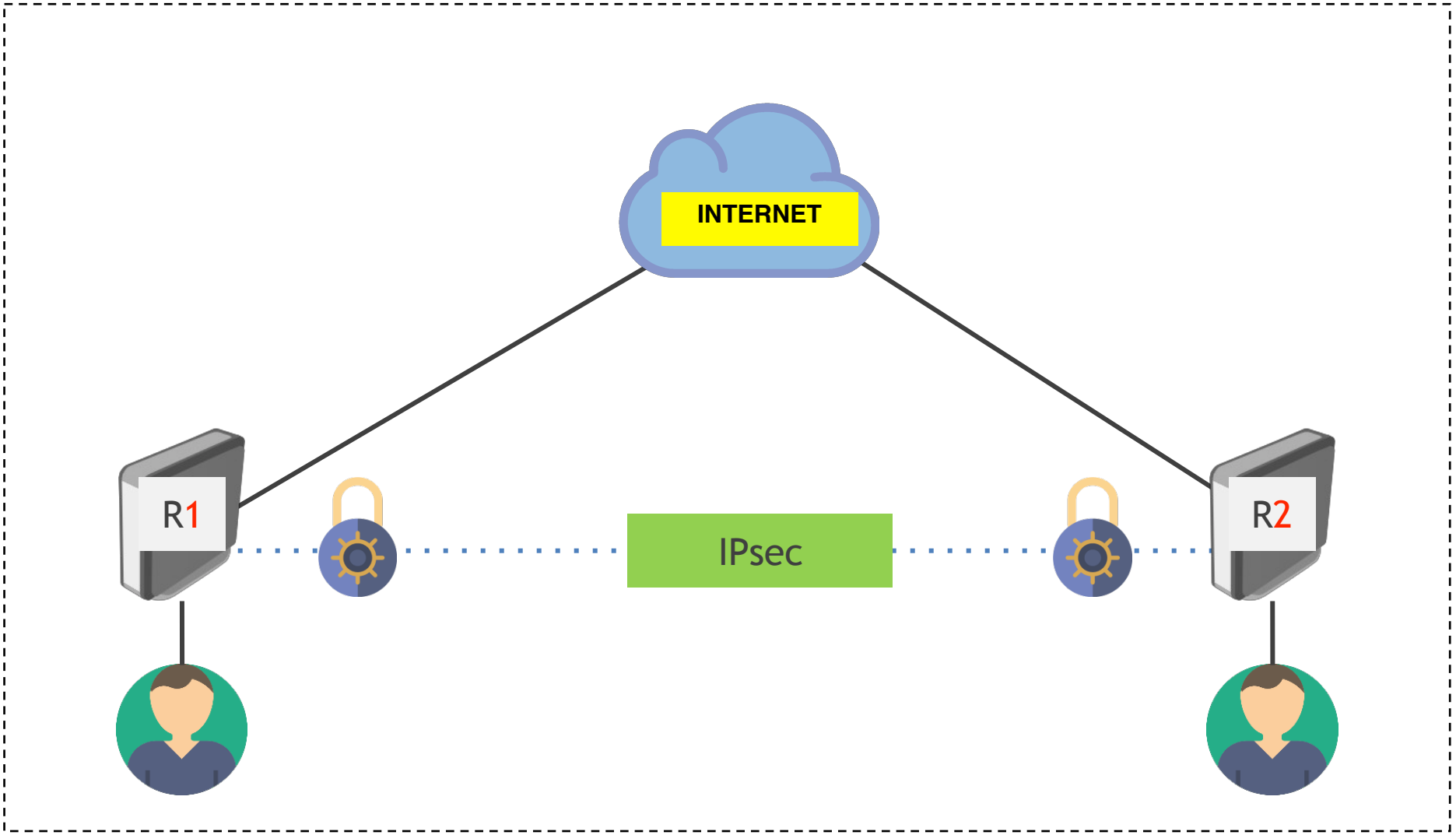
R1

- Public Address : 11.11.11.2/24
- Local Address: 192.168.1.0/24

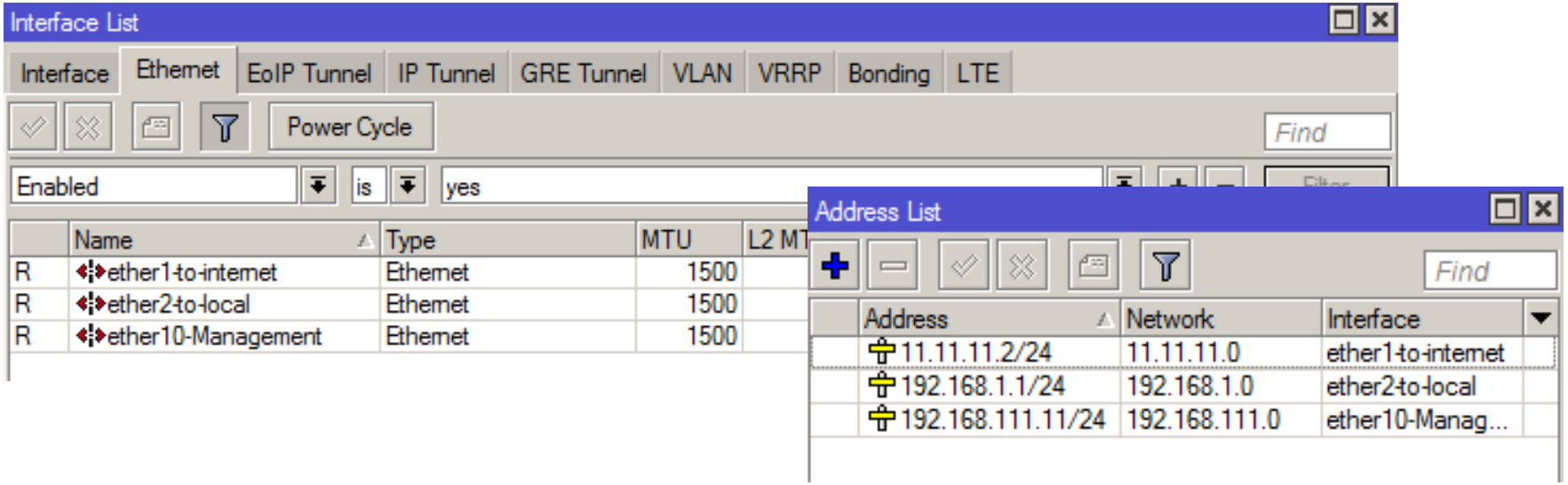
R2

- Public Address : 22.22.22.2/24
- Local Address: 192.168.2.0/24

Lab Setup

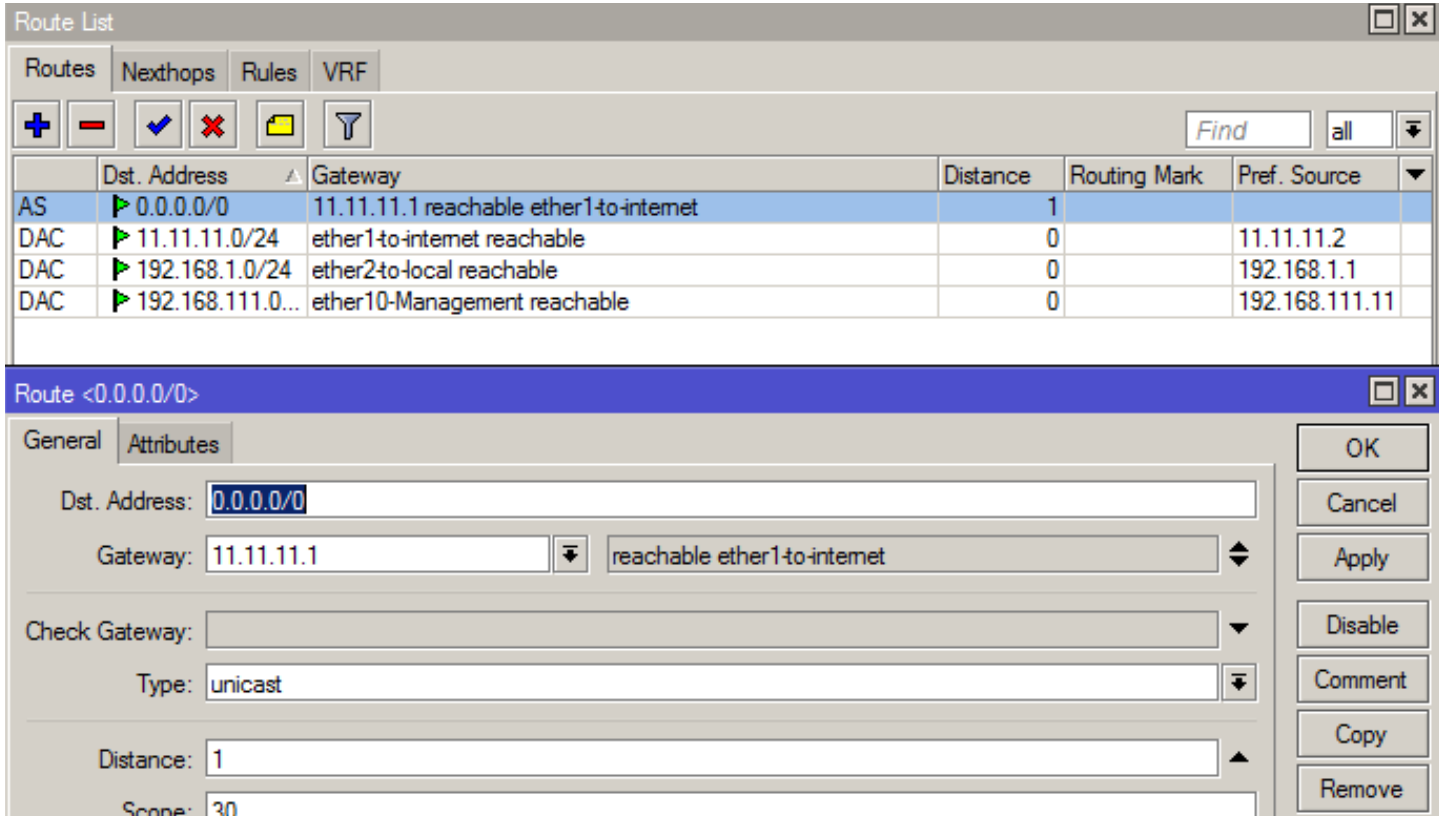


Setup IPsec R1



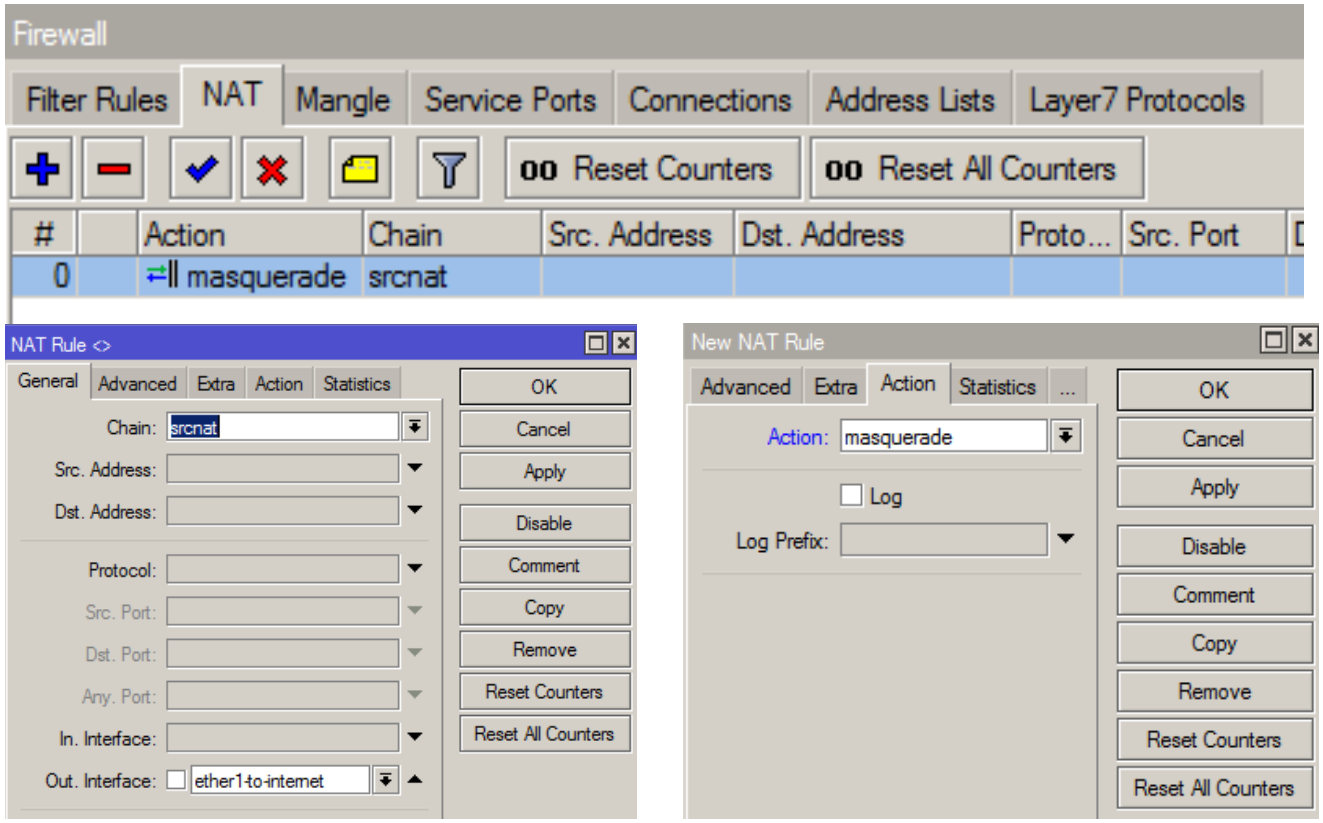

```
/ip address
add address=11.11.11.2/24 interface=ether1-to-internet network=11.11.11.0
add address=192.168.1.1/24 interface=ether2-to-local network=192.168.1.0
```

Setup IPsec R1



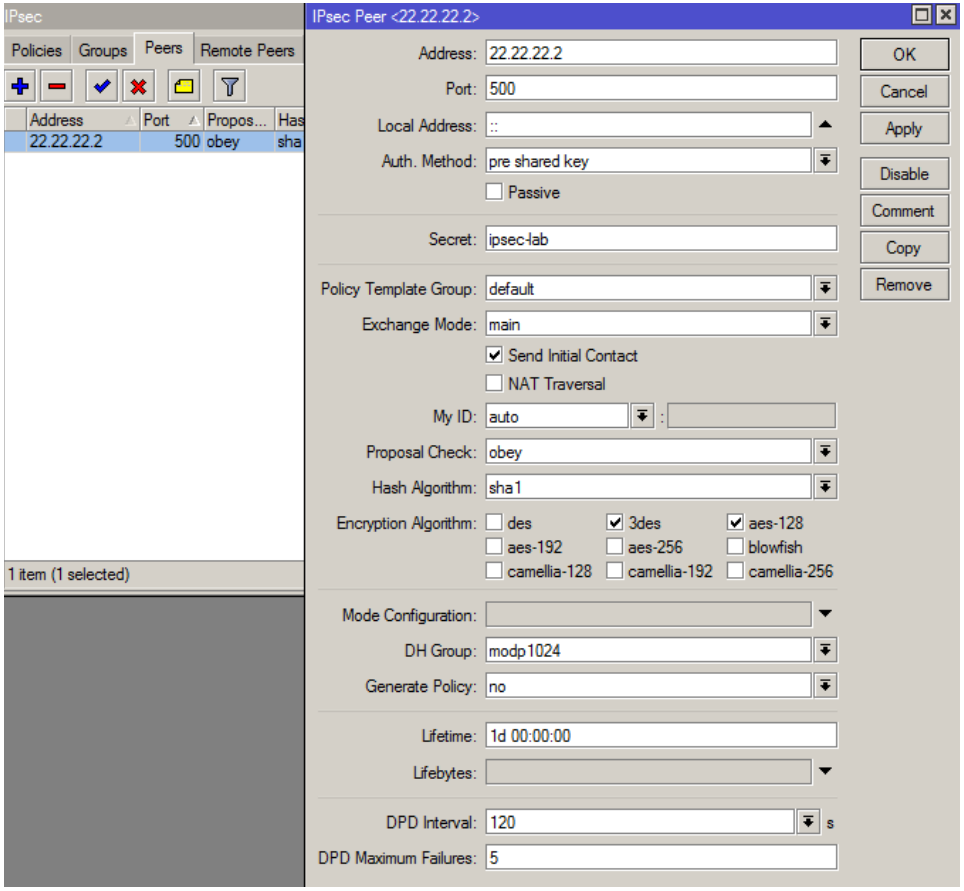

```
/ip route add distance=1 gateway=11.11.11.1
```

Setup IPsec R1



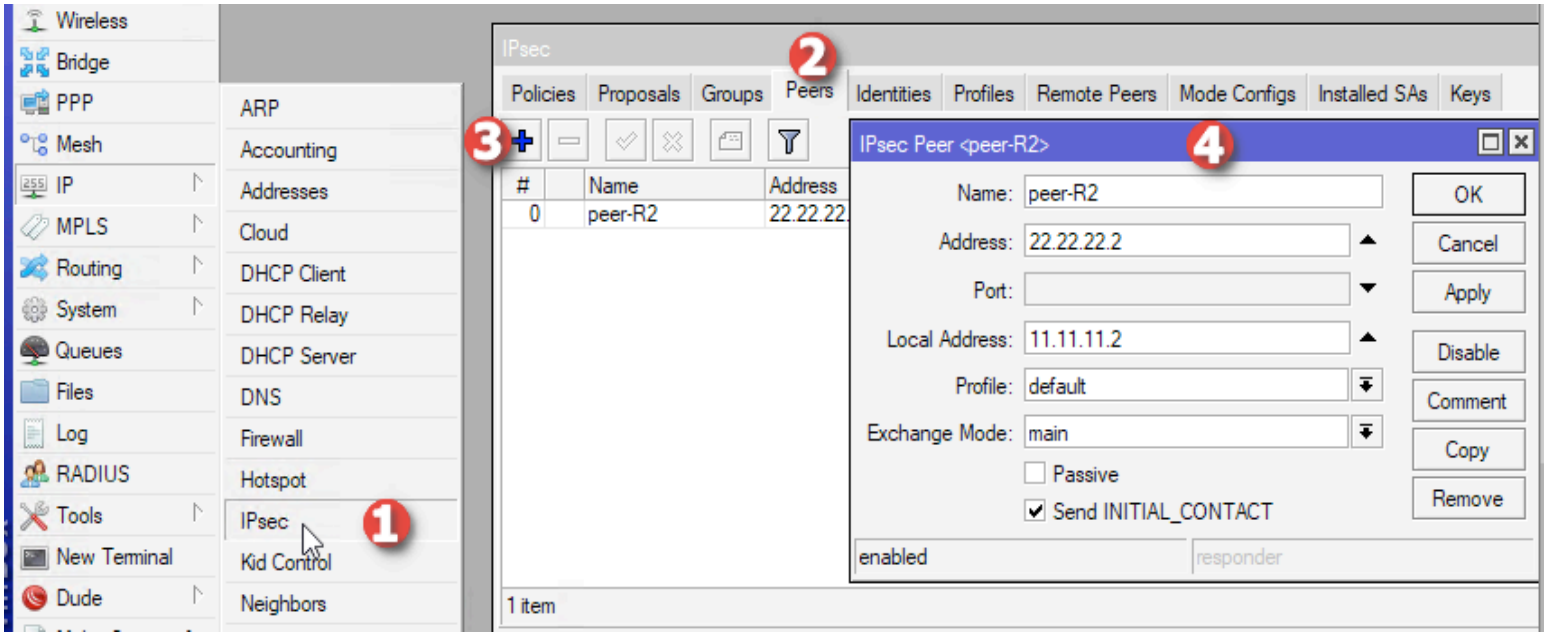

```
/ip firewall nat add action=masquerade chain=srcnat out-interface=ether1-to-internet
```

Setup IPsec R1



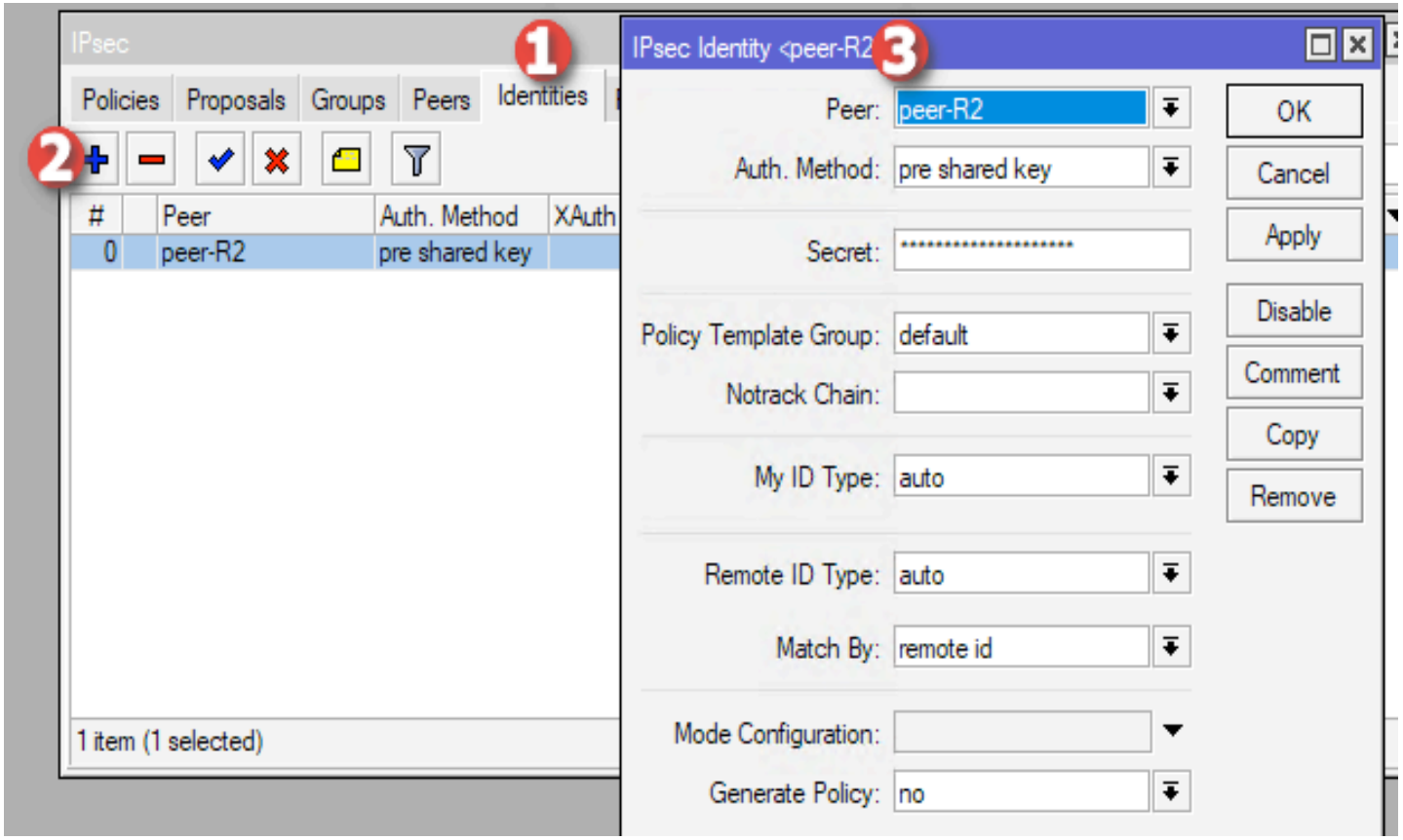

```
/ip ipsec peer add address=22.22.22.2/32 nat-traversal=no secret=ipsec-lab
```

Setup IPsec R1-NEW



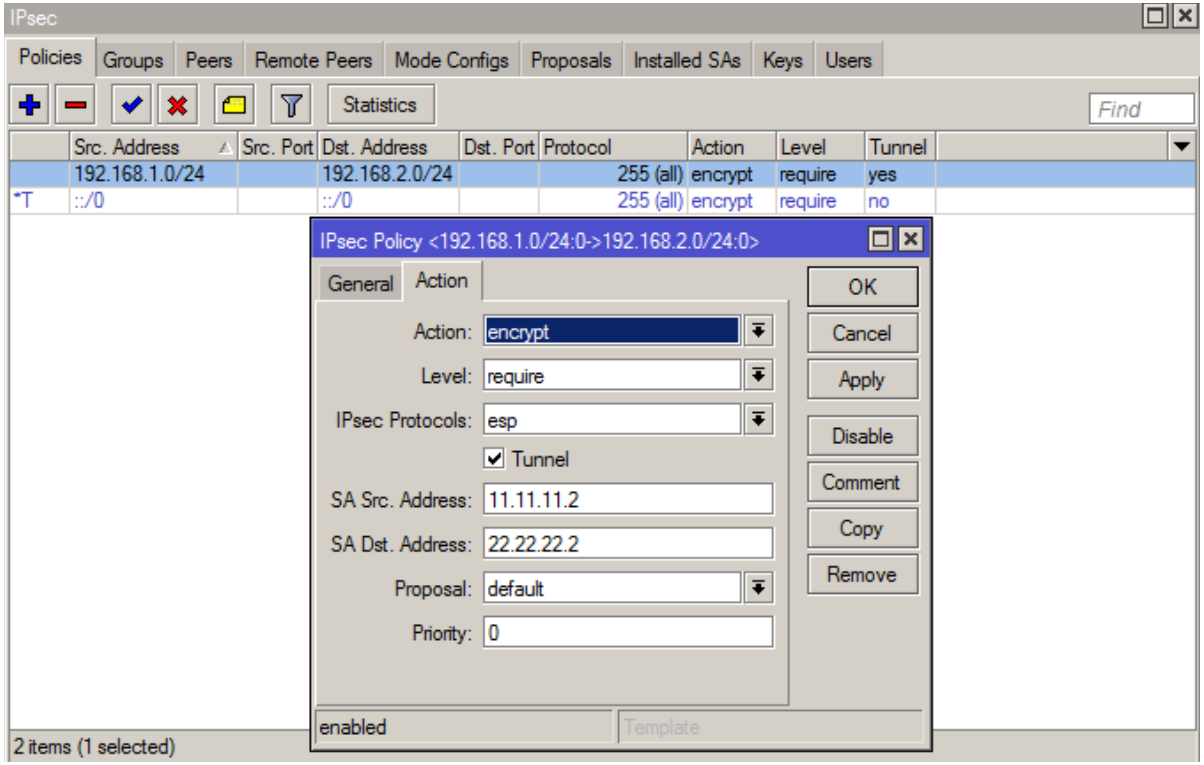

```
/ip ipsec peer add address=22.22.22.2/32 local-address=11.11.11.2 name=peer-R2
```

Setup IPsec R1-NEW



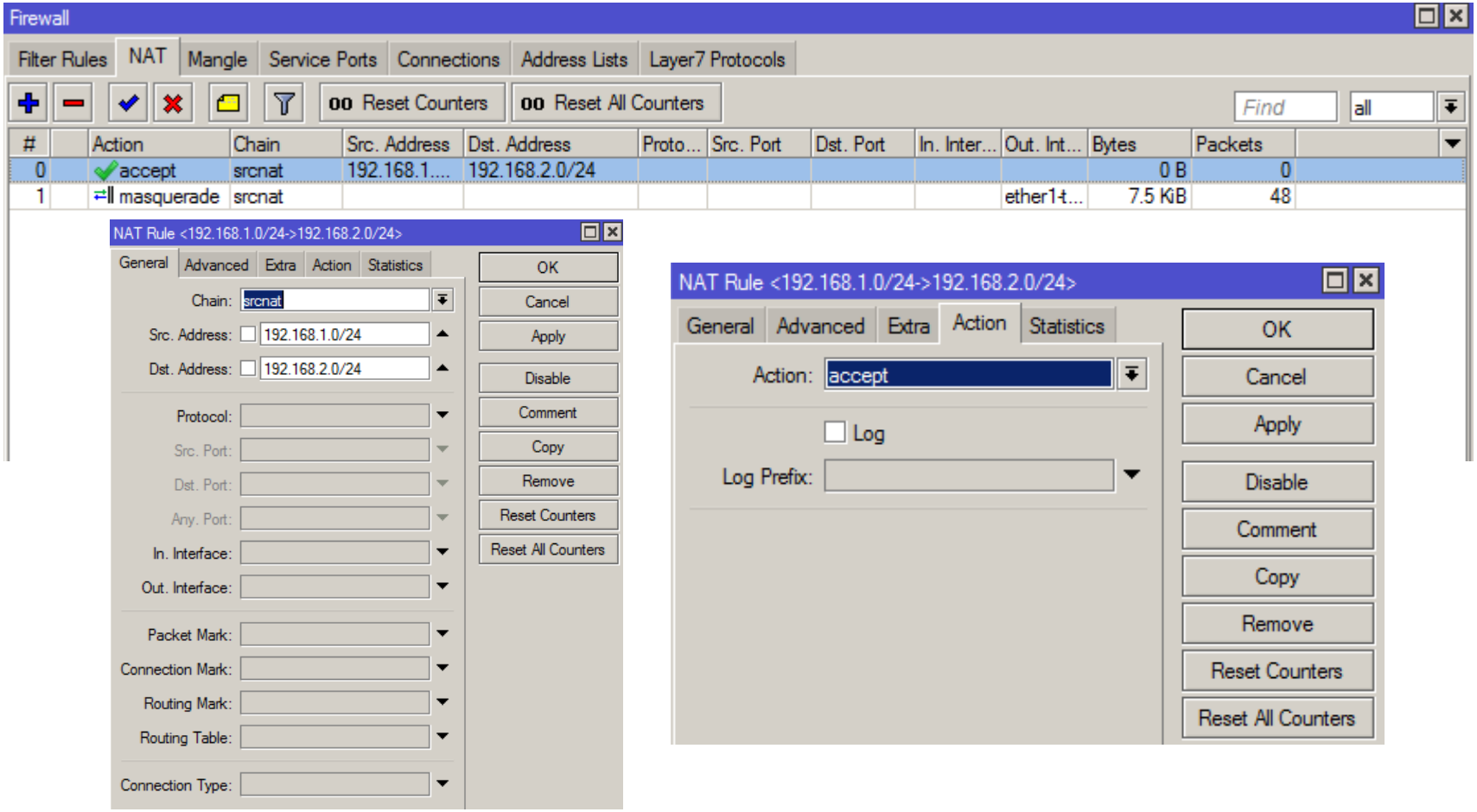

```
/ip ipsec identity add peer=peer-R2 secret=myIPSecLABsecret
```

Setup IPsec R1



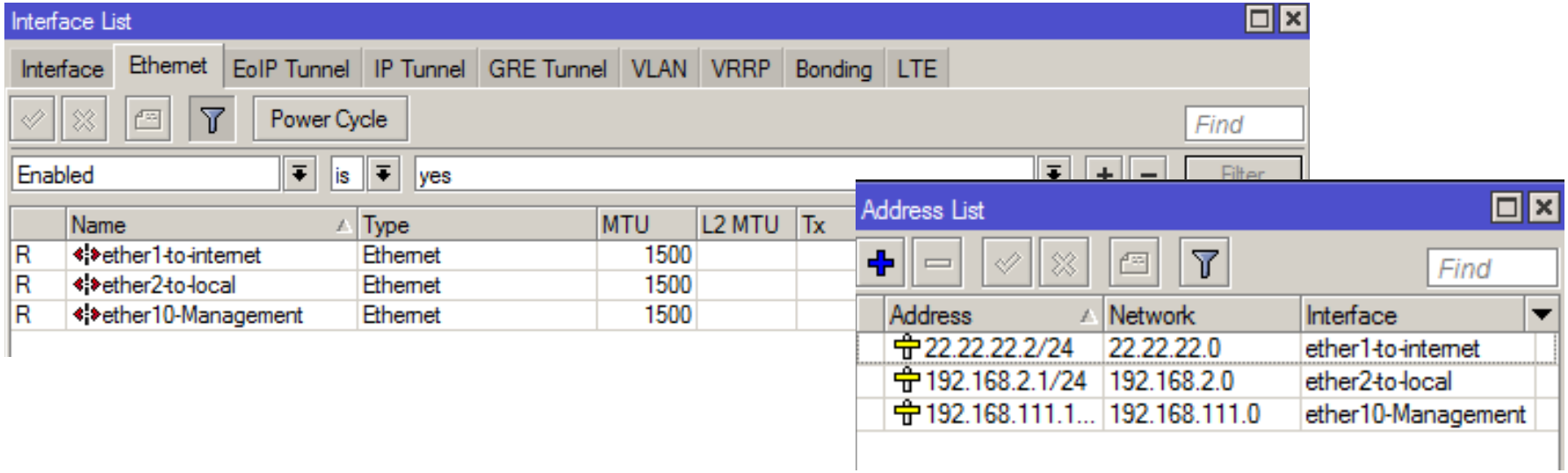

```
/ip ipsec policy
add dst-address=192.168.2.0/24 tunnel=yes sa-dst-address=22.22.22.2 \
sa-src-address=11.11.11.2 src-address=192.168.1.0/24
```

Setup IPsec R1



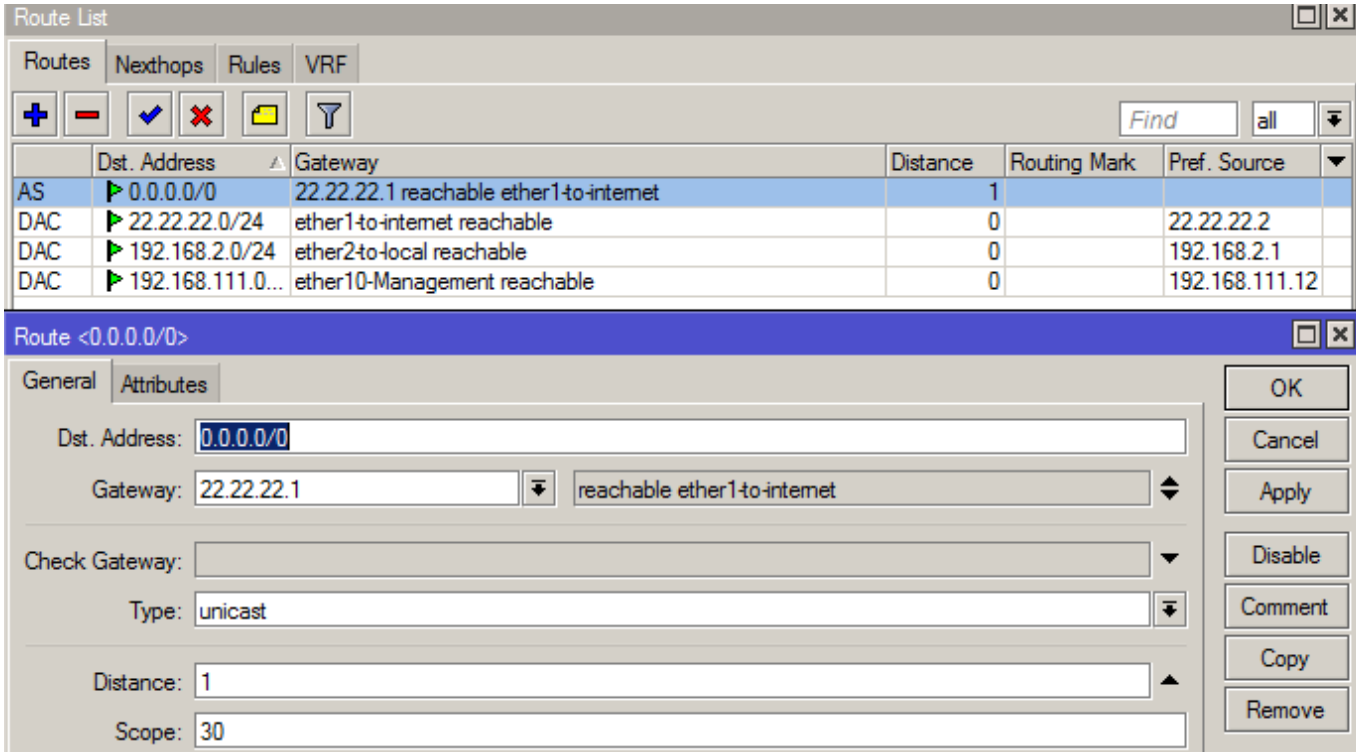

```
/ip firewall nat add chain=srcnat dst-address=192.168.2.0/24 src-address=192.168.1.0/24 place-before=0
```

Setup IPsec R2



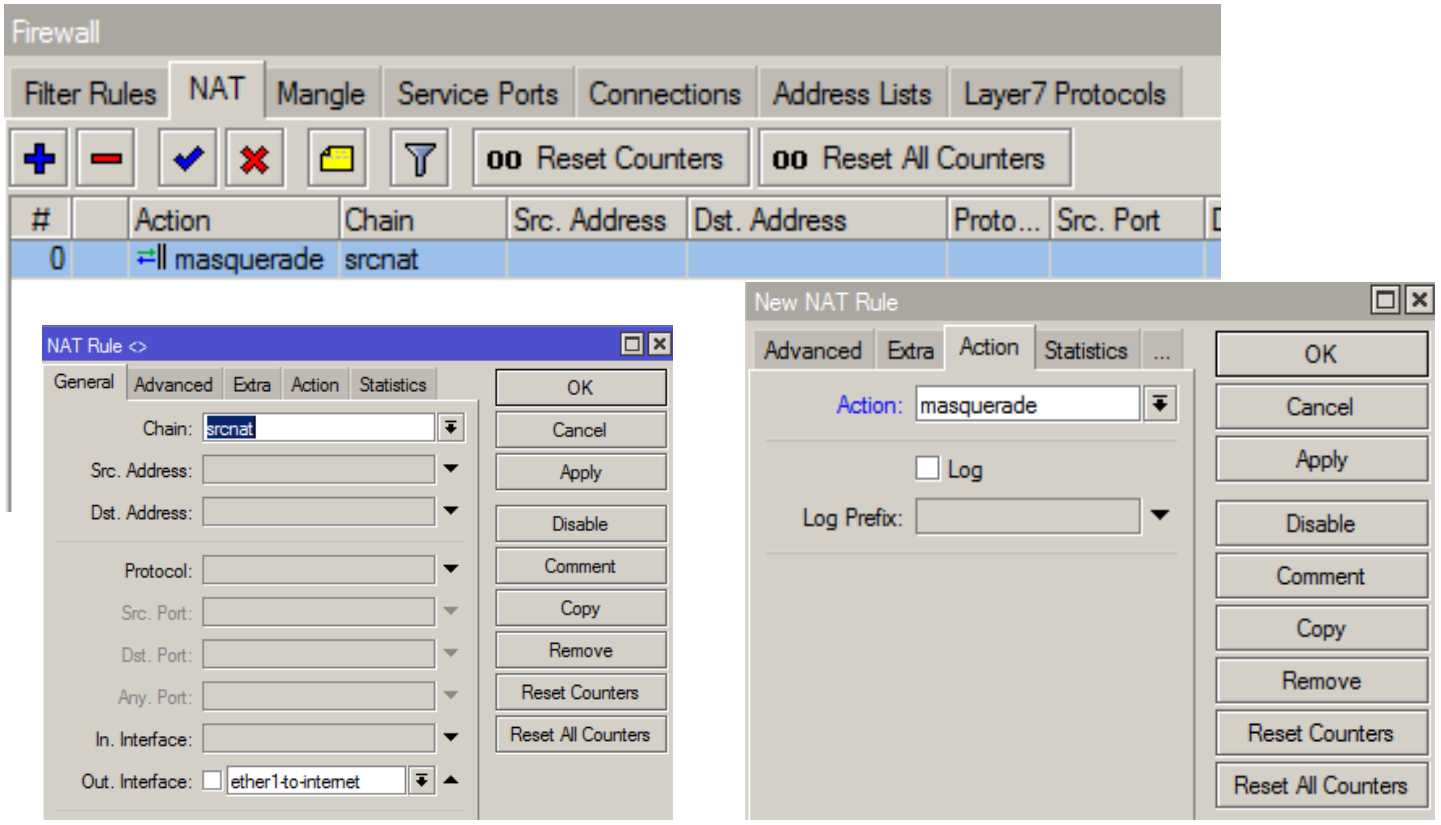

```
/ip address
add address=22.22.22.2/24 interface=ether1-to-internet network=22.22.22.0
add address=192.168.2.1/24 interface=ether2-to-local network=192.168.2.0
```

Setup IPsec R2



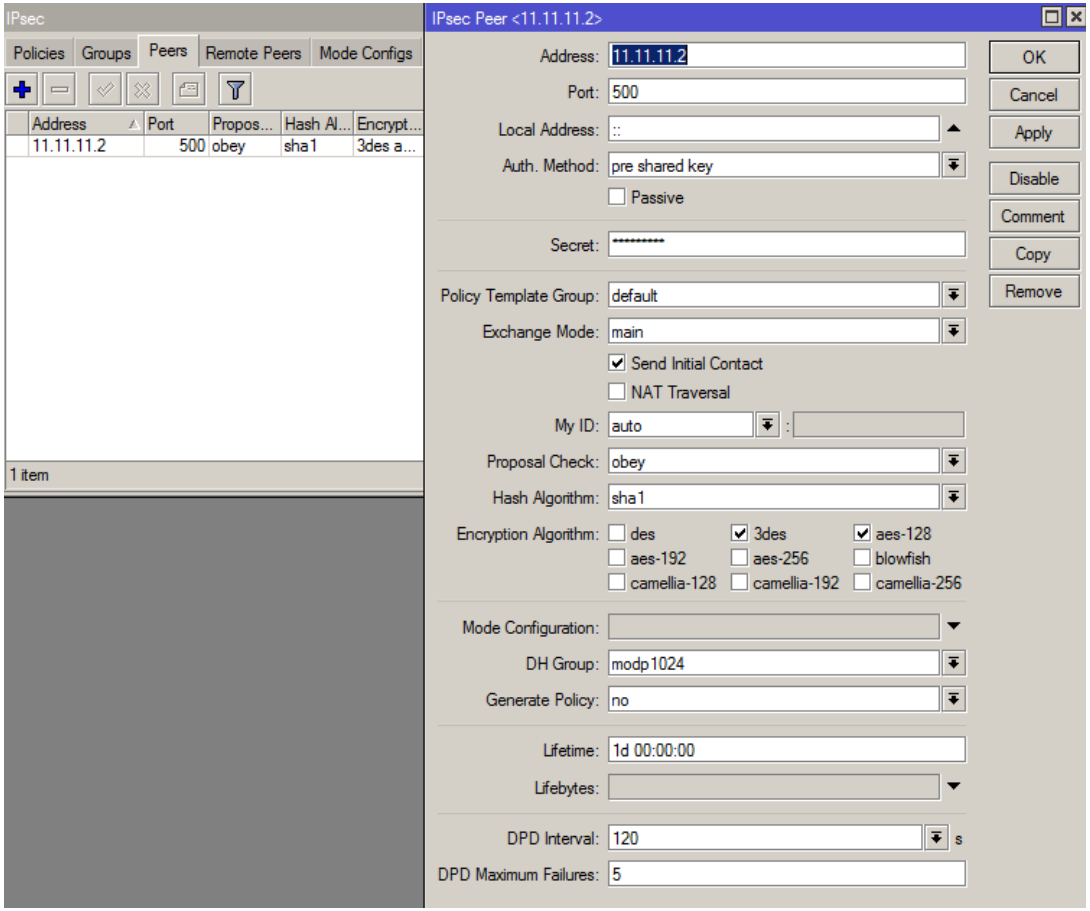

```
/ip route add distance=1 gateway=22.22.22.1
```

Setup IPsec R2



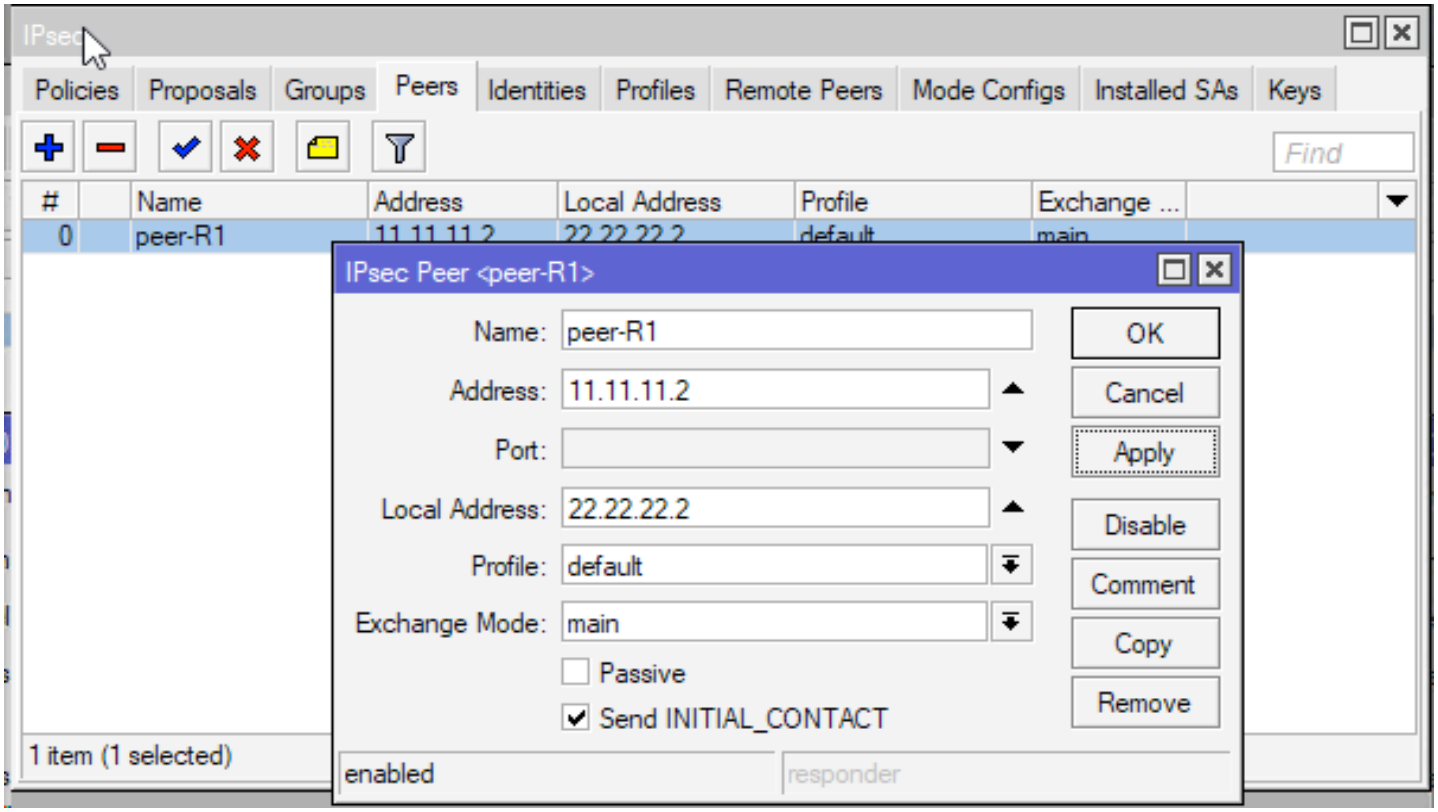

```
/ip firewall nat add action=masquerade chain=srcnat out-interface=ether1-to-internet
```

Setup IPsec R2-OLD



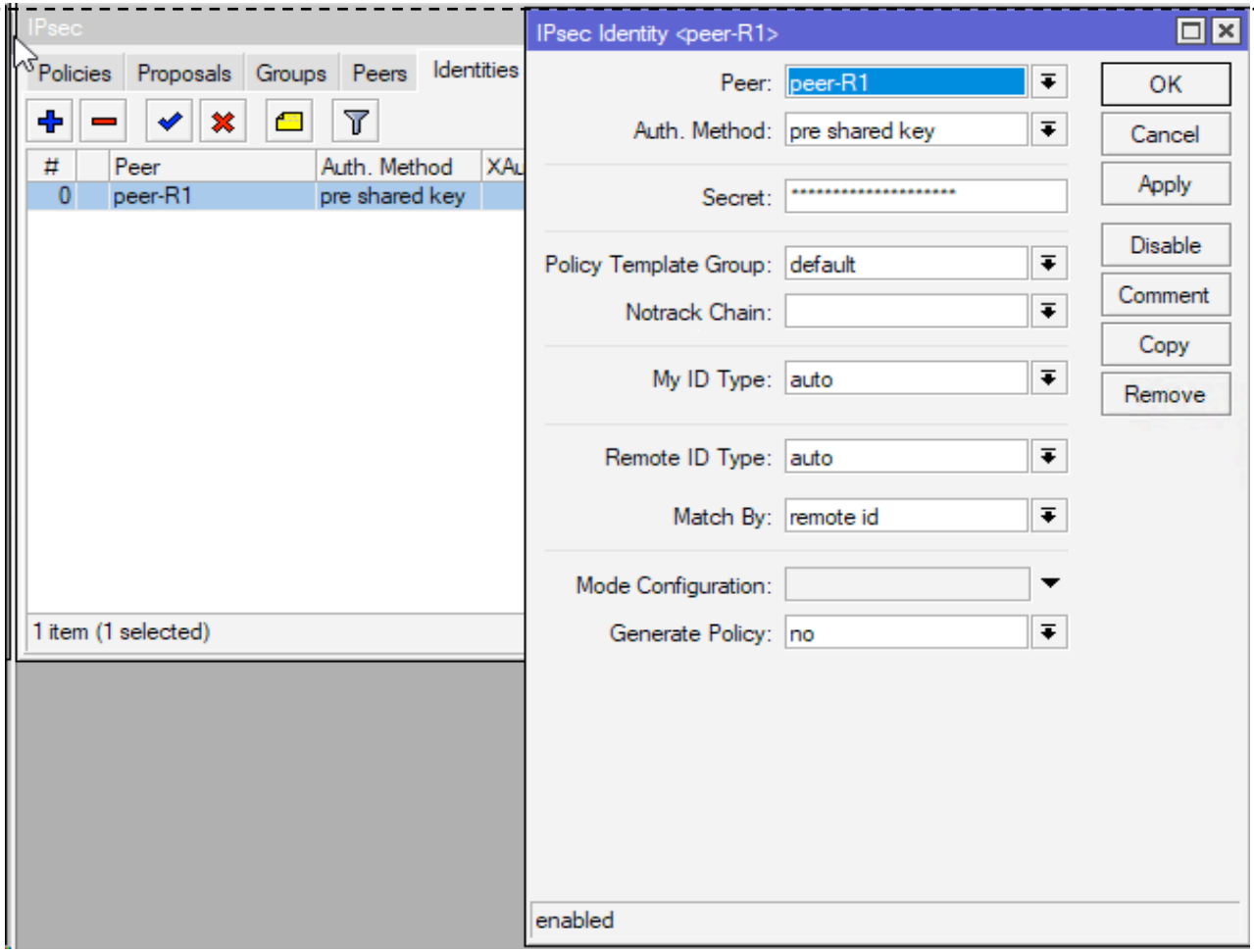

```
/ip ipsec peer add address=11.11.11.2/32 nat-traversal=no secret=ipsec-lab
```

Setup IPsec R2-NEW



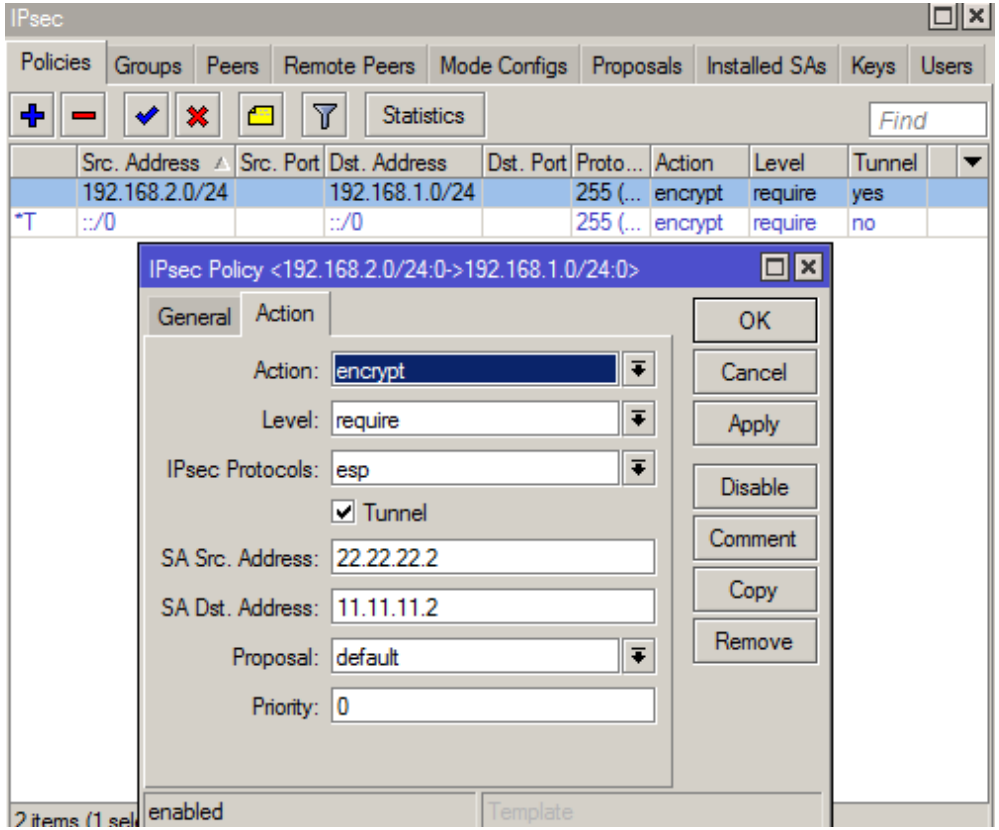

```
/ip ipsec peer add address=11.11.11.2/32 local-address=22.22.22.2 name=peer-R1
```

Setup IPsec R2-NEW



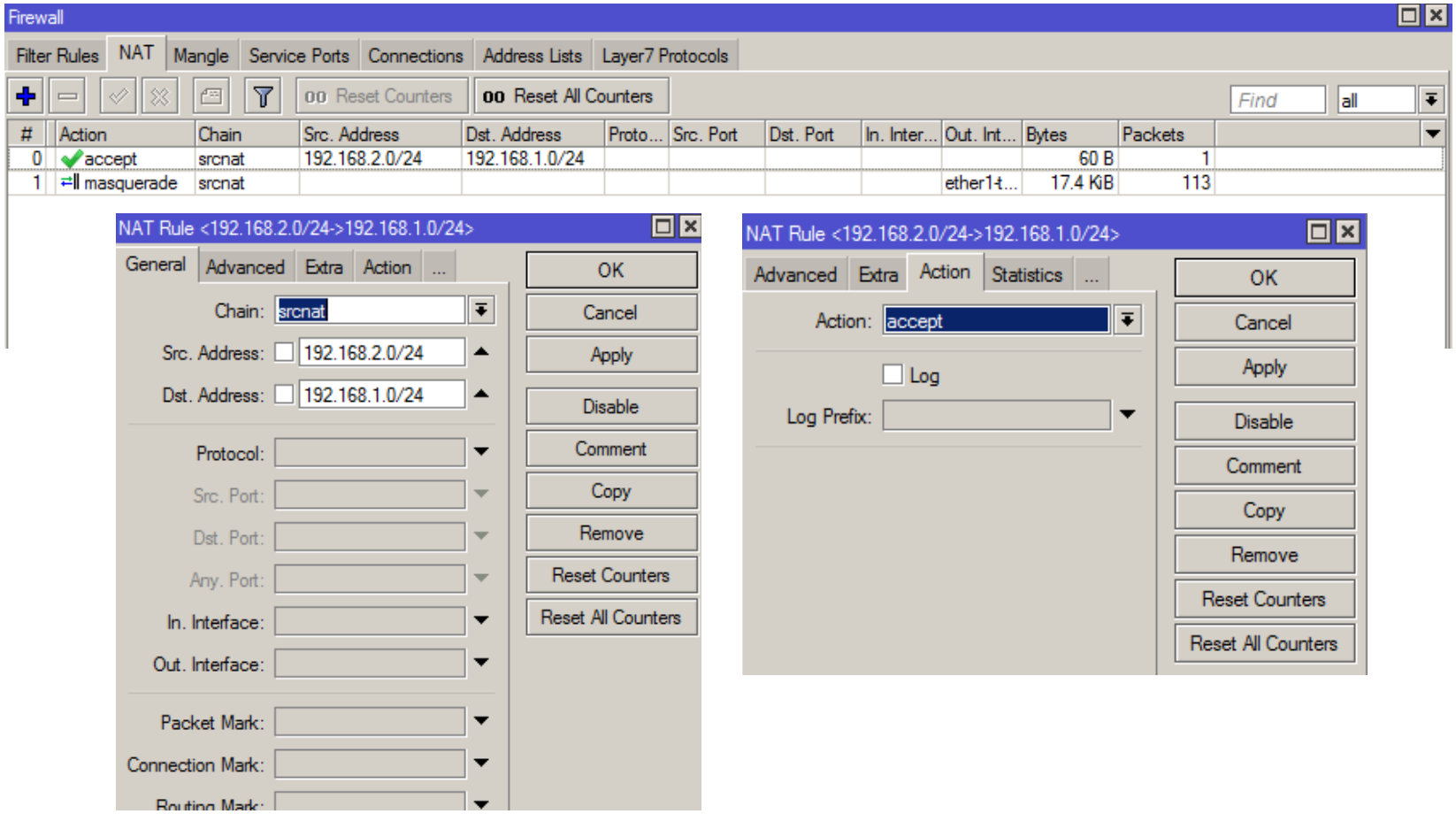
/ip ipsec identity add peer=peer-R1 secret=myIPSecLABsecret

Lab Setup



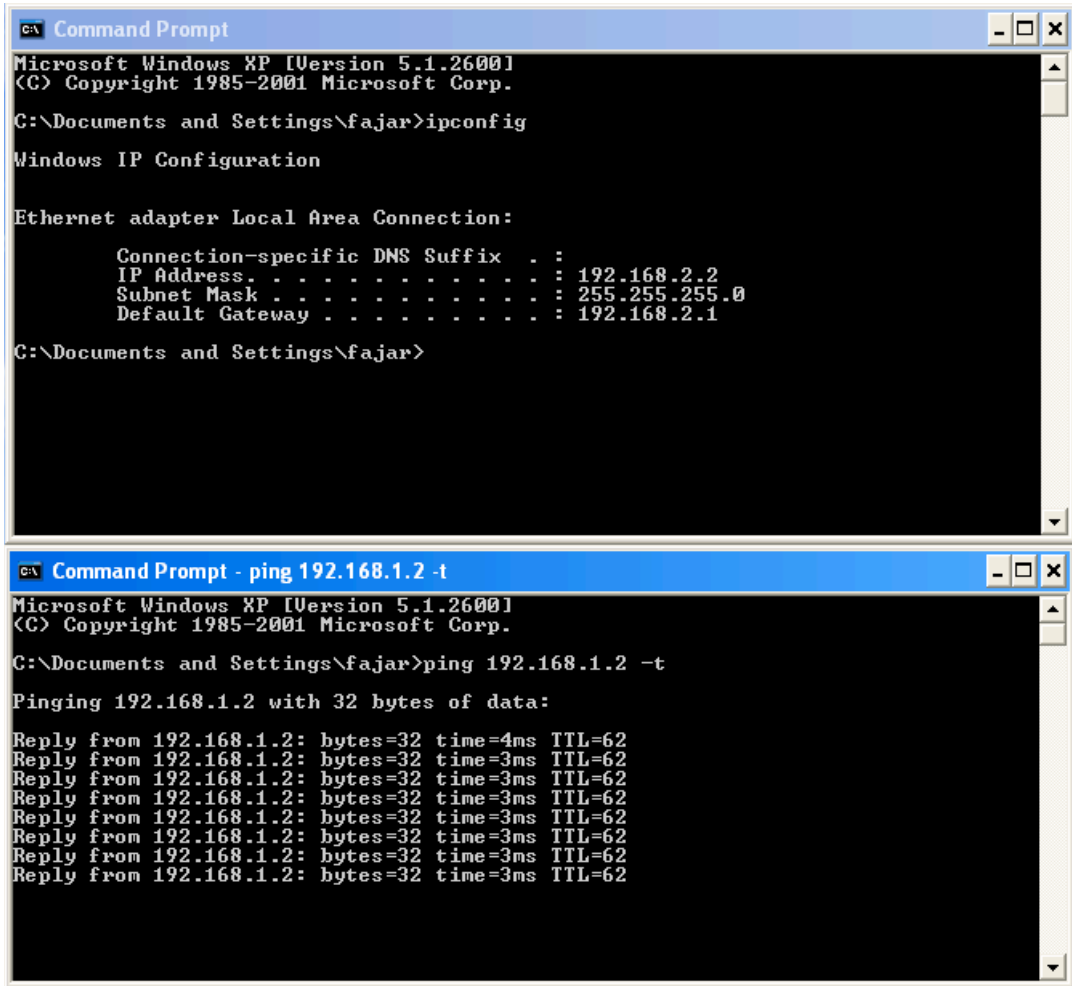

```
/ip ipsec policy
add dst-address=192.168.1.0/24 tunnel=yes sa-dst-address=11.11.11.2 \
sa-src-address=22.22.22.2 src-address=192.168.2.0/24
```

Lab Setup

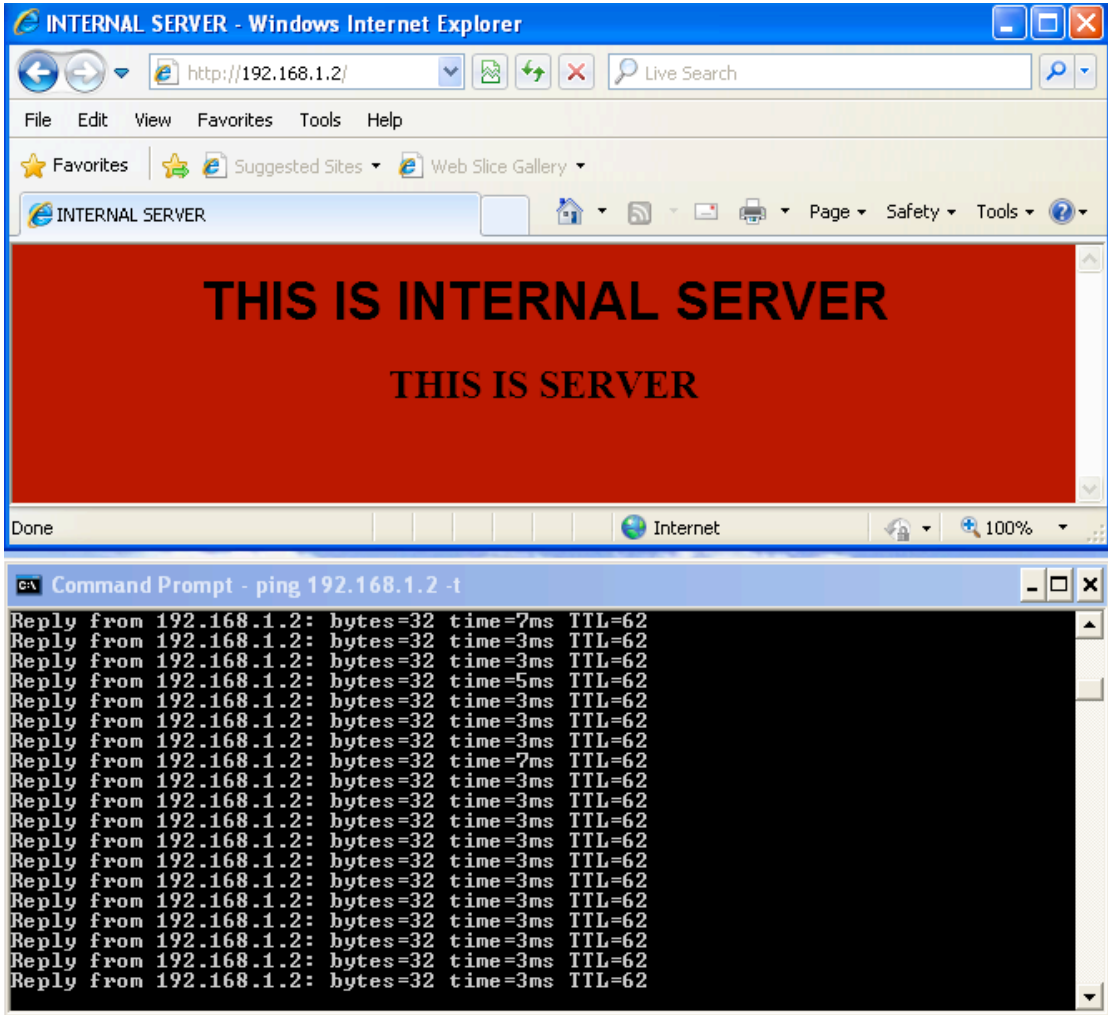


/ip firewall nat add chain=srcnat dst-address=192.168.1.0/24 src-address=192.168.2.0/24 place-before=0

Lab Setup



Lab Setup



MTCSE SUMMARY

Certification Test

- If needed reset router configuration and restore from a backup
- Make sure that you have an access to the www.mikrotik.com training portal
- Login with your account
- Choose **my training sessions**
- Good luck!