MikroTik lifehacking

Daniel Starnowski

About me

- Daniel Starnowski
- MikroTik user since 2008
- MikroTik trainer since 2011
- daniel@startik.net

What is lifehacking

MikroTik lifehacking

≠

MikroTik live hacking

Disclaimer: No RouterBOARDS were harmed in the making of this presentation

OK, so what lifehacking is?

Source: http://www.hometheaterequipment.com/general-discussion-10/lifehacking-5772/

Really, what is lifehacking about?

source: http://wonderfulengineering.com/100-clever-life-improving-ideas-that-you-can-use-in-everyday-life/

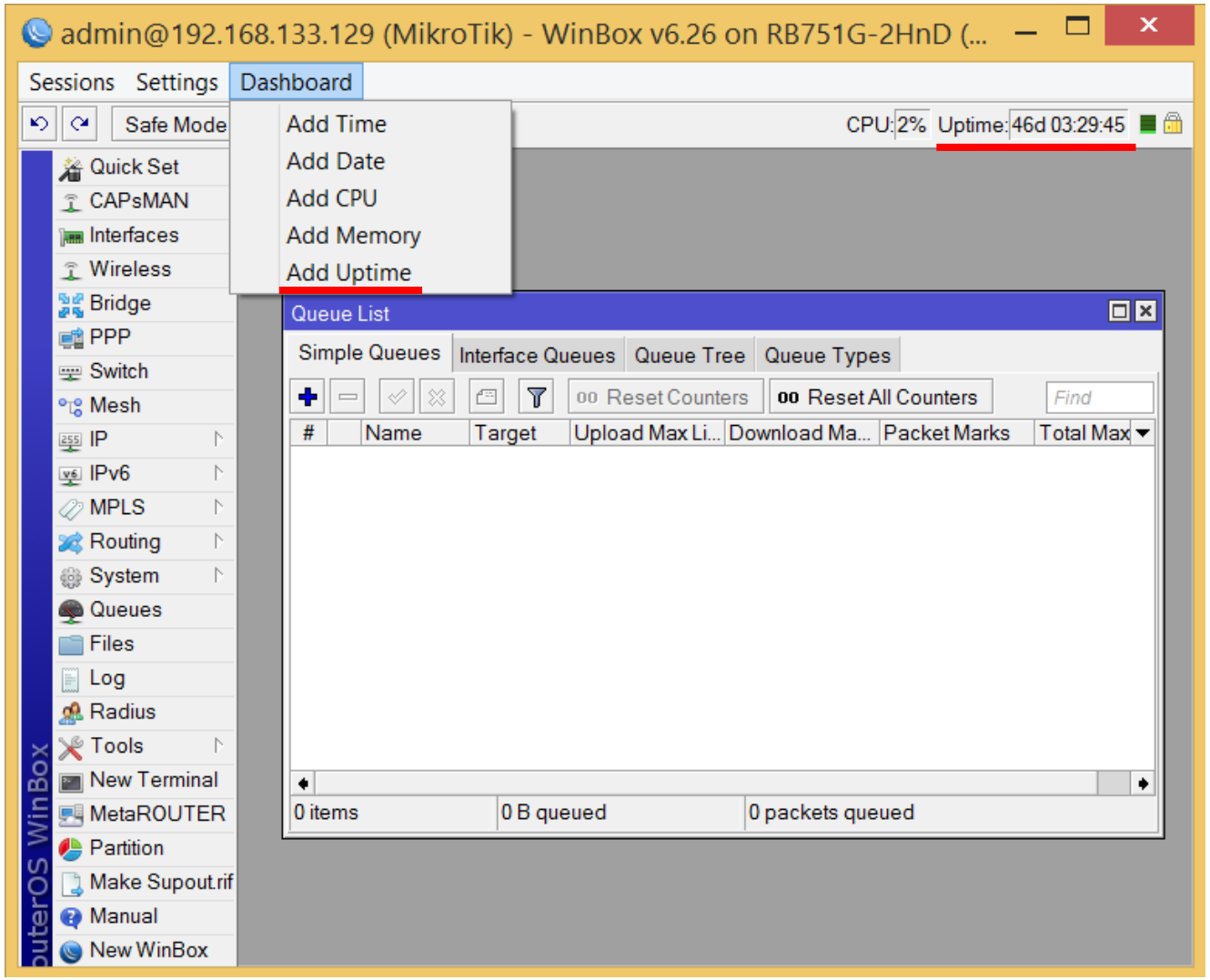
MikroTik lifehacking – why, for whom?

- The presentation target is to INSPIRE, not to teach ☺
- Some examples of my private MikroTik lifehacks, which – after I started using – made my life easier
- You may be using some of them already

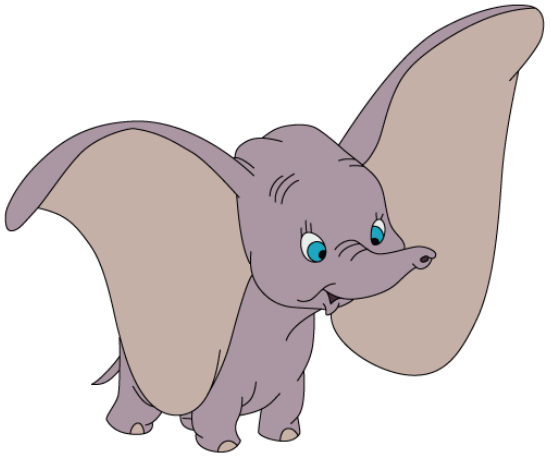
Problem example – **WinBox** connection frozen

- Sometimes, when loosing connectivity to MikroTik router, WinBox disconnects after time (~20 seconds)
- Before the time – **no sign** of connectivity problems
- Empty lists of items (IP addresses, etc.) in WinBox can be caused by temporary connection loss
- SOLUTION: always keep **uptime** shown in WinBox window! ☺

Problem example – **WinBox** connection frozen



And now for something completely different



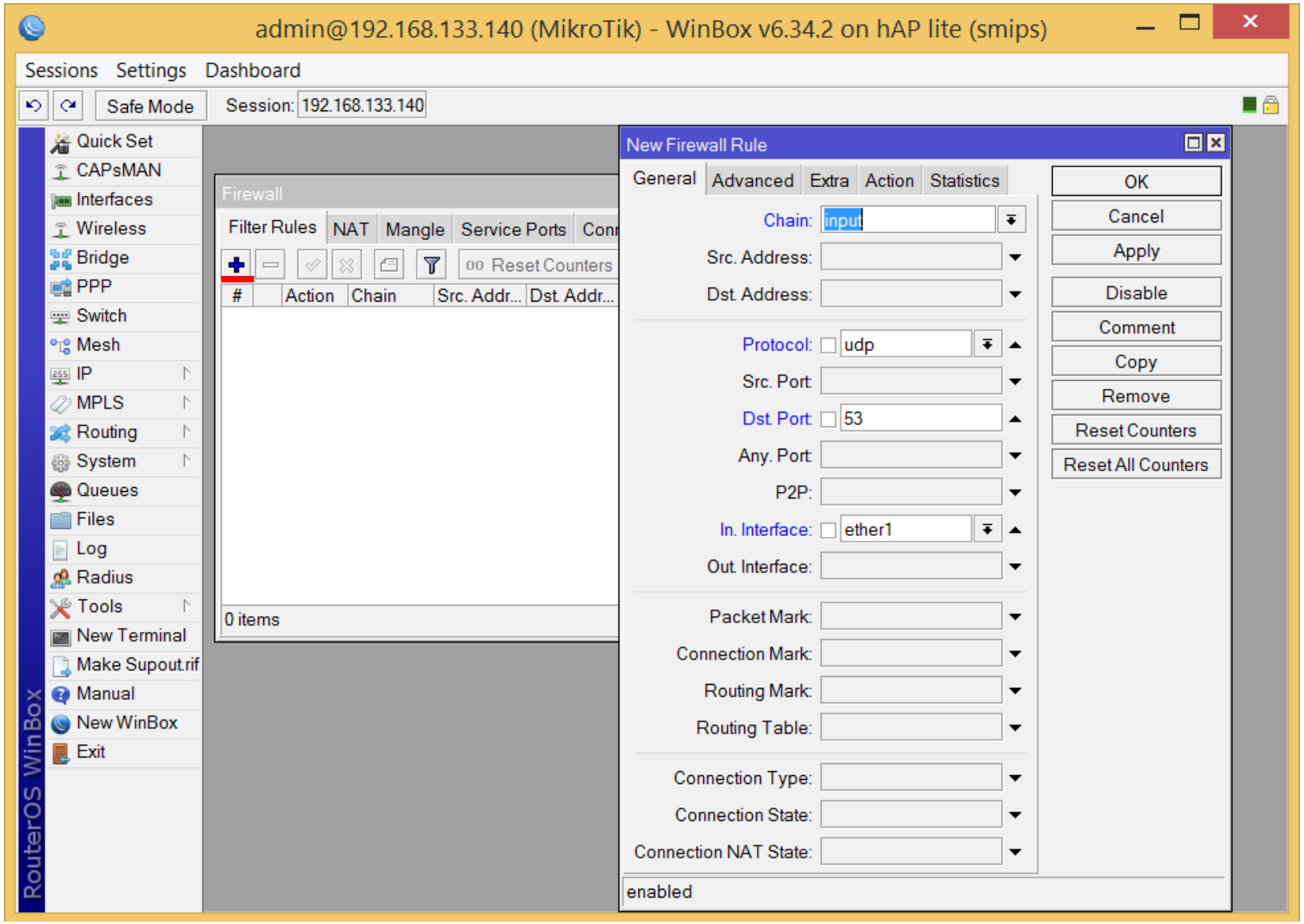
Copy – not a “hack”, but often forgotten

- WinBox has a “**Copy**” button, that speeds things up!

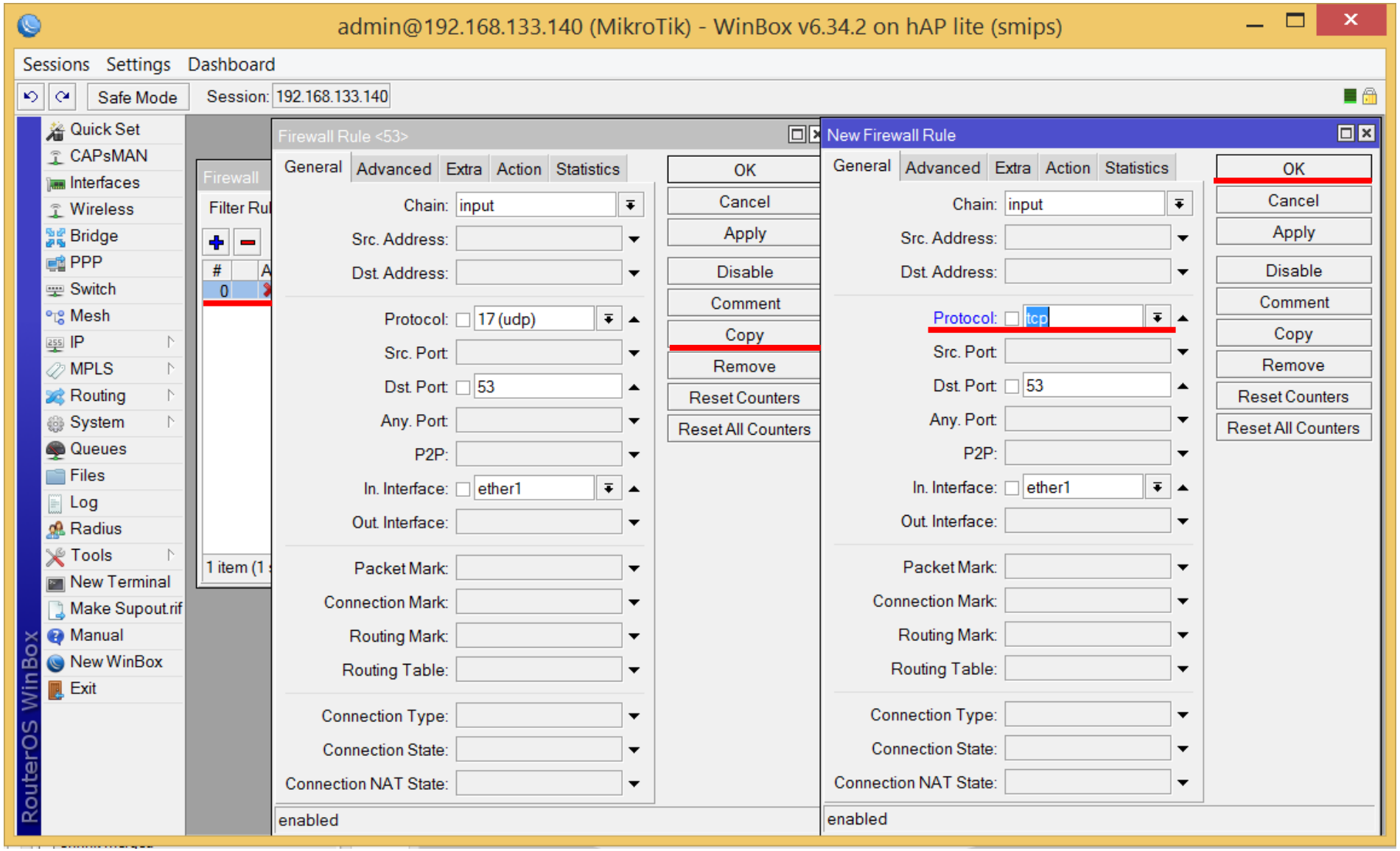
- Example – create rules blocking DNS connection from not trusted addresses
- Chain=input, protocol=udp, port=53, src-address-list, action…

- I said “rules”, DNS can also use TCP…
- How will you do this one? Add new one, or copy the existing**?** ☺

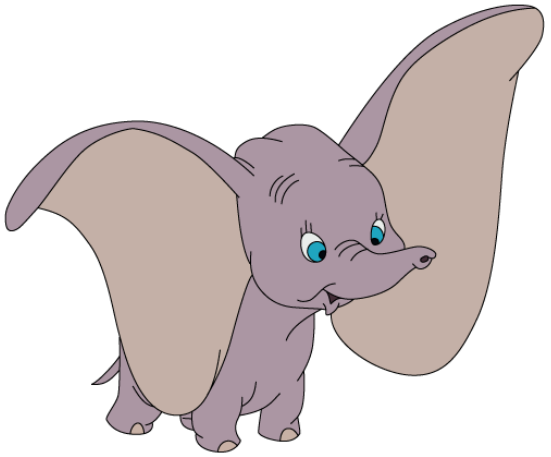
Copy – not a “hack”, but often forgotten



Copy – not a “hack”, but often forgotten



And now for something completely different

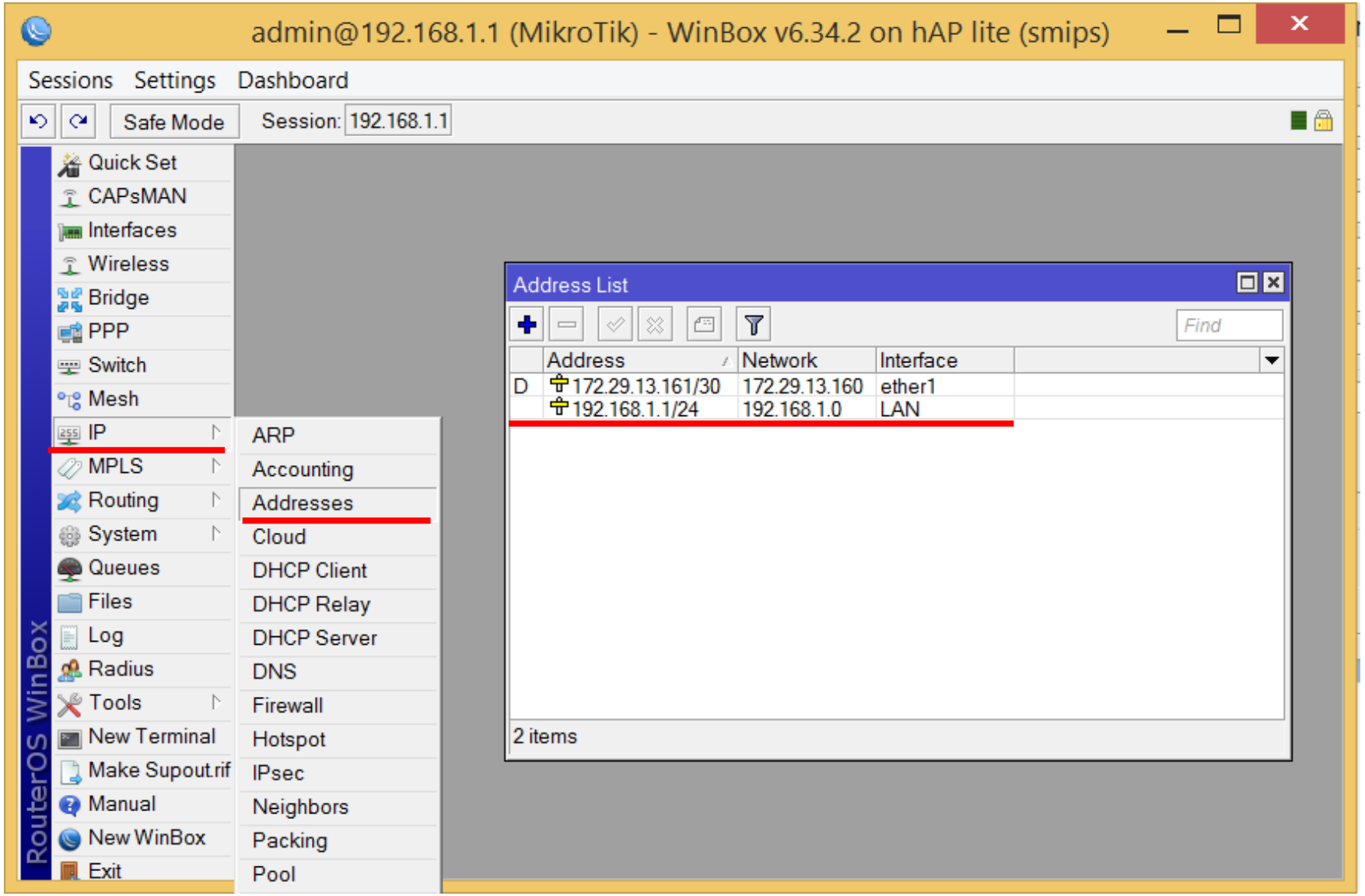


Learn how to use CLI, it’s not difficult!

- CLI can often speed things up – learn, how to use it ☺
- It’s not difficult – just configure something in WinBox, then go to New Terminal and see, how it was configured

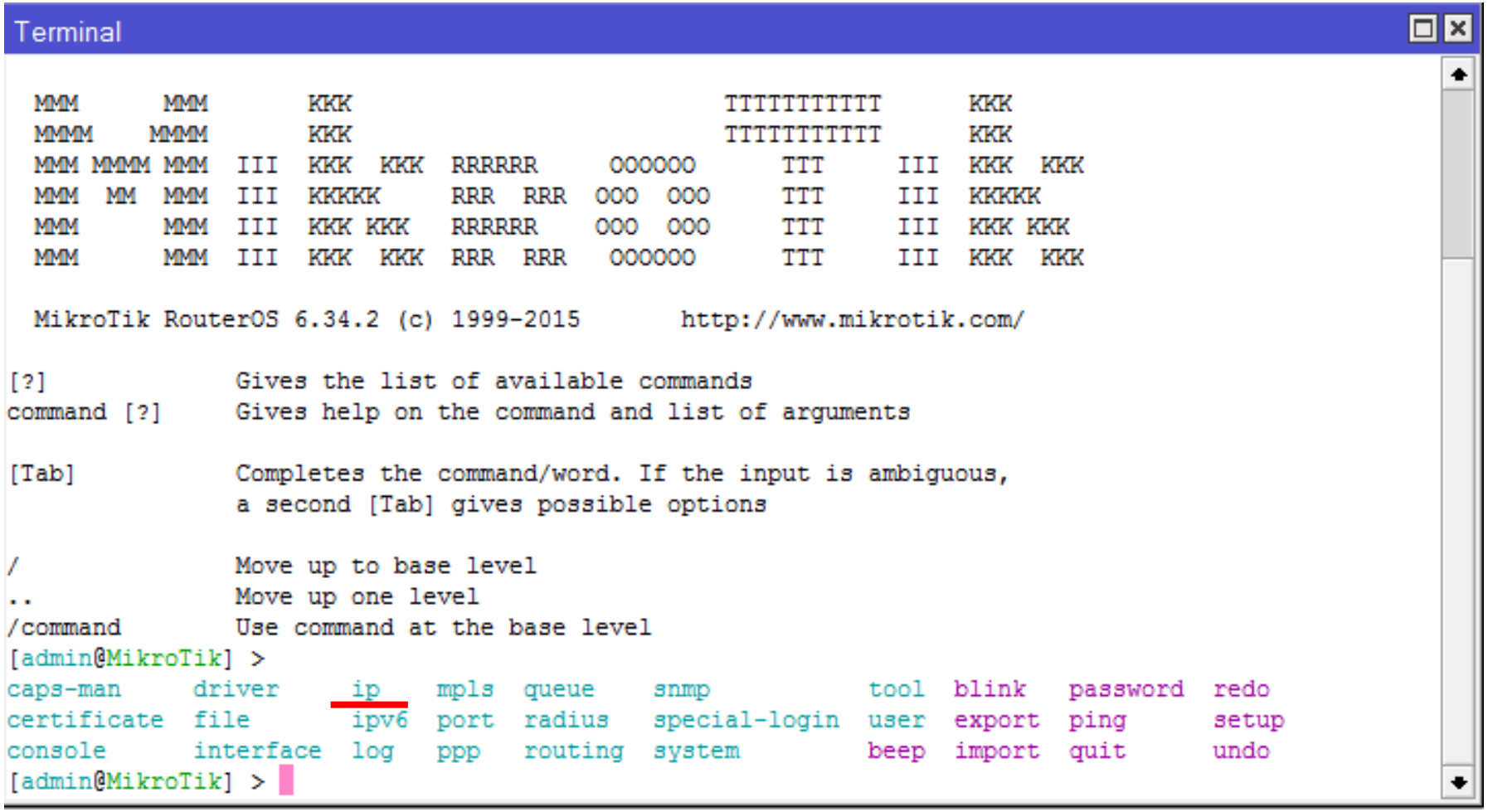
- Find the right section in terminal
- Use **print** to see, what is configured there
- Use **export** to see, how it is configured
- Use **<TAB>** and **?** to see available options

CLI – example with IP addresses



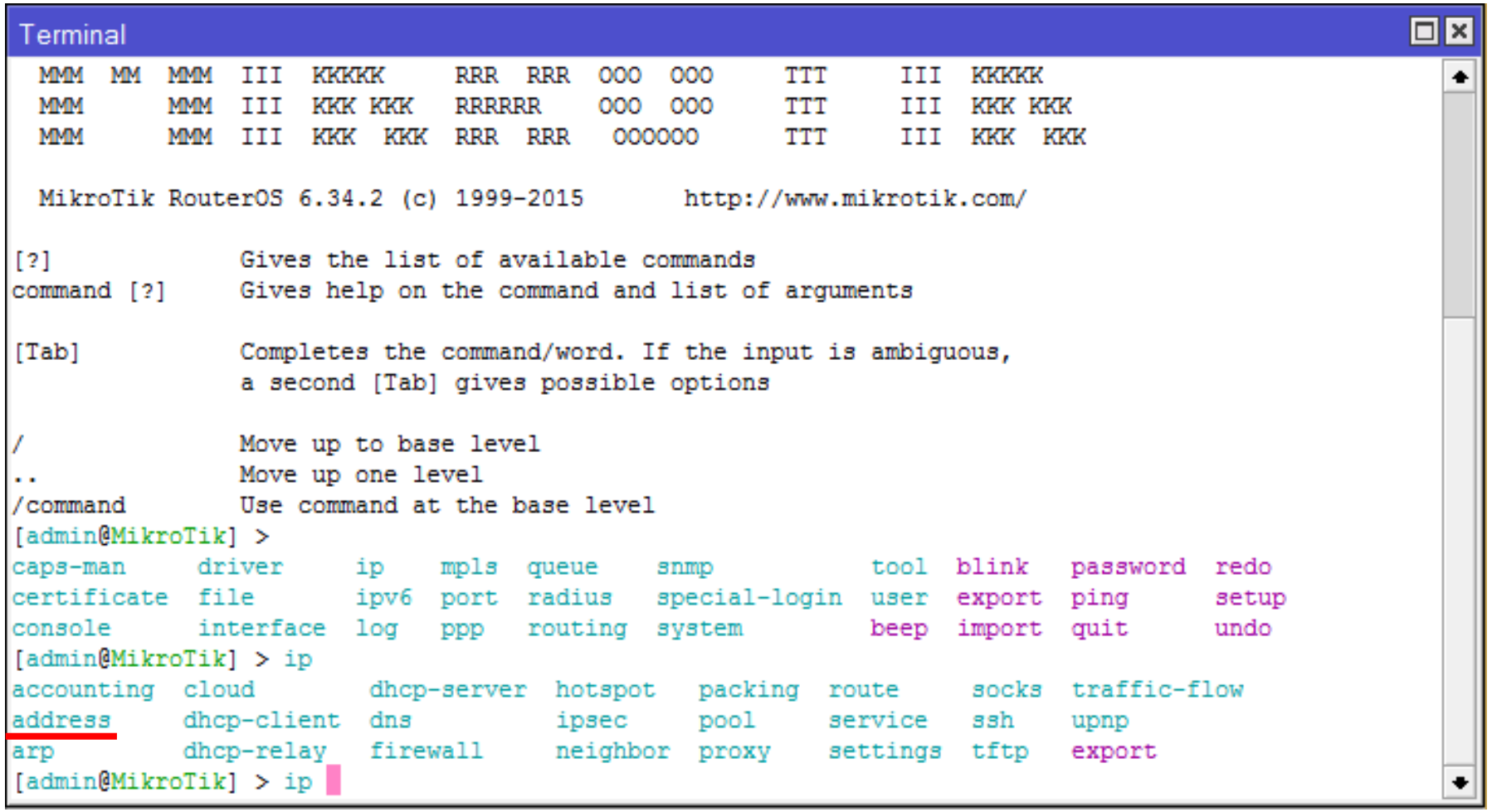
CLI – example with IP addresses

- Using <TAB>



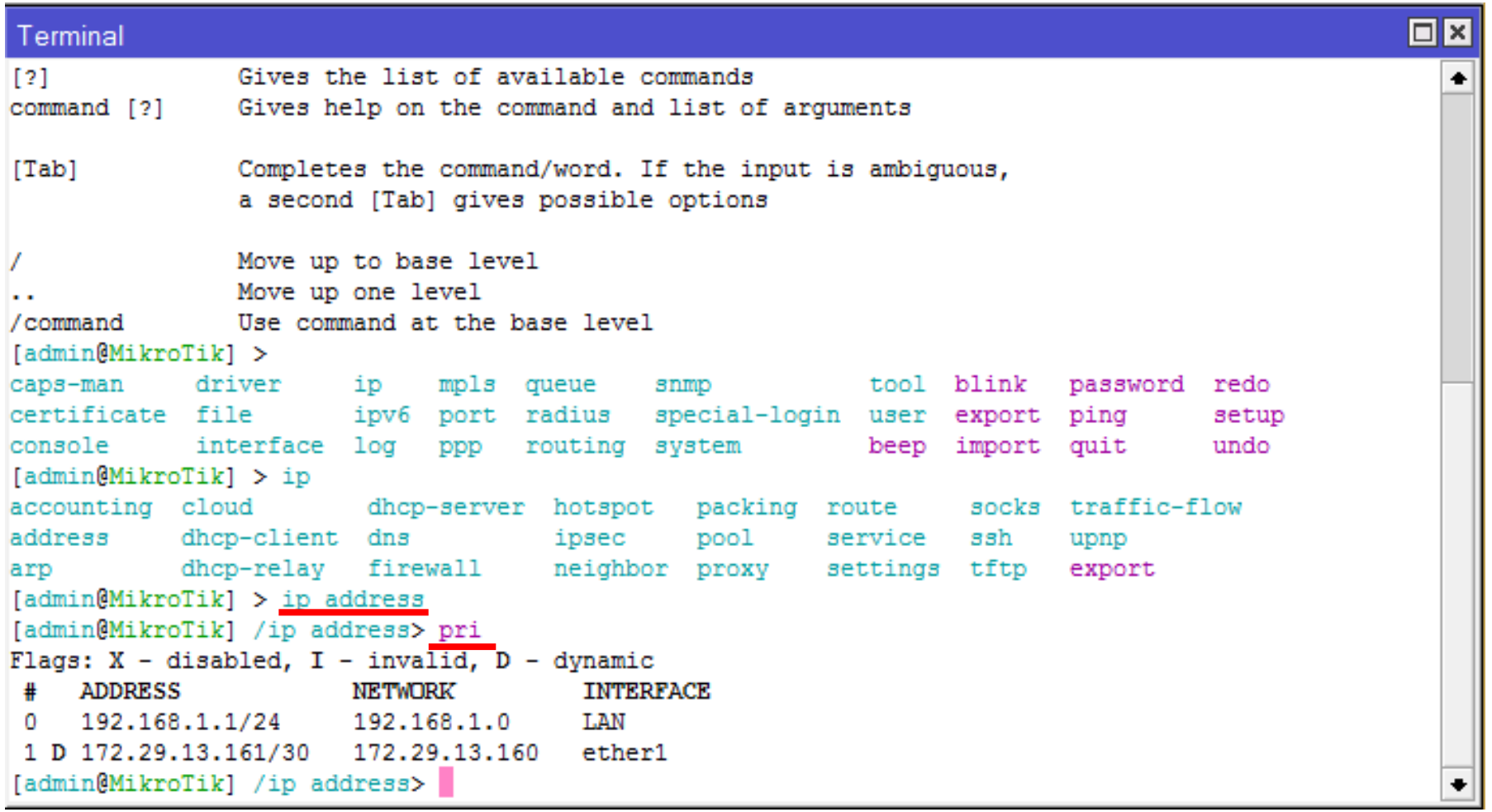
CLI – example with IP addresses

- Typing “ip” and using <TAB> again



CLI – example with IP addresses

- Using **print** in /ip address



CLI – example with IP addresses

- Using **export** in /ip address

```
Terminal
/               Move up to base level
..              Move up one level
/command        Use command at the base level
[admin@MikroTik] >
caps-man     driver     ip    mpls  queue    snmp           tool  blink   password  redo
certificate  file       ipv6  port  radius   special-login  user  export  ping      setup
console      interface  log   ppp   routing  system         beep  import  quit      undo
[admin@MikroTik] > ip
accounting  cloud        dhcp-server  hotspot   packing  route     socks  traffic-flow
address     dhcp-client  dns          ipsec     pool     service   ssh    upnp
arp         dhcp-relay   firewall     neighbor  proxy    settings  tftp   export
[admin@MikroTik] > ip address
[admin@MikroTik] /ip address> pri
Flags: X - disabled, I - invalid, D - dynamic
 #   ADDRESS            NETWORK         INTERFACE
 0   192.168.1.1/24     192.168.1.0     LAN
 1 D 172.29.13.161/30   172.29.13.160   ether1
[admin@MikroTik] /ip address> export
# feb/24/2016 23:37:32 by RouterOS 6.34.2
# software id = 1VWV-4NDH
#
/ip address
add address=192.168.1.1/24 interface=LAN network=192.168.1.0
[admin@MikroTik] /ip address>
```

Learn how to use CLI, really!

- Most sections are easy to find
- **IP -> Firewall -> Filter Rules** can be found in **/ip firewall filter**

- Sometimes small differences
- **Wireless** in WinBox, but **/interface wireless** in CLI
- **System -> Users** in WinBox, but **/user** in CLI

- Or just type **/export** to analyse whole configuration

Prepare CLI rules for “massive” configuration

- Imagine, that your MikroTik router is running DHCP server
- You get a spreadsheet with a list of 200 IP and MAC addresses of the devices that need to be added statically to the DHCP server
- Will you click “**+**”, copy-paste them, and click “**OK**” 200 times? ☺

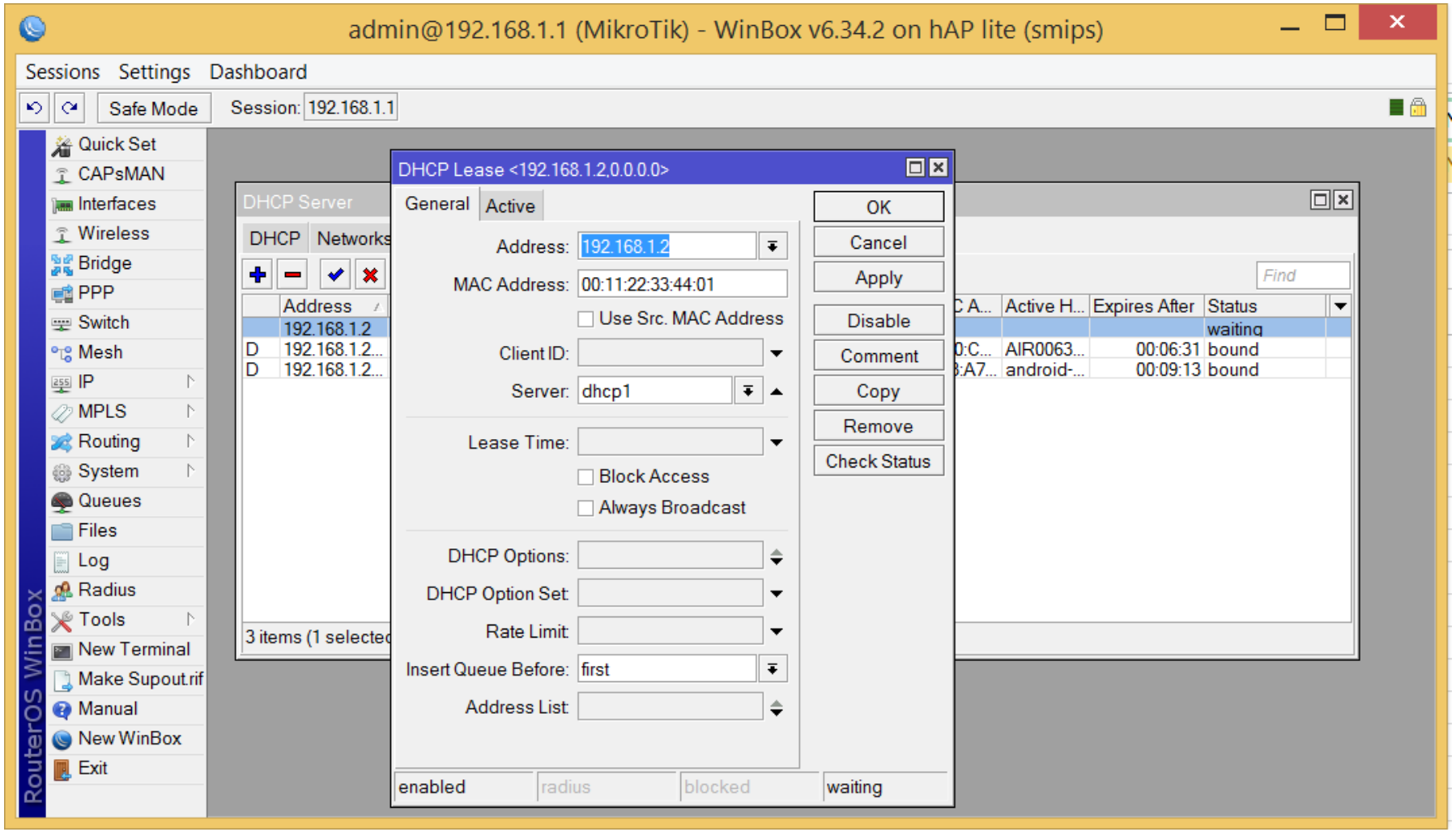
Prepare CLI rules for “massive” configuration

B3 | 192.168.1.4

	A	B	C	D	E	F
37		192.168.1.38	00:11:22:33:44:37			
38		192.168.1.39	00:11:22:33:44:38			
39		192.168.1.40	00:11:22:33:44:39			
40		192.168.1.41	00:11:22:33:44:40			
41		192.168.1.42	00:11:22:33:44:41			
42		192.168.1.43	00:11:22:33:44:42			
43		192.168.1.44	00:11:22:33:44:43			
44		192.168.1.45	00:11:22:33:44:44			
45		192.168.1.46	00:11:22:33:44:45			
46		192.168.1.47	00:11:22:33:44:46			
47		192.168.1.48	00:11:22:33:44:47			
48		192.168.1.49	00:11:22:33:44:48			
49		192.168.1.50	00:11:22:33:44:49			
50		192.168.1.51	00:11:22:33:44:50			
51		192.168.1.52	00:11:22:33:44:51			

Prepare CLI rules for “massive” configuration

- Add the first line with WinBox, if you’re not sure



Prepare CLI rules for “massive” configuration

- Check the syntax in CLI

Terminal

```
  MMM MMMM MMM  III  KKK  KKK  RRRRRR     OOOOOO      TTT     III  KKK  KKK
  MMM  MM  MMM  III  KKKKK     RRR  RRR  OOO  OOO     TTT     III  KKKKK
  MMM      MMM  III  KKK KKK   RRRRRR    OOO  OOO     TTT     III  KKK KKK
  MMM      MMM  III  KKK  KKK  RRR  RRR   OOOOOO      TTT     III  KKK  KKK

  MikroTik RouterOS 6.34.2 (c) 1999-2015       http://www.mikrotik.com/

[?]             Gives the list of available commands
command [?]     Gives help on the command and list of arguments

[Tab]           Completes the command/word. If the input is ambiguous,
                a second [Tab] gives possible options

/               Move up to base level
..              Move up one level
/command        Use command at the base level
[admin@MikroTik] > ip dhcp-server lease
[admin@MikroTik] /ip dhcp-server lease> export
# feb/24/2016 23:24:25 by RouterOS 6.34.2
# software id = 1VWV-4NDH
#
/ip dhcp-server lease
add address=192.168.1.2 mac-address=00:11:22:33:44:01 server=dhcp1
[admin@MikroTik] /ip dhcp-server lease>
```

Prepare CLI rules for “massive” configuration

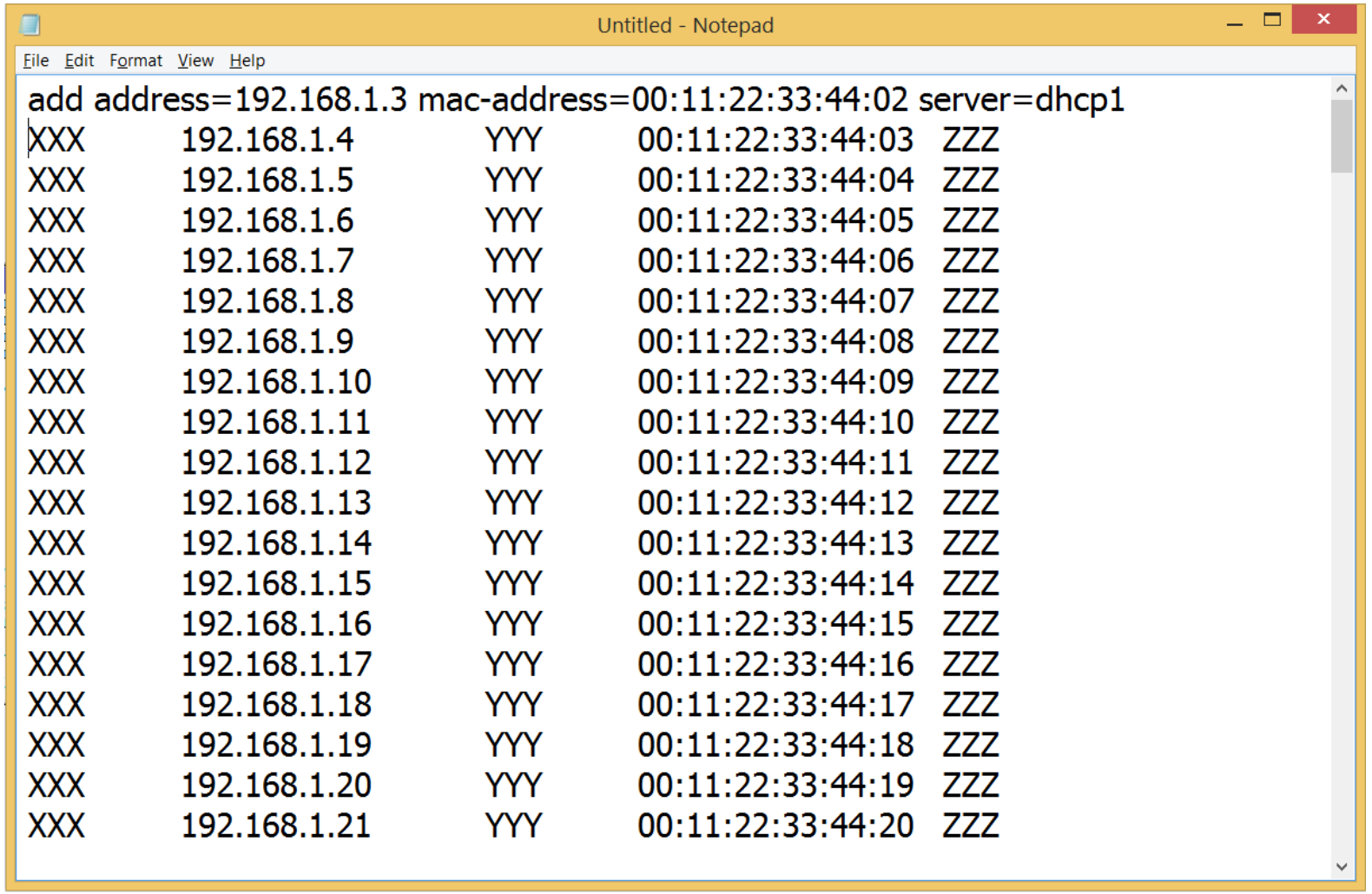
- Create “dummy” columns with unique values

B6 | 192.168.1.7

	A	B	C	D	E	F
37	XXX	192.168.1.38	YYY	00:11:22:33:44:37	ZZZ	
38	XXX	192.168.1.39	YYY	00:11:22:33:44:38	ZZZ	
39	XXX	192.168.1.40	YYY	00:11:22:33:44:39	ZZZ	
40	XXX	192.168.1.41	YYY	00:11:22:33:44:40	ZZZ	
41	XXX	192.168.1.42	YYY	00:11:22:33:44:41	ZZZ	
42	XXX	192.168.1.43	YYY	00:11:22:33:44:42	ZZZ	
43	XXX	192.168.1.44	YYY	00:11:22:33:44:43	ZZZ	
44	XXX	192.168.1.45	YYY	00:11:22:33:44:44	ZZZ	
45	XXX	192.168.1.46	YYY	00:11:22:33:44:45	ZZZ	
46	XXX	192.168.1.47	YYY	00:11:22:33:44:46	ZZZ	
47	XXX	192.168.1.48	YYY	00:11:22:33:44:47	ZZZ	
48	XXX	192.168.1.49	YYY	00:11:22:33:44:48	ZZZ	
49	XXX	192.168.1.50	YYY	00:11:22:33:44:49	ZZZ	
50	XXX	192.168.1.51	YYY	00:11:22:33:44:50	ZZZ	
51	XXX	192.168.1.52	YYY	00:11:22:33:44:51	ZZZ	

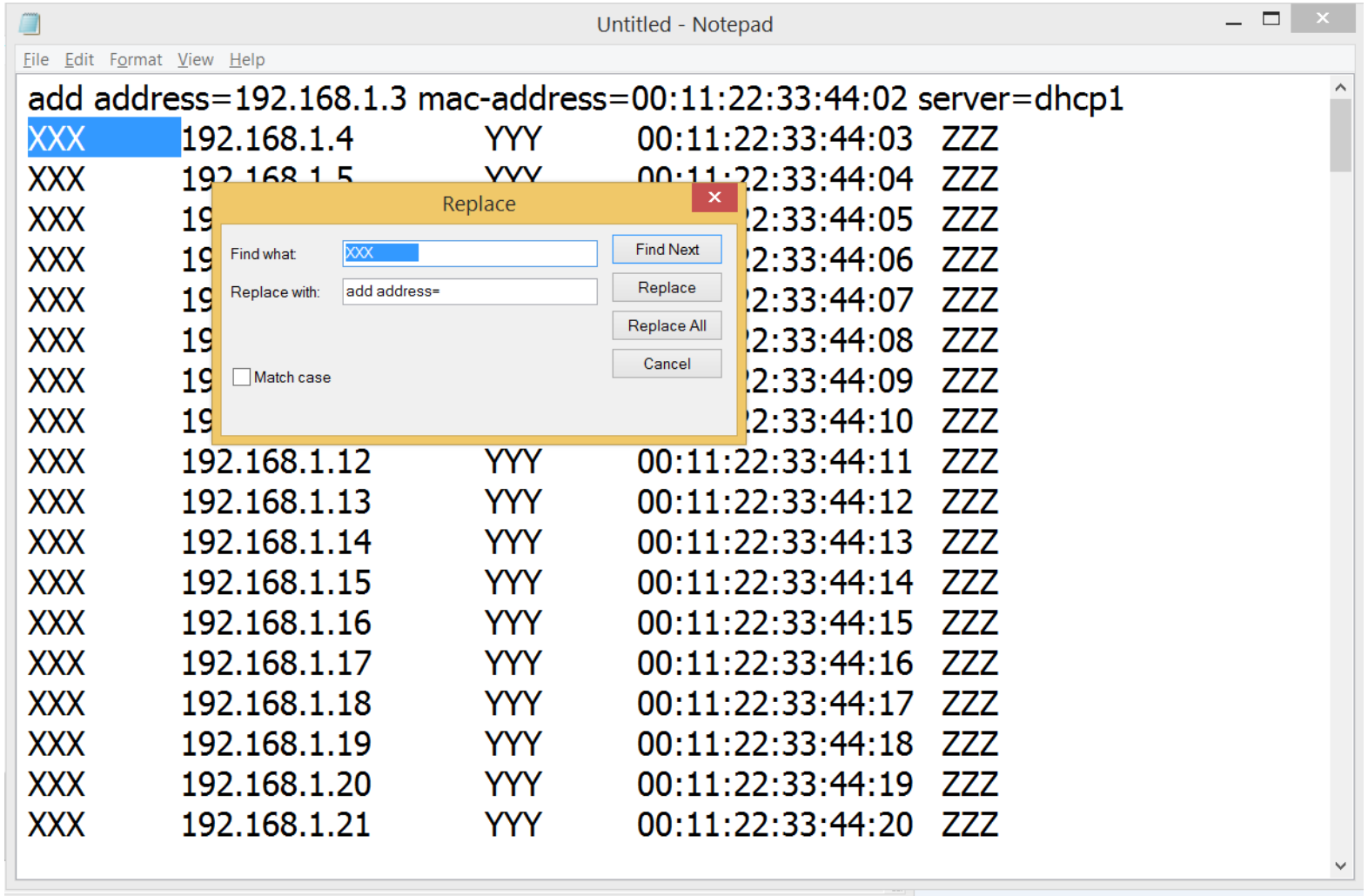
Prepare CLI rules for “massive” configuration

- Copy the spreadsheet to the notepad, prepare the first line



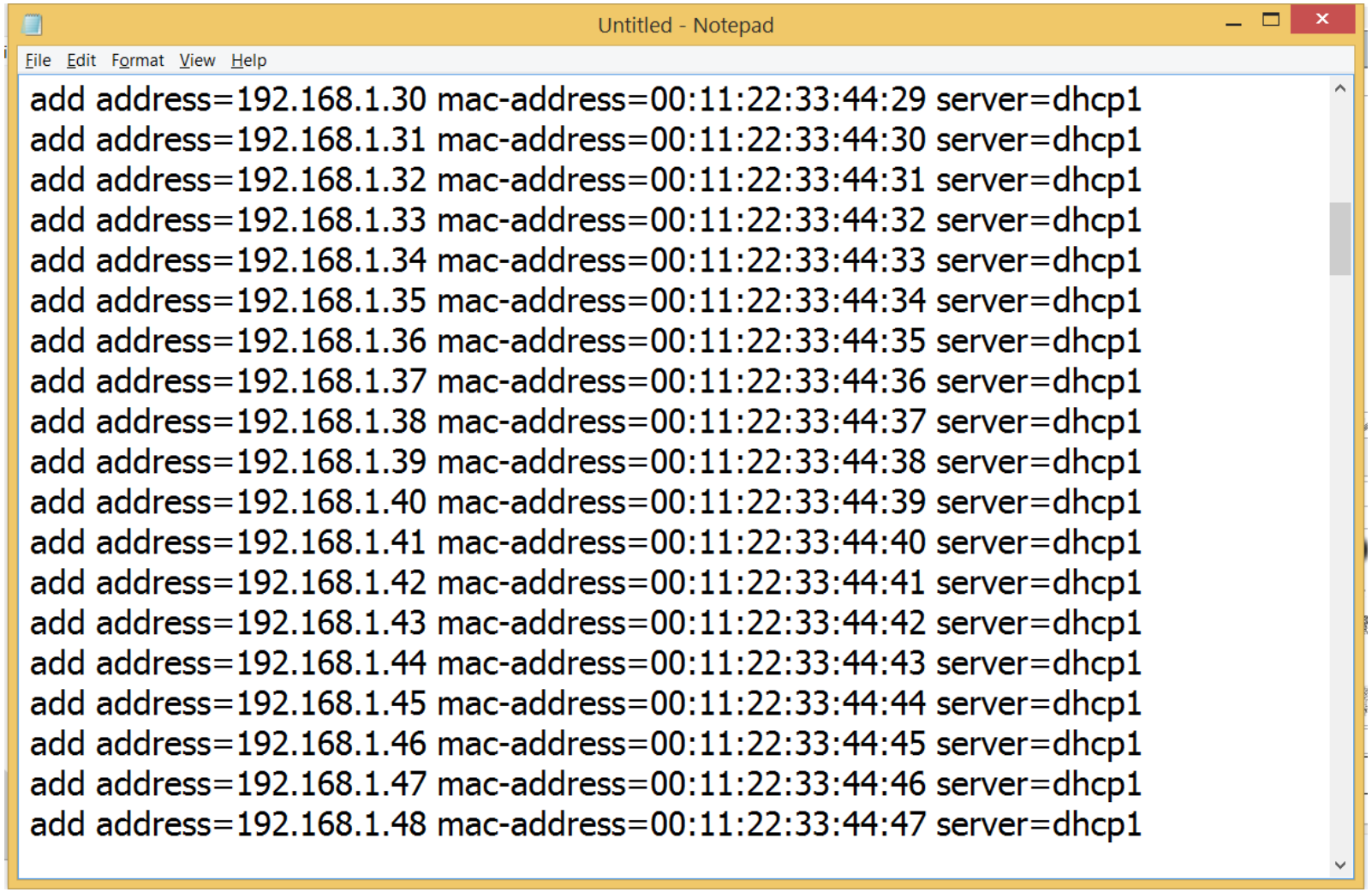
Prepare CLI rules for “massive” configuration

- Use “Replace” function

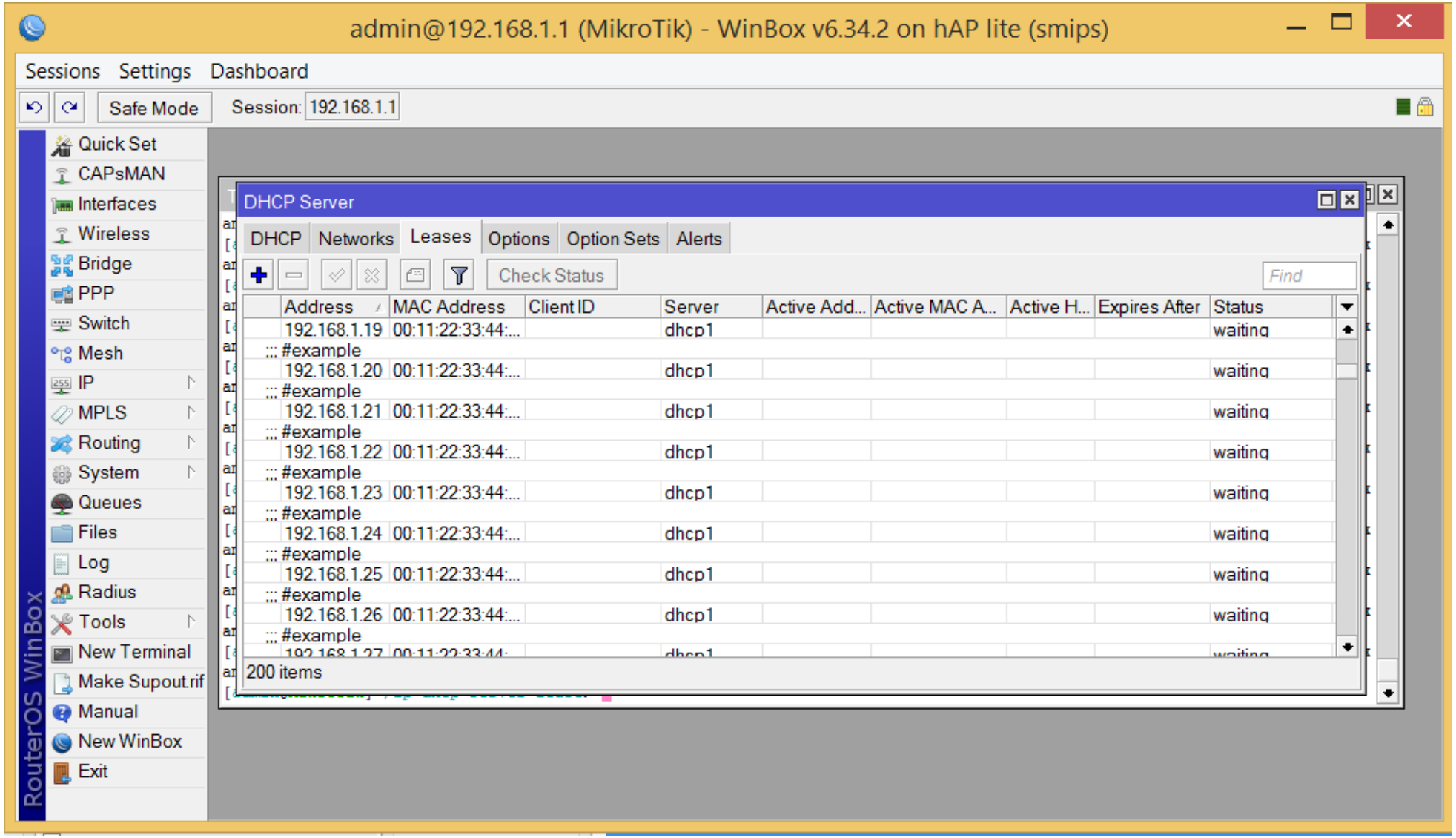


Prepare CLI rules for “massive” configuration

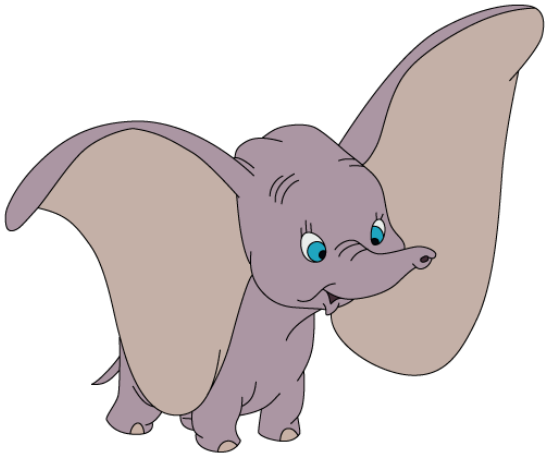
- Voila!



Prepare CLI rules for “massive” configuration



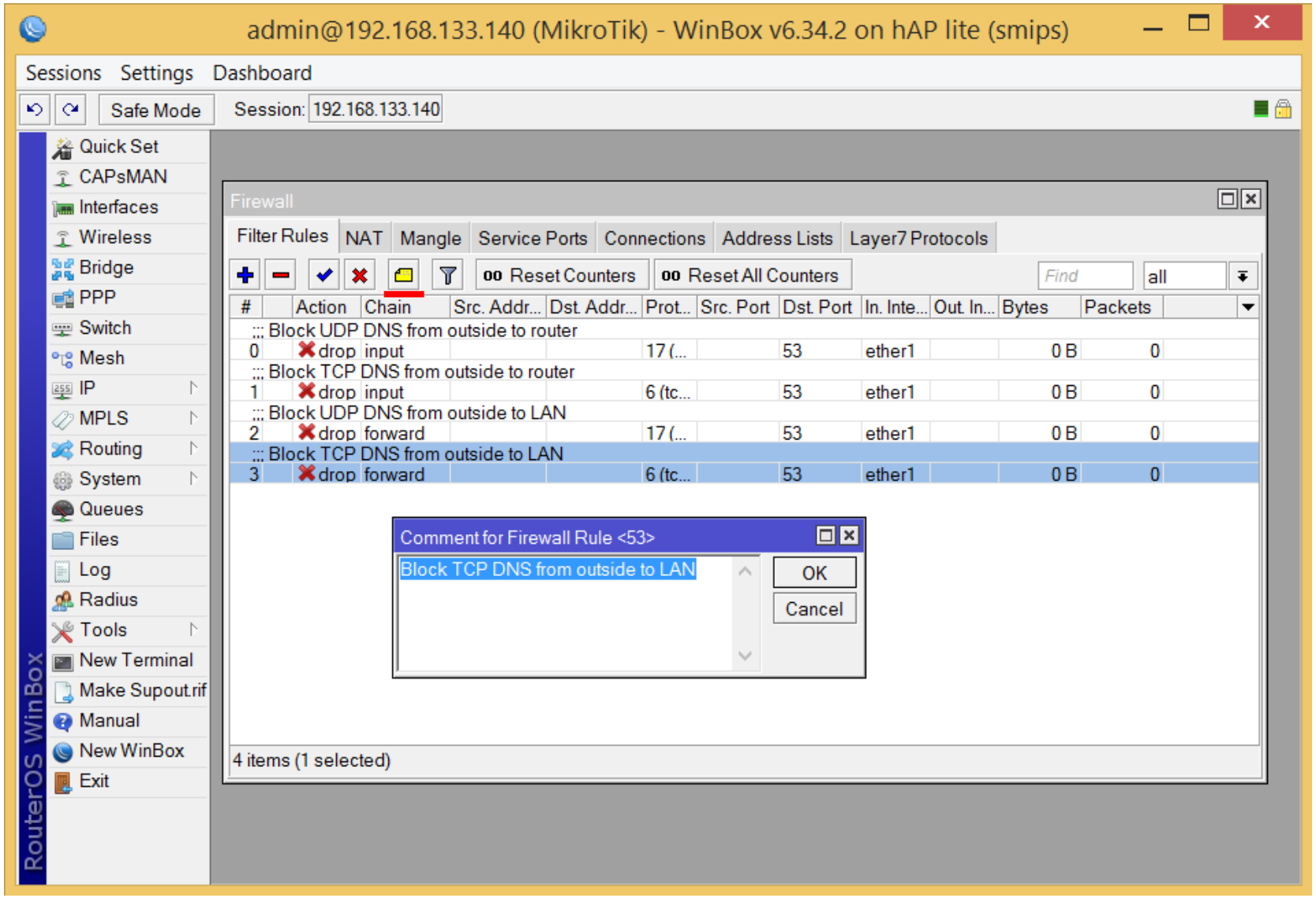
And now for something completely different



Comment your configuration

- Adding comments is very useful!
- Questions after few years – “why the h*ll did I put this rule here?” ☺
- Especially useful in Firewall – where it’s impossible to list all attributes

Comment your configuration



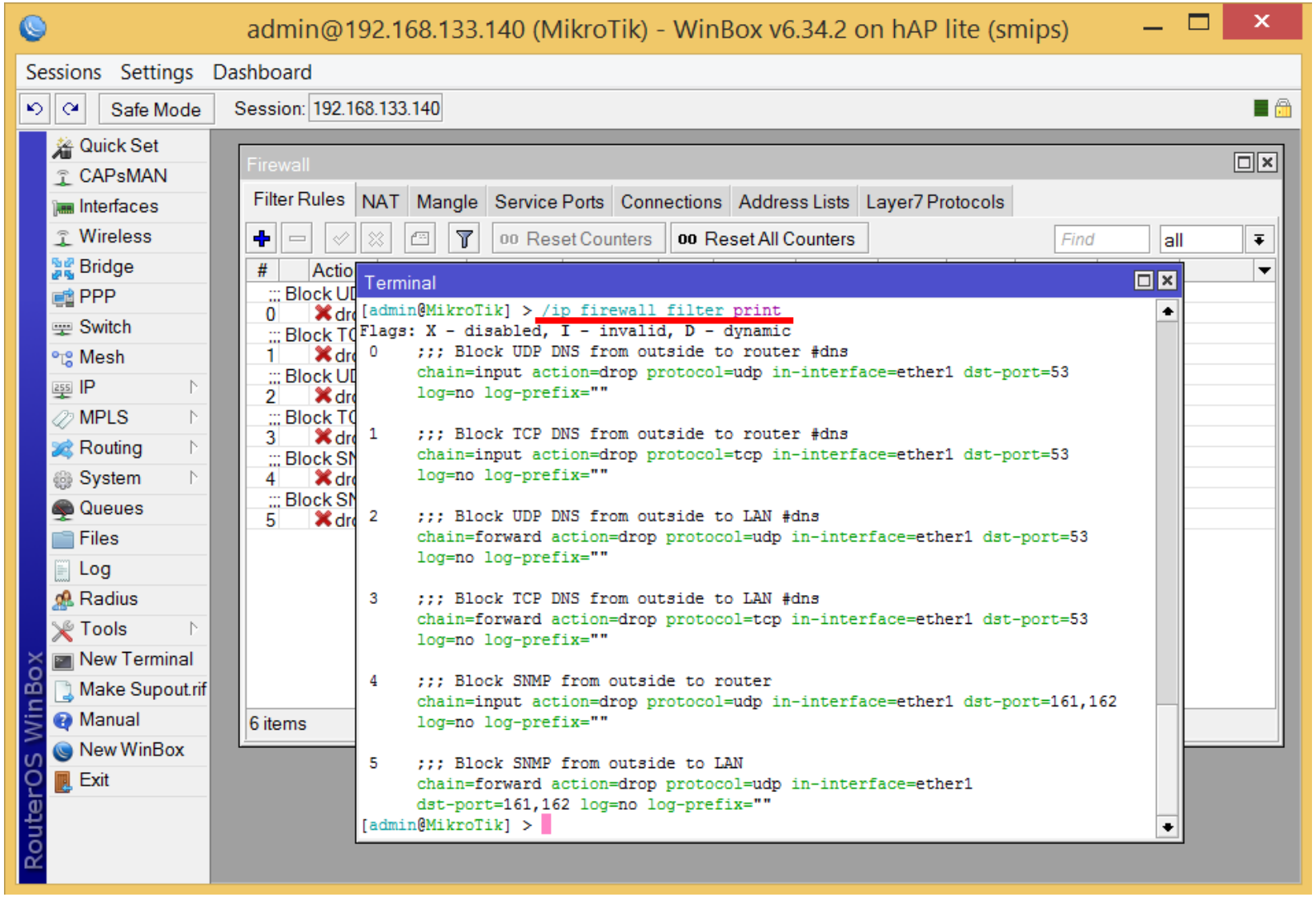
Use #hashtags to group your rules

- Having many firewall rules for specific hosts or services?
- Having many DHCP leases for specific users?
- Having many firewall address-list addresses for some users?

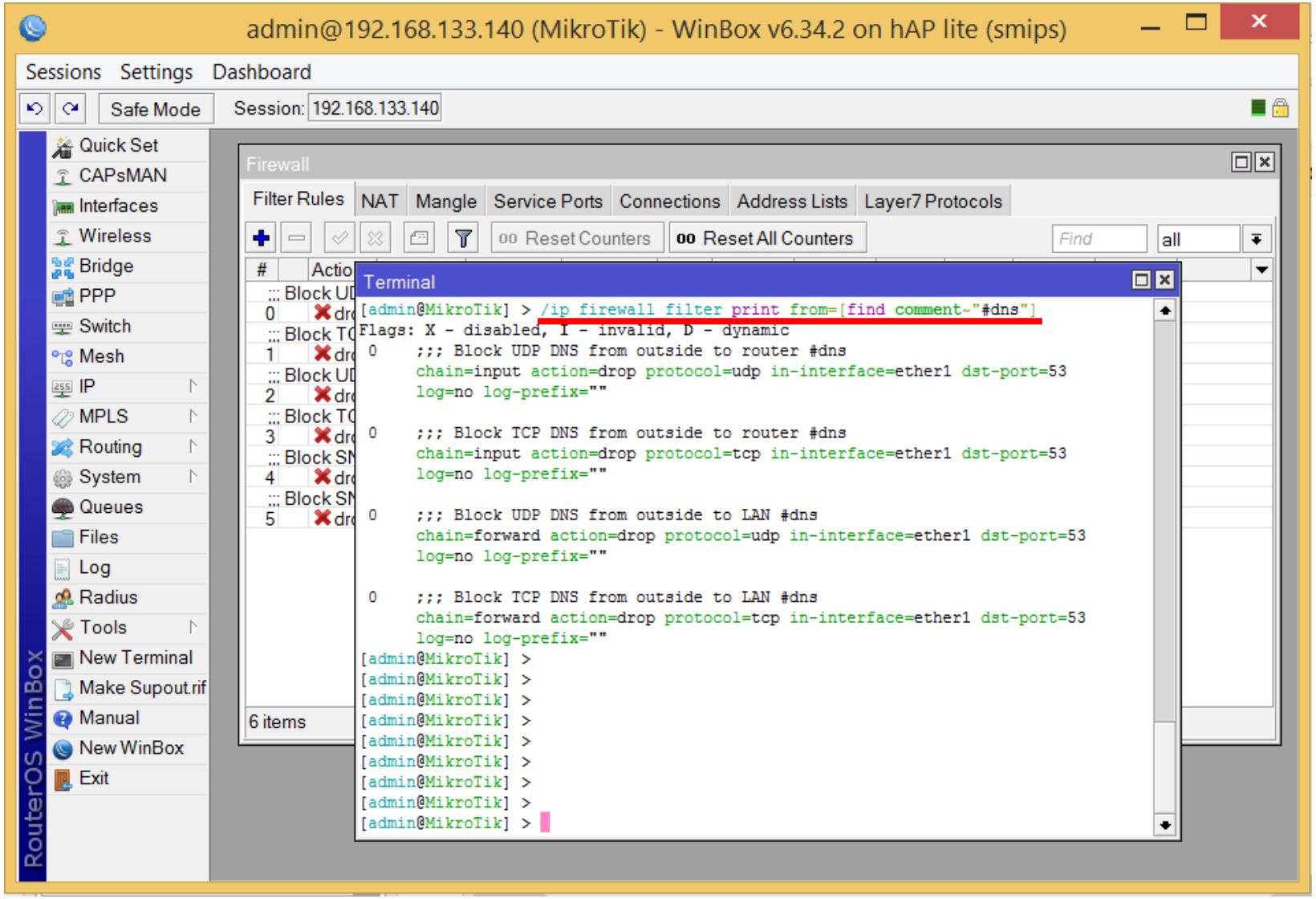
- Add hashtags in comments, like:
- #server1, #server1, #dstarnowski, #jsmith

- Easy to find by **[find comment~"#hashtag"]**

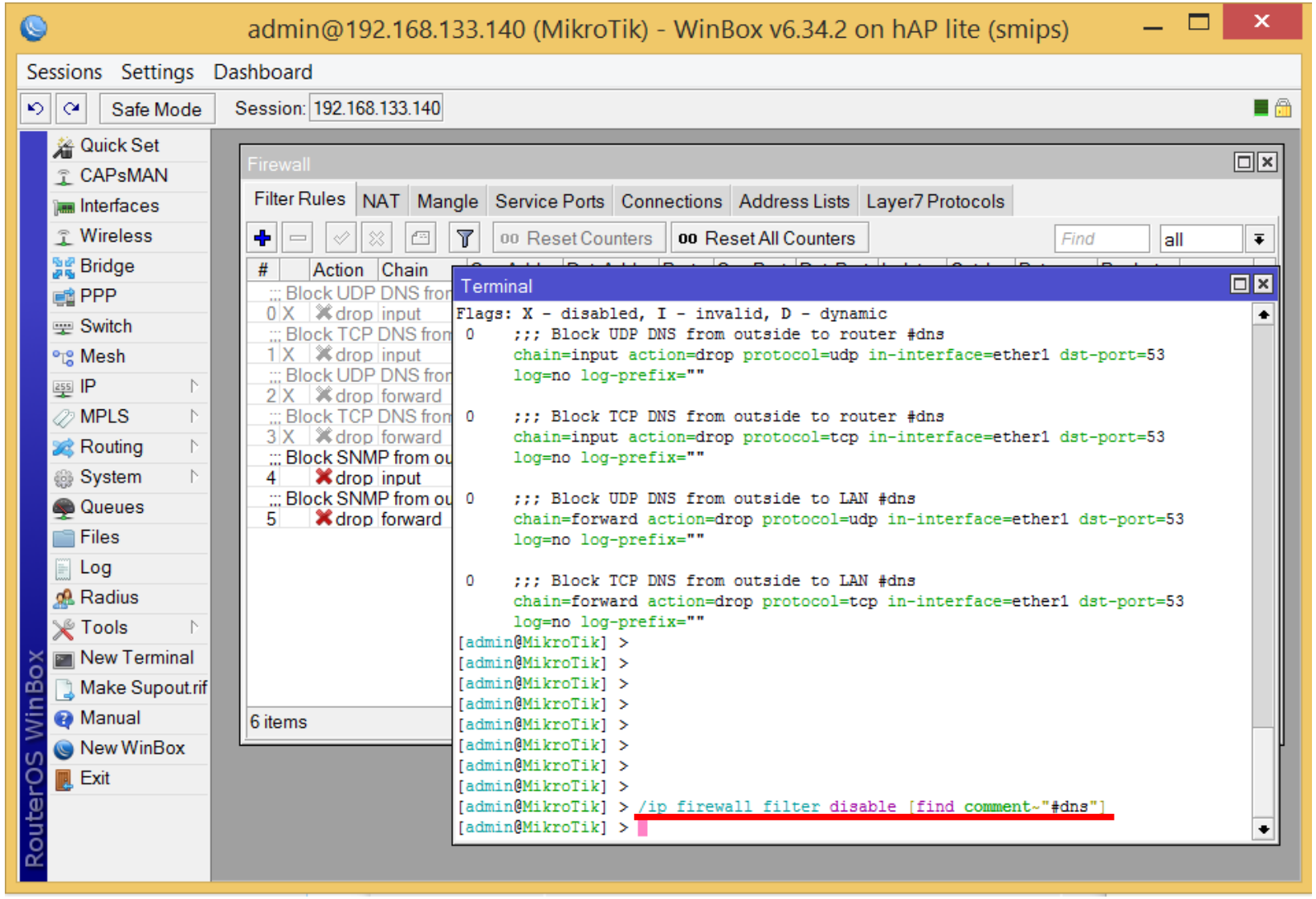
Use #hashtags to group your rules



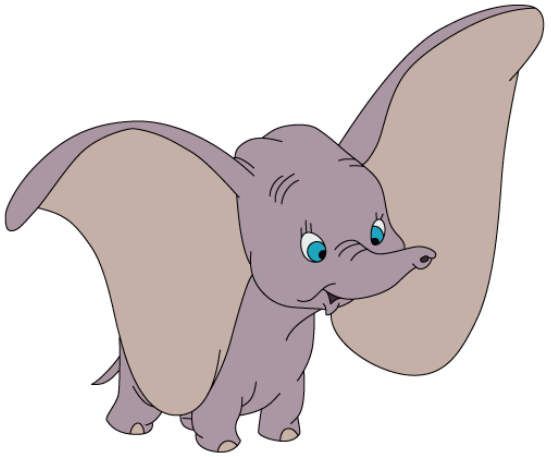
Use #hashtags to group your rules



Use #hashtags to group your rules



And now for something completely different



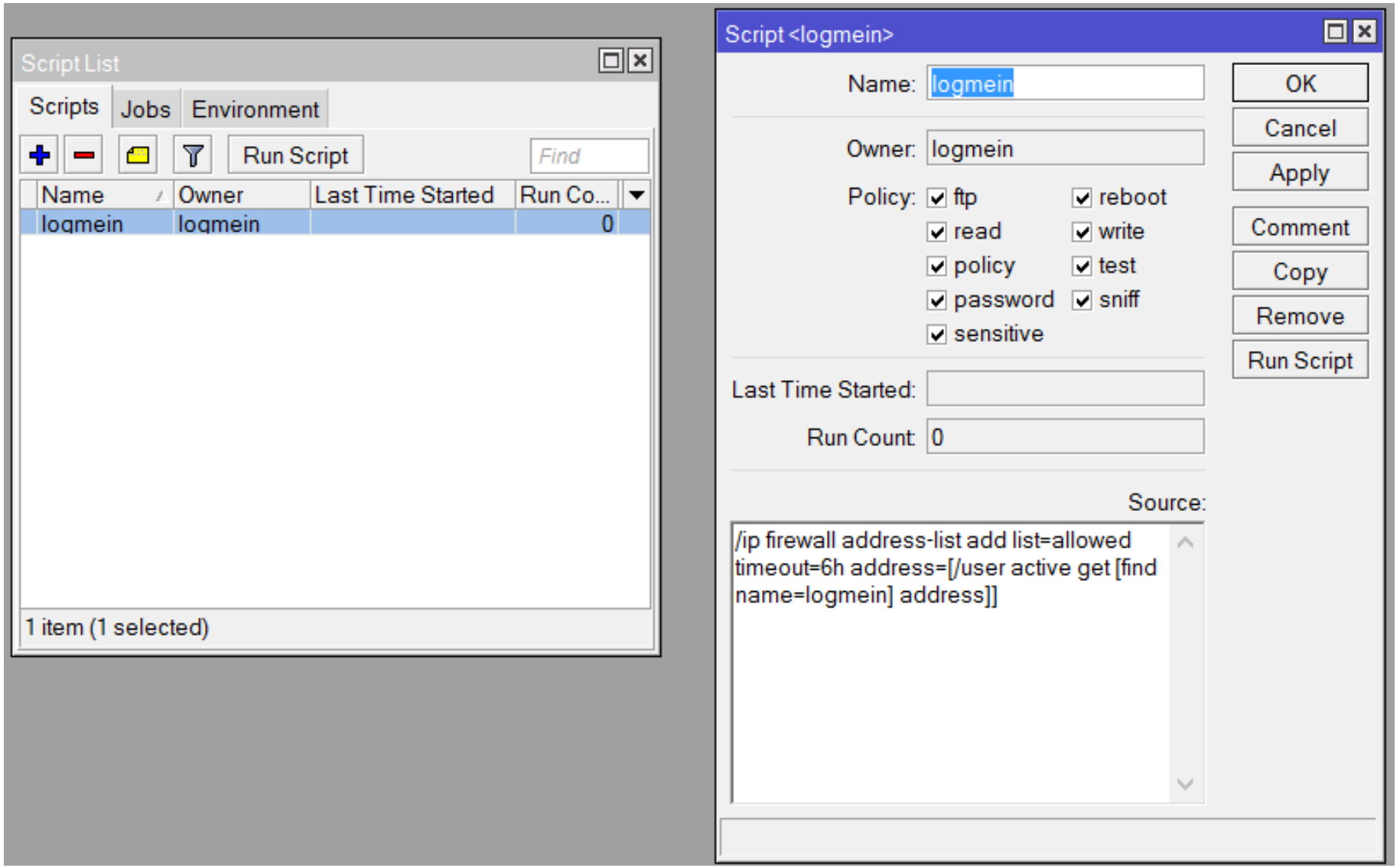
Remember you can write your scripts

- Scripts can be useful to speed things up

- My example – home MikroTik with public IP address
- Only SSH (on a non-standard port) open to the world
- Some servers inside, a network disk with FTP, etc. – with dst-nat rules (port forwarding) only for trusted IP addresses from address list

- One simple script to add my current IP address

Remember you can write your scripts



Remember you can write your scripts

- To access my local servers, I log in via SSH with username **logme**
- I run the script: **/system script run logmein**
- The script does the following:

```
/ip firewall address-list add
list=allowed
timeout=6h
address=[/user active get [find name=logme] address]]
```

- My IP address is added to the list for 6 hours
- Of course – I can also add a “logmeout” script to delete it earlier

SNMP – I want to see everything!

- Only some values can be monitored with SNMP
- All values can be monitored with CLI, WinBox, API…
- But I don’t want my NMS to open NNNN TCP connections to my NNNN MikroTik routers every minute, I want SNMP!
- Let’s take an example – hw_frames in wireless connection

SNMP – I want to see everything!

Terminal

```
[Tab]           Completes the command/word. If the input is ambiguous,
                a second [Tab] gives possible options

/               Move up to base level
..              Move up one level
/command        Use command at the base level
[admin@MikroTik] > interface wireless registration-table
[admin@MikroTik] /interface wireless registration-table> print stats
 0 interface=wlan1 mac-address=48:51:B7:C0:C3:E9 ap=no wds=no bridge=no
   rx-rate="150Mbps-40MHz/1S/SGI" tx-rate="135Mbps-40MHz/1S" packets=49011,32901
   bytes=53740690,6247165 frames=45063,32913 frame-bytes=53906360,6051344 hw-frames=57888,33809
   hw-frame-bytes=70539749,7493037 tx-frames-timed-out=0 uptime=5m52s last-activity=0ms
   signal-strength=-48dBm@HT40-7 signal-to-noise=59dB signal-strength-ch0=-48dBm
   strength-at-rates=-45dBm@1Mbps 4m43s730ms,-50dBm@2Mbps 18m16s400ms,-44dBm@5.5Mbps 20m3s550ms,-
                     42dBm@6Mbps 1s,-47dBm@9Mbps 19m56s920ms,-50dBm@12Mbps 27m42s870ms,-51dBm@18Mbps
                     27m24s700ms,-53dBm@24Mbps 27m12s400ms,-45dBm@36Mbps 20m9s470ms,-44dBm@48Mbps
                     2m15s710ms,-46dBm@54Mbps 5m46s220ms,-58dBm@HT40-1 27m27s20ms,-48dBm@HT40-2
                     19m56s910ms,-47dBm@HT40-3 19m56s940ms,-47dBm@HT40-4 5m44s460ms,-45dBm@HT40-5
                     5s720ms,-46dBm@HT40-6 4s930ms,-48dBm@HT40-7 0ms
   tx-ccq=81% p-throughput=94472 distance=3 last-ip=192.168.1.254 802.1x-port-enabled=yes
   authentication-type=wpa2-psk encryption=aes-ccm group-encryption=aes-ccm
   management-protection=no wmm-enabled=yes
   tx-rate-set="CCK:1-11 OFDM:6-54 BW:1x-2x SGI:1x-2x HT:0-7"
[admin@MikroTik] /interface wireless registration-table>
```

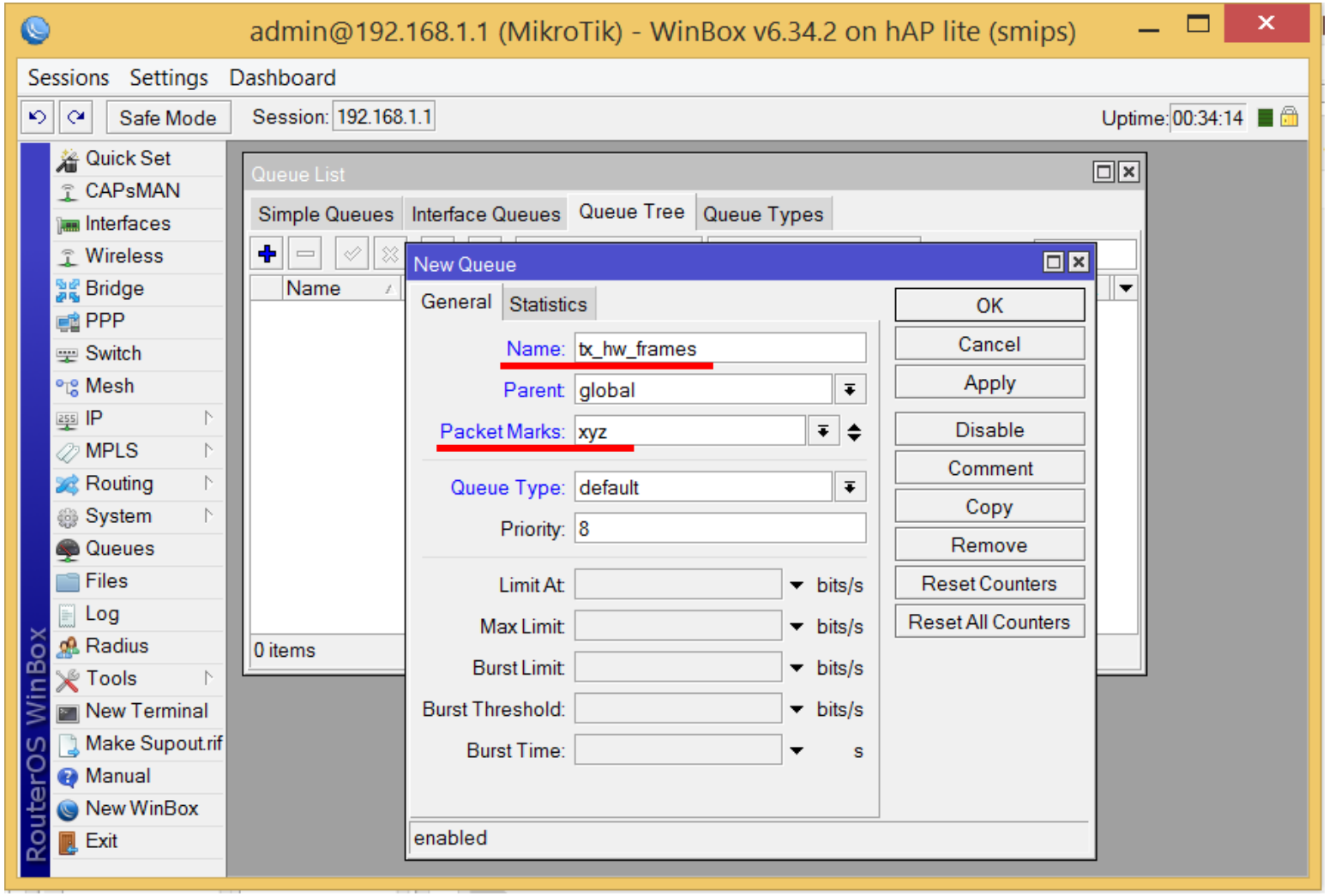
SNMP – I want to see everything!

Terminal

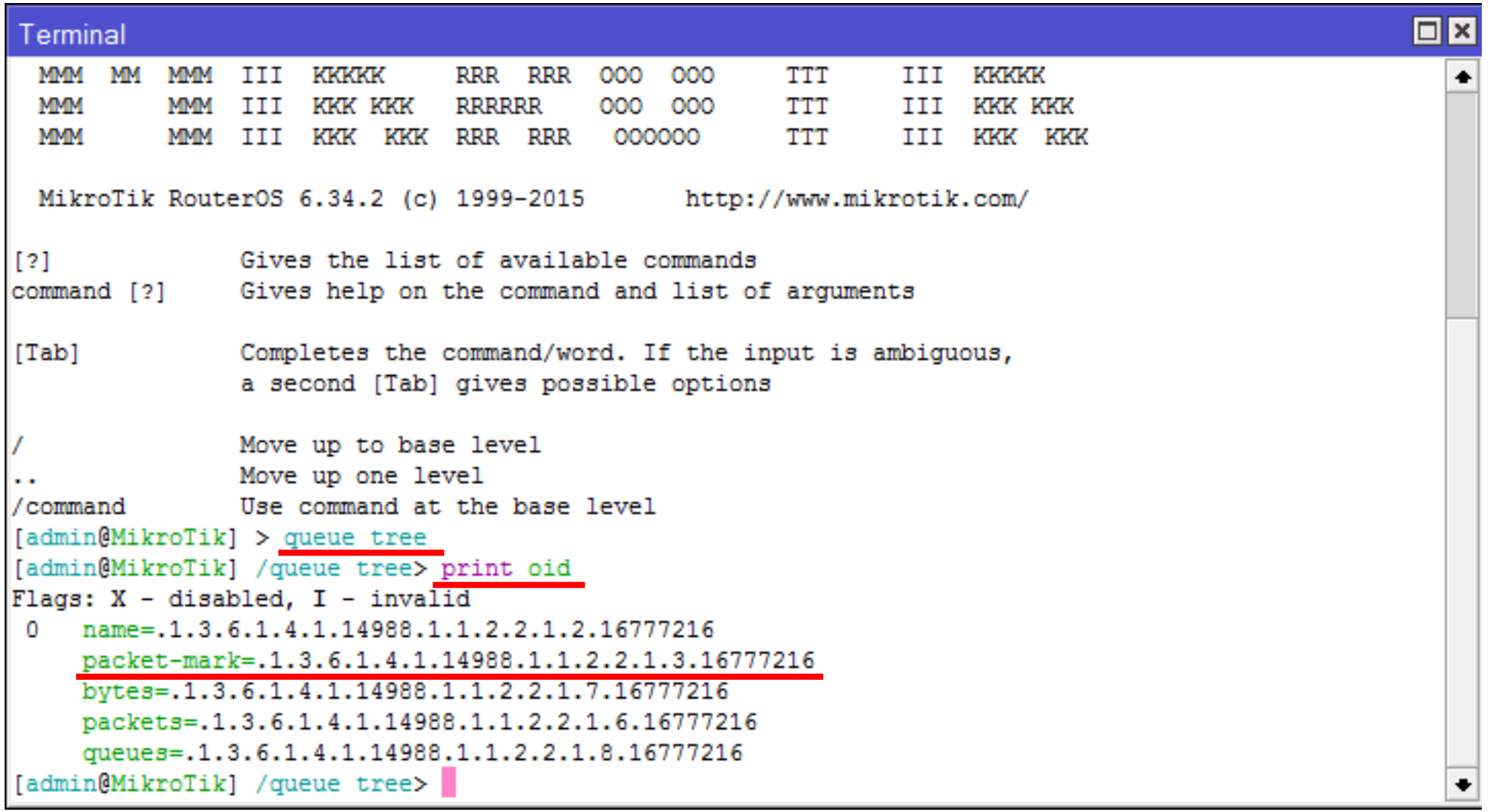
```
/               Move up to base level
..              Move up one level
/command        Use command at the base level
[admin@MikroTik] > interface wireless registration-table
[admin@MikroTik] /interface wireless registration-table> print oid
 0 signal-strength=.1.3.6.1.4.1.14988.1.1.1.2.1.3.72.81.183.192.195.233.5
   tx-signal-strength=.1.3.6.1.4.1.14988.1.1.1.2.1.19.72.81.183.192.195.233.5
   tx-bytes=.1.3.6.1.4.1.14988.1.1.1.2.1.4.72.81.183.192.195.233.5
   rx-bytes=.1.3.6.1.4.1.14988.1.1.1.2.1.5.72.81.183.192.195.233.5
   tx-packets=.1.3.6.1.4.1.14988.1.1.1.2.1.6.72.81.183.192.195.233.5
   rx-packets=.1.3.6.1.4.1.14988.1.1.1.2.1.7.72.81.183.192.195.233.5
   tx-rate=.1.3.6.1.4.1.14988.1.1.1.2.1.8.72.81.183.192.195.233.5
   rx-rate=.1.3.6.1.4.1.14988.1.1.1.2.1.9.72.81.183.192.195.233.5
   routeros-version=.1.3.6.1.4.1.14988.1.1.1.2.1.10.72.81.183.192.195.233.5
   uptime=.1.3.6.1.4.1.14988.1.1.1.2.1.11.72.81.183.192.195.233.5
   signal-to-noise=.1.3.6.1.4.1.14988.1.1.1.2.1.12.72.81.183.192.195.233.5
   tx-signal-strength-ch0=.1.3.6.1.4.1.14988.1.1.1.2.1.13.72.81.183.192.195.233.5
   signal-strength-ch0=.1.3.6.1.4.1.14988.1.1.1.2.1.14.72.81.183.192.195.233.5
   tx-signal-strength-ch1=.1.3.6.1.4.1.14988.1.1.1.2.1.15.72.81.183.192.195.233.5
   signal-strength-ch1=.1.3.6.1.4.1.14988.1.1.1.2.1.16.72.81.183.192.195.233.5
   tx-signal-strength-ch2=.1.3.6.1.4.1.14988.1.1.1.2.1.17.72.81.183.192.195.233.5
   signal-strength-ch2=.1.3.6.1.4.1.14988.1.1.1.2.1.18.72.81.183.192.195.233.5
[admin@MikroTik] /interface wireless registration-table>
```

SNMP – I want to see everything!

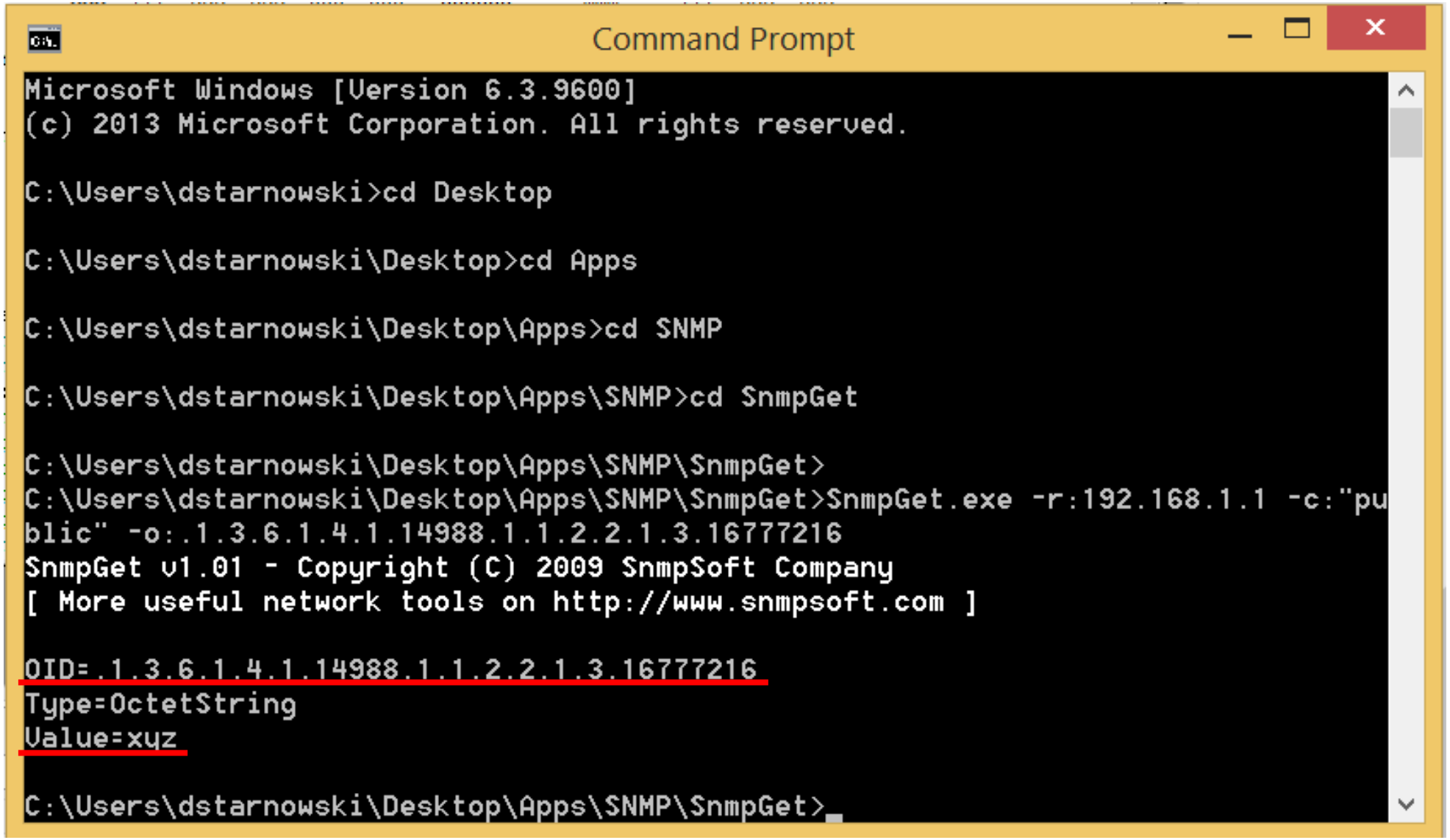
- Use **/queue tree** and some “dummy” entries!



SNMP – I want to see everything!

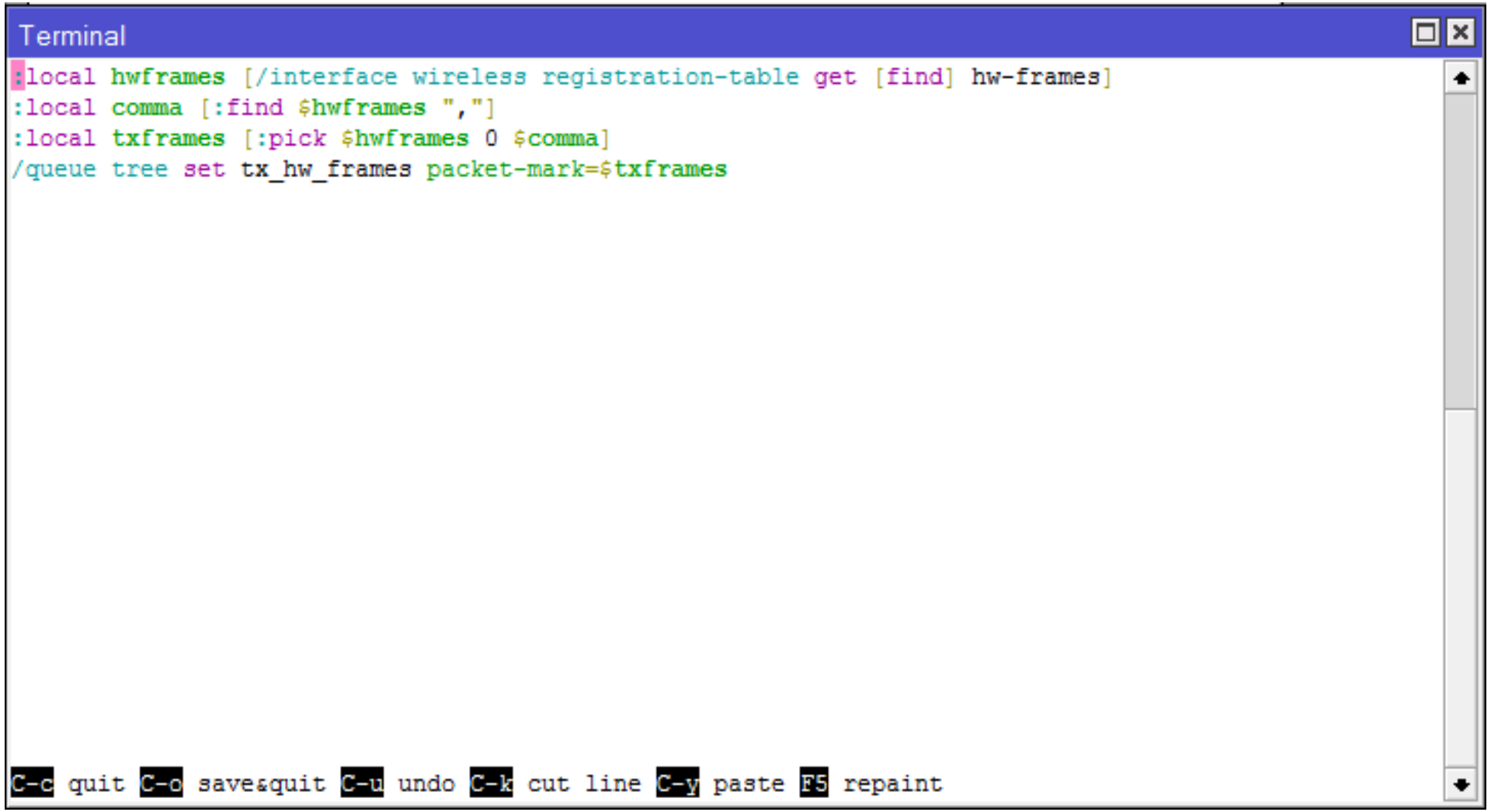


SNMP – I want to see everything!

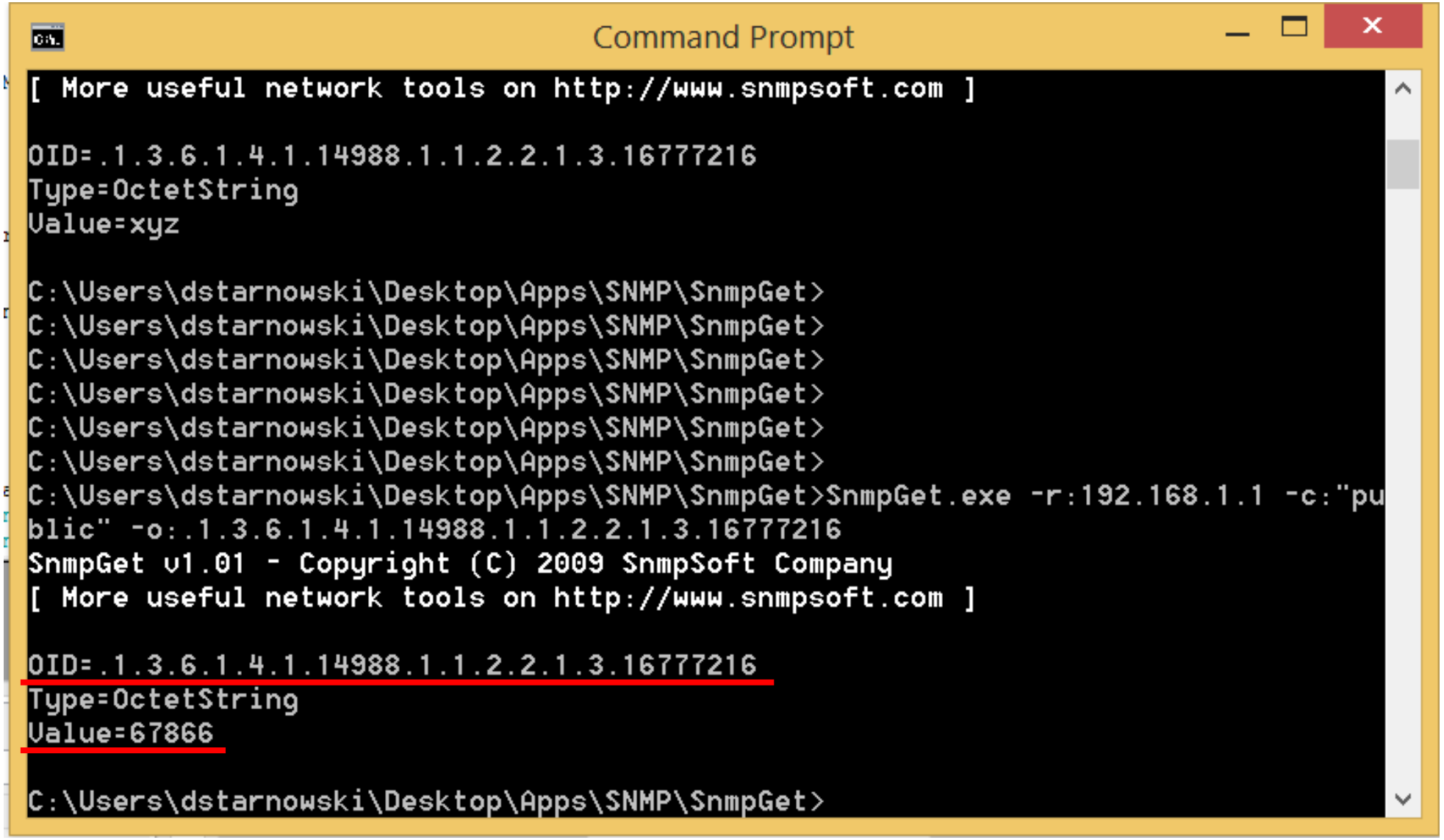


SNMP – I want to see everything!

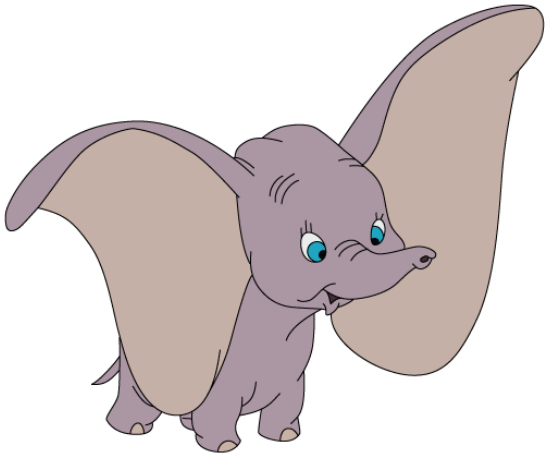
- Now let's create a script to put hw_frames value to packet mark ☺



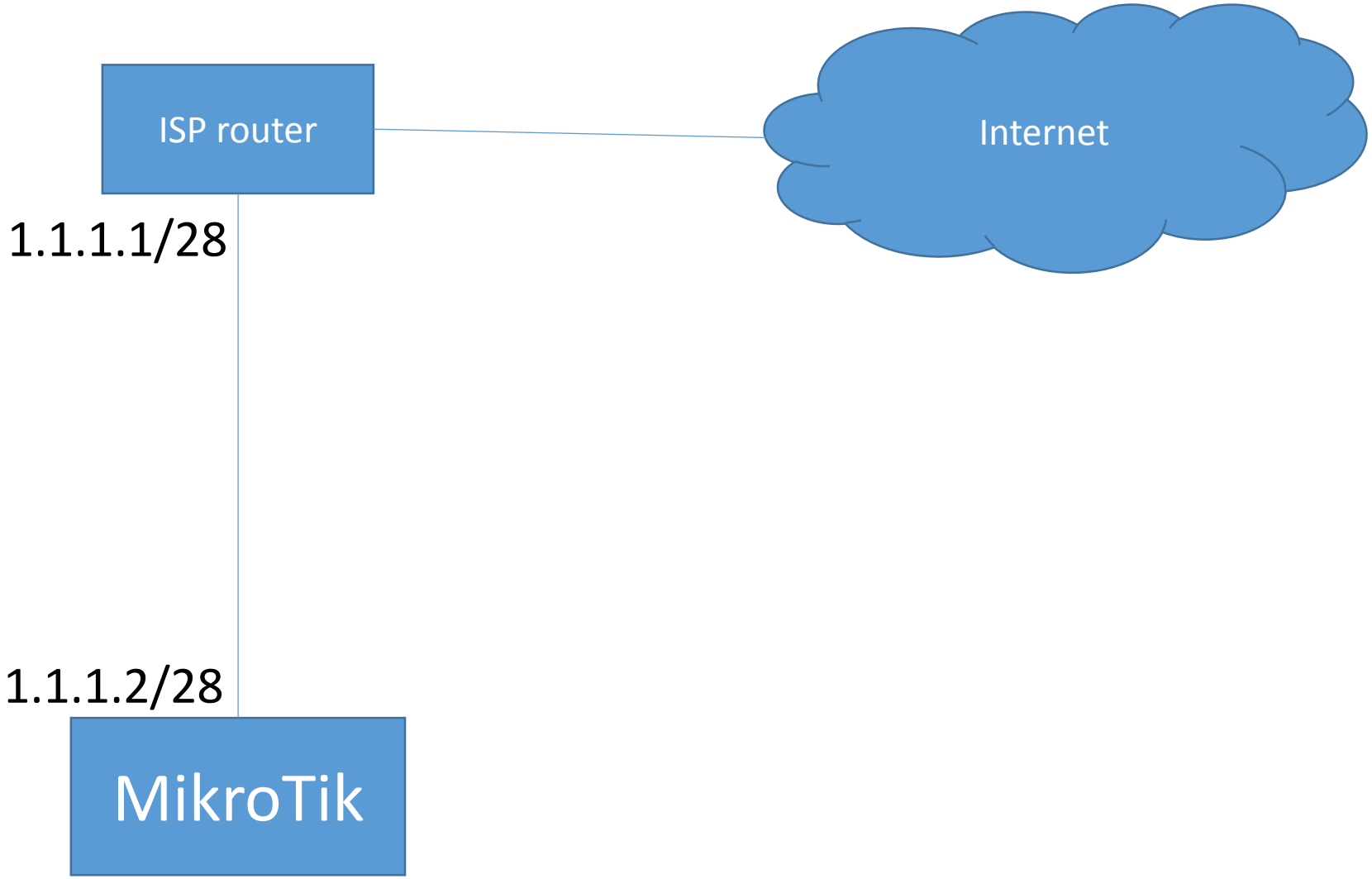
SNMP – I want to see everything!



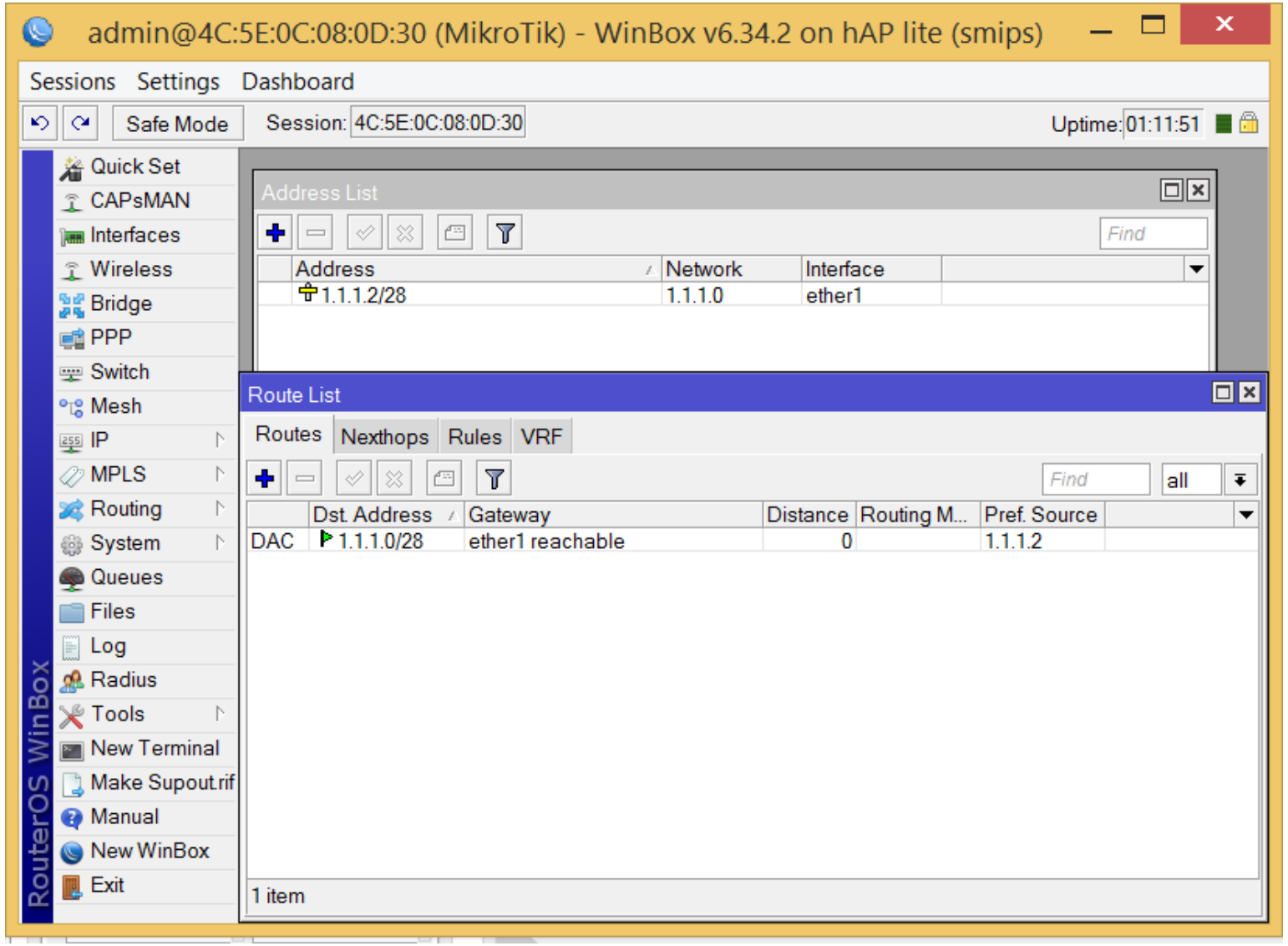
And now for something completely different



IP and ARP – short introduction



IP and ARP – short introduction

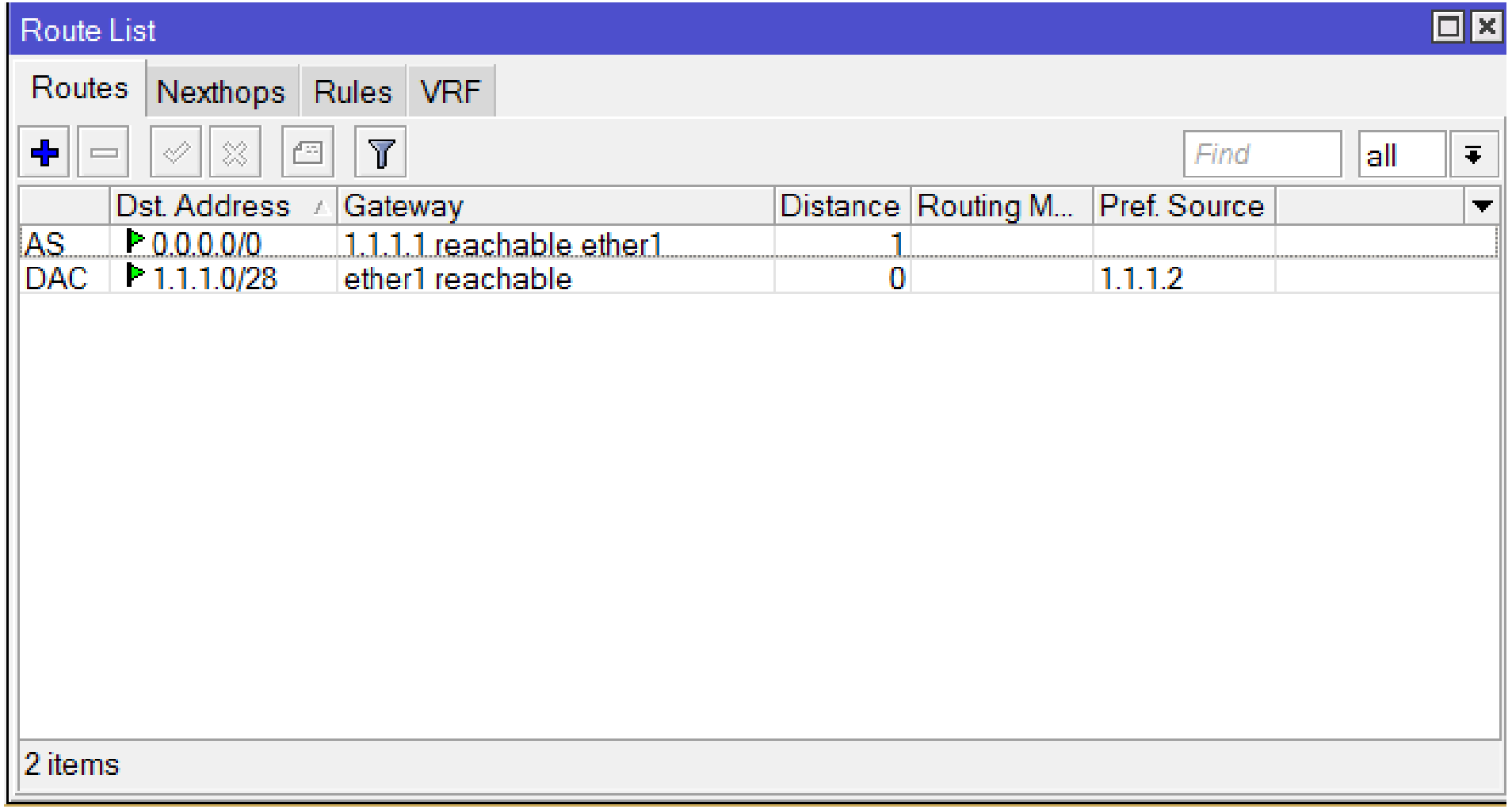


IP and ARP – short introduction

- We configure IP address 1.1.1.2/28 on ether1 interface

- This means three things:
- 1. The address 1.1.1.2 is one of our router's IP addresses
- 2. The network on ether1 is 1.1.1.0/28
- 3. Packets going to this network, originated on our router, will have source IP 1.1.1.2

IP and ARP – short introduction



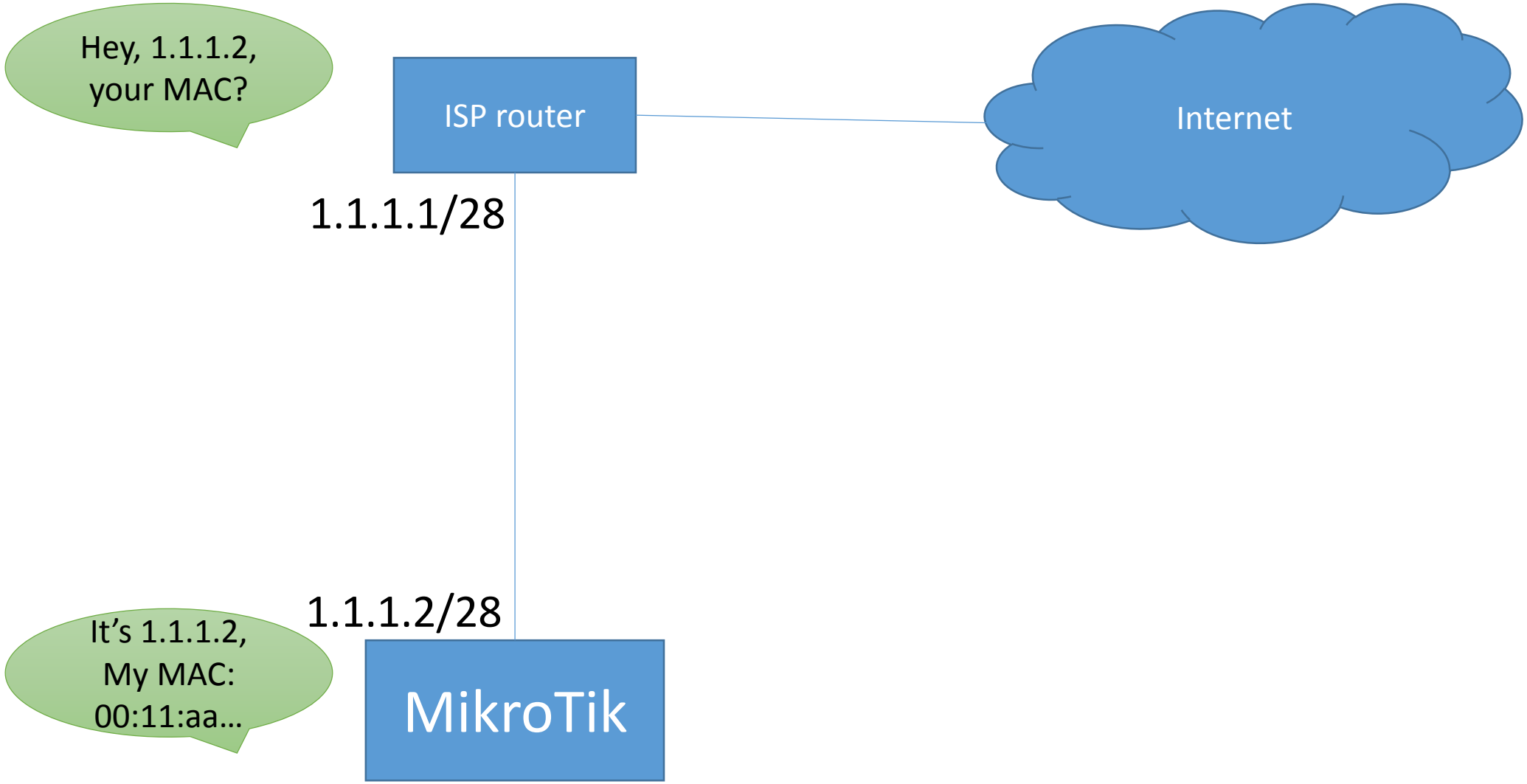
IP and ARP – short introduction

- We add 1.1.1.1 as our default gateway (static route to 0.0.0.0/0)
- We want to send a packet to 8.8.8.8
- MikroTik checks route to 8.8.8.8 – it fits 0.0.0.0/0, so it's via 1.1.1.1
- MikroTik checks route to 1.1.1.1 – it fits 1.1.1.0/28, so it's on ether1
- MikroTik checks 1.1.1.1's MAC address with ARP request
- MikroTik sends the packet to:
  - Destination IP: 8.8.8.8
  - Destination MAC: MAC of 1.1.1.1

IP and ARP – short introduction

- The provider's router (1.1.1.1) does the same.
- When a packet comes to 1.1.1.2, it looks for it in routing table
- It's directly connected (1.1.1.0/28) on its ethernet interface
- It asks for MAC address of 1.1.1.2 using ARP request
- It sends packet to 1.1.1.2 using the MAC address it got

IP and ARP – short introduction

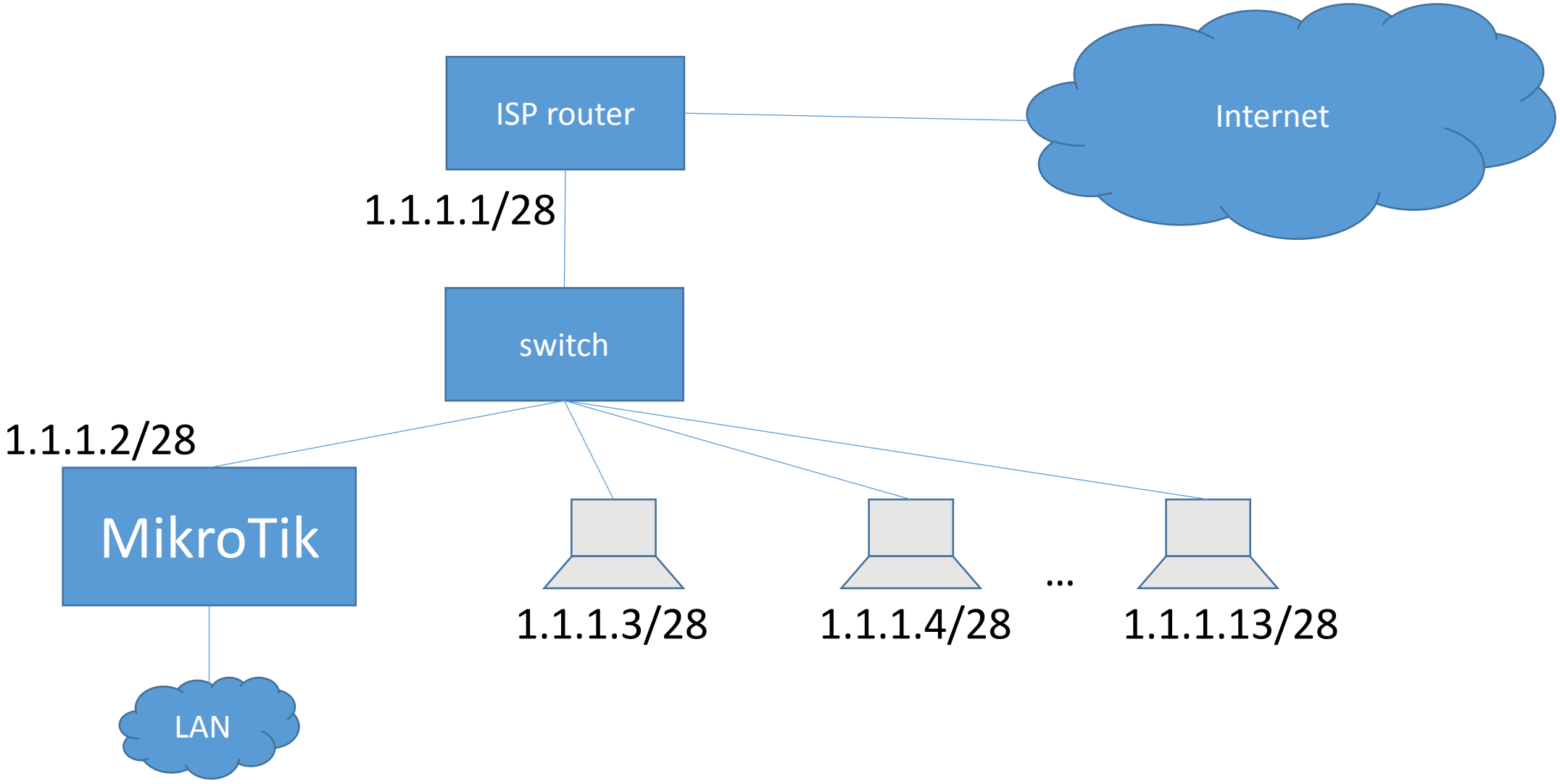


Problem 1 – IP pool from provider

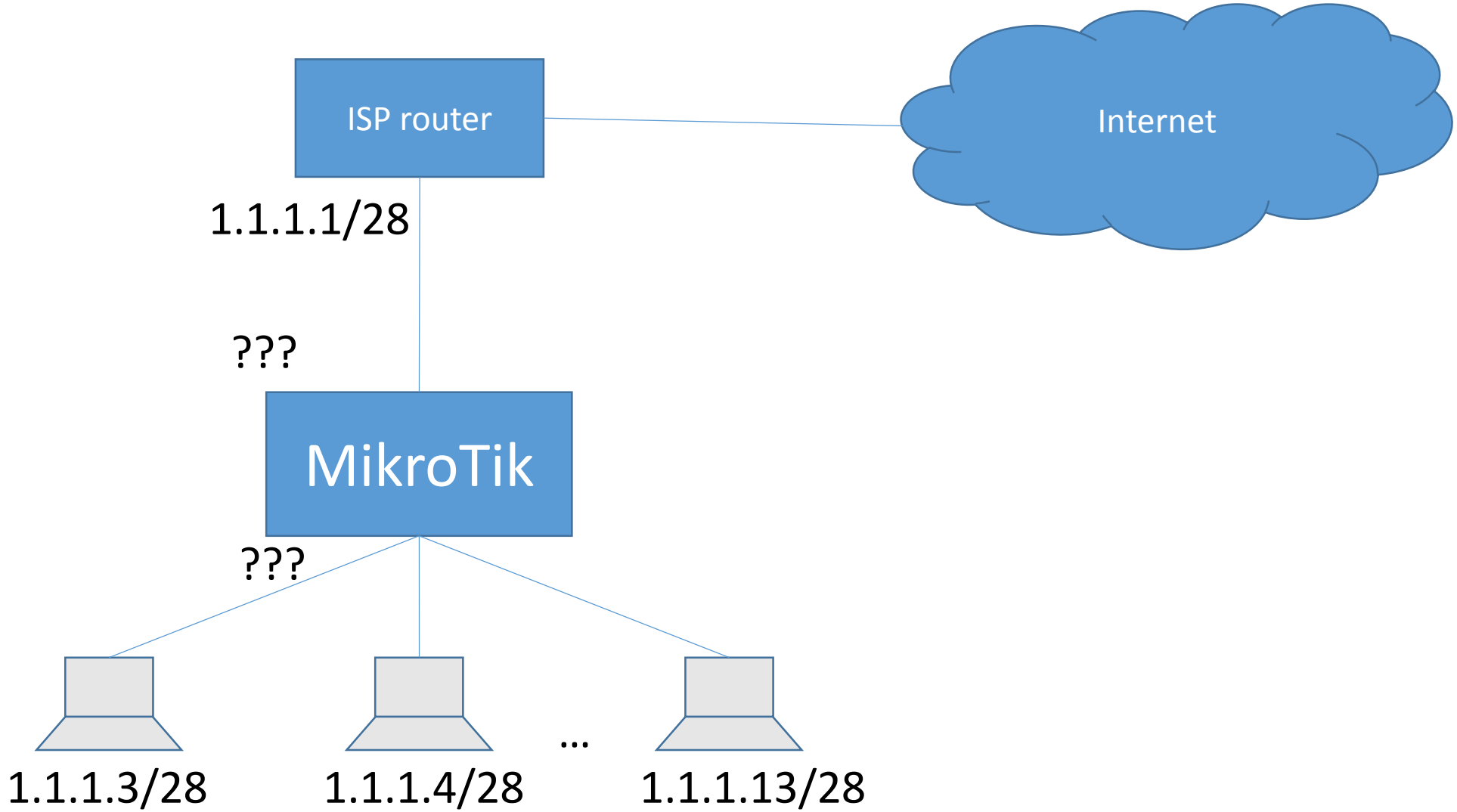
- ISP gives us an IP pool 1.1.1.0/28, with 1.1.1.1 being provider's router
- We can use any IP between .2 and .14
- It gives us 13 "usable" IP addresses

- Only one problem – the IP addresses are in one LAN

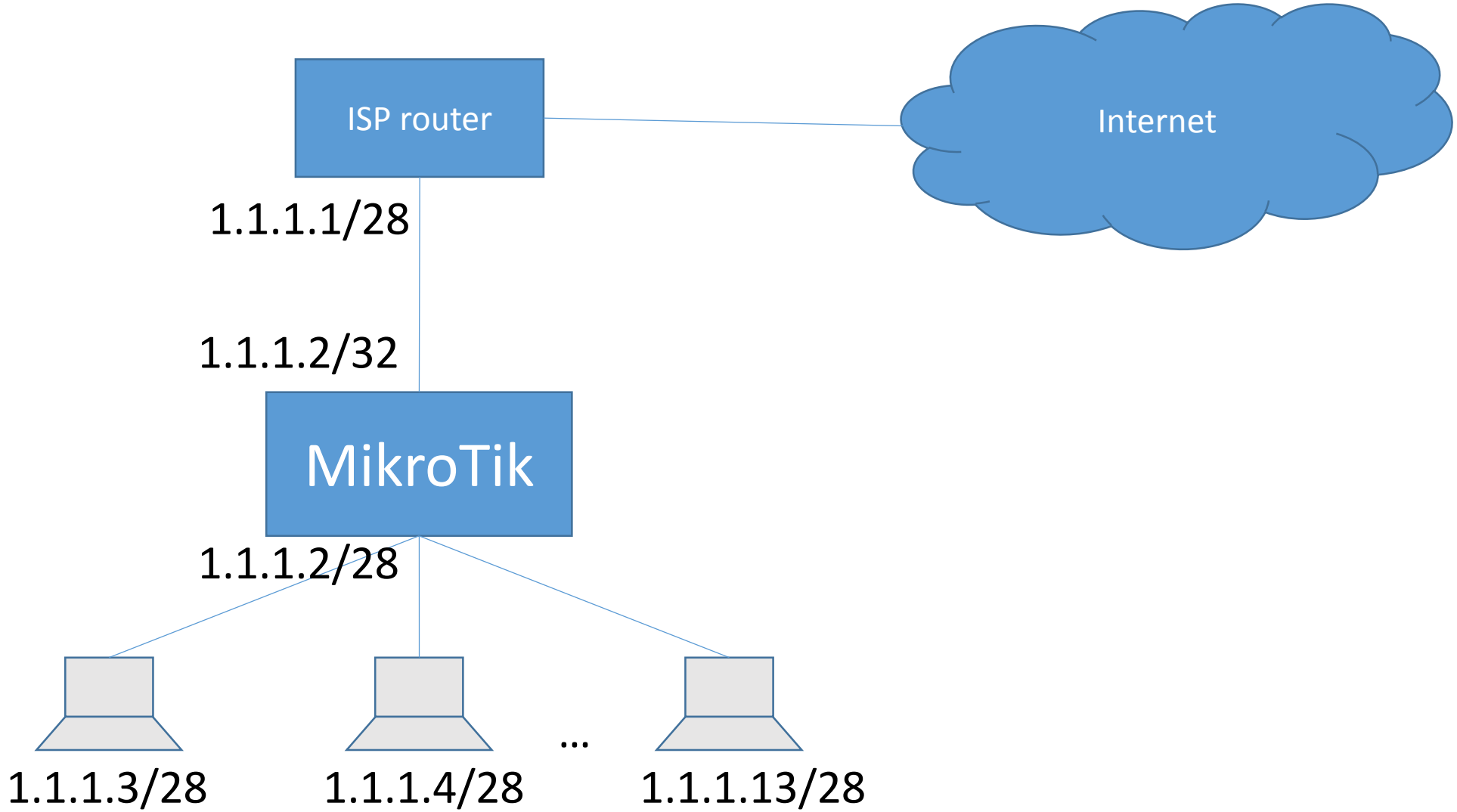
Problem 1 – IP pool from provider



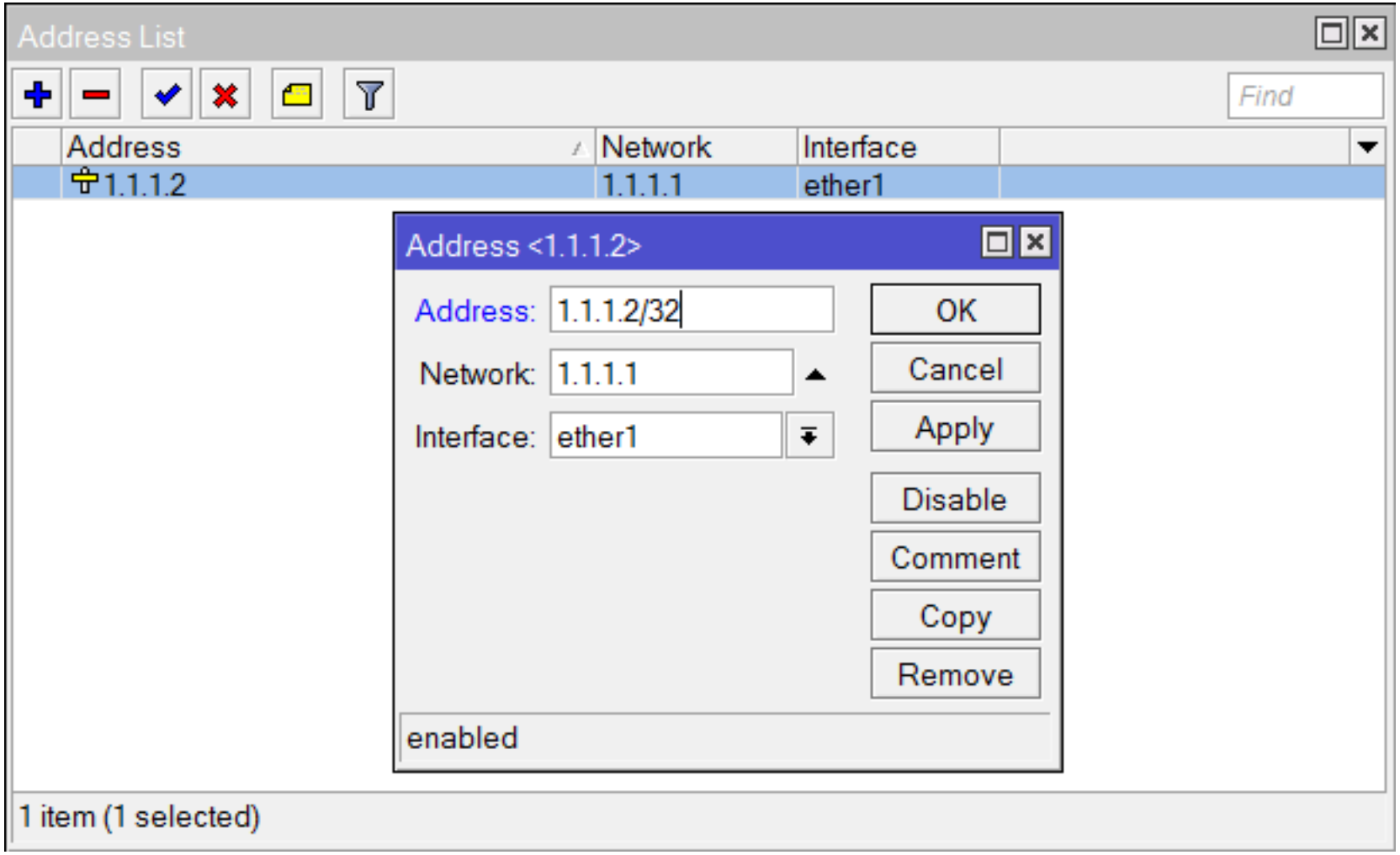
Problem 1 – IP pool from provider



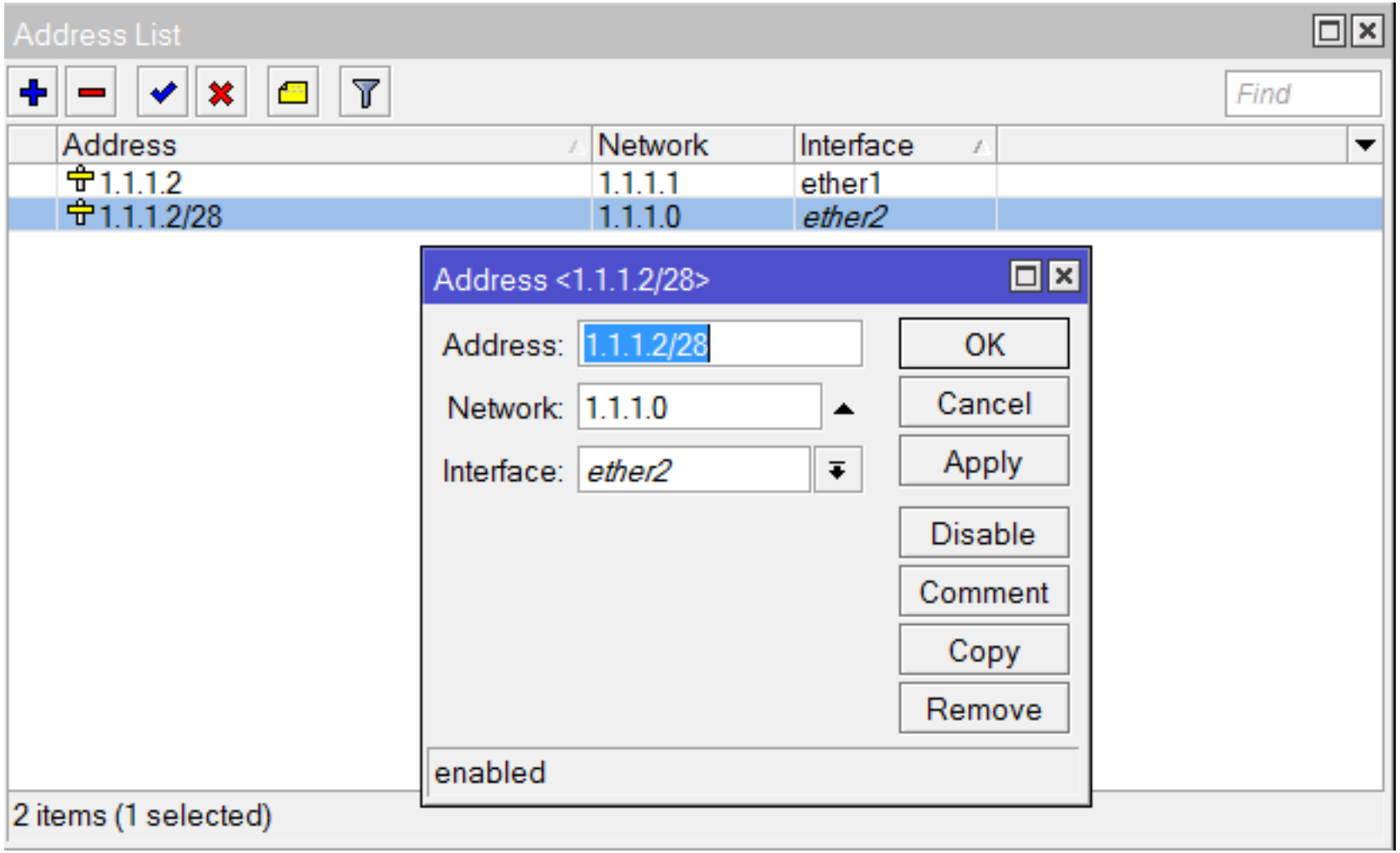
Problem 1 – IP pool from provider



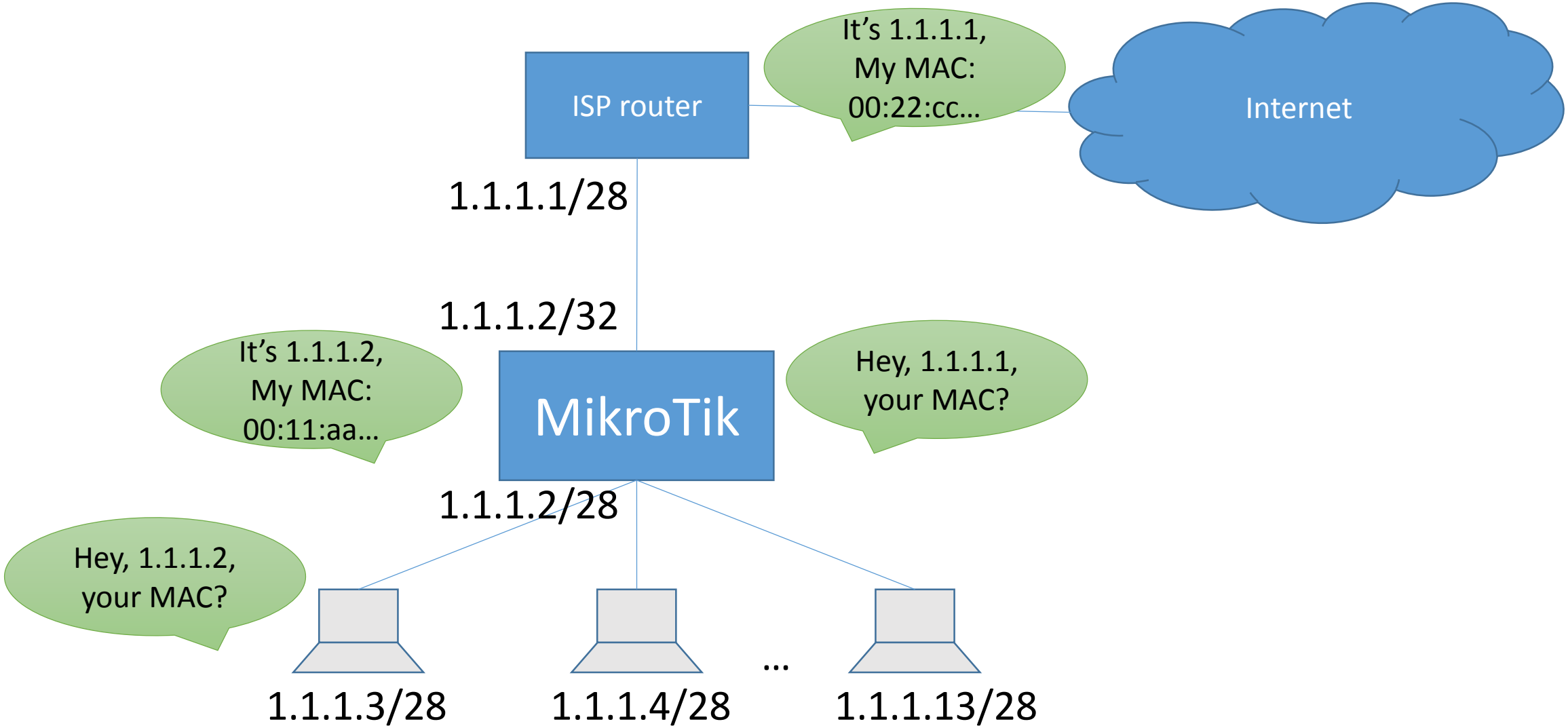
Problem 1 – IP pool from provider



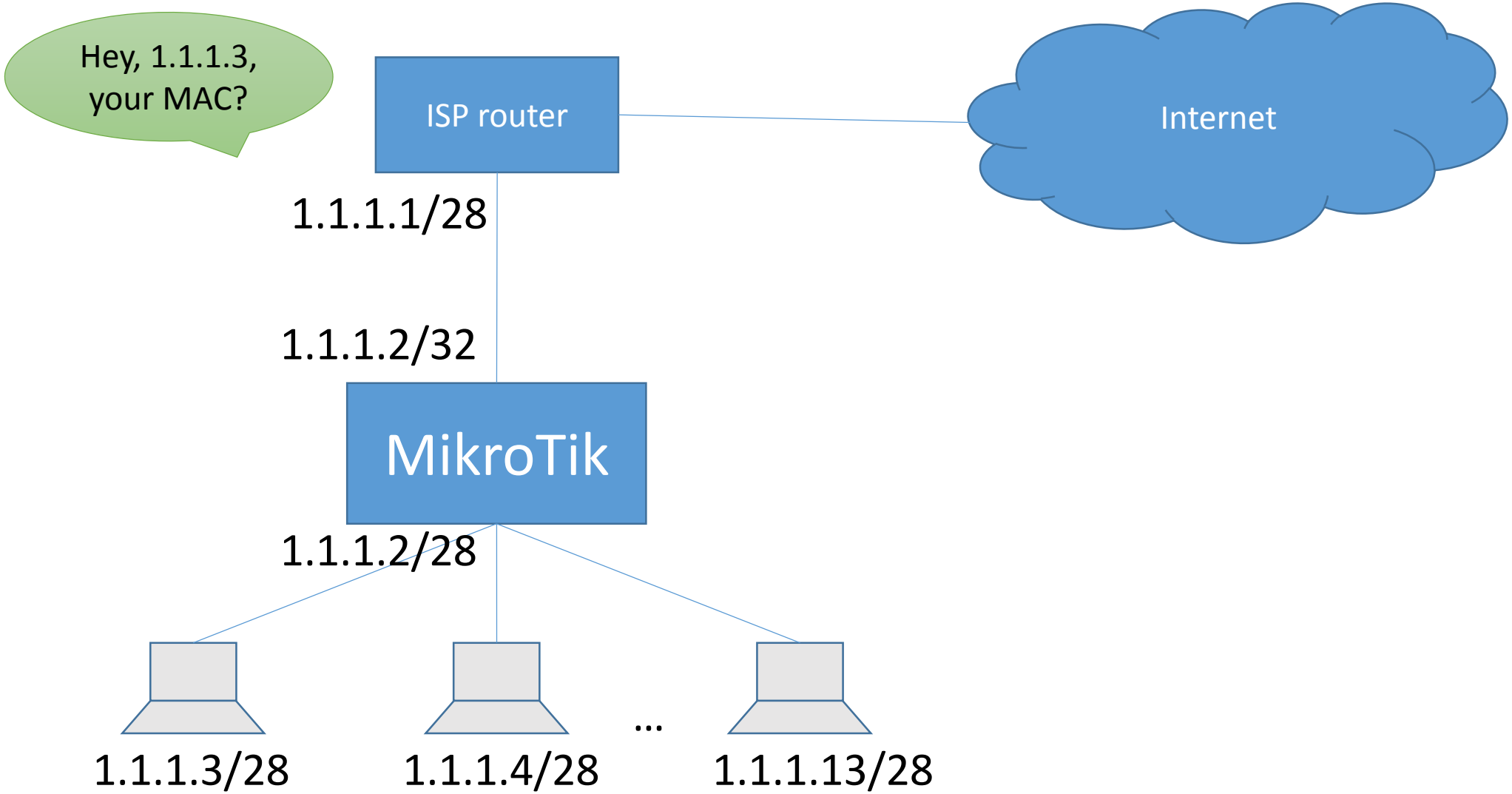
Problem 1 – IP pool from provider



Problem 1 – IP pool from provider



Problem 1 – IP pool from provider

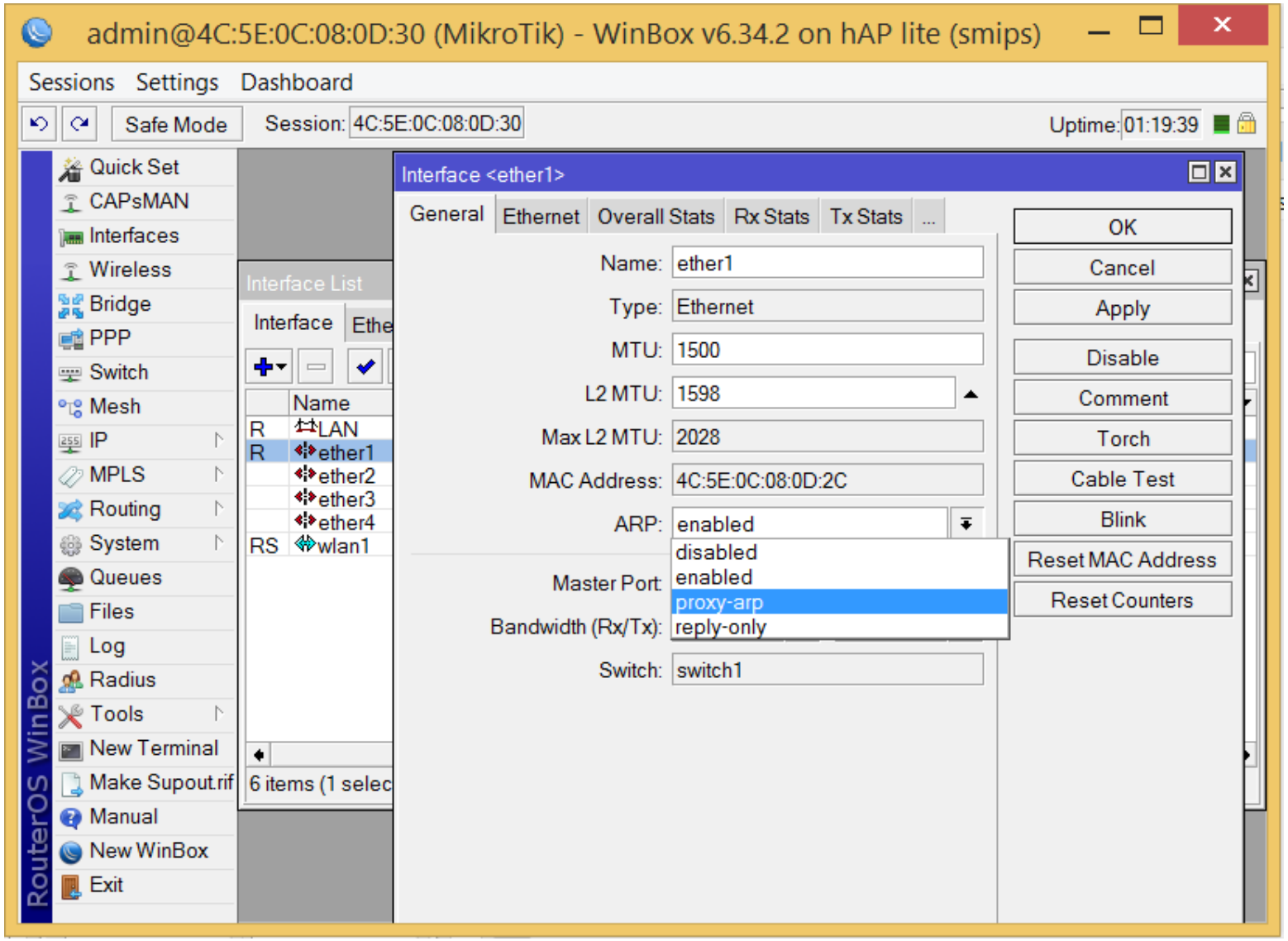


Problem 1 – IP pool from provider

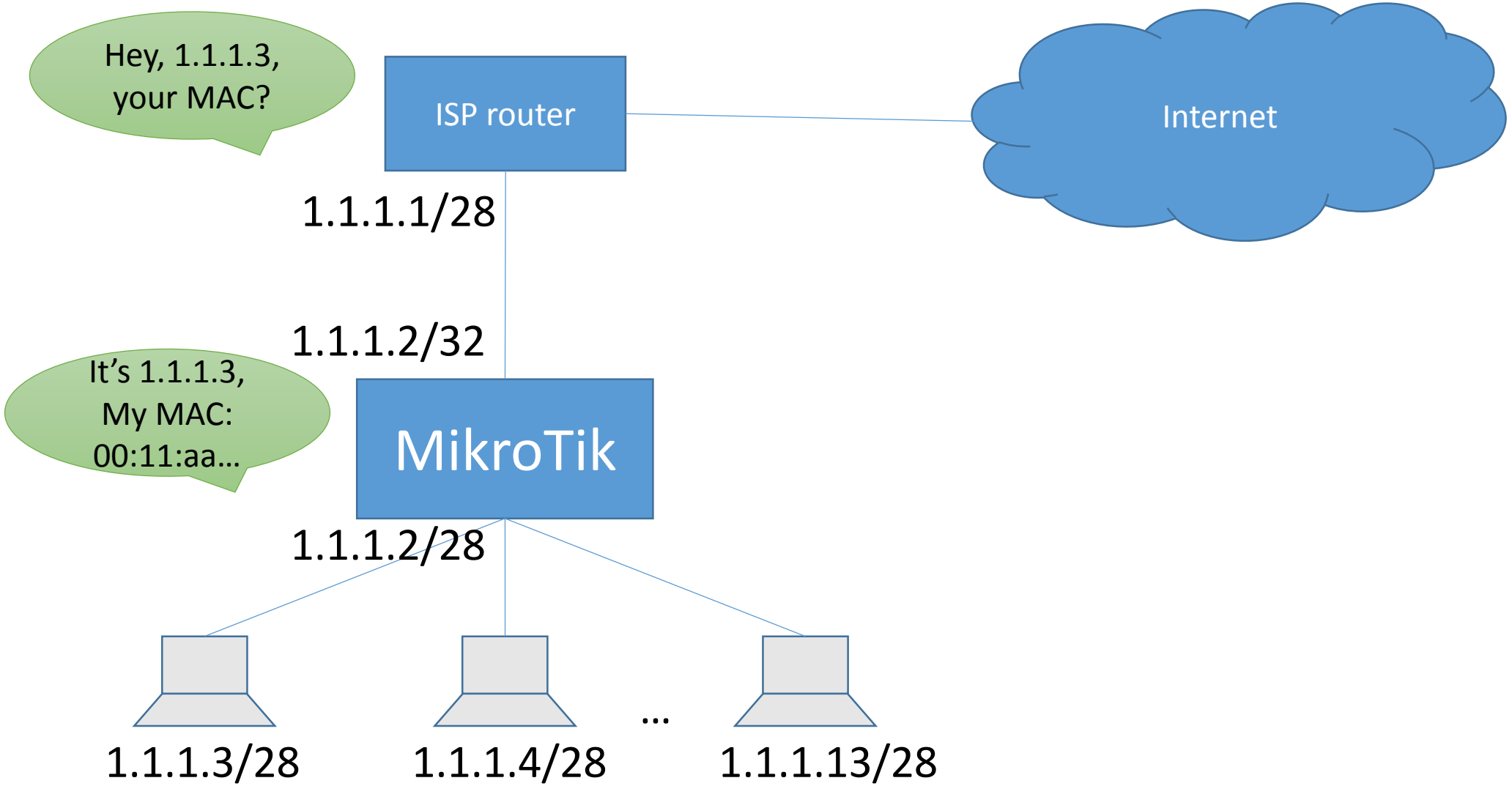
- Only thing we miss is proxy-arp on ether1

- MikroTik will respond to any ARP request on ether1 with its own MAC address if the requested IP address is on a network directly connected to the MikroTik

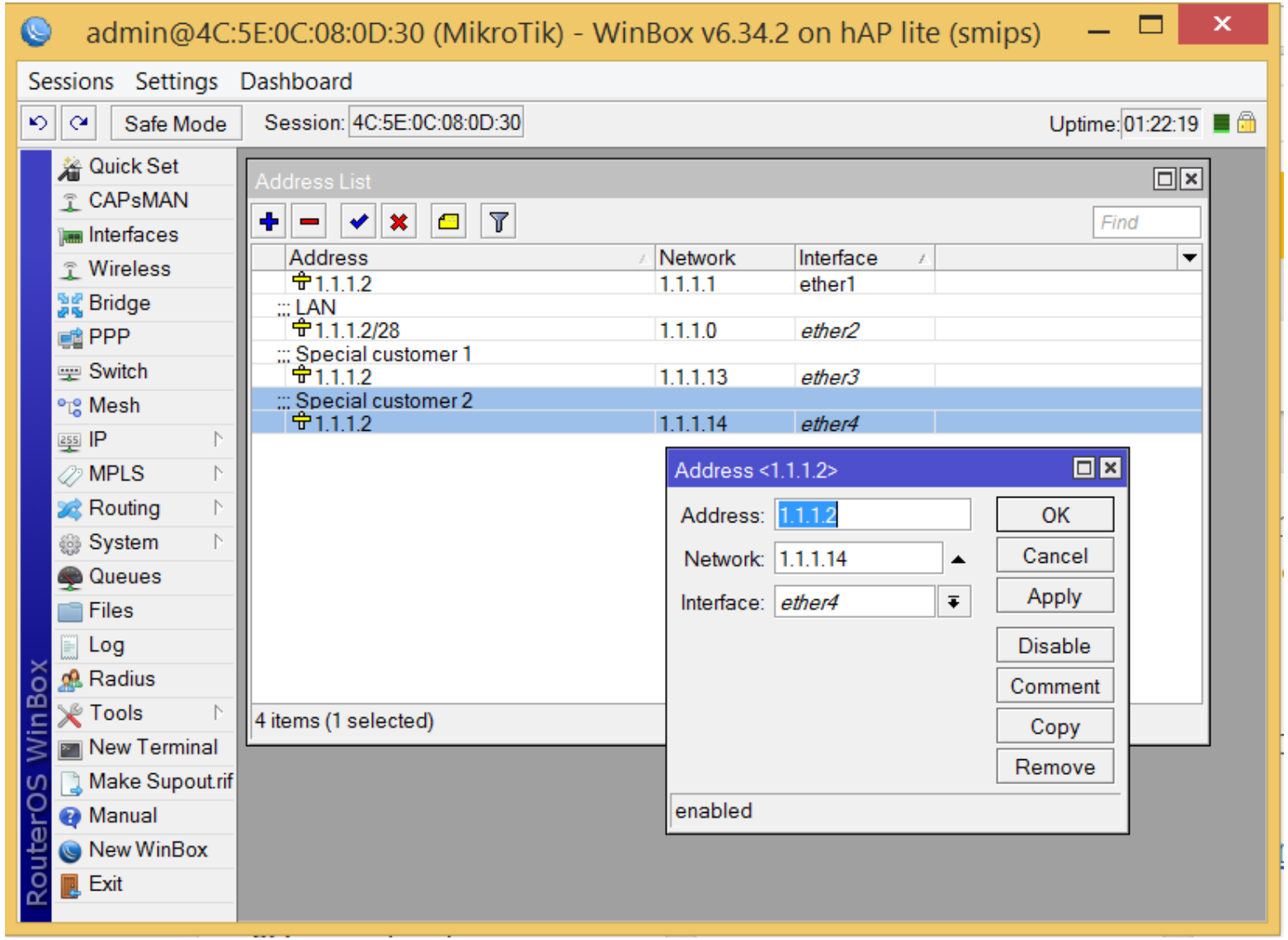
Problem 1 – IP pool from provider



Problem 1 – IP pool from provider



We can do even more!

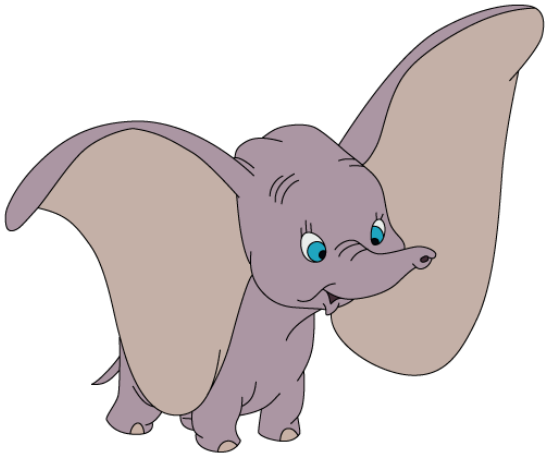


We can do even more!

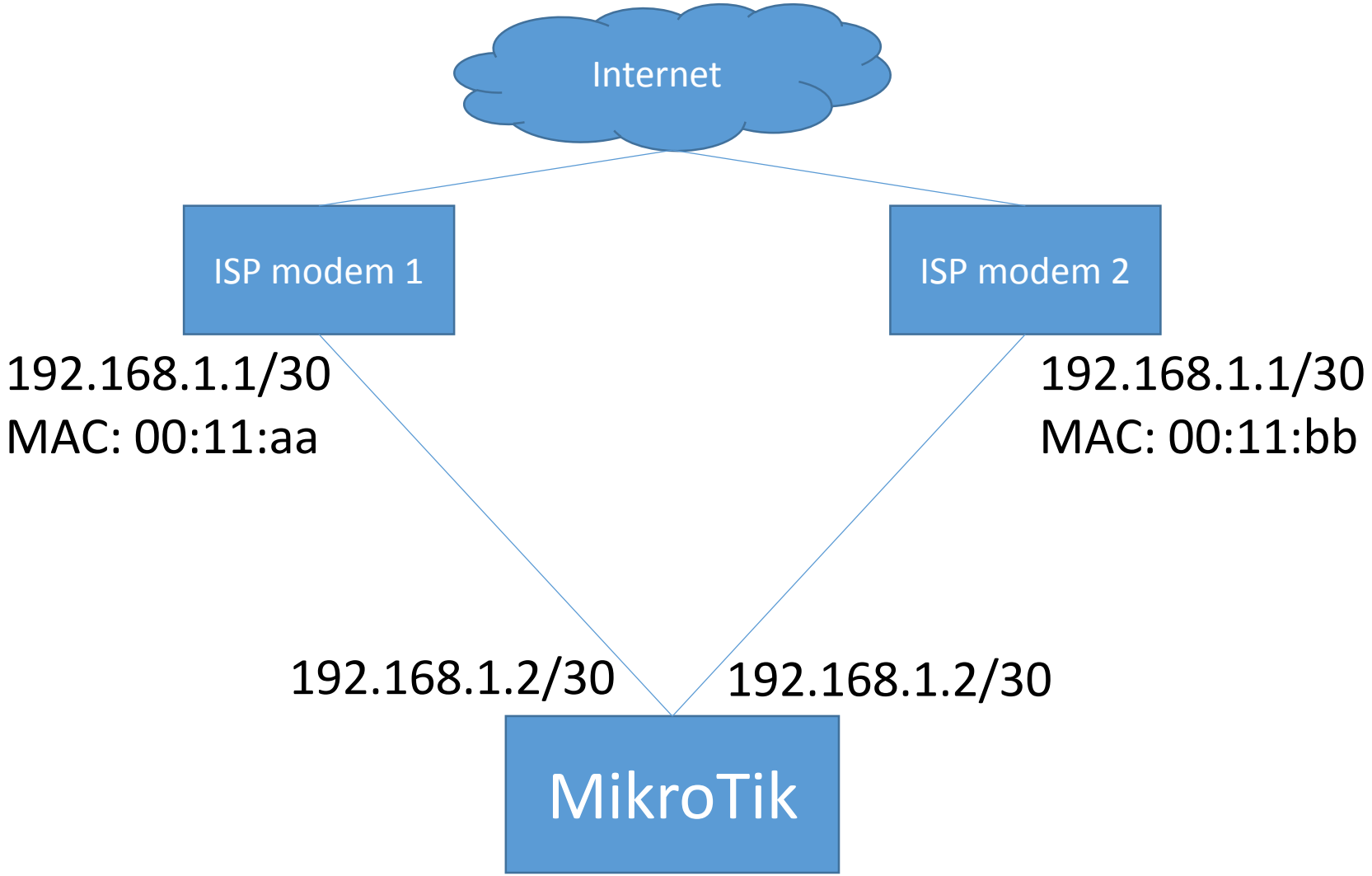
- We can add /32 addresses and proxy-arp for some special machines on separate ethernet ports
- The machines will think they are part of /28 network, we will treat them as individual networks
- The ethernet ports will not be bridged, treated as separate interfaces

- Example – we configure different DHCP server on each interface
- DHCP server giving only 1 IP address on 1 port, to any connected MAC

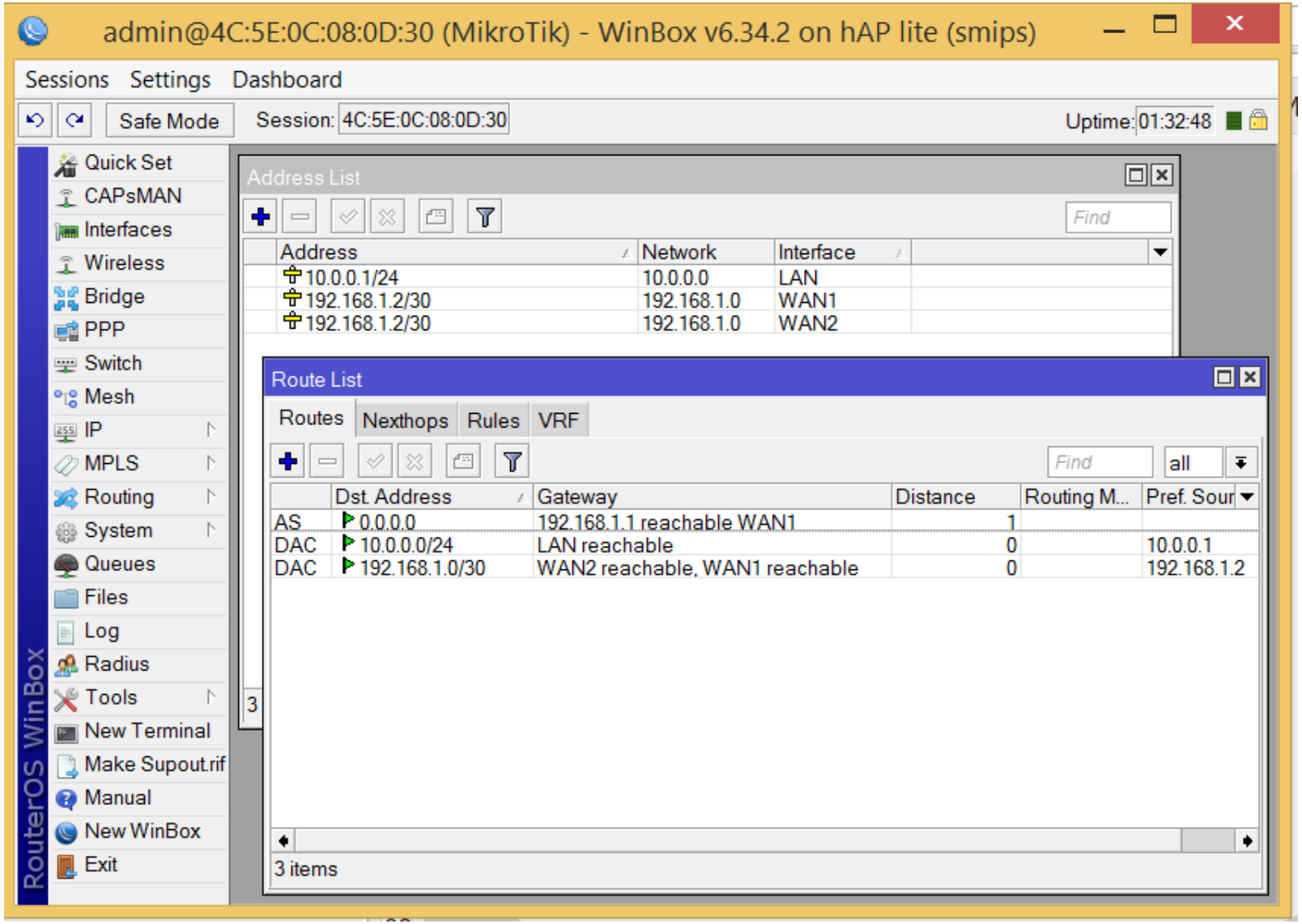
And now for something completely… crazy!



2 interfaces with the same IP address?



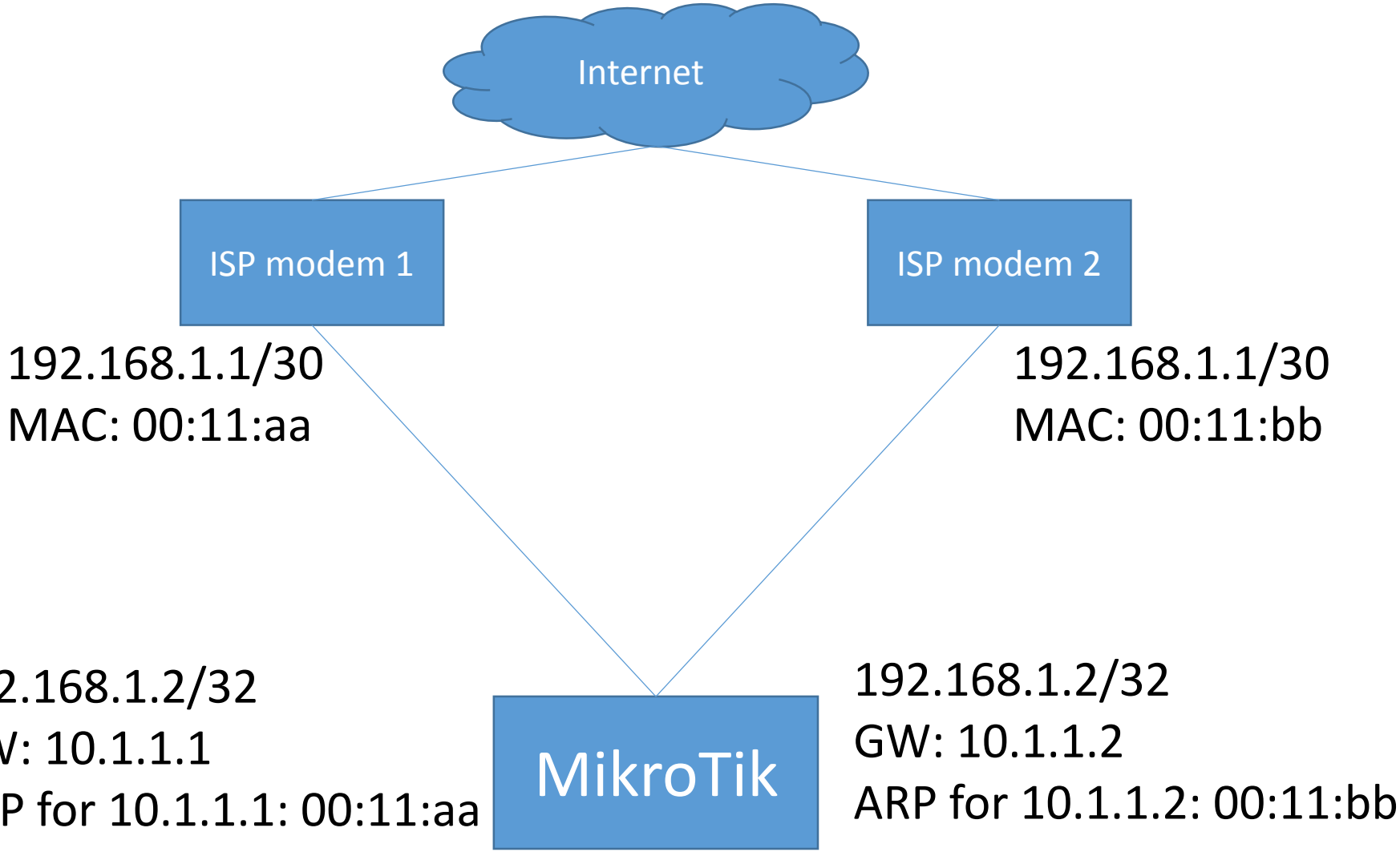
2 interfaces with the same IP address?



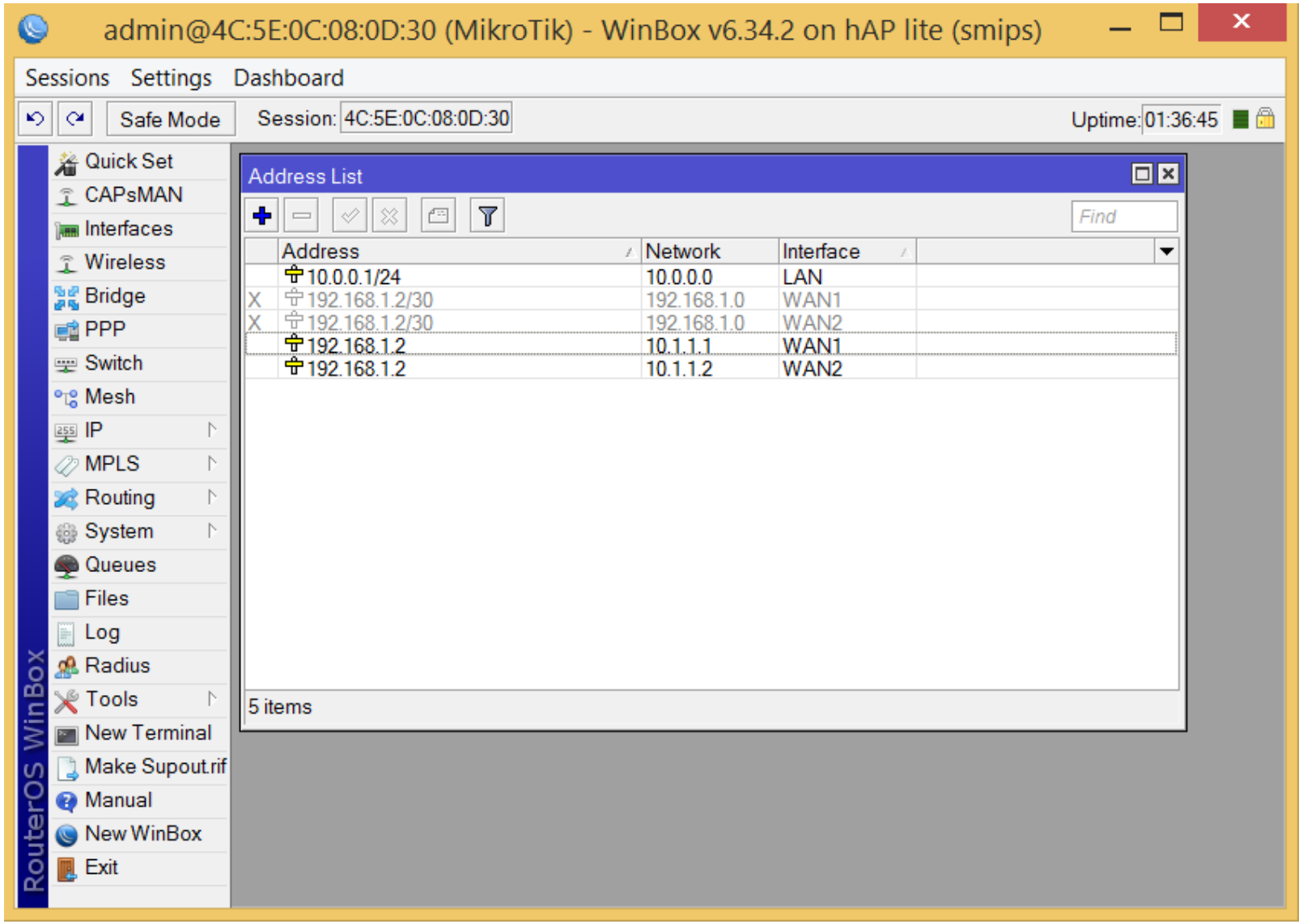
IP and ARP – short introduction (reminder)

- We add 1.1.1.1 as our default gateway (static route to 0.0.0.0/0)
- We want to send a packet to 8.8.8.8
- MikroTik checks route to 8.8.8.8 – it fits 0.0.0.0/0, so it's via 1.1.1.1
- MikroTik checks route to 1.1.1.1 – it fits 1.1.1.0/28, so it's on ether1
- MikroTik checks 1.1.1.1's MAC address with ARP request
- MikroTik sends the packet to:
  - Destination IP: 8.8.8.8
  - Destination MAC: MAC of 1.1.1.1
- **We configure IP of gateway only to use it's MAC!!!**

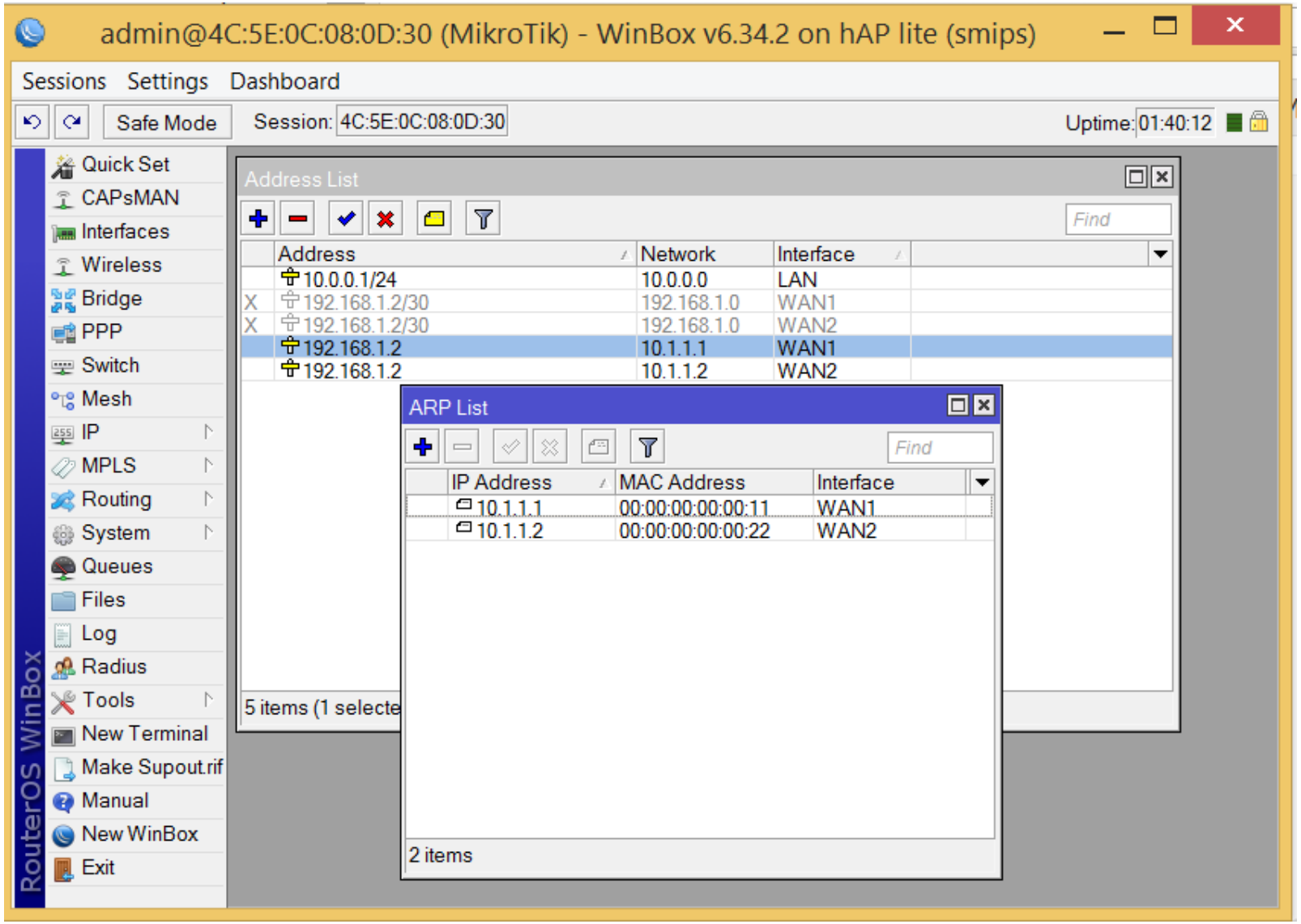
2 interfaces with the same IP address?



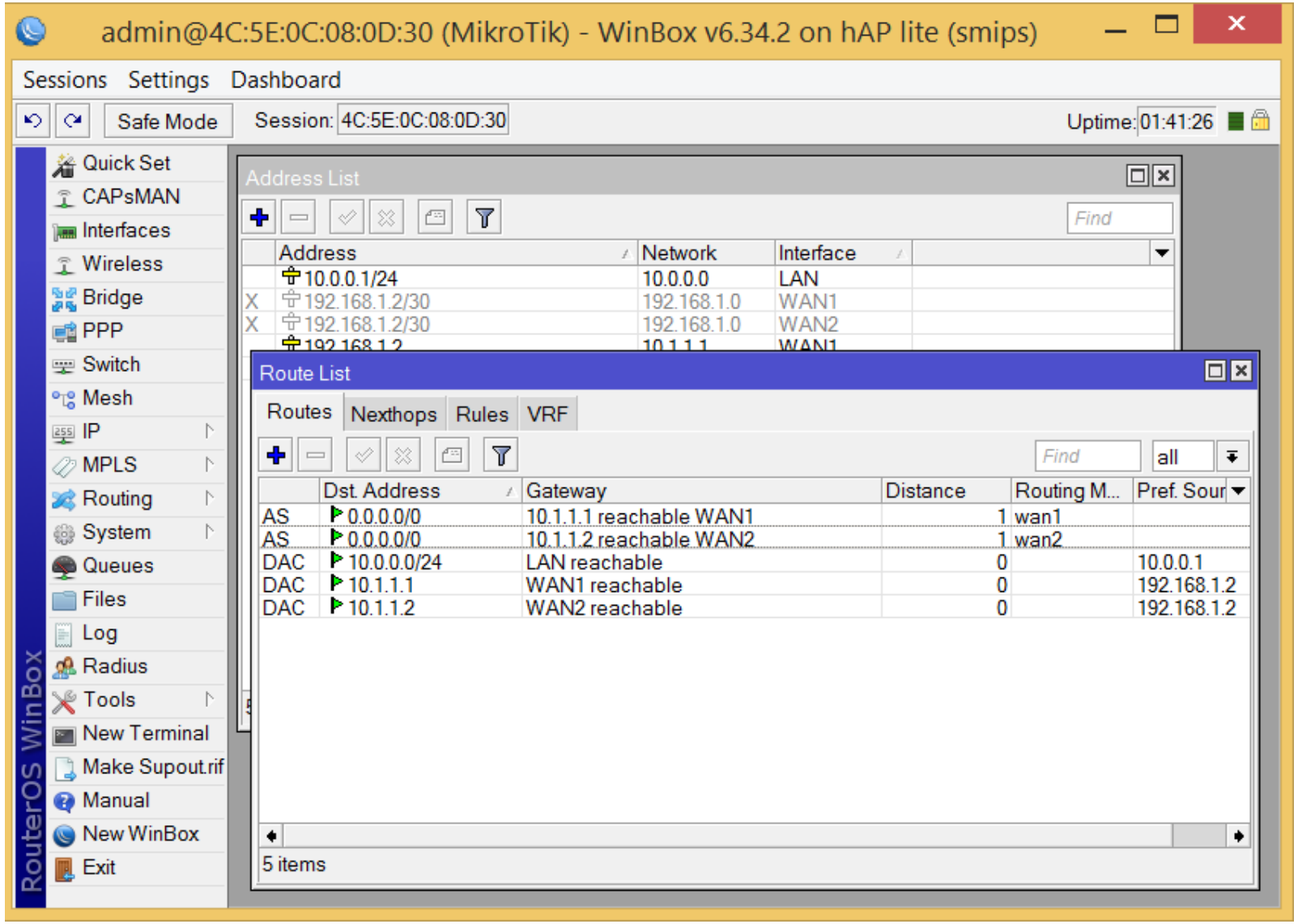
2 interfaces with the same IP address?



2 interfaces with the same IP address?



2 interfaces with the same IP address?

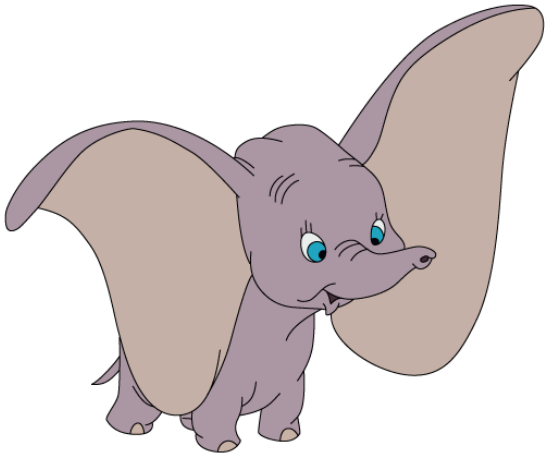


2 interfaces with the same IP address?

- We can use the 2 gateways simultaneously
- We can use them for different routing marks and do load balancing
- We can do a normal failover if one of the links goes down (but not using “check-gateway”)

- We can write a script to periodically check the MAC addresses for ARP
- If the IP addresses are given by DHCP, we can also write the script to set the 2 gateways.

Questions?
Comments?



Thanks!