IPv6 Fundamentals in LAN

SEE 3 RIPE NCC Regional Meeting
14-15 April 2014, Sofia, Bulgaria

Matjaž Straus Istenič
Aviat Networks, SINOG

matjaz@njetwork.si

http://twitter.yfrog.com/obn9ykktj?sa=0

On Friday, 14 September 2012, the RIPE NCC, the Regional Internet Registry (RIR) for Europe, the Middle East and parts of Central Asia, distributed the last blocks of IPv4 address space from the available pool. ...

And here's our RS team as the mouse button was clicked...the moment we reached the last /8 of...

RIPE NCC

http://twitter.yfrog.com/obn9ykktj?sa=0

Topics

- IPv6 Address
- ICMP and Automatic Configuration
- Fundamental Network Services
- Basic Security Considerations
- A Peek at Transition mechanisms

Topics

- IPv6 Address
- ICMP and Automatic Configuration
- Fundamental Network Services
- Basic Security Considerations
- A Peek at Transition mechanisms

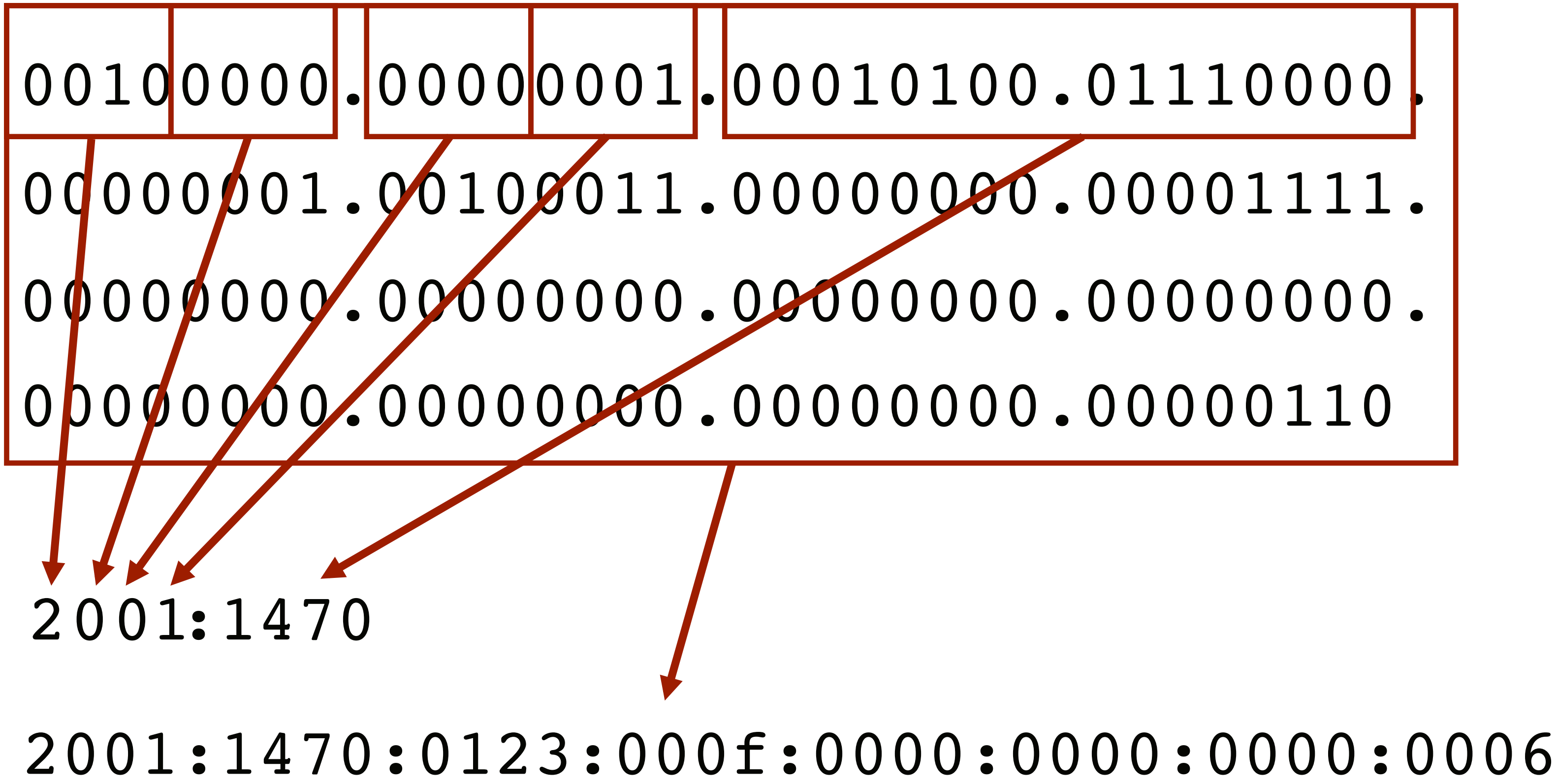
IPv6 Address

- Textual Representation
- Address Types
- IPv6 Address Distribution
- Addressing within Organisation

Textual Representation

- 128 bits in 8 groups,16 bits per group
- Hexadecimal notation
- Shortening (compressing)
- Canonical textual representation format

IPv6 Address – 128 bits

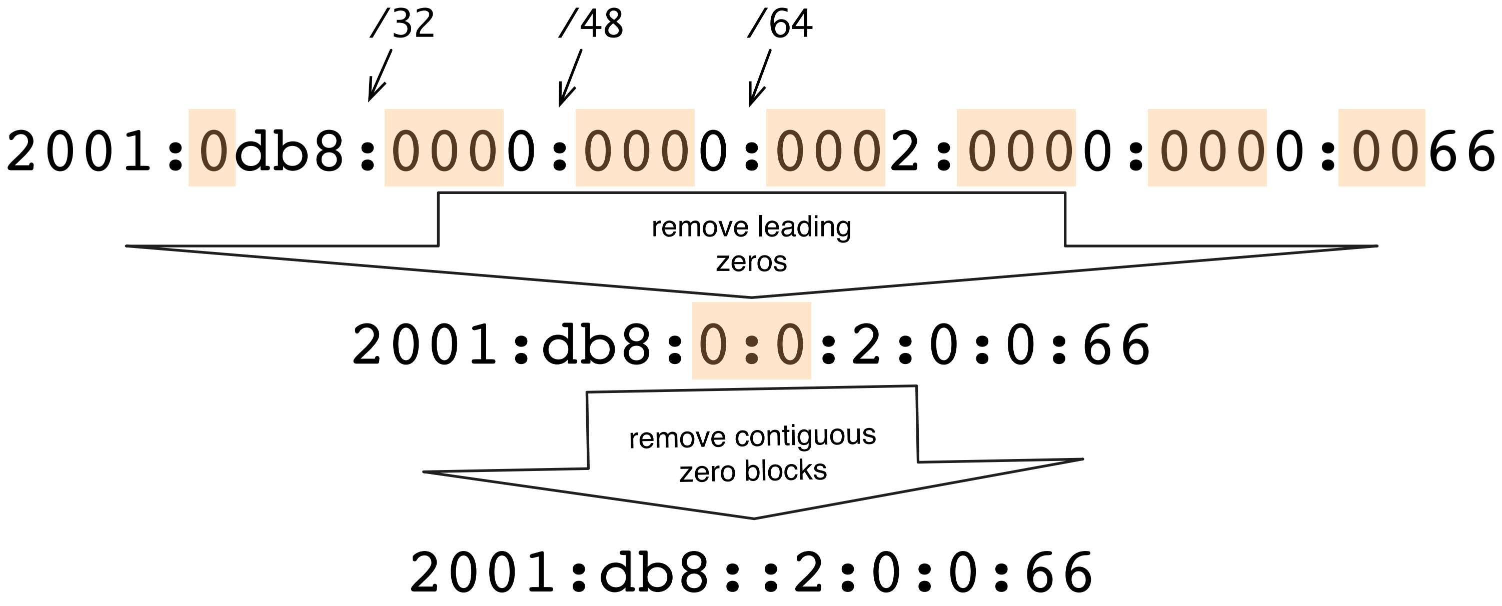


```
00100000.00000001.00010100.01110000.
00000001.00100011.00000000.00001111.
00000000.00000000.00000000.00000000.
00000000.00000000.00000000.00000110
```

```
2001:1470
```

```
2001:1470:0123:000f:0000:0000:0000:0006
```

Compressing the IPv6 Address



/32 /48 /64

2001:0db8:0000:0000:0002:0000:0000:0066

remove leading zeros

2001:db8:0:0:2:0:0:66

remove contiguous zero blocks

2001:db8::2:0:0:66

IPv6 Address – Recommendation for Textual Representation

- Always remove leading zeros
- Shorten as much as possible
- Do not replace one single 16-bit zero field with ::
- In case of alternative choices use :: at first zero sequence (left one)
- Lowercase

RFC 5952 – A Recommendation for IPv6 Address Text Representation

Address Types

- Usage – how do we communicate?
  - Unicast
  - Multicast
- Scope – where do we communicate?
  - Locally
  - Globally

Address Types

- Usage – **how** do we communicate
  - Between the two – *unicast*
  - With many – *multicast*
  - With the closest one – *anycast*
    (same address format as *unicast*)
- Scope – **where** do we communicate
  - Locally on the link – *link-local*
  - Locally within administrative domain – *unique local (ULA)*
  - Globally, publicly

Address Types – Address per Purpose

- Is is obvious from the address itself for what purpose and where is to be used
  - Local communication
  - Global communication
  - Communication within a group
    - Group scope

Address Types – some examples

Address	Address Range	Where is to be used
Loopback	::1	With the system
Link-local	fe80::/10	On the link
Unique local	fc00::/7	At location/organisation
Public and global	2000::/3	Globally
Group address (multicast)	ff00::/8	Locally or globally

127.0.0.1

169.254.0.0/16

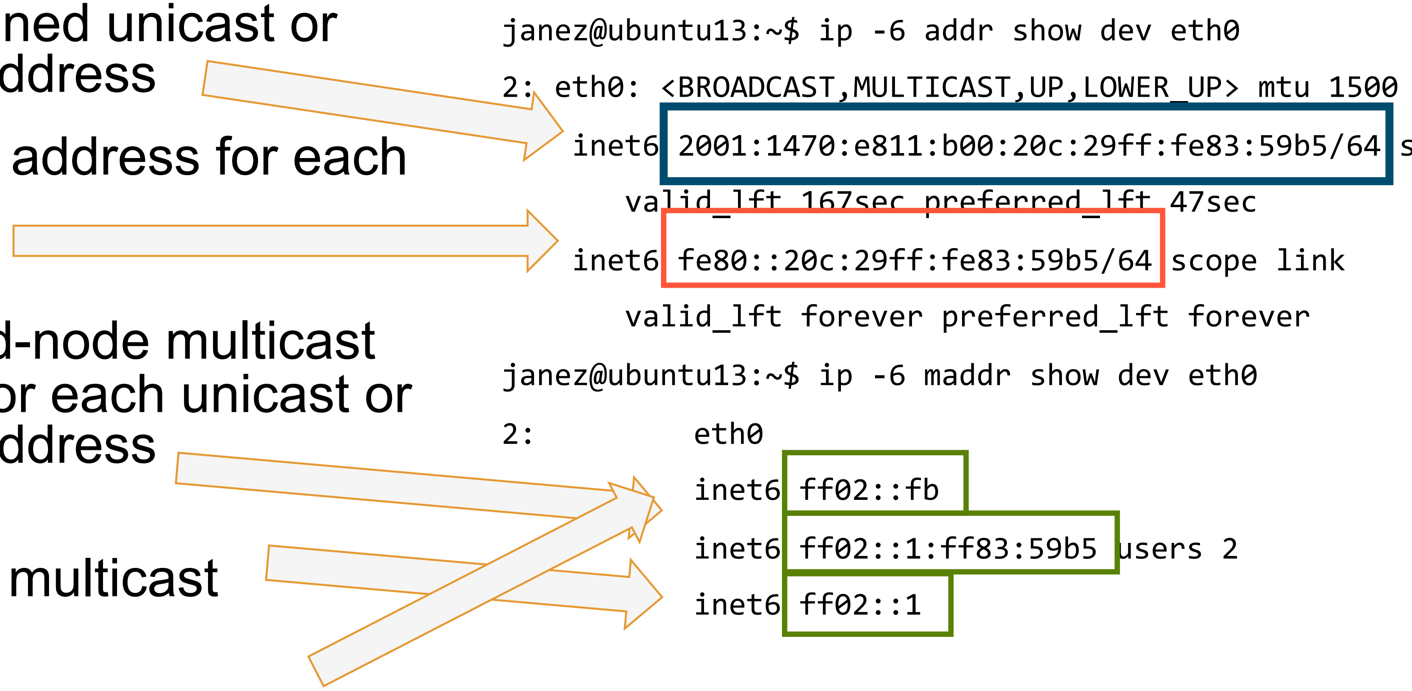
10.0.0.0/8
192.168.0.0/16

224.0.0.0/4

Address Types – multiple addresses per interface

At least 4 addresses:

- Any assigned unicast or anycast address
- Link-local address for each interface
- A solicited-node multicast address for each unicast or anycast address
- All-nodes multicast
- ... other multicast grups



```
janez@ubuntu13:~$ ip -6 addr show dev eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
    inet6 2001:1470:e811:b00:20c:29ff:fe83:59b5/64 s
       valid_lft 167sec preferred_lft 47sec
    inet6 fe80::20c:29ff:fe83:59b5/64 scope link
       valid_lft forever preferred_lft forever
janez@ubuntu13:~$ ip -6 maddr show dev eth0
2:          eth0
            inet6 ff02::fb
            inet6 ff02::1:ff83:59b5 users 2
            inet6 ff02::1
```

Address Types – IPv4-embedded

Address	Address Range	Example	Purpose
IPv4-mapped	`::ffff:0:0/96`	`::ffff:193.2.1.66`	Simplifies TCP/IP stack: IPv6-only functions/ applications
IPv4-embedded - IPv4-translatable - IPv4-converted	`<nat64-prefix>:<ipv4>/ (64|96)+ipv4_pfxlen`	`2001:db8::193.2.1.66`	NAT64

Address Types – Special Addresses

Address	Address Range	Example	Description
6to4	`2002::/16`	`2002:c102:142::/48`	6to4 tunneling: prefix for 193.2.1.66
Teredo	`2001::/32`	`2001:0:9d38:6abd:47f:2c9b:66fa:f7b`	Teredo tunneling: IPv4 address of the server, client, UDP port, ...
Documentation	`2001:db8::/32`	`2001:db8::66`	Documentation, books, examples, workshops, LABs ... (debate |dɪˈbeɪt)

http://www.ripe.net/lir-services/new-lir/ipv6_reference_card.pdf

Task 1 – IPv6 Addresses

- Which ones are valid and appropriate?

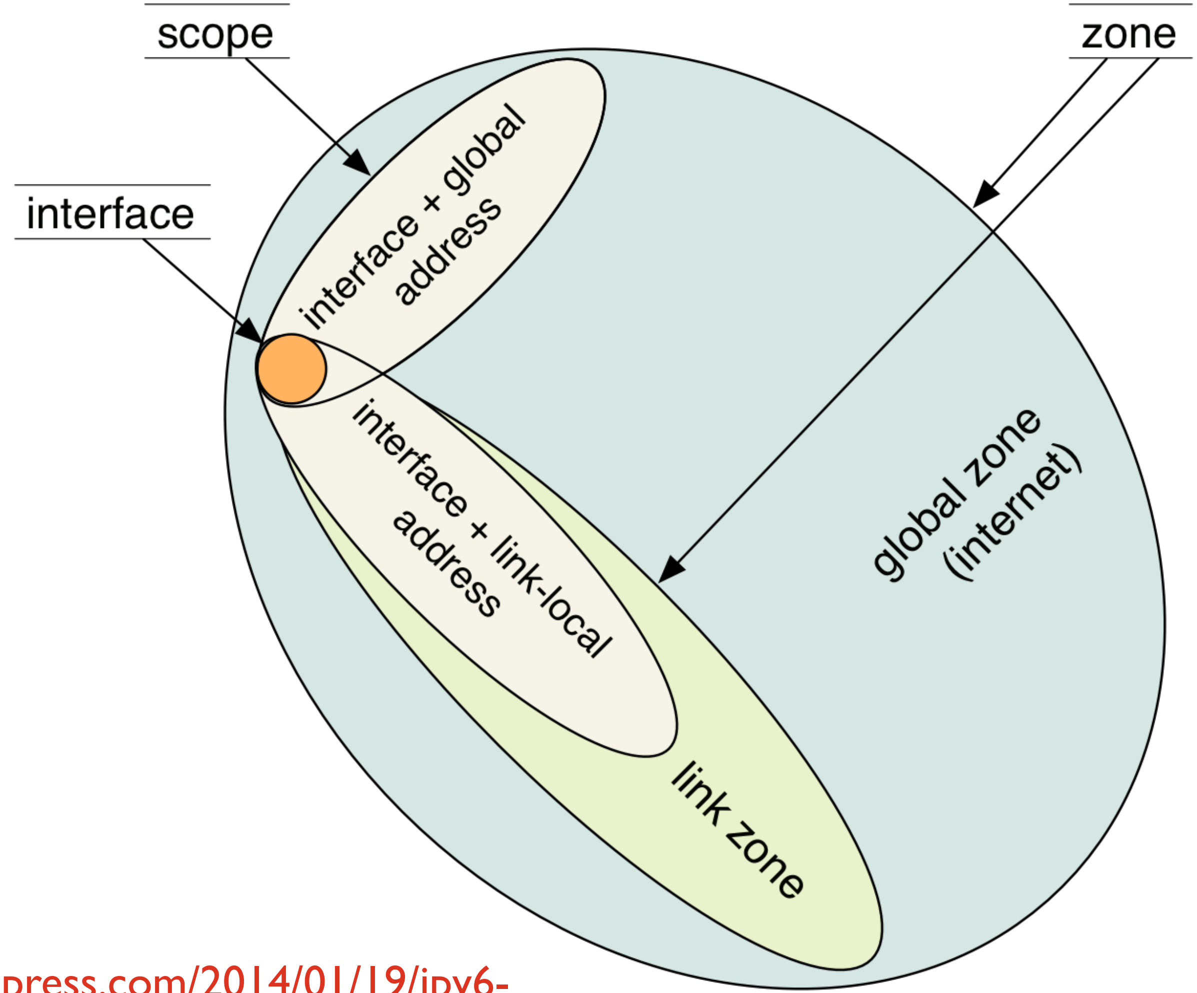
Address Types – Scopes

- Scope is define by an address
  - interface-local
  - link-local
  - unique local (site-local is deprecated)
  - global

Scope

- Device (node)
- Link
- Organisation/administrative domain (site)
- Internet (global)

Interface belongs to a certain **zone** of each possible scope



http://njetwork.wordpress.com/2014/01/19/ipv6-scopes-and-zones/

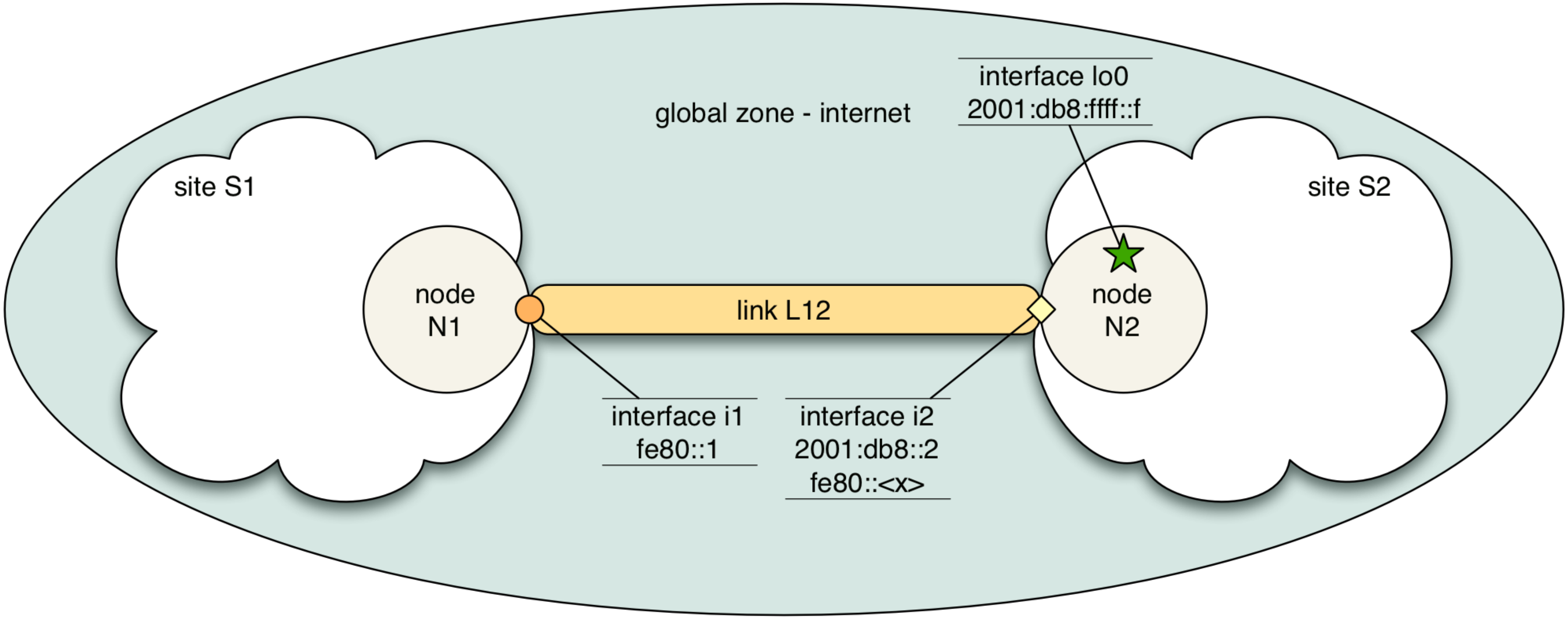
Zone Isolation Principle

- Packet with a source or destination address from a particular zone must stay within that zone boundary
- To comply with the principle it is Required and Sufficient that ...
  - Source interface (the sender of the packet) is in the same zone as destination address and
  - Destination interface (the receiver) is in the same zone as source address

Zone Isolation Principle – will these *pings* work?

ping from node N1: fe80::1→ 2001:db8::2

ping from node N1: fe80::1→ 2001:db8::ffff:f



Which address will be used?

Yes, there are strict rules ☺

- Source selection for as f(destination)

1. Prefer same address
2. Prefer appropriate scope
3. Avoid deprecated
4. Prefer home address
5. Prefer outgoing interface
   Prefer address in a prefix advertised by the next-hop
6. Prefer matching label
7. Prefer temporary
8. Use longest matching prefix

- Destination list sorted according to:

1. Avoid unusable
2. Prefer matching scope
3. Avoid deprecated
4. Prefer home address
5. Prefer matching label
6. Prefer higher precedence
7. Prefer native transport
8. Prefer smaller scope
9. Use longest matching prefix

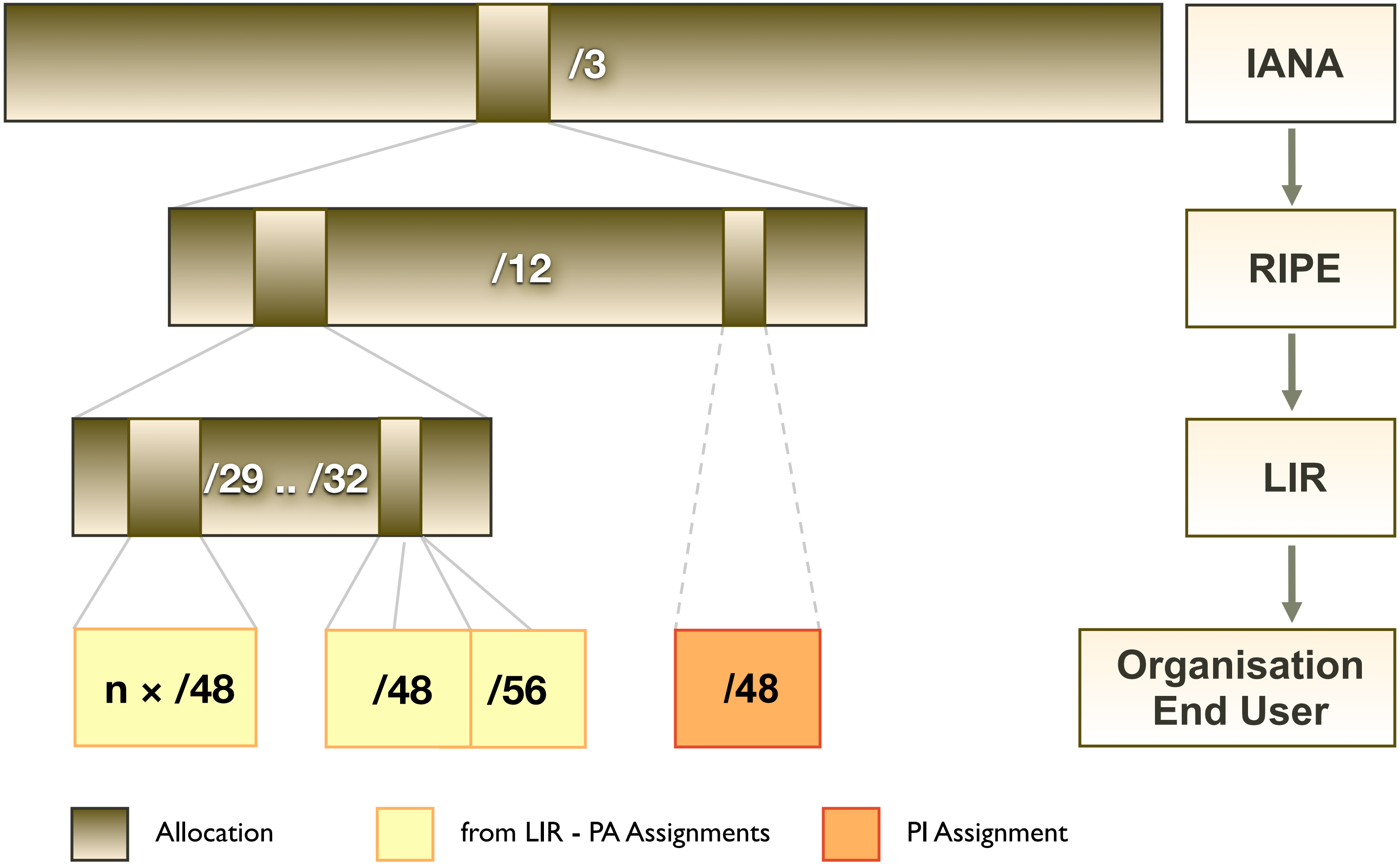
RFC 6724 - Default Address Selection for Internet Protocol Version 6

Which address will be used? (simplified version)

- Prefer equal scope or type
- Prefer smaller scopes over larger ones for the destination
- Prefer non-deprecated source
- Avoid transitional addresses such as tunnels
- Prefer pair with longest common prefix
- Prefer temporary source over public one
- Prefer home address

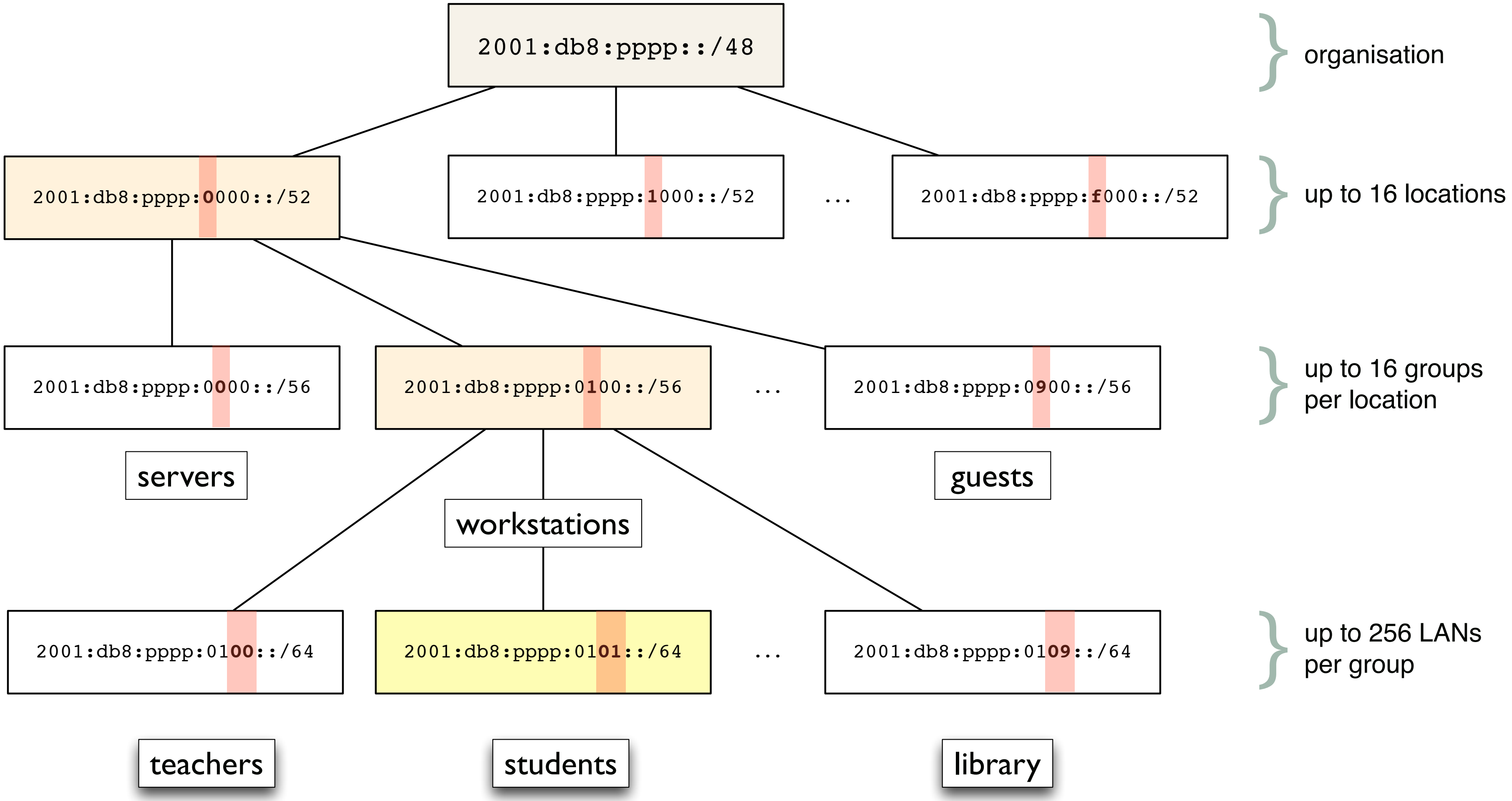
RFC 6724 - Default Address Selection for Internet Protocol Version 6

IPv6 Address Distribution



source: RIPE NCC's IPv6 for LIRs training

Example Address Plan



2001:db8:pppp::/48

2001:db8:pppp:**0**000::/52

2001:db8:pppp:**1**000::/52

...

2001:db8:pppp:**f**000::/52

2001:db8:pppp:0**0**00::/56

servers

2001:db8:pppp:0**1**00::/56

workstations

...

2001:db8:pppp:0**9**00::/56

guests

2001:db8:pppp:01**00**::/64

teachers

2001:db8:pppp:01**01**::/64

students

...

2001:db8:pppp:01**09**::/64

library

organisation

up to 16 locations

up to 16 groups per location

up to 256 LANs per group

Is it really so difficult?

- Divide 192.168.0.0/23 into 5 subnets
- Choose 5 /64 subnets from 2001:db8:1234::/48

Is it really so difficult?

192.168.0.0/23

- 192.168.0.0/26
- 192.168.0.64/26
- 192.168.0.128/25
- 192.168.1.0/25
- 192.168.1.128/25

2001:db8:1234::/48

- 2001:db8:1234:**1**::/64
- 2001:db8:1234:**2**::/64
- 2001:db8:1234:**3**::/64
- 2001:db8:1234:**4**::/64
- 2001:db8:1234:**5**::/64

Address Plan

LIR	User				Device ID		
/32	/48	/52	/56	/64		/112	/128
2001:db8:	pppp:	L	G	NN:		DHCP	

- pppp – assigned by provider (LIR)
- L = 0 or location
- G – group, per usage/service (security policy)
- NN – subnet in group G
- DHCP – mark for DHCPv6, for example “da” – *Dynamically Allocated*

Address Plan (option 2 – more groups)

LIR	User					Device ID	
/32	/48	/52	/60	/64		/112	/128
2001:db8:	pppp:	L	GG	N:		DHCP	

- pppp – assigned by provider (LIR)
- L = 0 or location
- GG– group
- N – subnet in group GG
- DHCP – mark for DHCPv6 Addresses

Address Plan (option 3 - flat)

LIR	User			Device ID	
/32	/48	/52	/64	/112	/128
2001:db8:	pppp:	L	GGG:	DHCP	

- pppp – assigned by provider (LIR)
- L = 0 or location
- GGG – group: subnet ID, number, VLAN ID ...
- DHCP – mark for DHCPv6 Addresses

Address Plan – starting points

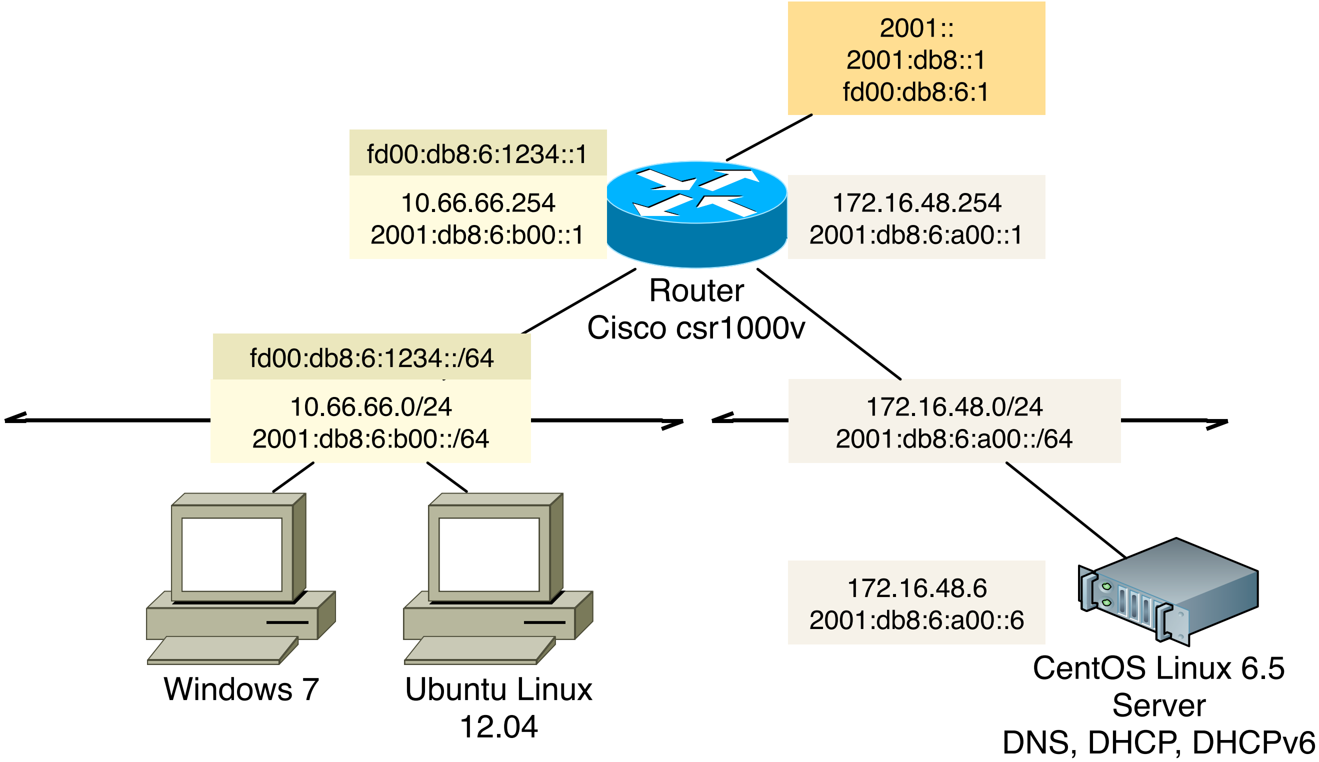
LIR	User					Device ID	
/32	/48	/52	/56	/64		/112	/128
2001:db8:	pppp:	L	G	NN:		DHCP	

- How can we use L?
- How to split into group, how many groups (G)?
- How to ID the subnets, how many of them per group (N)?
- How to distinguish DHCPv6 addresses?
  - How big will the DHCPv6 pool be?
- How to align with the existing network topology?

Task 2 – Address Plan

- Make an Address Plan
  Keep in mind:
  - Hierarchic subneting
  - Transparent and clean design
  - Alignment with existing network topology
  - Divide on (n × 4)-bit boundaries
  - LAN is always /64

Demo Topology



Task 3 – my little LAB

- Startup the virtual machines
- Open console windows
- Use SSH to connect to Linux VMs and the router
- Check default network settings

Topics

- IPv6 Address
- ICMP and Automatic Configuration
- Fundamental Network Services
- Basic Security Considerations
- A Peek at Transition mechanisms

ICMP and Auto-Configuration

- ICMPv6
- Local network mechanisms – Link operations
  (ND - Neighbour Discovery)
- Automatic configuration for hosts
  (SLAAC - Stateless Address Autoconfiguration)

ICMPv6

- Very important protocol
- Particularly in LAN
  - *Self configuration with network parameters (Address, GW) – including DAD*
  - *Establishing neighbourship*
  - *Router solicitations*
  - *Reachability checks*
- **no ICMPv6 = no IPv6 connectivity**

Some ICMPv6 Messages

- Neighbour, where are you?
- Neighbour, are you still there?
- Anybody with the same address?
- Router, where are you?
- I’m your router
- ...and I’m sending data for you to auto-configure
- How large can my packets be?
- Packet is too big

Local Network Mechanisms – (ND - Neighbour Discovery)

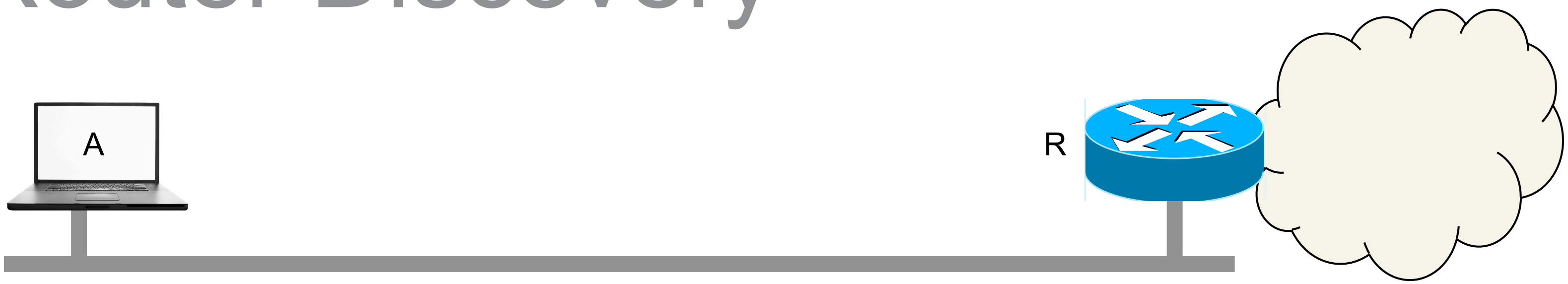
Task (Link operation)	IPv4	IPv6
Router discovery	DHCP	ND
Prefix discovery	-	ND for hosts DHCP-PD for routers
Getting network parameters	DHCP	ND (MTU) DHCP (DNS, NTP, ...)
Address assignment	DHCP	ND (SLAAC) DHCP (global addresses only)
Duplicate address detection (DAD)	ARP	ND
L2 address resolution (ethernet MAC), finding neighbours	ARP	ND
Reachability check (NUD)	ARP	ND
Redirection	ICMP	ND

inspired by Cisco’s BRKSEC 3003, Advanced IPv6 Security in the LAN

Link Operations – important ICMPv6 messages in ND

- ND takes care for most operations on the link
- ND is based on ICMPv6
  - *neighbour solicitation*
  - *neighbour advertisement*
  - *router solicitation*
  - *router advertisement*
  - *redirect*
- Operations rely on *link-local multicast*

Router Discovery



ICMP 133 – router solicitation

Source = link-local address(A)

Dest = all-routers multicast ff02::2

Query = send RA

RS

RA

ICMP 134 – router advertisement

Source = link-local address(R)

Dest = all-nodes multicast ff02::1

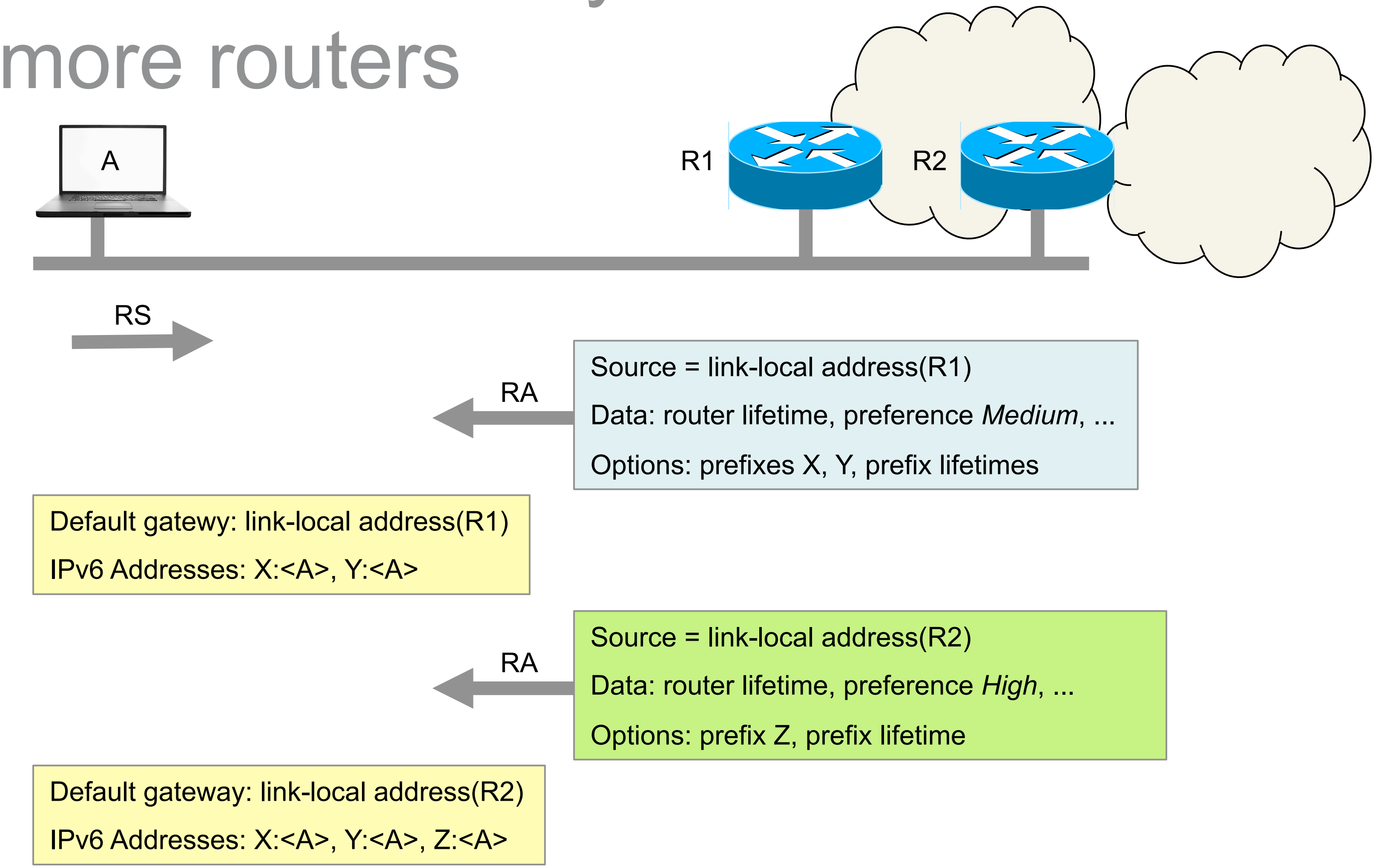
Data: router lifetime, preference, ...

Options: prefix X, prefix lifetime

Default gateway: link-local address(R)

IPv6 Address: X:<A>

Router Discovery – more routers



Source = link-local address(R1)

Data: router lifetime, preference *Medium*, ...

Options: prefixes X, Y, prefix lifetimes

Default gatewy: link-local address(R1)

IPv6 Addresses: X:<A>, Y:<A>

Source = link-local address(R2)

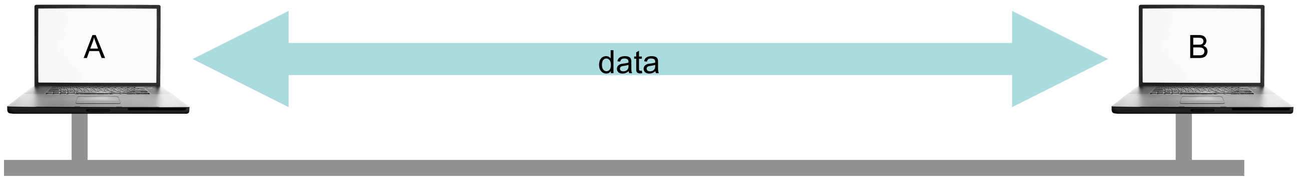
Data: router lifetime, preference *High*, ...

Options: prefix Z, prefix lifetime

Default gateway: link-local address(R2)

IPv6 Addresses: X:<A>, Y:<A>, Z:<A>

Address Resolution – Establishing neighbourship



ICMP 135 – neighbour solicitation

Source = link-local address(A), MAC(A)

Dest = solicited-node multicast za B

Query = what is MAC(B)?

NS-lookup →

Neighbour cache

B *incomplete*

← NA

ICMP 136 – neighbour advertisement

Source = one of B's addresses

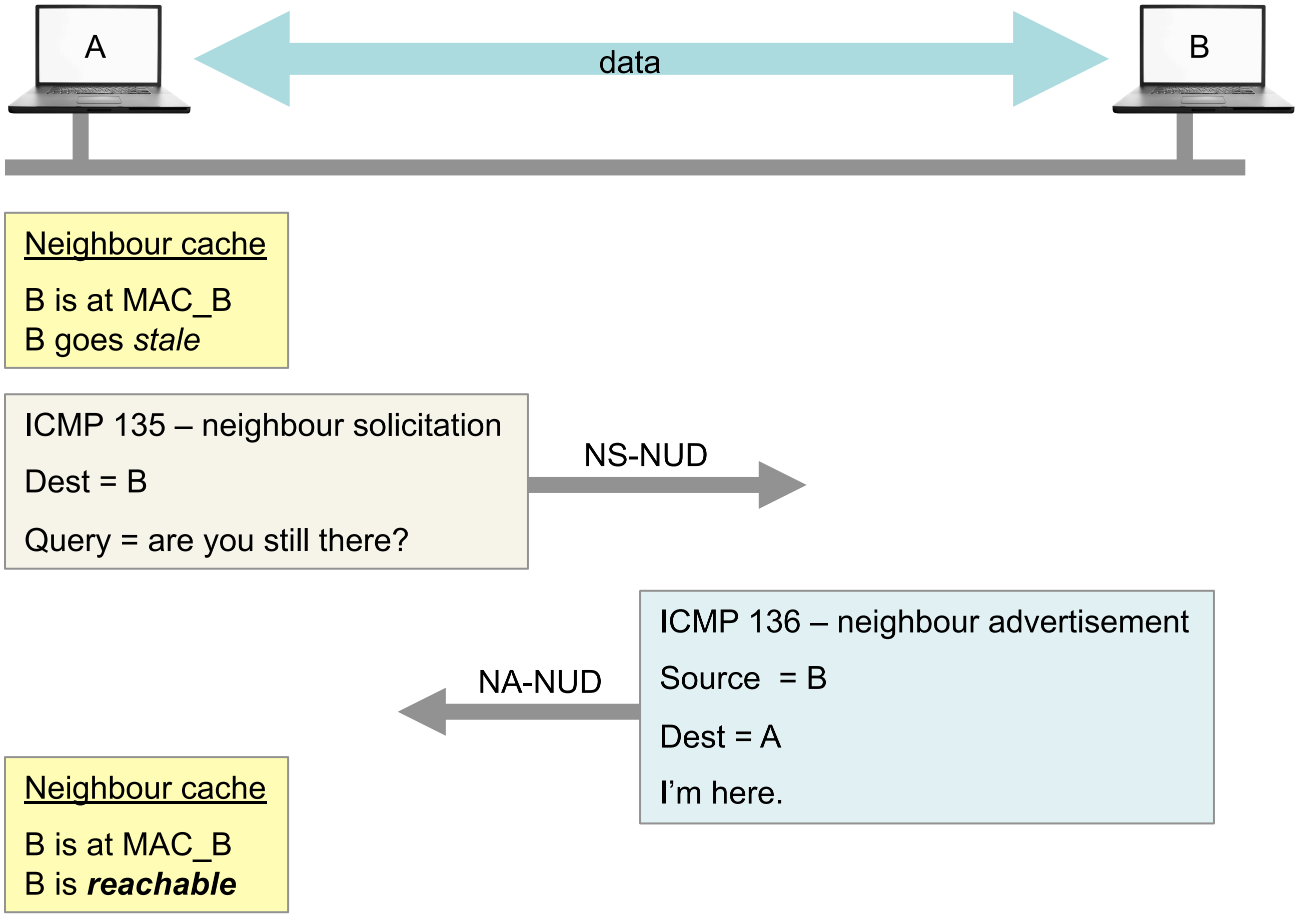
Dest = A

Options: ethernet address is MAC_B

Neighbour cache

B is at MAC_B
B is ***reachable***

Confirming neighbourship



Neighbour cache

B is at MAC_B
B goes *stale*

ICMP 135 – neighbour solicitation

Dest = B

Query = are you still there?

ICMP 136 – neighbour advertisement

Source = B

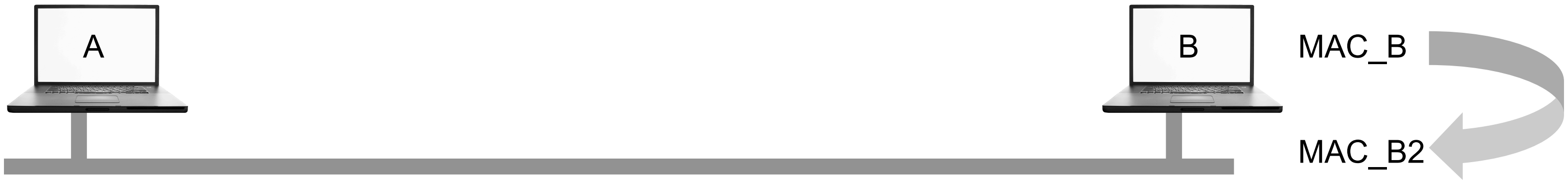
Dest = A

I'm here.

Neighbour cache

B is at MAC_B
B is ***reachable***

Notification when *ethernet* (MAC) address is changed



<u>Neighbour cache</u>

B is at MAC_B
B is ***reachable***

NA-override unsolicited

ICMP 136 – neighbour advertisement

Source = B

Dest = all-nodes multicast

Options: new ethernet address is MAC_B2

<u>Neighbour cache</u>

B is now at MAC_B2
B is ***reachable***

Link-local operations – ND instead of ARP

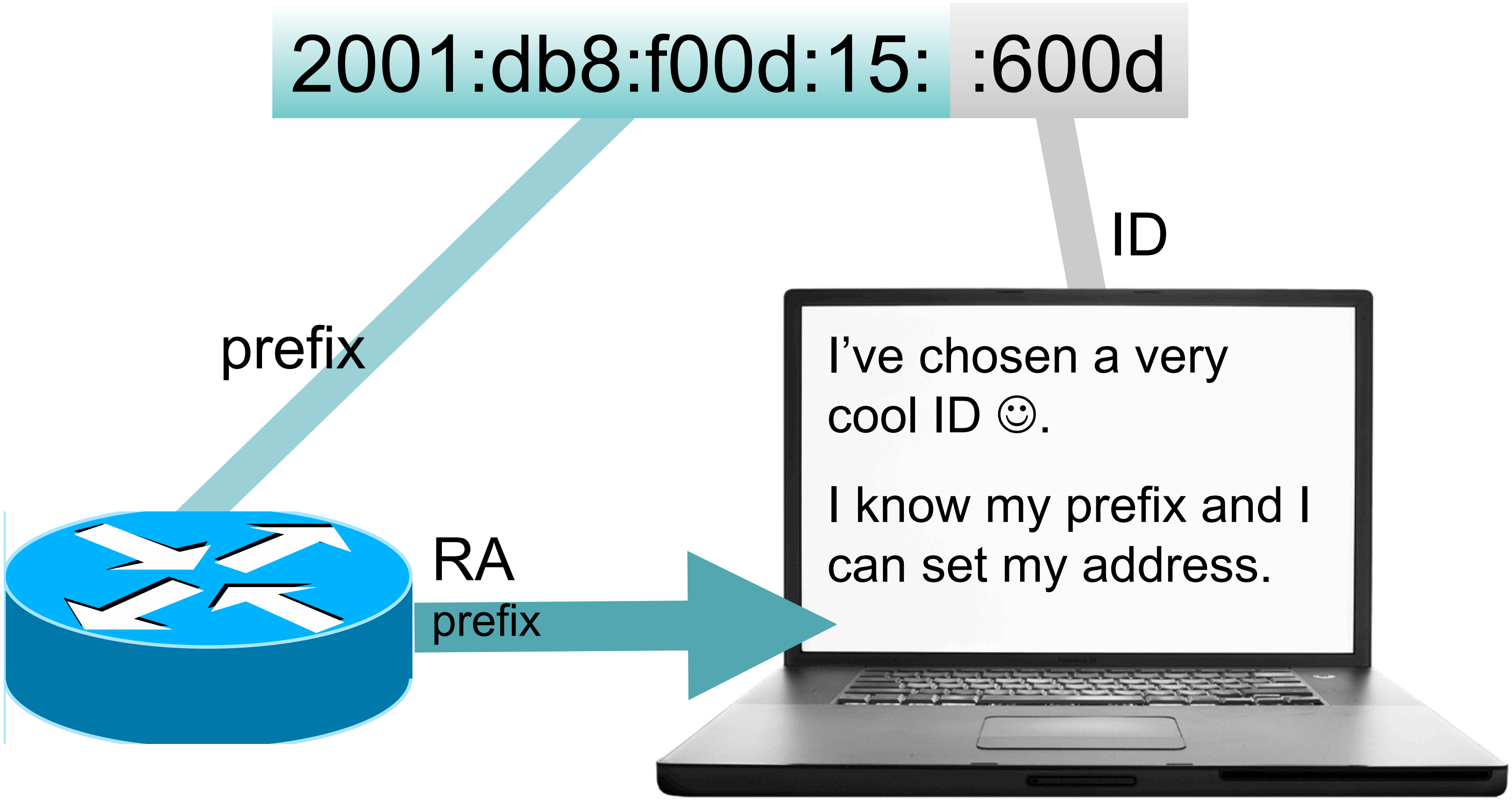
Key takeaways

- ARP ✗ ND ✓
- *neighbour cache* table *(NC)*
- ICMP is used for all messages
- SLAAC
  - Prefix ⇒ Address
  - Default gateway
  - MTU

Stateless Address Auto-configuration – SLAAC

1. Host chooses its own ID
2. Host configures its local address and check if unique (DAD)
3. Host sends RS query
4. Host receives RA reply
5. For each prefix from RA
   - Use prefix to configure global address (run DAD as well)
     only if A = 1, otherwise prefix is not used for addressing
6. Based on RA:
   - Set default gateway to *link-local* address of RA sender
   - Flag M and O = 0: no DHCP
   - Flag M = 1: DHCP for IPv6 address
   - Flag O = 1: DHCP for other network parameters, like DNS

SLAAC



2001:db8:f00d:15: :600d

prefix

ID

RA

prefix

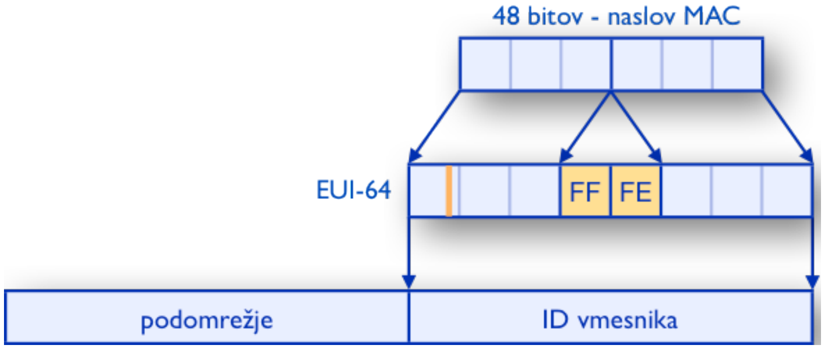
I’ve chosen a very cool ID ☺.

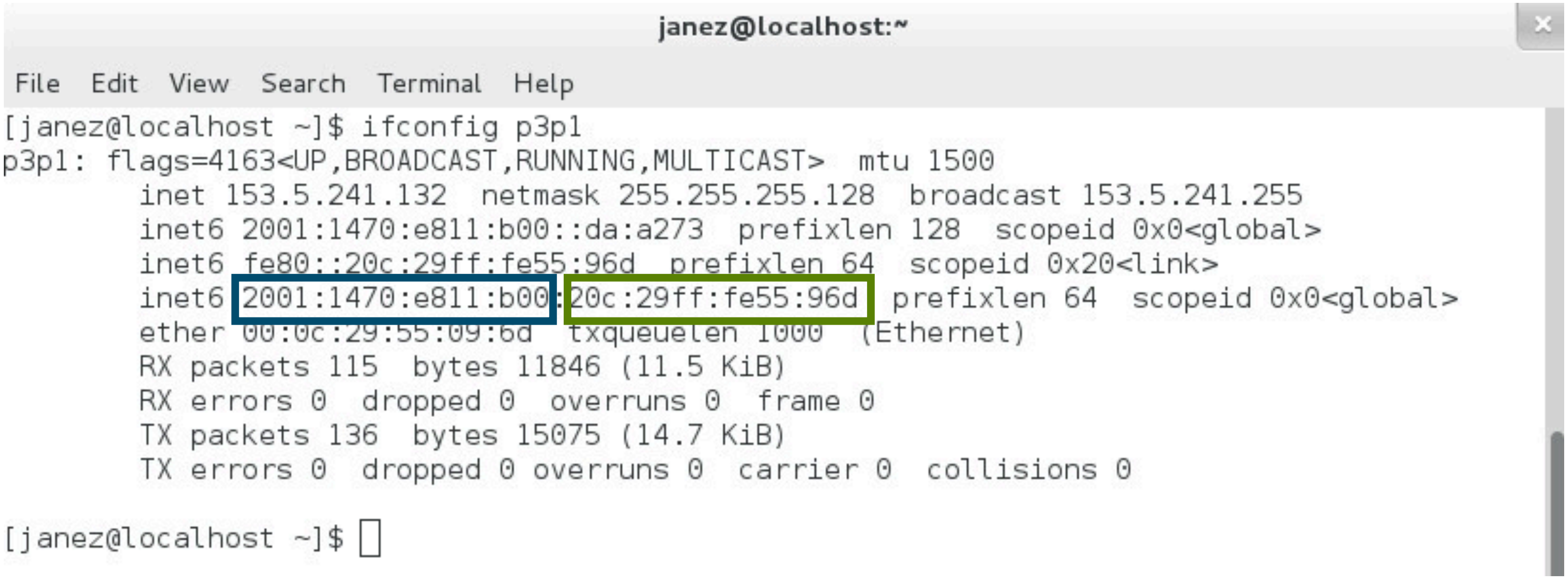
I know my prefix and I can set my address.

SLAAC – some facts

- Prefix is always /64
- Various possibilities for an ID:
  - Hardware based – *modified EUI-64*
  - Private (*privacy extensions*)
  - Crypto-generated (CGA)
  - Stable privacy
- Always run DAD

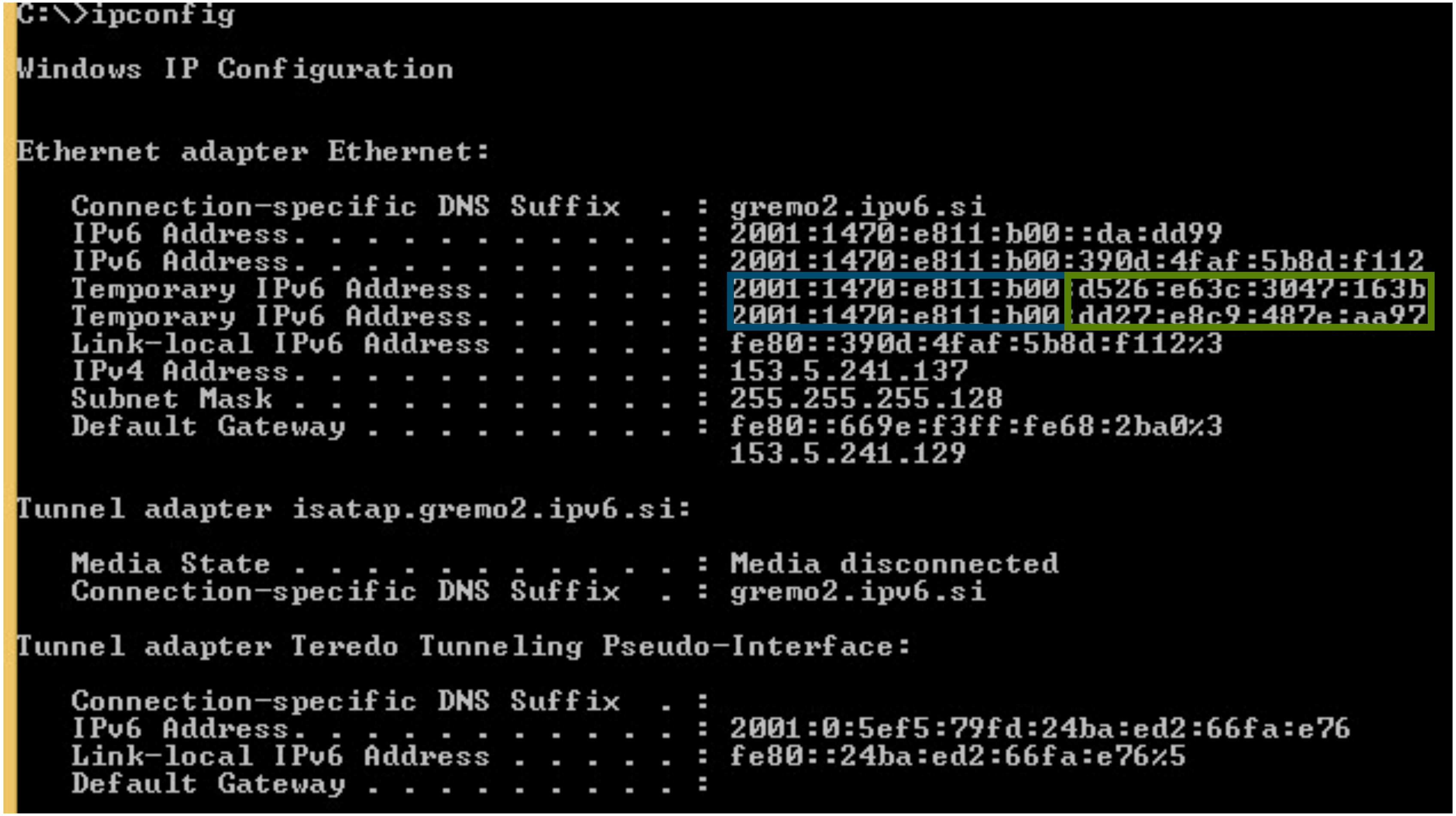
SLAAC – EUI-64-based Address





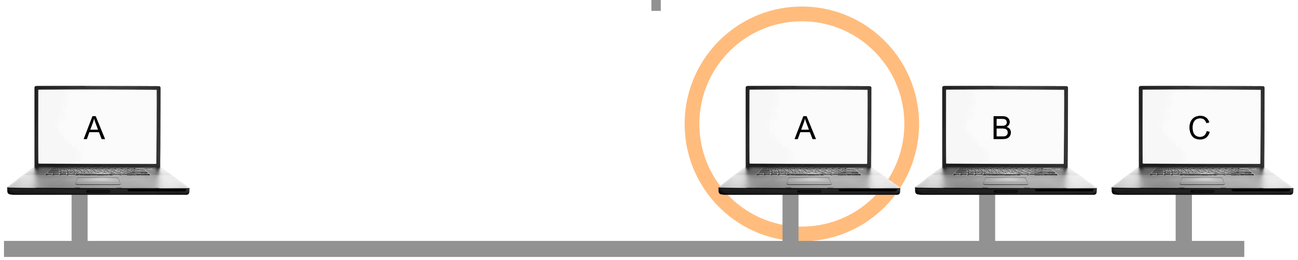
```
[janez@localhost ~]$ ifconfig p3p1
p3p1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 153.5.241.132  netmask 255.255.255.128  broadcast 153.5.241.255
        inet6 2001:1470:e811:b00::da:a273  prefixlen 128  scopeid 0x0<global>
        inet6 fe80::20c:29ff:fe55:96d  prefixlen 64  scopeid 0x20<link>
        inet6 2001:1470:e811:b00:20c:29ff:fe55:96d  prefixlen 64  scopeid 0x0<global>
        ether 00:0c:29:55:09:6d  txqueuelen 1000  (Ethernet)
        RX packets 115  bytes 11846 (11.5 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 136  bytes 15075 (14.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[janez@localhost ~]$ 
```

SLAAC – Private and Temporary Addresses



```
C:\>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . : gremo2.ipv6.si
   IPv6 Address. . . . . . . . . . . : 2001:1470:e811:b00::da:dd99
   IPv6 Address. . . . . . . . . . . : 2001:1470:e811:b00:390d:4faf:5b8d:f112
   Temporary IPv6 Address. . . . . . : 2001:1470:e811:b00:d526:e63c:3047:163b
   Temporary IPv6 Address. . . . . . : 2001:1470:e811:b00:dd27:e8c9:487e:aa97
   Link-local IPv6 Address . . . . . : fe80::390d:4faf:5b8d:f112%3
   IPv4 Address. . . . . . . . . . . : 153.5.241.137
   Subnet Mask . . . . . . . . . . . : 255.255.255.128
   Default Gateway . . . . . . . . . : fe80::669e:f3ff:fe68:2ba0%3
                                       153.5.241.129

Tunnel adapter isatap.gremo2.ipv6.si:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . : gremo2.ipv6.si

Tunnel adapter Teredo Tunneling Pseudo-Interface:

   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . : 2001:0:5ef5:79fd:24ba:ed2:66fa:e76
   Link-local IPv6 Address . . . . . : fe80::24ba:ed2:66fa:e76%5
   Default Gateway . . . . . . . . . :
```

Check for Uniqueness – DAD



ICMP 135 – neighbour solicitation

Source: :: (undefined)

Dest = solicited-node multicast for A

Query = is somebody using A?

NS-DAD →

I can use address A ✓

← NA

ICMP 136 – neighbour advertisement

Source = A

Dest = A

A is taken and I can **not** use it! ✗

Task 4 – Enable IPv6 with SLAAC

- Start Wireshark
- Enable IPv6 on the router
- Check network settings on Windows and Linux
- Analyse ND traffic
- Check connectivity (ping the gateway)
- Disable tunnels and privacy extensions on Windows 7

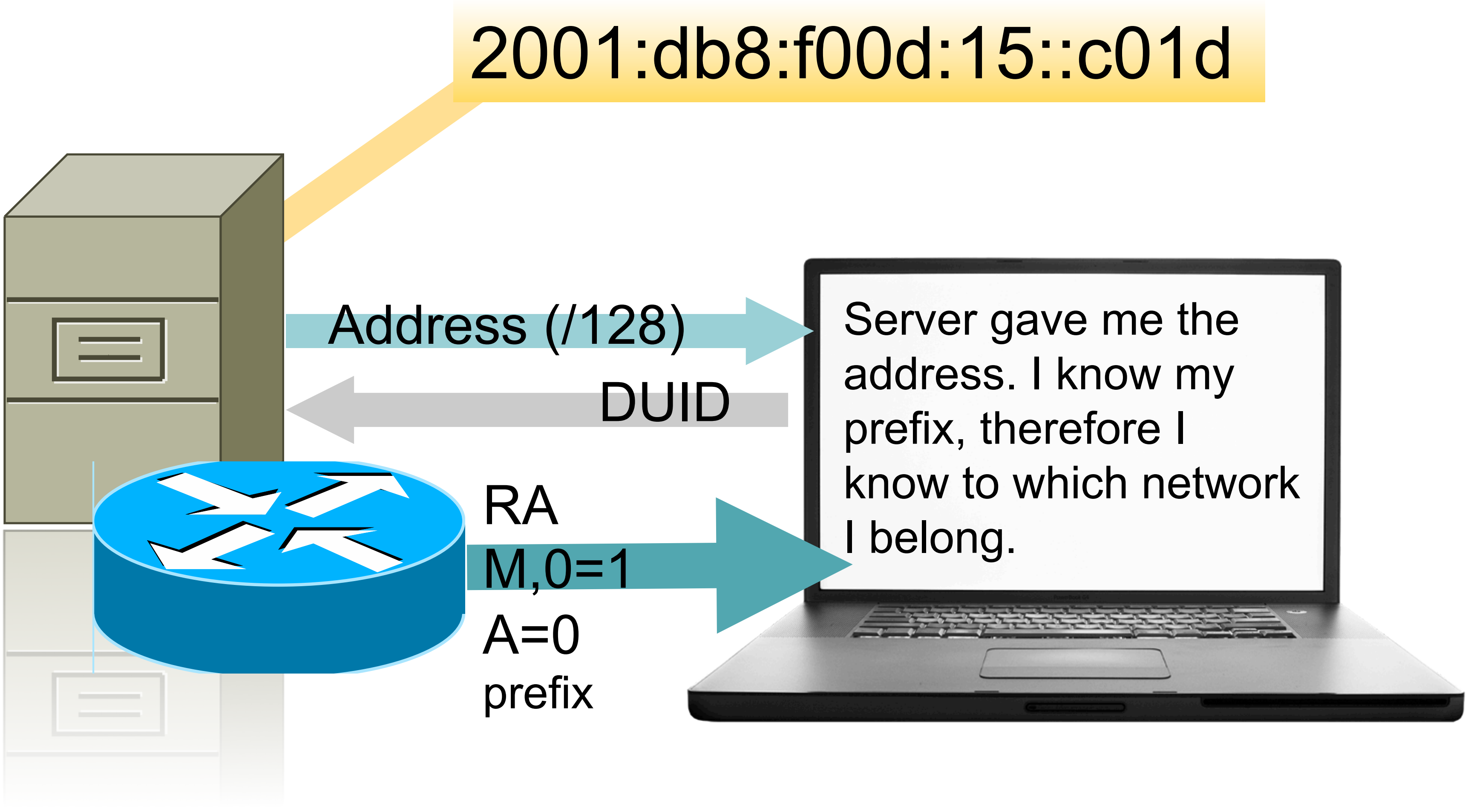
Topics

- IPv6 Address
- ICMP and Automatic Configuration
- Fundamental Network Services
- Basic Security Considerations
- A Peek at Transition mechanisms

Fundamental network services

- DHCP
- DNS

DHCP



2001:db8:f00d:15::c01d

Address (/128)

DUID

RA

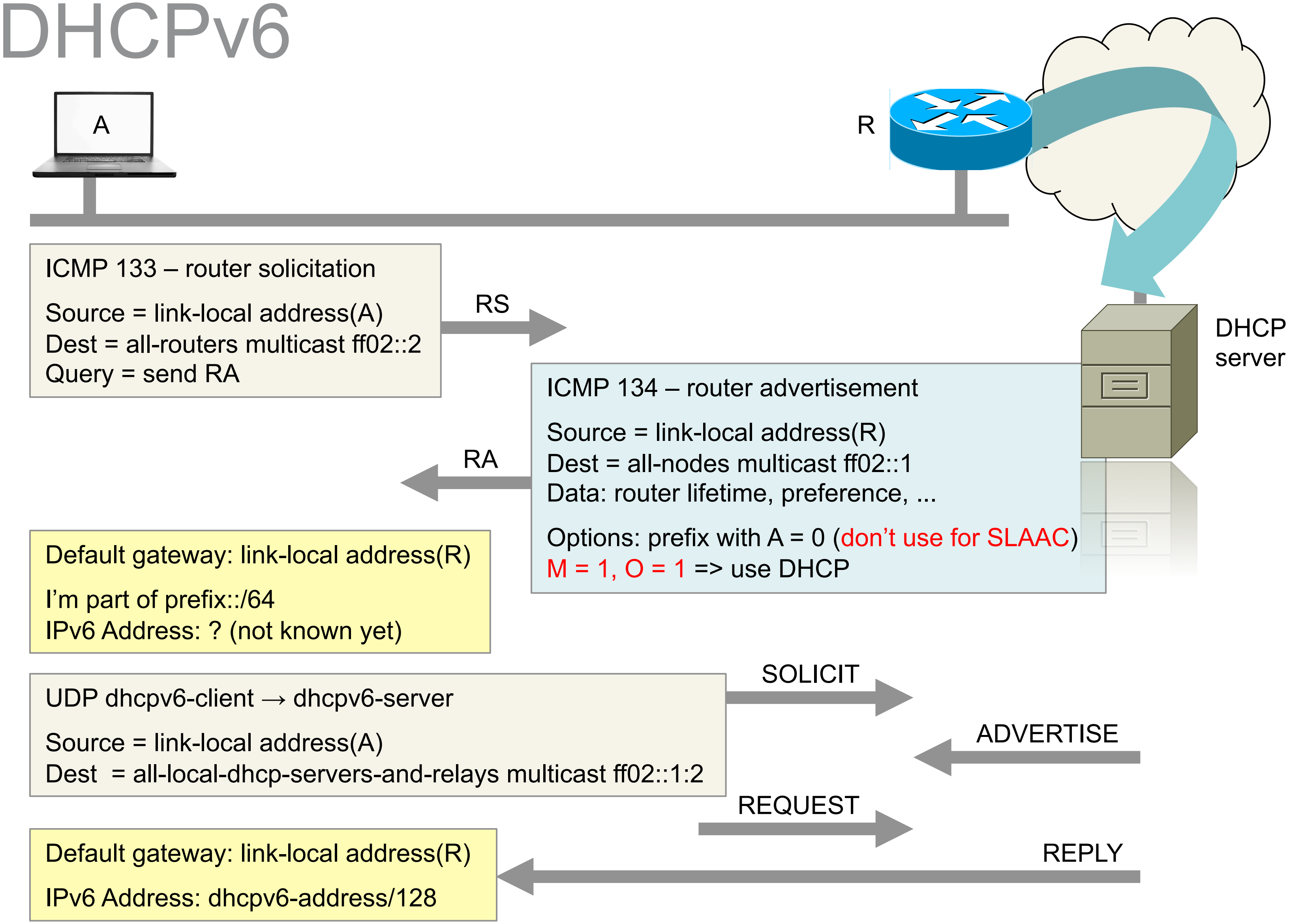
M,0=1

A=0

prefix

Server gave me the address. I know my prefix, therefore I know to which network I belong.

Router Solicitation and DHCPv6



ICMP 133 – router solicitation

Source = link-local address(A)
Dest = all-routers multicast ff02::2
Query = send RA

ICMP 134 – router advertisement

Source = link-local address(R)
Dest = all-nodes multicast ff02::1
Data: router lifetime, preference, ...

Options: prefix with A = 0 (don’t use for SLAAC)
M = 1, O = 1 => use DHCP

Default gateway: link-local address(R)

I’m part of prefix::/64
IPv6 Address: ? (not known yet)

UDP dhcpv6-client → dhcpv6-server

Source = link-local address(A)
Dest = all-local-dhcp-servers-and-relays multicast ff02::1:2

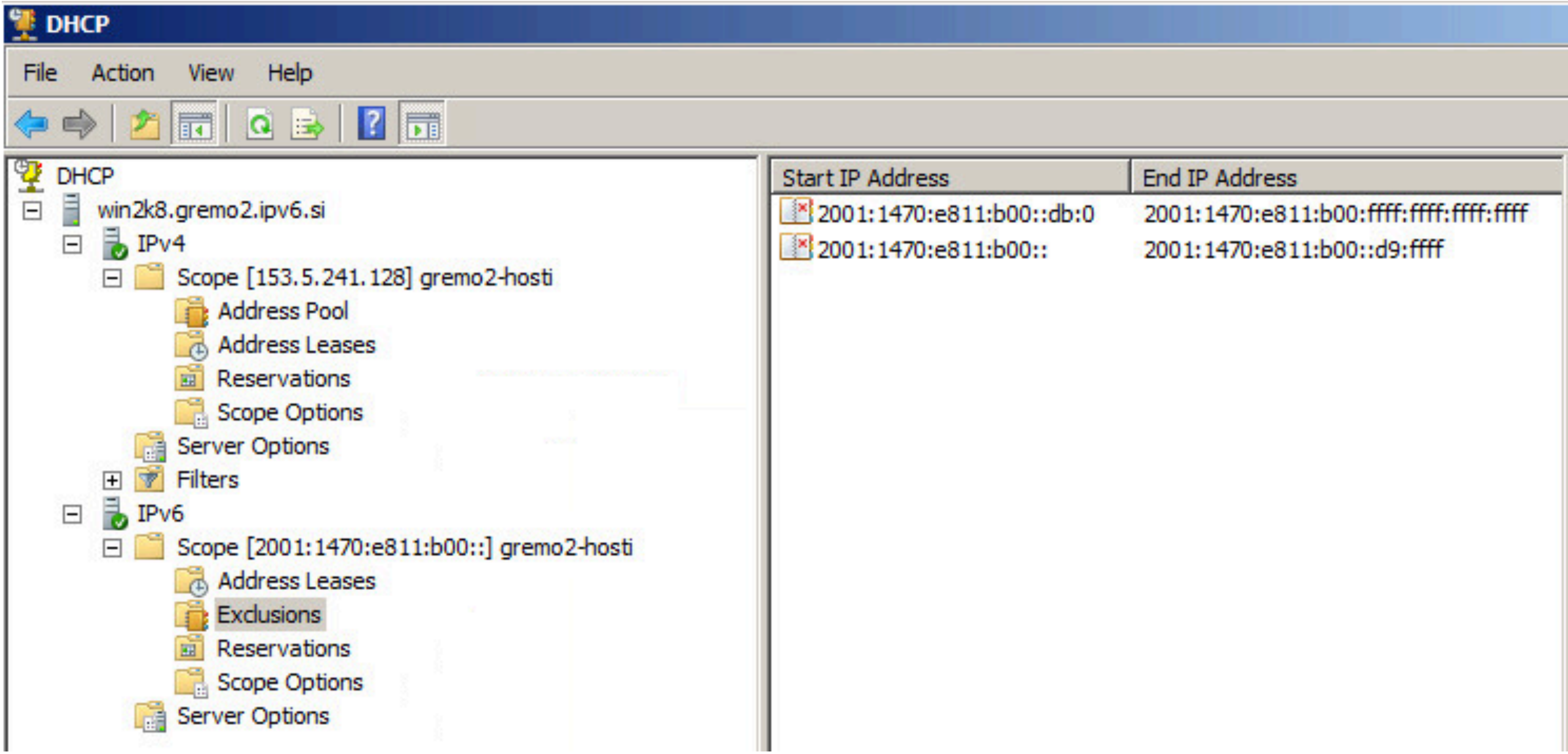
Default gateway: link-local address(R)

IPv6 Address: dhcpv6-address/128

DHCPv6 – Key takeaways

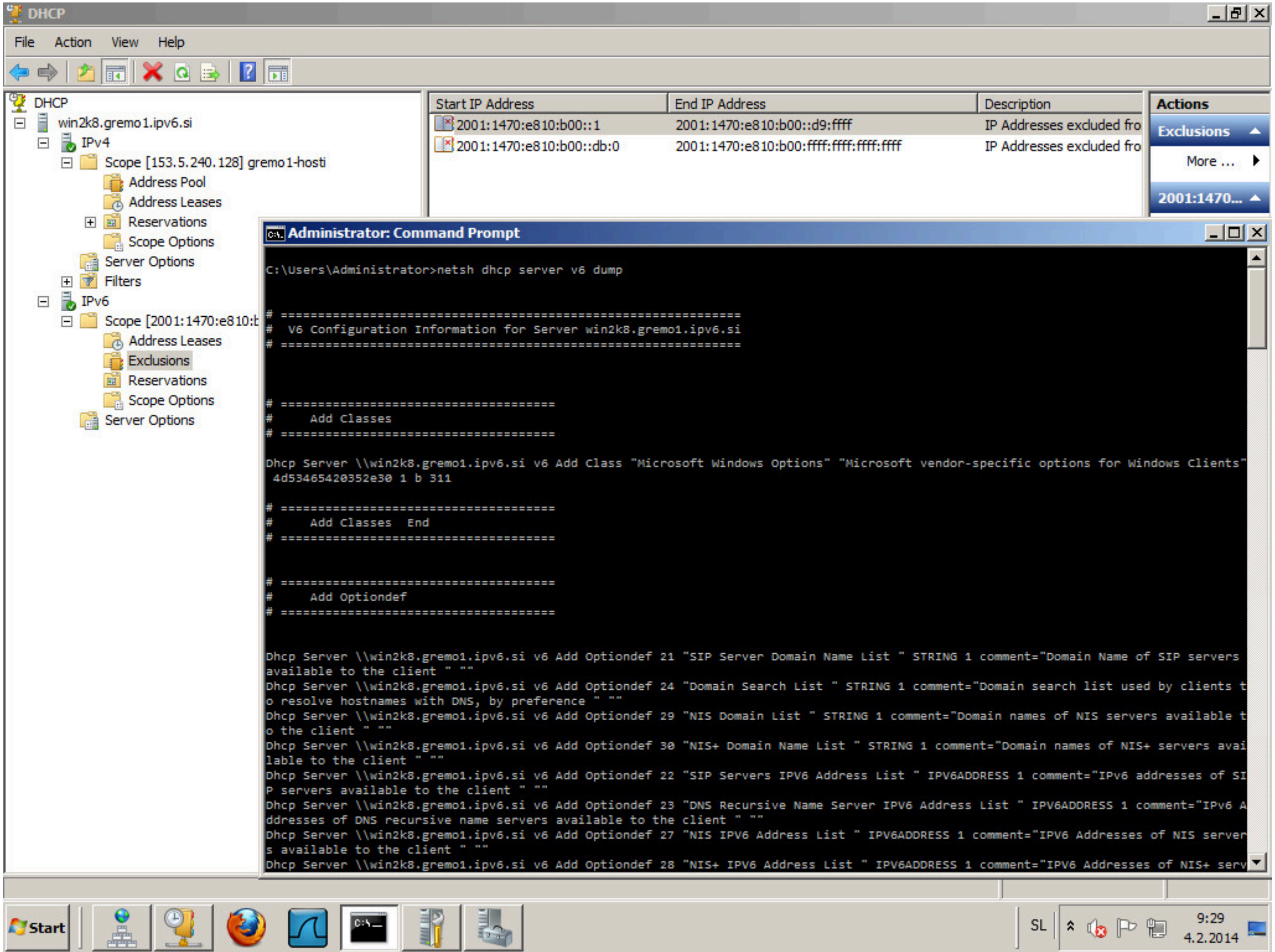
- Server and relay are part of *multicast* group ff02::1:2
- DHCPv6 does not give you the default gateway
- DHCPv6 assigns addresses only, not prefixes
- System DUID is used to identify clients and servers (instead of interface MAC)
- Relay forwards original DHCP-query
- Triggered by flags M and O in RA messages (*stateful* M=1 and *stateless* DHCP M=0)
- Dynamic updates in DNS
- Best practice: use /64

DHCPv6 – Windows 2008 Server



Start IP Address	End IP Address
2001:1470:e811:b00::db:0	2001:1470:e811:b00:ffff:ffff:ffff:ffff
2001:1470:e811:b00::	2001:1470:e811:b00::d9:ffff

DHCP on Windows Server – some screenshots



Start IP Address	End IP Address	Description
2001:1470:e810:b00::1	2001:1470:e810:b00::d9:ffff	IP Addresses excluded fro
2001:1470:e810:b00::db:0	2001:1470:e810:b00:ffff:ffff:ffff:ffff	IP Addresses excluded fro

```
C:\Users\Administrator>netsh dhcp server v6 dump


# ================================================================
#  V6 Configuration Information for Server win2k8.gremo1.ipv6.si
# ================================================================




# ======================================
#     Add Classes
# ======================================

Dhcp Server \\win2k8.gremo1.ipv6.si v6 Add Class "Microsoft Windows Options" "Microsoft vendor-specific options for Windows Clients"
 4d53465420352e30 1 b 311

# ======================================
#     Add Classes  End
# ======================================



# ======================================
#     Add Optiondef
# ======================================



Dhcp Server \\win2k8.gremo1.ipv6.si v6 Add Optiondef 21 "SIP Server Domain Name List " STRING 1 comment="Domain Name of SIP servers
available to the client " ""
Dhcp Server \\win2k8.gremo1.ipv6.si v6 Add Optiondef 24 "Domain Search List " STRING 1 comment="Domain search list used by clients t
o resolve hostnames with DNS, by preference " ""
Dhcp Server \\win2k8.gremo1.ipv6.si v6 Add Optiondef 29 "NIS Domain List " STRING 1 comment="Domain names of NIS servers available t
o the client " ""
Dhcp Server \\win2k8.gremo1.ipv6.si v6 Add Optiondef 30 "NIS+ Domain Name List " STRING 1 comment="Domain names of NIS+ servers avai
lable to the client " ""
Dhcp Server \\win2k8.gremo1.ipv6.si v6 Add Optiondef 22 "SIP Servers IPV6 Address List " IPV6ADDRESS 1 comment="IPv6 addresses of SI
P servers available to the client " ""
Dhcp Server \\win2k8.gremo1.ipv6.si v6 Add Optiondef 23 "DNS Recursive Name Server IPV6 Address List " IPV6ADDRESS 1 comment="IPv6 A
ddresses of DNS recursive name servers available to the client " ""
Dhcp Server \\win2k8.gremo1.ipv6.si v6 Add Optiondef 27 "NIS IPV6 Address List " IPV6ADDRESS 1 comment="IPV6 Addresses of NIS server
s available to the client " ""
Dhcp Server \\win2k8.gremo1.ipv6.si v6 Add Optiondef 28 "NIS+ IPV6 Address List " IPV6ADDRESS 1 comment="IPV6 Addresses of NIS+ serv
```

DHCPv6 – Example Configuration for ISC DHCP Server

/etc/dhcp/dhcpd6.conf

```
default-lease-time 2592000;
preferred-lifetime 604800;
option dhcp-renewal-time 3600;
option dhcp-rebinding-time 7200;
allow leasequery;
option dhcp6.preference 255;
option dhcp6.rapid-commit;
option dhcp6.info-refresh-time 21600;

# Static addressing for servers
subnet6 2001:1470:e812:a00::/64 { }

subnet6 2001:1470:e812:b00::/64 {
          # Range (DAA as „Dynamic Address Assignment“)
          range6 2001:1470:e812:b00::daa:0
                 2001:1470:e812:b00::daa:ffff;

          # Domain
          option dhcp6.domain-search "gremo3.ipv6.si.";

          # DNS server
          option dhcp6.name-servers 2001:1470:e812:a00::d25;
}
```

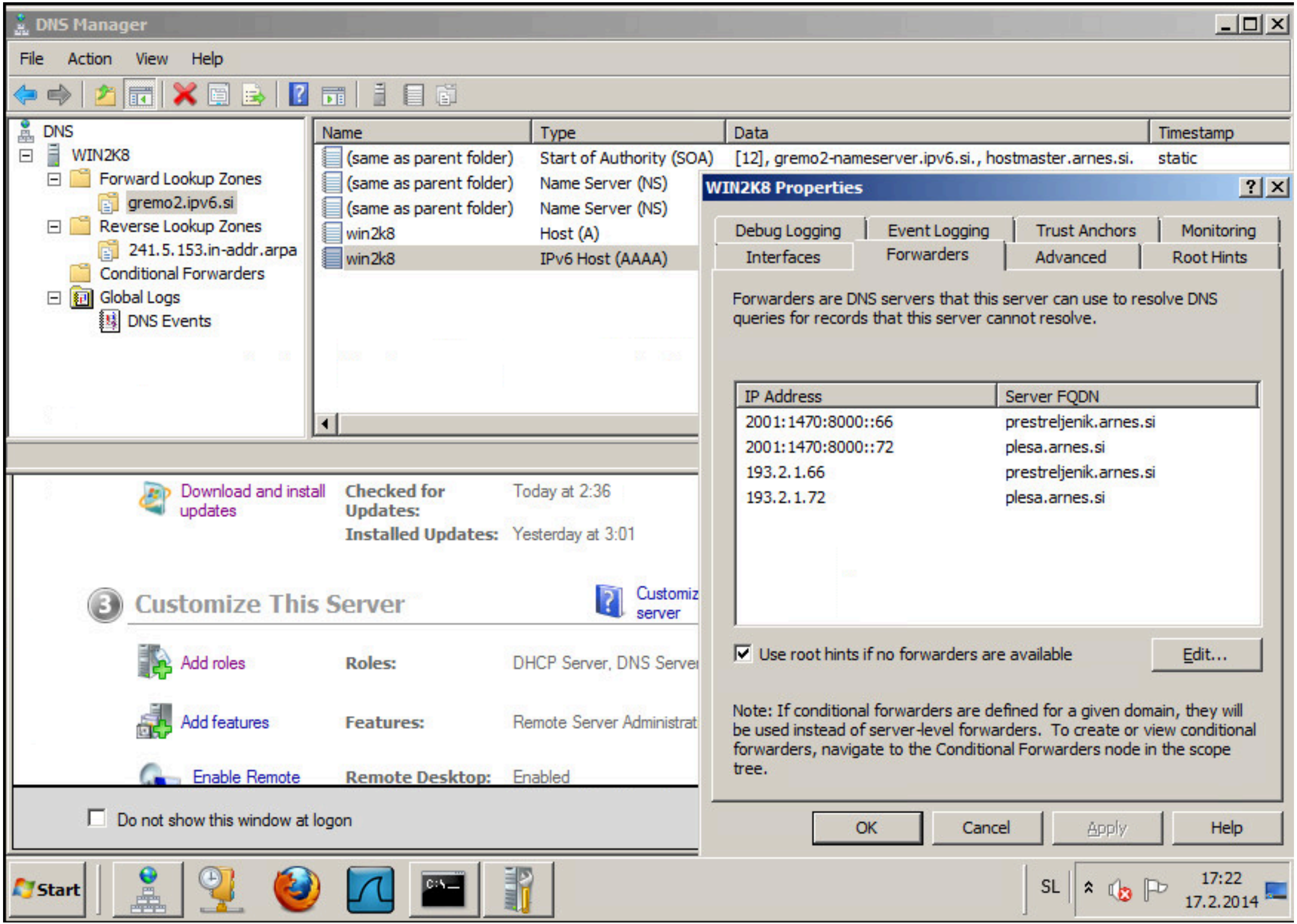
Task 5 – DHCP

- Enable DHCPv6 server
- Start Wireshark and filter ICMPv6 and dhcpv6 messages
- Configure the router withs flags M and O
- Check network settings on Windows and Linux
- Analyse ND traffic

DNS

- No big deal
- A for IPv4: name → IPv4 address
- AAAA (*quad A*): name → IPv6 address
- PTR: IP address → name
  - ip6.arpa (instead of in-addr.arpa)

DNS – Windows 2008 Server



Name	Type	Data	Timestamp
(same as parent folder)	Start of Authority (SOA)	[12], gremo2-nameserver.ipv6.si., hostmaster.arnes.si.	static
(same as parent folder)	Name Server (NS)		
(same as parent folder)	Name Server (NS)		
win2k8	Host (A)		
win2k8	IPv6 Host (AAAA)		

Download and install updates

Checked for Updates: Today at 2:36

Installed Updates: Yesterday at 3:01

3 Customize This Server

Customiz server

Add roles — Roles: DHCP Server, DNS Server

Add features — Features: Remote Server Administrat

Enable Remote — Remote Desktop: Enabled

□ Do not show this window at logon

WIN2K8 Properties

Forwarders are DNS servers that this server can use to resolve DNS queries for records that this server cannot resolve.

IP Address	Server FQDN
2001:1470:8000::66	prestreljenik.arnes.si
2001:1470:8000::72	plesa.arnes.si
193.2.1.66	prestreljenik.arnes.si
193.2.1.72	plesa.arnes.si

☑ Use root hints if no forwarders are available

Note: If conditional forwarders are defined for a given domain, they will be used instead of server-level forwarders. To create or view conditional forwarders, navigate to the Conditional Forwarders node in the scope tree.

DNS – Sample Forward Zone File

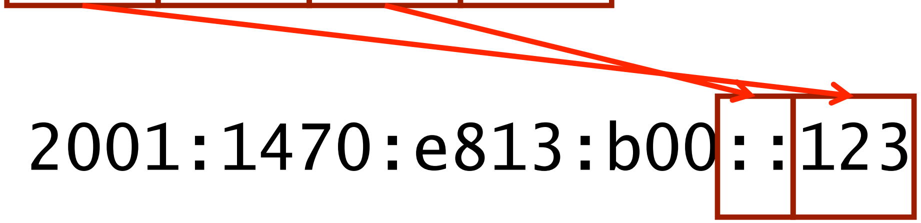
/var/named/gremo4.ipv6.si.zone

```
$ORIGIN .
$TTL 3600 ; 1 hour
gremo4.ipv6.si                IN SOA    gremo4-nameserver.ipv6.si.
                                           hostmaster.arnes.si. (
                                        1263531586 ; serial
                                        900        ; refresh (15 minutes)
                                        600        ; retry (10 minutes)
                                        86400      ; expire (1 day)
                                        3600       ; minimum (1 hour)
                                        )
                              NS        gremo4-nameserver.ipv6.si.
$ORIGIN gremo4.ipv6.si.
$TTL 1800 ; 30 minutes
lan4-windows7                 A         153.5.243.136
test                          AAAA      2001:1470:e813:b00::123
ubuntu1204                    A         153.5.243.135
windowsxp           A         153.5.243.134
```

DNS – Sample Reverse Zone File

/var/named/0.0.b.0.3.1.8.e.0.7.4.1.1.0.0.2.ip6.arpa.zone

```
$ORIGIN .
$TTL 3600 ; 1 hour

0.0.b.0.3.1.8.e.0.7.4.1.1.0.0.2.ip6.arpa  IN SOA  gremo4-nameserver.ipv6.si.
                                                    hostmaster.arnes.si. (
              1263527840 ; serial
              900        ; refresh (15 minutes, was 3 hours/10800)
              600        ; retry (10 minutes, was 1 hour/3600)
              86400      ; expire (1 day, was 1 week/604800)
              3600       ; minimum/negative TTL (1 hour, was 10 hours 40 minutes/38400)
              )

          NS  gremo4-nameserver.ipv6.si.

$ORIGIN 0.0.b.0.3.1.8.e.0.7.4.1.1.0.0.2.ip6.arpa.
$TTL 900    ; 15 minutes
3.2.1.0.0.0.0.0.0.0.0.0.0.0.0.0         PTR       test.gremo4.ipv6.si.
```



2001:1470:e813:b00::123

Task 6 – DNS

- Configure DNS for IPv6
- Add a test AAAA in PTR record
  - test6.go6.example.org = 2001:db8:6::abcd
- Verify with *dig*, *host* or *nslookup*

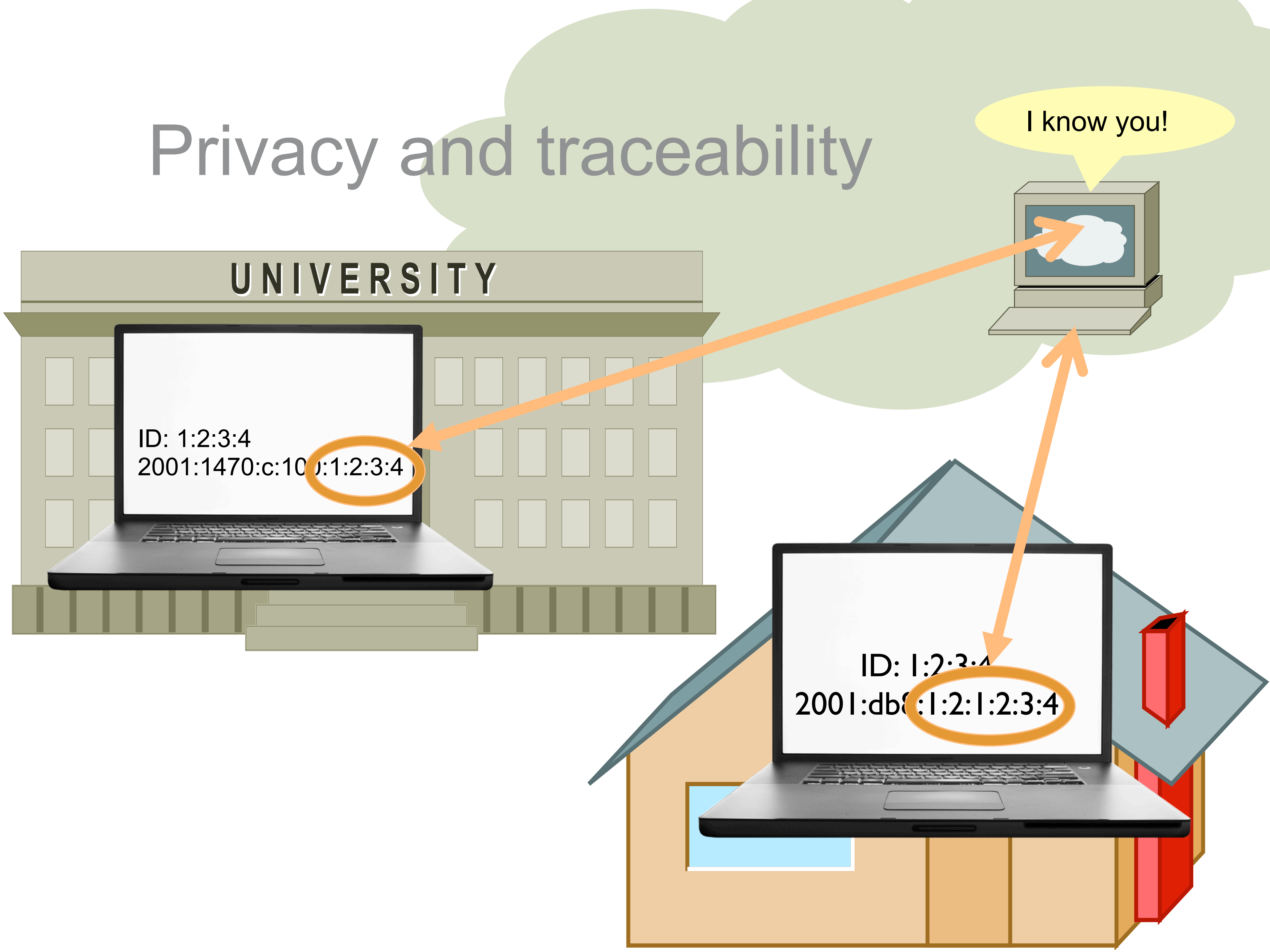
Topics

- IPv6 Address
- ICMP and Automatic Configuration
- Fundamental Network Services
- Basic Security Considerations
- A Peek at Transition mechanisms

Security

- Privacy and traceability
- Security at First Hop
  (securing link-local operations – ND)

Privacy and traceability



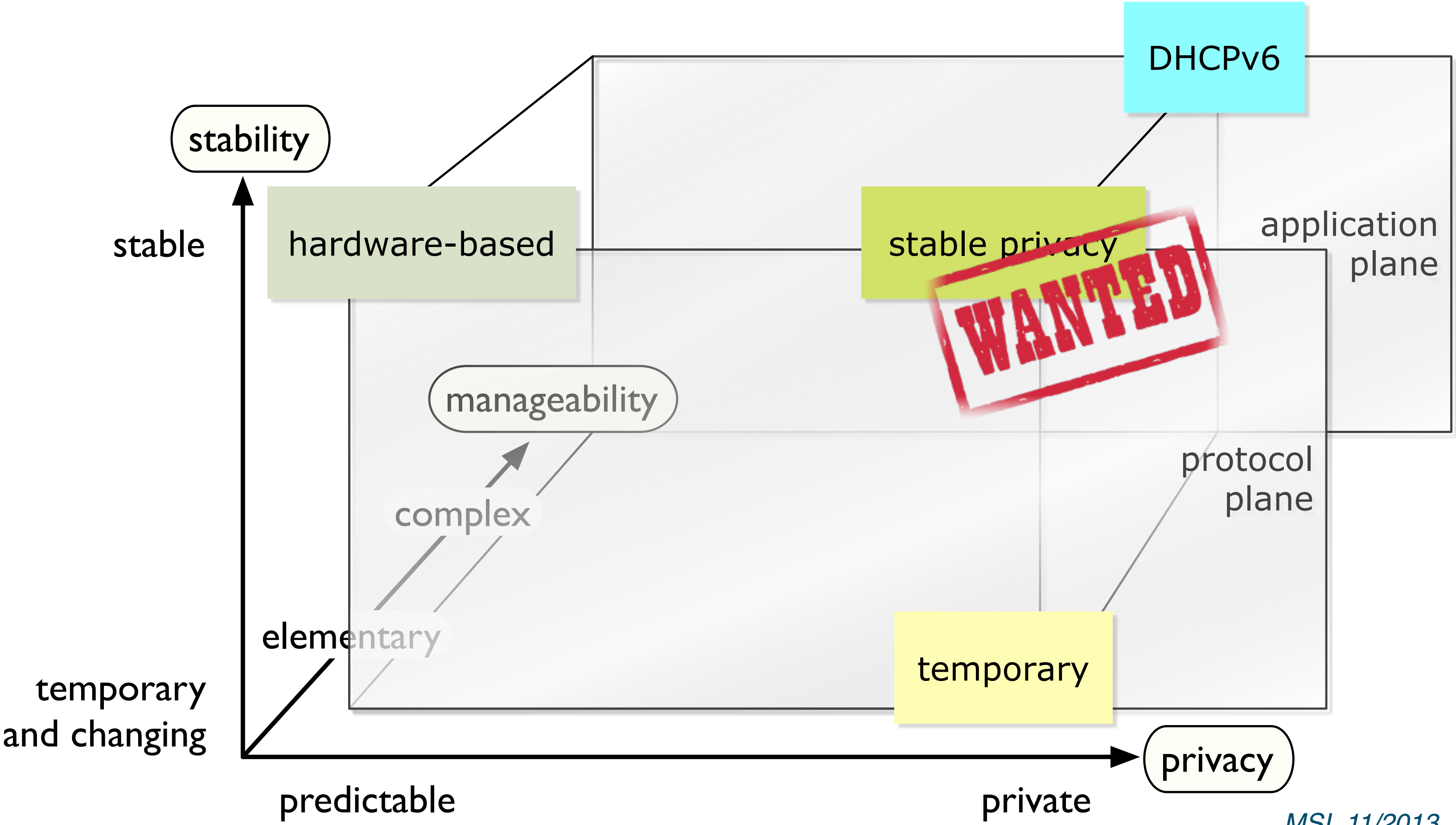
Privacy and traceability

- Hardware based address (*EUI-64 based*) is traceable
  - Privacy *vs* security?
    - Fine, no anonymous systems in LAN. Traceable (identifiable, trackable).
    - But ...
      is this really OK and secure enough?

Privacy and traceability

- Random and temporary addresses
  IPv6 Privacy Extensions
- Good for privacy
- ...but unwanted in LAN
- How to disable anonymous address usage in LAN?
  - ...or, at least, monitor and LOG (audit)

The "Privacy-stability-manageability" Cube

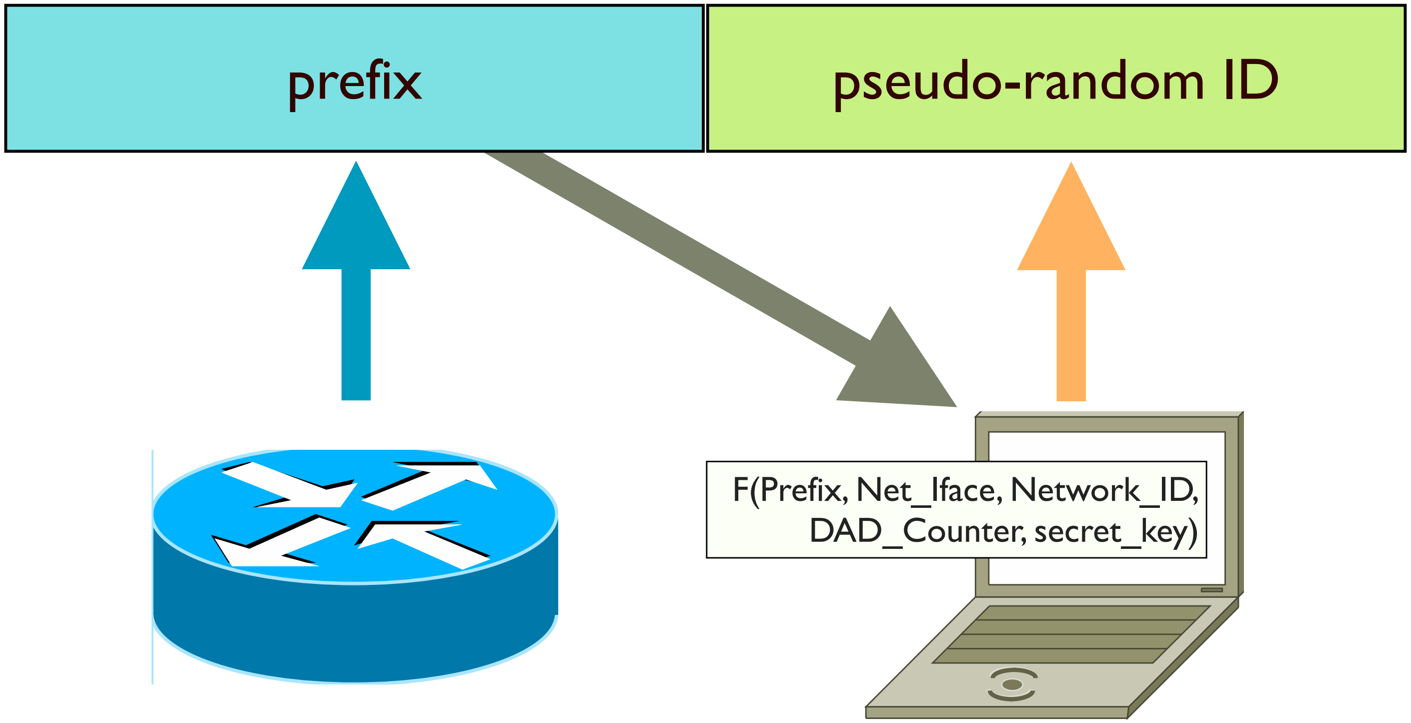


MSI, 11/2013

http://njetwork.wordpress.com/2013/11/03/to-slaac-or-not-to-slaac/

Privacy and traceability – Stable Privacy Address

`2001:db8:f00d:15::6a49:536e:76b8:2ce8`



prefix

pseudo-random ID

F(Prefix, Net_Iface, Network_ID, DAD_Counter, secret_key)

A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC), draft-ietf-6man-stable-privacy-addresses-14

Privacy and traceability

- A modest advise:
  - Use DHCP and allow only addresses from a fixed pool
    - Filter with ACLs
  - Disable SLAAC
    - Configure the router properly (mind the A-flag)
    - Additional security measure:
      - Disable privacy extensions
      - Disable tunnels

Task 7 – Disable SLAAC

- Configure the router with A = 0 (`no-autoconfig`)
- Hosts should not use prefix for SLAAC
- Verify that

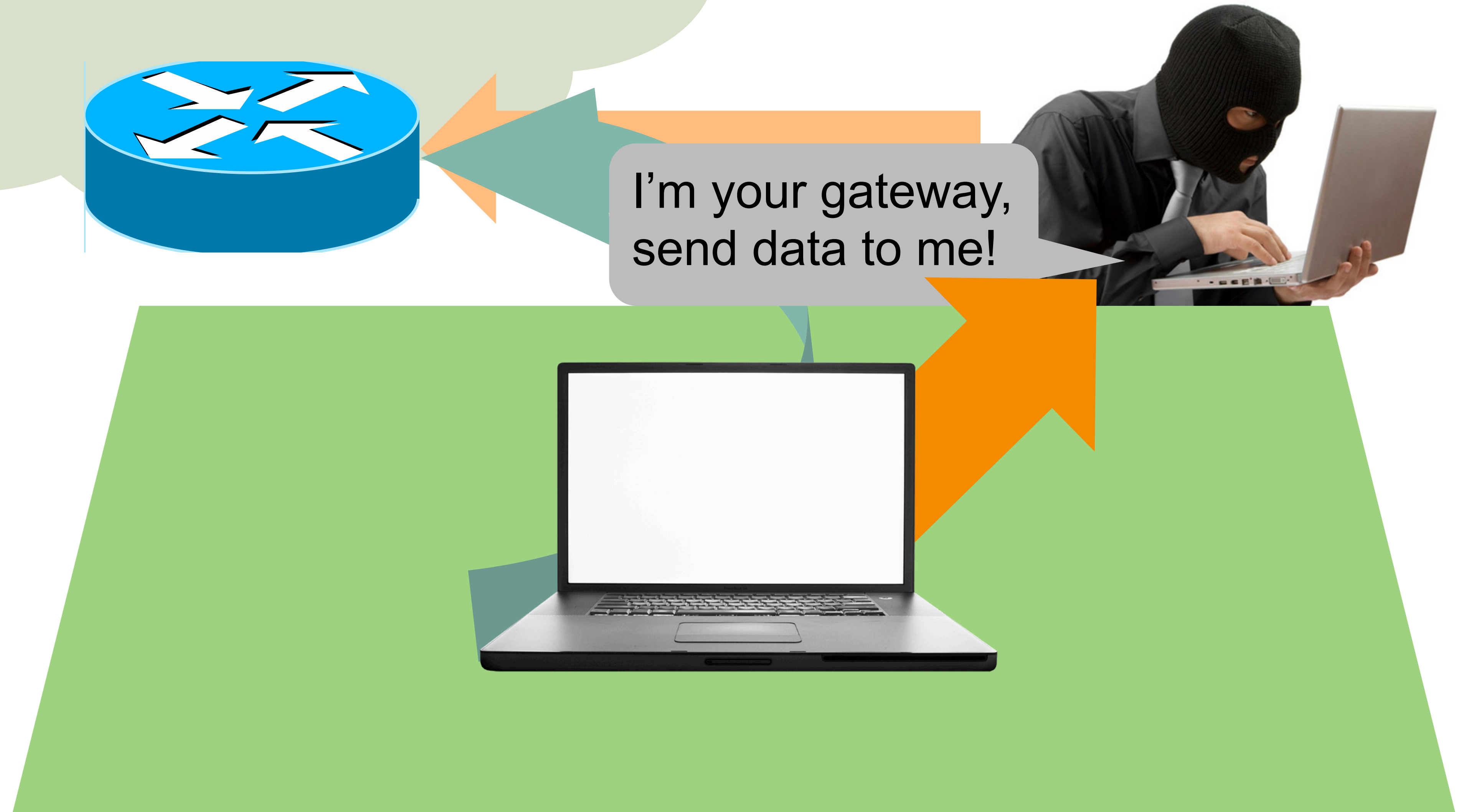
Security at First Hop

- ND is a vulnerable protocol
  - Fake RA
  - Fake DHCPv6
  - NDP spoofing
- DoS
  - ND cache overflow
  - DAD attacks
  - DHCPv6 attacks

Security at First Hop

- Tools
  - The Hacker's Choice https://www.thc.org/thc-ipv6/
  - SI6 Networks' IPv6 Toolkit http://www.si6networks.com/tools/ipv6toolkit/
  - nmap + NSE scripts http://nmap.org/nsedoc/scripts/

Security at First Hop – Example
Rogue Router



Security at First Hop

- Some vendors already offer FHS:
  - RA Guard
  - DHCPv6 Guard
  - Snooping and device tracking (binding integrity guard)
  - Source Guard
  - Destination Guard
  - Prefix Guard

http://docwiki.cisco.com/wiki/FHS

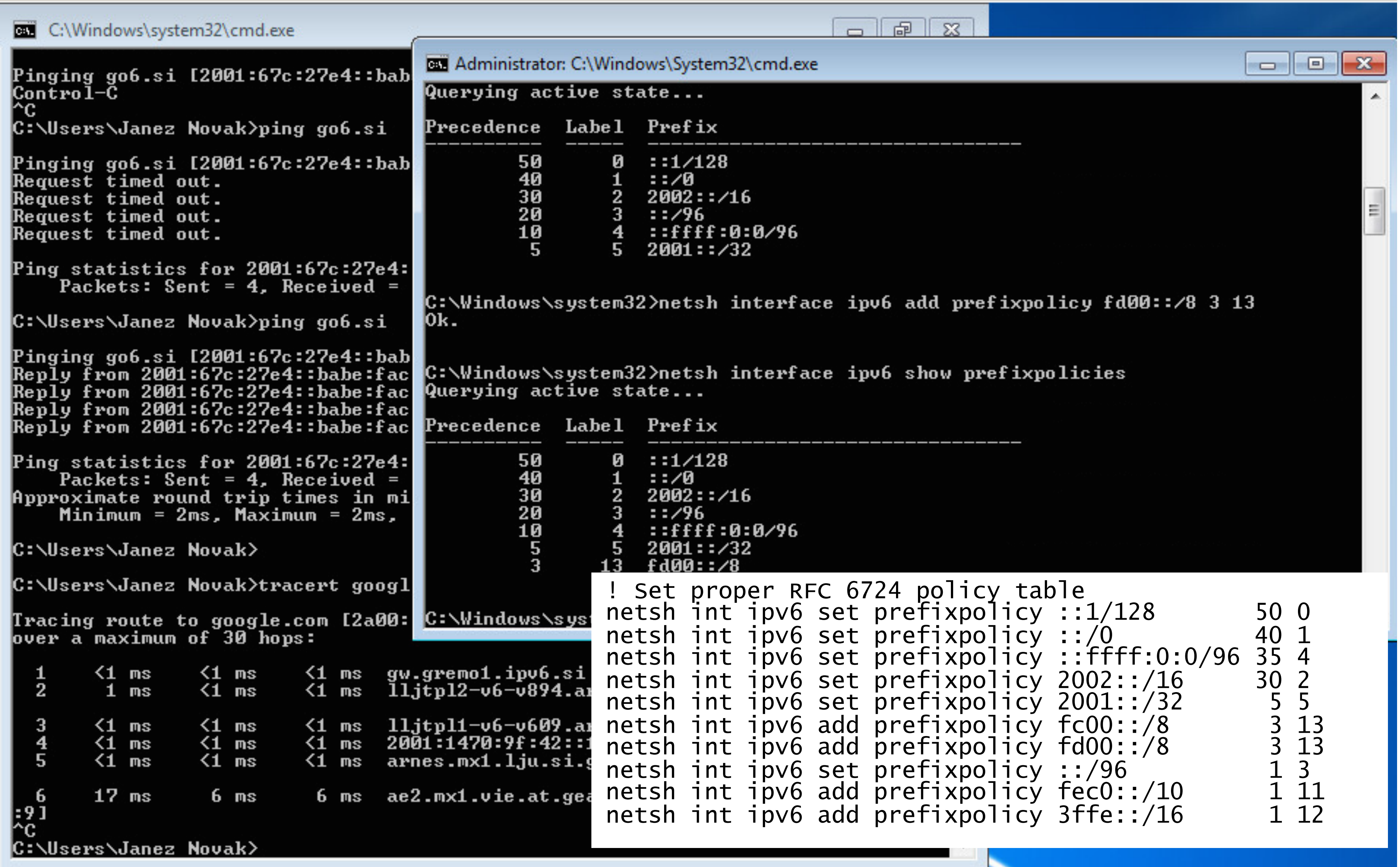
Security at First Hop – basic requirements in public tenders

- Surveil and filter announcements from local IPv6 routers - *Router Advertisement (RA) guard [RFC6105]*
- Filter DHCPv6 traffic
- Monitor IPv6 ND traffic (Neighbor Discovery) and DHCPv6 in local network and dynamically maintain the list of registered IPv6 systems (ND table or "IPv6 First-Hop Security binding table") – *Dynamic IPv6 Neighbor solicitation/advertisement inspection*
- Block traffic from sources which are not registered in the ND table – *IPv6 Source Guard*

Task 8 – ULA

- Add ULA addresses on LAN
- Verify the source address selection algorithm
    - Use
        - `ping 2001:db8:6::1`
        - `ping fd00:db8:6::1`
    - ...and check with Wireshark

Policy on your OS might not be fully compliant with the RFC 6724 – ULA on Windows 7



```
Pinging go6.si [2001:67c:27e4::bab
Control-C
^C
C:\Users\Janez Novak>ping go6.si

Pinging go6.si [2001:67c:27e4::bab
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 2001:67c:27e4:
    Packets: Sent = 4, Received =

C:\Users\Janez Novak>ping go6.si

Pinging go6.si [2001:67c:27e4::bab
Reply from 2001:67c:27e4::babe:fac
Reply from 2001:67c:27e4::babe:fac
Reply from 2001:67c:27e4::babe:fac
Reply from 2001:67c:27e4::babe:fac

Ping statistics for 2001:67c:27e4:
    Packets: Sent = 4, Received =
Approximate round trip times in mi
    Minimum = 2ms, Maximum = 2ms,

C:\Users\Janez Novak>

C:\Users\Janez Novak>tracert googl

Tracing route to google.com [2a00:
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  gw.gremo1.ipv6.si
  2     1 ms    <1 ms    <1 ms  lljtpl2-v6-v894.ar

  3    <1 ms    <1 ms    <1 ms  lljtpl1-v6-v609.ar
  4    <1 ms    <1 ms    <1 ms  2001:1470:9f:42::1
  5    <1 ms    <1 ms    <1 ms  arnes.mx1.lju.si.g

  6    17 ms     6 ms     6 ms  ae2.mx1.vie.at.gea
:9]
^C
C:\Users\Janez Novak>
```

```
Querying active state...

Precedence  Label  Prefix
----------  -----  --------------------------------
        50      0  ::1/128
        40      1  ::/0
        30      2  2002::/16
        20      3  ::/96
        10      4  ::ffff:0:0/96
         5      5  2001::/32


C:\Windows\system32>netsh interface ipv6 add prefixpolicy fd00::/8 3 13
Ok.


C:\Windows\system32>netsh interface ipv6 show prefixpolicies
Querying active state...

Precedence  Label  Prefix
----------  -----  --------------------------------
        50      0  ::1/128
        40      1  ::/0
        30      2  2002::/16
        20      3  ::/96
        10      4  ::ffff:0:0/96
         5      5  2001::/32
         3     13  fd00::/8


C:\Windows\sys
```

```
! Set proper RFC 6724 policy table
netsh int ipv6 set prefixpolicy ::1/128       50 0
netsh int ipv6 set prefixpolicy ::/0          40 1
netsh int ipv6 set prefixpolicy ::ffff:0:0/96 35 4
netsh int ipv6 set prefixpolicy 2002::/16     30 2
netsh int ipv6 set prefixpolicy 2001::/32      5 5
netsh int ipv6 add prefixpolicy fc00::/8       3 13
netsh int ipv6 add prefixpolicy fd00::/8       3 13
netsh int ipv6 set prefixpolicy ::/96          1 3
netsh int ipv6 add prefixpolicy fec0::/10      1 11
netsh int ipv6 add prefixpolicy 3ffe::/16      1 12
```

Topics

- IPv6 Address
- ICMP and Automatic Configuration
- Fundamental Network Services
- Basic Security Considerations
- A Peek at Transition mechanisms

Transition Mechanisms

- Tunnels
  - 6in4, 6to4 (IPv6-over-IPv4)
  - 6rd (IPv6-over-IPv4, *stateless*)
  - DS-Lite (IPv4-over-IPv6, *stateful*)
- MAP – Mapping of Address and Port (*stateless*)
- Translation
  - *stateless* NAT64
  - NAT64/DNS64 (*stateful*)

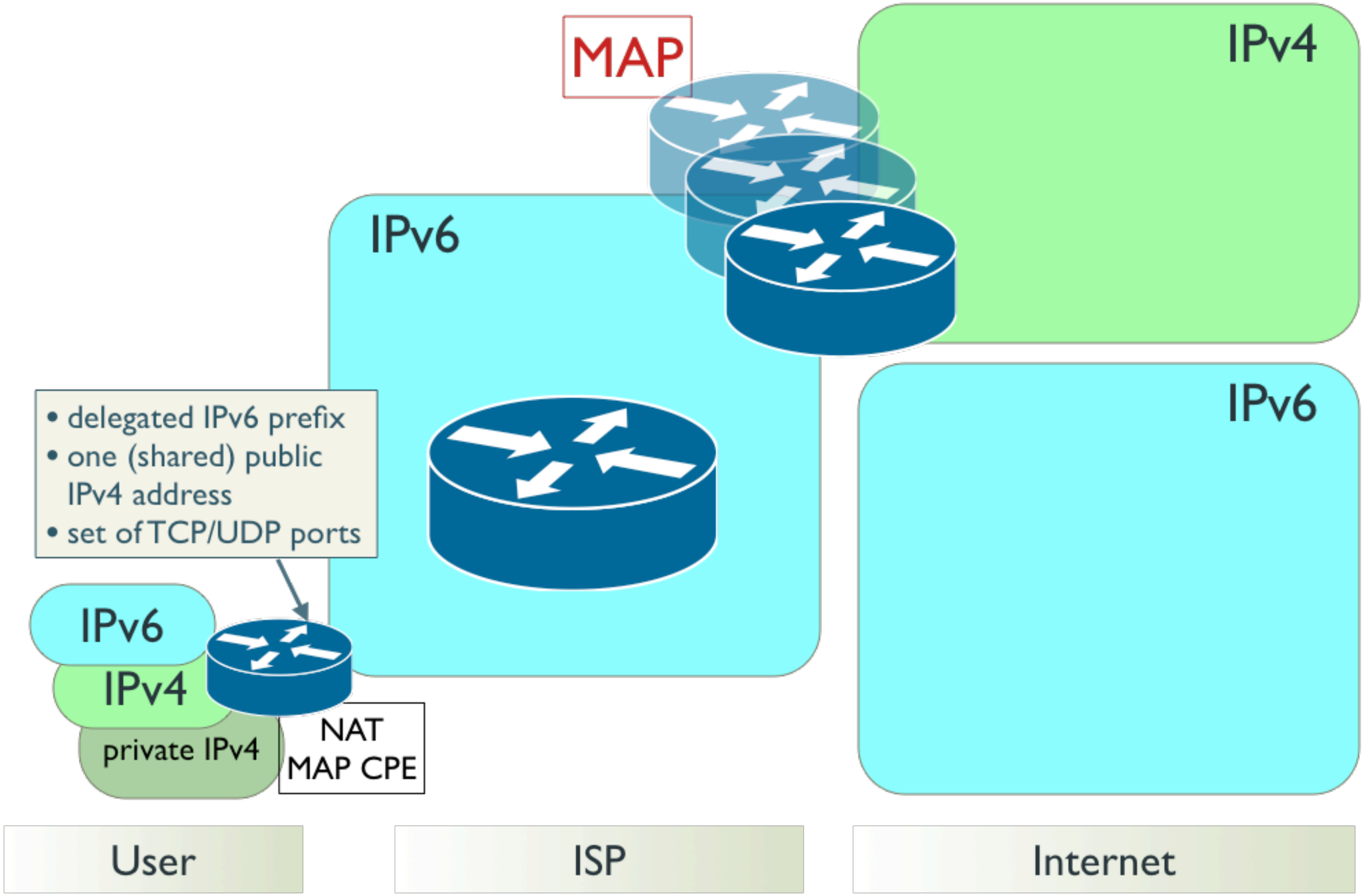
Transition Mechanisms

- Tunnels
  - 6in4, 6to4 (IPv6-over-IPv4)
  - 6rd (IPv6-over-IPv4, *stateless*)
  - DS-Lite (IPv4-over-IPv6, *stateful*)
- MAP – Mapping of Address and Port (*stateless*)
- Translation
  - *stateless* NAT64
  - NAT64/DNS64 (*stateful*)

Transition mechanisms – which technology is the right one?

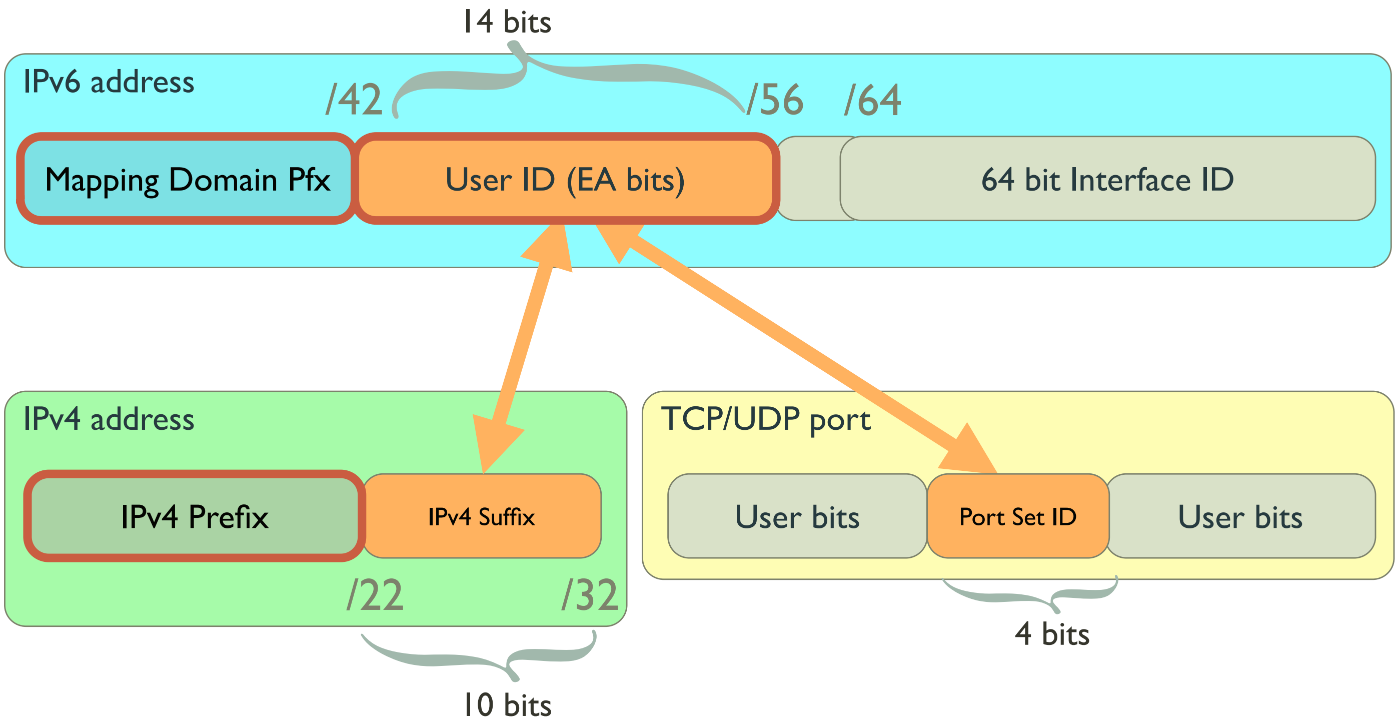
- Ask yourself:
  - Is IPv6 preferred?
  - Can we eliminate IPv4 in the long run?
  - Does it maintain the quality during the transition period?
  - Is it expandable, upgradable, scalable?
  - How much €?

Transition mechanisms – MAP



Transition mechanisms – MAP

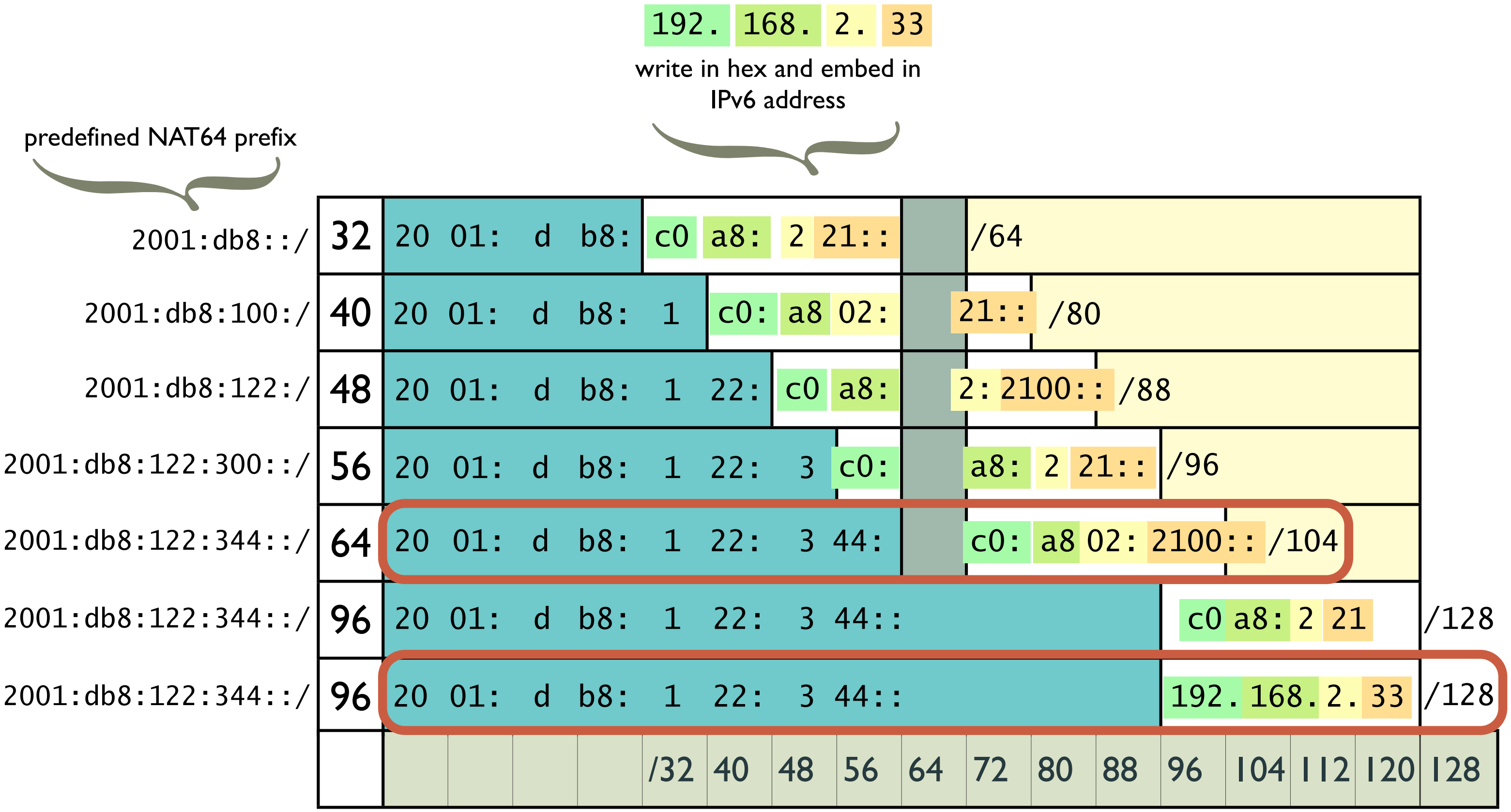
Mapping IPv6 address ⟷ IPv4 address + ports



http://map46.cisco.com/MAP.php

Transition mechanisms – NAT64

mapping IPv6 address ⟷ IPv4 address



192. 168. 2. 33

write in hex and embed in IPv6 address

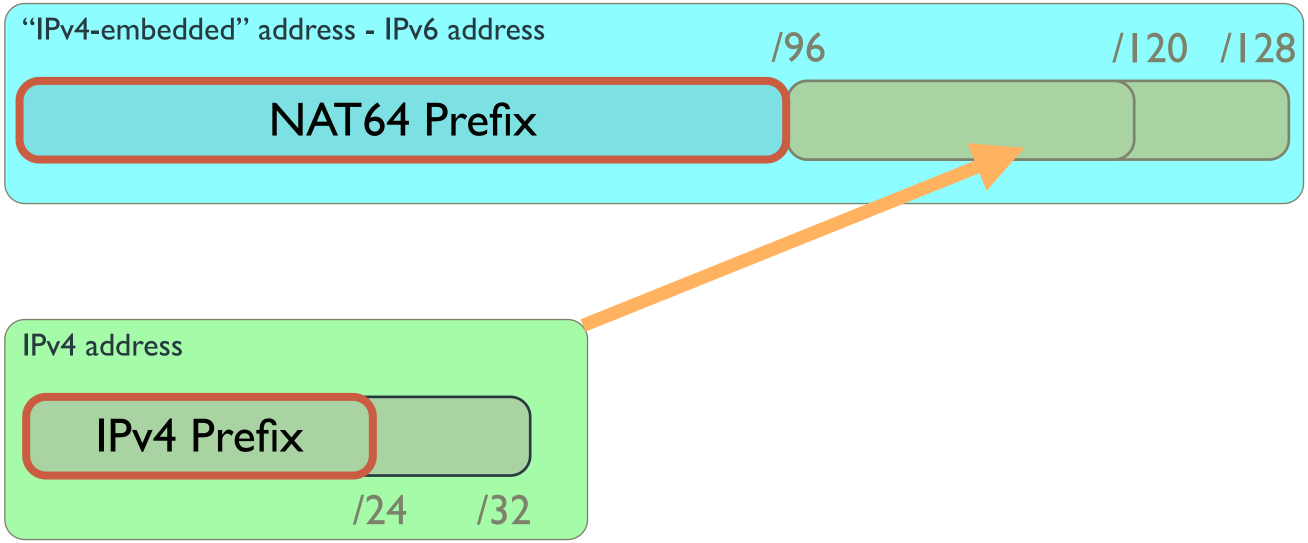
predefined NAT64 prefix

2001:db8::/	32	20 01: d b8:	c0 a8: 2 21::		/64
2001:db8:100:/	40	20 01: d b8: 1	c0: a8 02:	21::	/80
2001:db8:122:/	48	20 01: d b8: 1 22:	c0: a8:	2: 2100::	/88
2001:db8:122:300::/	56	20 01: d b8: 1 22: 3	c0:	a8: 2 21::	/96
2001:db8:122:344::/	64	20 01: d b8: 1 22: 3 44:		c0: a8 02: 2100::	/104
2001:db8:122:344::/	96	20 01: d b8: 1 22: 3 44::		c0 a8: 2 21	/128
2001:db8:122:344::/	96	20 01: d b8: 1 22: 3 44::		192. 168. 2. 33	/128

/32 40 48 56 64 72 80 88 96 104 112 120 128

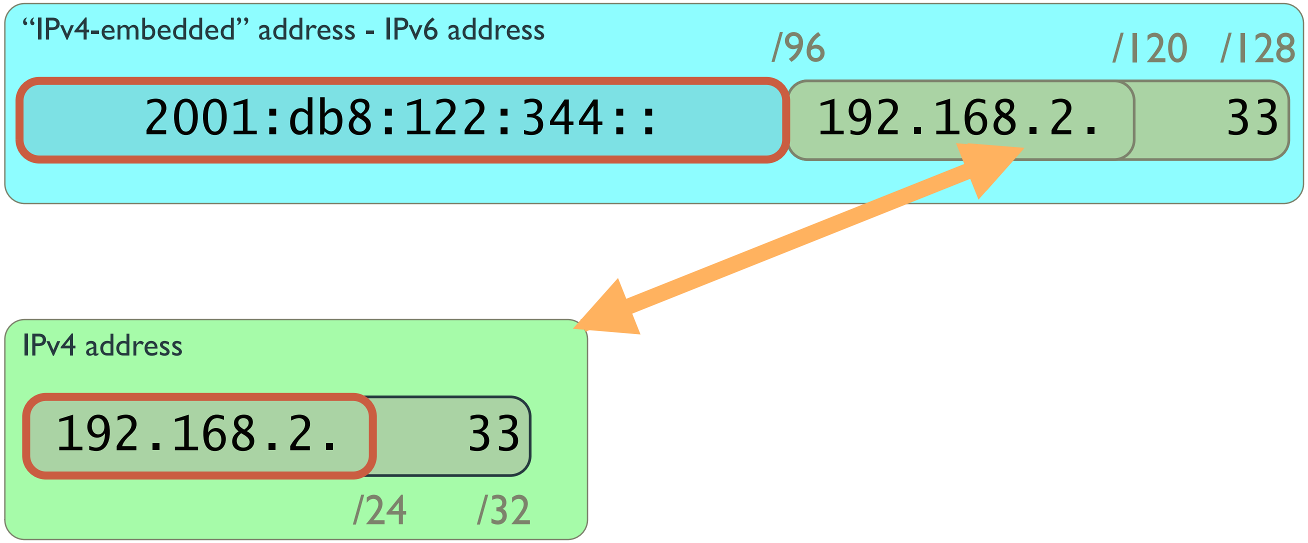
Transition mechanisms – NAT64

embedding IPv4 address in IPv6 address

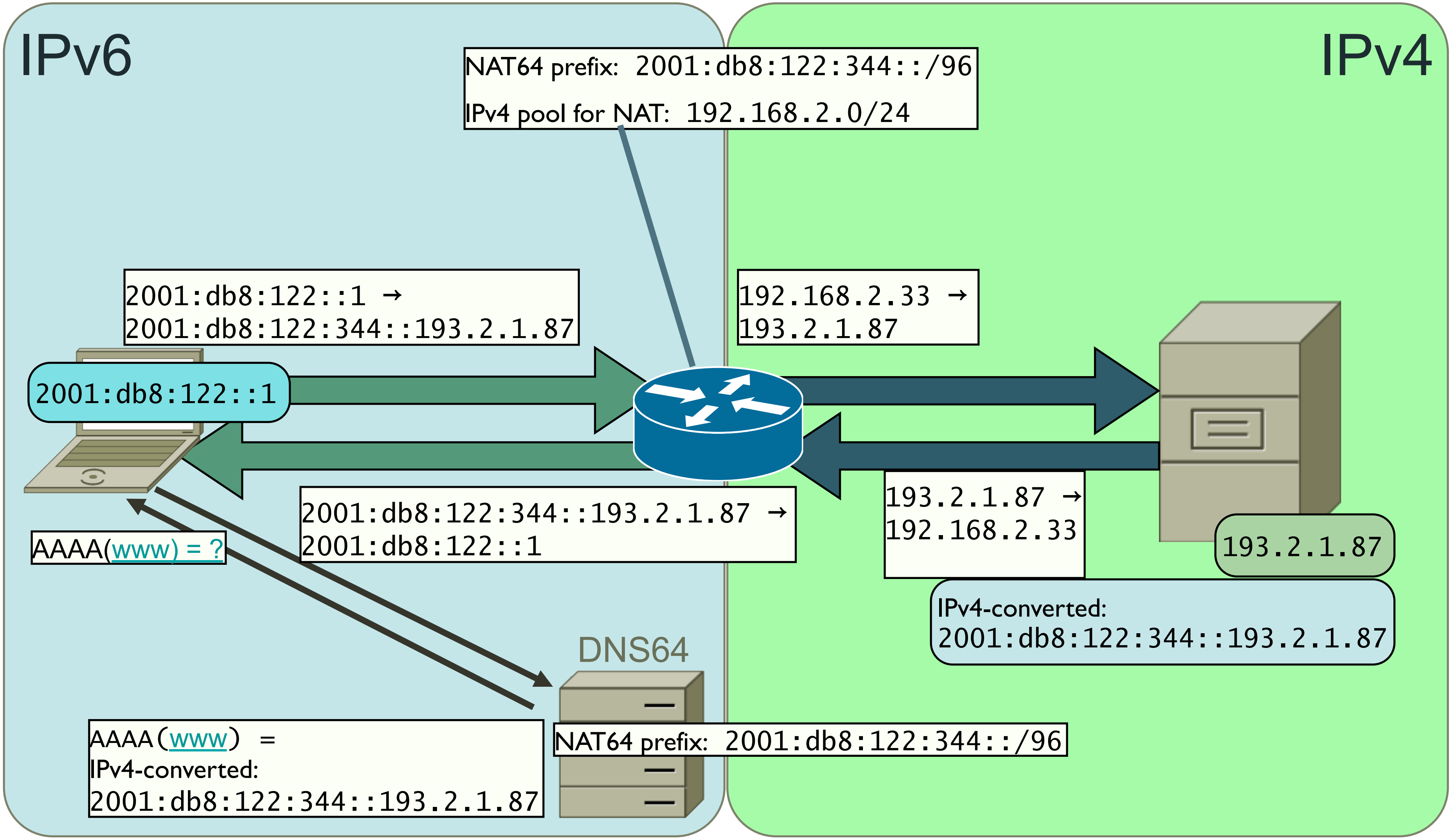


Transition mechanisms – NAT64

embedding IPv4 address in IPv6 address



Transition mechanisms – *stateful* NAT64 in front of IPv6-only clients



IPv6

IPv4

NAT64 prefix: 2001:db8:122:344::/96

IPv4 pool for NAT: 192.168.2.0/24

2001:db8:122::1 →
2001:db8:122:344::193.2.1.87

192.168.2.33 →
193.2.1.87

2001:db8:122::1

2001:db8:122:344::193.2.1.87 →
2001:db8:122::1

193.2.1.87 →
192.168.2.33

193.2.1.87

IPv4-converted:
2001:db8:122:344::193.2.1.87

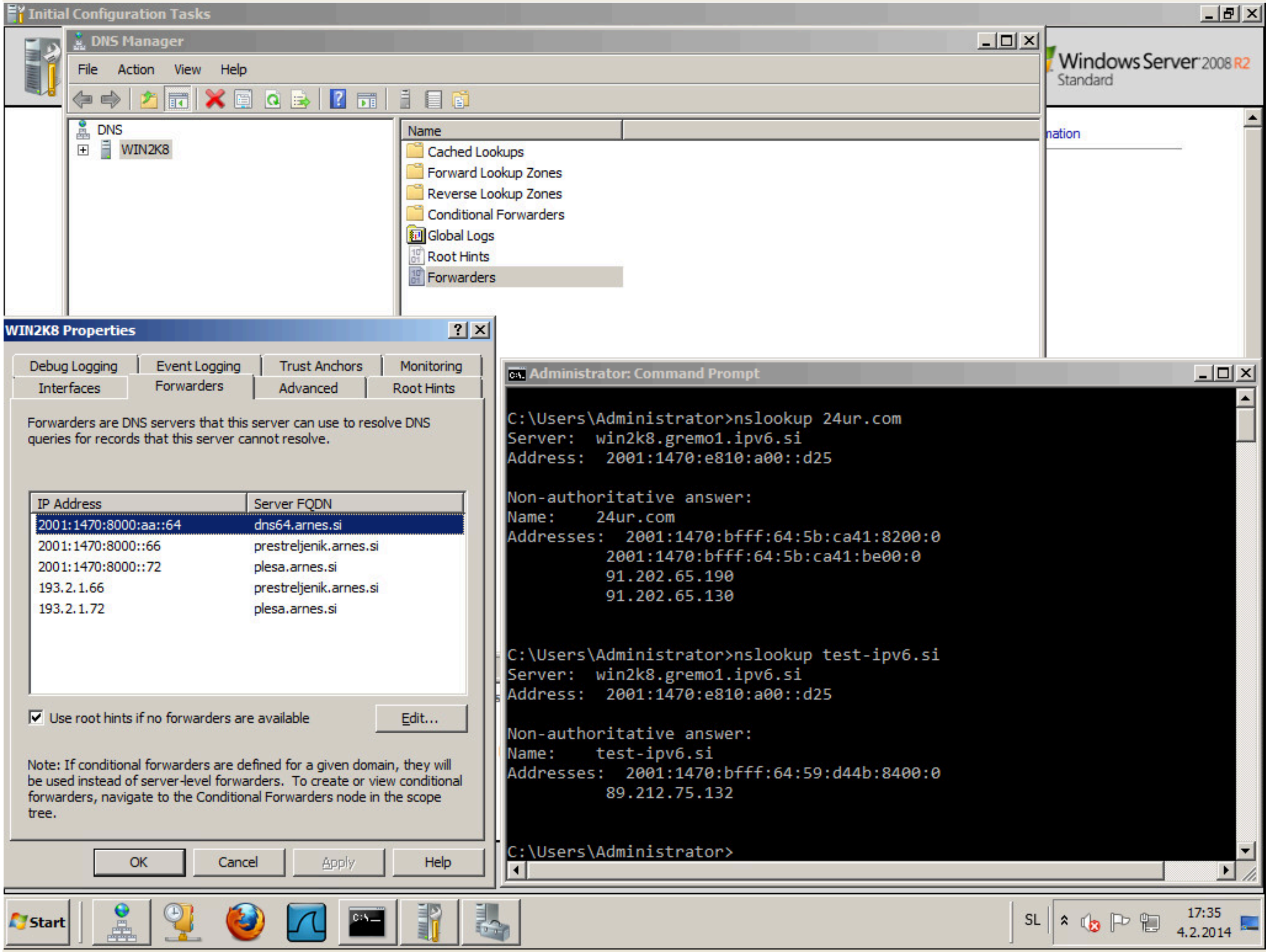
AAAA(www) = ?

DNS64

AAAA(www) =
IPv4-converted:
2001:db8:122:344::193.2.1.87

NAT64 prefix: 2001:db8:122:344::/96

DNS64 – Example:
forwarders for Windows 2008 Server



Forwarders are DNS servers that this server can use to resolve DNS queries for records that this server cannot resolve.

IP Address	Server FQDN
2001:1470:8000:aa::64	dns64.arnes.si
2001:1470:8000::66	prestreljenik.arnes.si
2001:1470:8000::72	plesa.arnes.si
193.2.1.66	prestreljenik.arnes.si
193.2.1.72	plesa.arnes.si

☑ Use root hints if no forwarders are available

Edit...

Note: If conditional forwarders are defined for a given domain, they will be used instead of server-level forwarders. To create or view conditional forwarders, navigate to the Conditional Forwarders node in the scope tree.

OK Cancel Apply Help

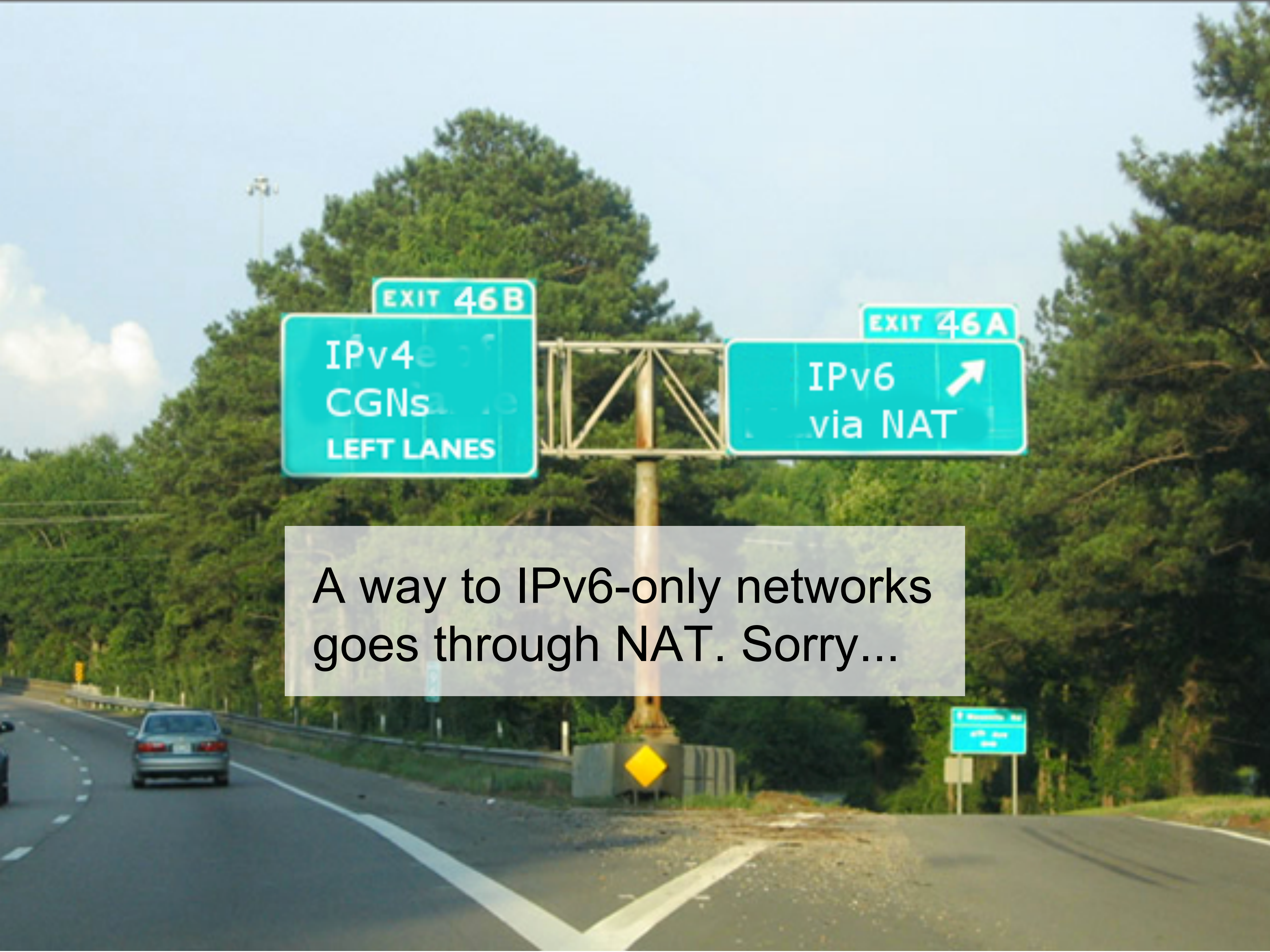
```
C:\Users\Administrator>nslookup 24ur.com
Server:  win2k8.gremo1.ipv6.si
Address:  2001:1470:e810:a00::d25

Non-authoritative answer:
Name:    24ur.com
Addresses:  2001:1470:bfff:64:5b:ca41:8200:0
          2001:1470:bfff:64:5b:ca41:be00:0
          91.202.65.190
          91.202.65.130


C:\Users\Administrator>nslookup test-ipv6.si
Server:  win2k8.gremo1.ipv6.si
Address:  2001:1470:e810:a00::d25

Non-authoritative answer:
Name:    test-ipv6.si
Addresses:  2001:1470:bfff:64:59:d44b:8400:0
          89.212.75.132


C:\Users\Administrator>
```

Task 9 (optional) – DNS64

NAT64/DNS64 public test

http://go6lab.si/current-ipv6-tests/nat64dns64-public-test/

- DNS64
  - BIND9 2001:67c:27e4::60
- NAT64 – various implementations:
  - Cisco ASR 1k
  - PaloAlto Networks
  - Ecdysis
- Ask DNS64 for AAAA of IP4-only servers
- Figure out the NAT64 prefix



A way to IPv6-only networks goes through NAT. Sorry...

Thank you!

Aviat NETWORKS

matjaz@njetwork.si