

# IPv6 Security

27th ANNUAL  
**FIRST BERLIN**  
CONFERENCE  
14-19 JUNE 2015



# SWITCH

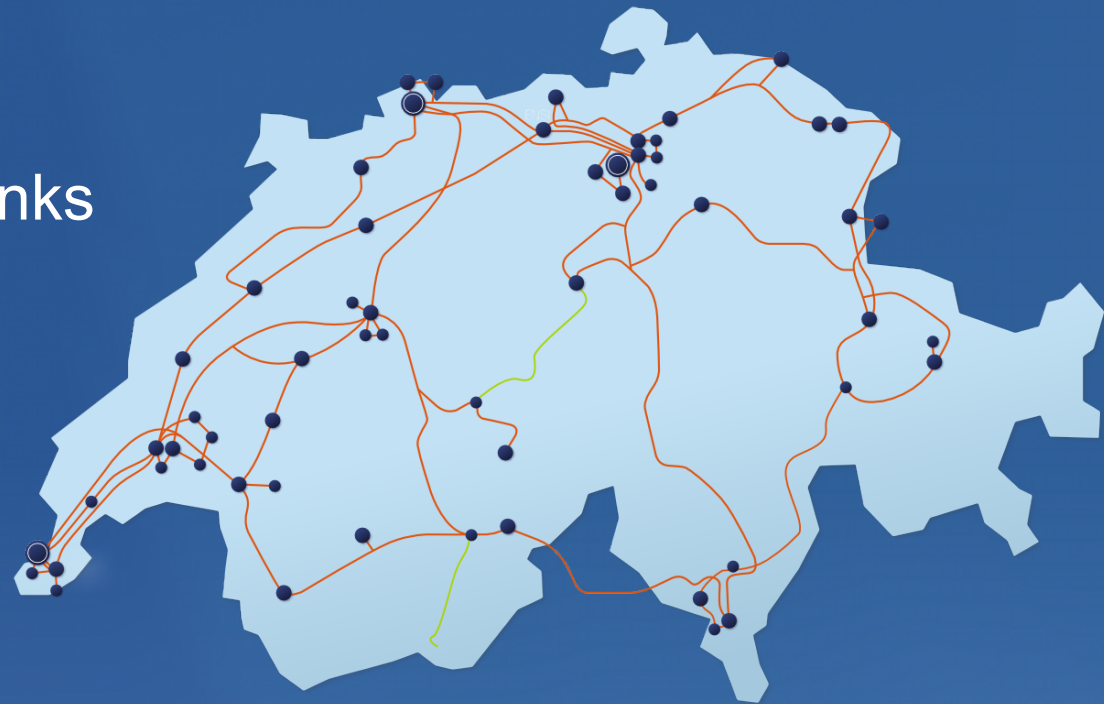
Frank Herberg  
[frank.herberg@switch.ch](mailto:frank.herberg@switch.ch)

Berlin, 18 June 2015



# SWITCH Security

- 12 employees
- Operates SWITCH-CERT
- Main customers:
  - NREN CH/LI
  - Registry CH/LI
  - Some Swiss Banks





# Agenda

- **Warm-up:** A (very) short introduction to IPv6
- **Part 1: Introduction to IPv6 Security**
  - Why IPv6 is an extensive security topic
  - Overview of the differences to IPv4, relating to Security
- **Part 2: It's Demo time! Selected IPv6 attacks**
  - Local Protocol Attacks
  - Remote Protocol Attacks
- **Part 3: Wrap-up**
  - Recommendations, Resources and Tools
  - Q & A

# IPv4 address pool is empty since 2011



Registry	Area Covered
AFRINIC	Africa Region
APNIC	Asia/Pacific Region
ARIN	North America Region
LACNIC	Latin America and some Caribbean Islands
RIPE NCC	Europe, the Middle East, and Central Asia

- IANAs global pool of available IPv4 addresses was exhausted on 1 February, 2011
- The five Regional Internet Registries each received one of the IANA's five reserved /8 blocks
- Policy: A LIR may **receive only 1,024 IPv4 addresses**, even if they can justify a larger allocation



## ...but the Internet is growing

That's why IPv6 was developed

- **1994:** RFC 1631 “Short term” solution: NAT
- **1995:** IETF starts with IPng
- **1998:** Initial RFC 2460, Internet Protocol, Version 6 (IPv6) Specification

# A word about NAT

## Quotation from RFC 1631, May 1994

### 4. Conclusions

NAT may be a good short term solution to the address depletion and scaling problems. This is because it requires very few changes and can be installed incrementally.

NAT has several negative characteristics that make it inappropriate as a long term solution, and may make it inappropriate even as a short term solution.

# Internet Protocol Version 6 Address Space

- IPv6 addresses are 128 bits long
- Address space:  $2^{128}$  addresses

340.282.366.920.938.463.463.374.607.431.768.211.456  
(IPv4: 4.294.967.296)

- $2^{96}$  times the size of the IPv4 address space

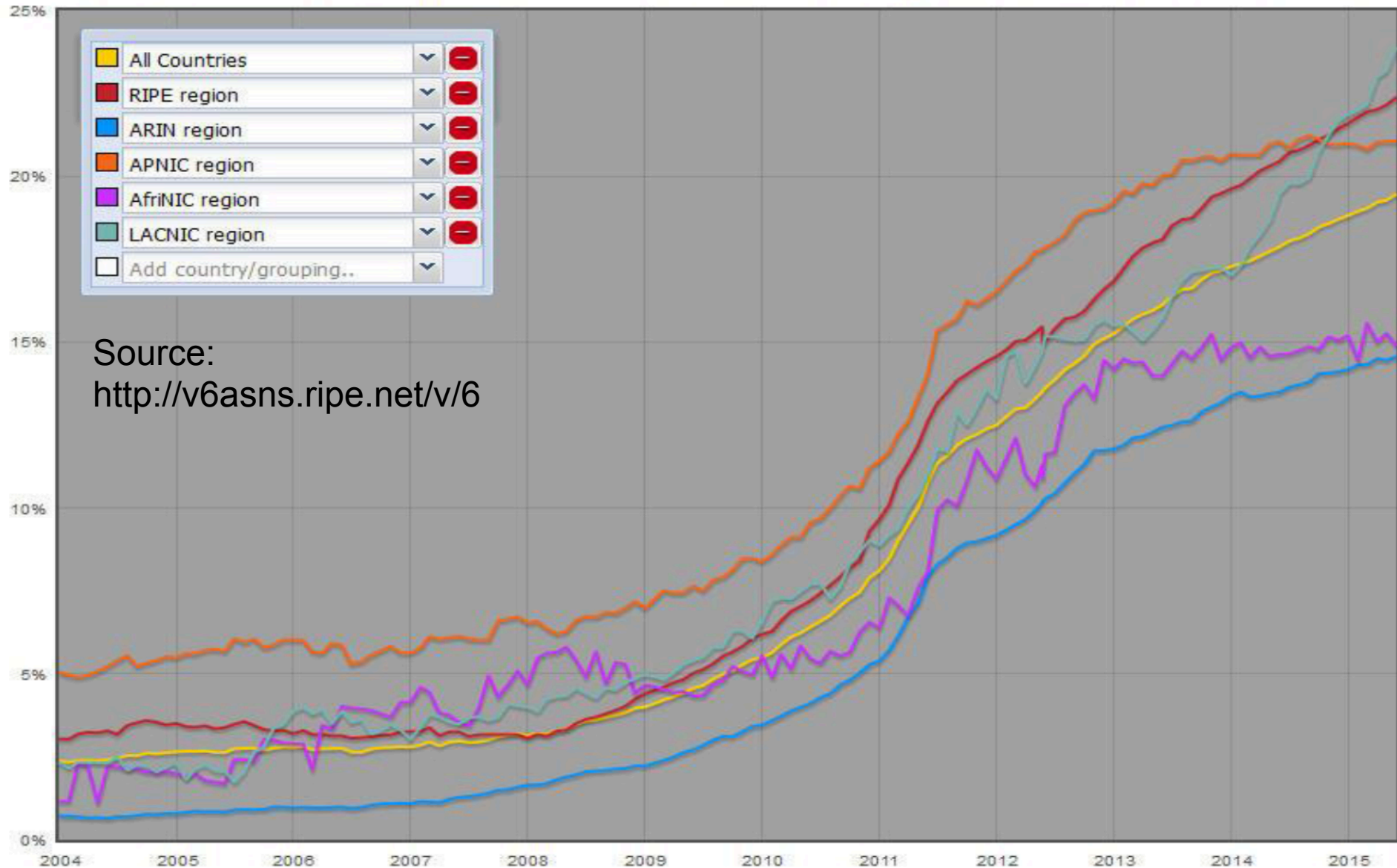
**So what's the status today?**

# Percentage of users who access Google over IPv6





# Percentage of networks (AS) that announce an IPv6 prefix



www.worldipv6launch.org

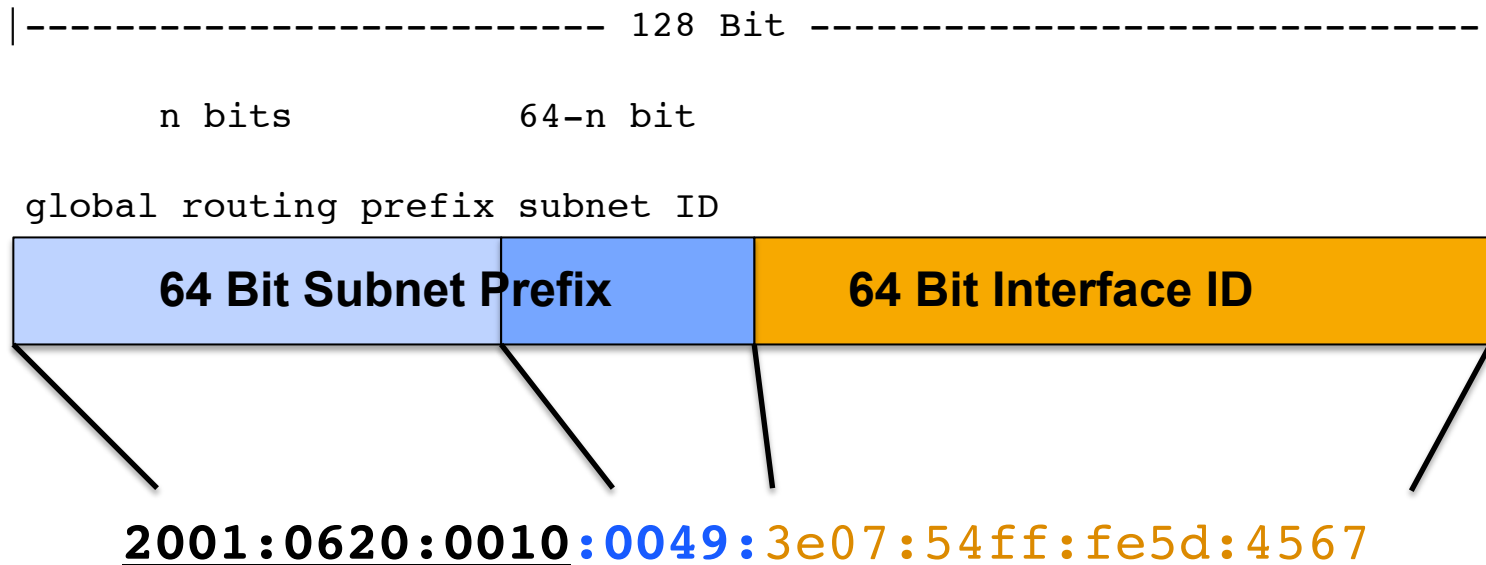
HOME MEASUREMENTS PARTICIPANTS BLOG JOIN THE LAUNCH DOWNLOADS

## IPV6 IS THE NEW NORMAL

Major Internet service providers (ISPs), home networking equipment manufacturers, and web companies around the world are permanently enabling IPv6 for their products and services.

Global IPv6 traffic has grown more than 500% since World IPv6 Launch began on 6 June 2012, and this year – the 2nd “Launchiversary” – marks the fourth straight year IPv6 use has doubled. If current trends continue, more than half of all Internet users will be IPv6-connected in less than four years!

# Global Unicast Address Example



ISP gets from RIR (RIPE NCC): 2001:0620::/32

Client gets from the ISP: 2001:0620:0010::/48

Client has 16 Bits for Subnetting (65536 Subnets)

Prefix for a Subnet: 2001:0620:0010:0049::/64

# Part 1: Introduction to IPv6 Security



# Multiple IPv6 addresses per interface (plus the IPv4 address)

<b>IPv4</b>	173.194.32.119
<b>Link Local</b>	fe80::3e07:54ff:fe5d:abcd
<b>Global</b>	2001:610::41:3e07:54ff:fe5d:abcd*
Privacy Extensions = random / temporary	
<b>Global PE</b>	2001:610::41:65d2:e7eb:d16b:a761**
Unique Local Address = 'private' IPv6 address	
<b>ULA</b>	fd00:1232:ab:41:3e07:54ff:fe5d:abcd

\* Privacy Issue (64 Bit IID the same all over the world)

\*\* Traceability Issue (every hour/day new IP address)



# Unpredictable source address choice

The screenshot illustrates the 'Meine IP-Adresse' tool on the Heise Netze website. The tool displays the source IP address of the request. In the first instance, the IP address is 130.59.1. In the second instance, the IP address is 2001:0620:0000:0049:541d:184f:.

The website header includes the Heise Netze logo and navigation links for News, Artikel, and Foren. The main content area shows the tool's output and a list of available tools.

Tools		
Netalyzr	Bandbreitenrechner	DNS-Abfrage
Fernwartung	MAC-Adressen	Meine IP-Adresse
Netzwerk-Rechner	Ping	Punycode
RFCs	Traceroute	Webcheck
Whois	IPv4-Adressen	IPv6-Adressen

AKTUELLE ARTIKEL »

**So funktioniert LTE**  
Long Term Evolution ist schneller und einfacher als seine Vorgänger

# Certain Mobile devices configure new IPv6 address each time they wake up

- 10:35 Wake up to poll for information

**2001:610::41:65d2:e7eb:d16b:a761**

- 10:37 Entering power-save mode

- 10:40 Wake up to poll for information

**2001:610::41:b5db:3745:463b:57a1**

- 10:42 Entering power-save mode

- 10:47 Wake up to poll for information

**2001:610::41:11c2:abeb:d12a:17fa**

- ...



- Multiple source addresses
- Changing source addresses
- Two protocol stacks

**Correlation can be difficult for...**

**...logging (changing IPs)**

**...monitoring (different views for IPv4/6)**

**...detection (attacks distributed over 4/6)**

# IPv6 address notation isn't unique

## full form:

2001:0db8:0000:08d3:0000:8a2e:0070:7344

## drop leading zeroes:

2001:db8:0:8d3:0:8a2e:70:7344

## collapse multiple zeroes to '::' (once):

2001:db8::8d3:0:8a2e:70:7344

## represent an IPv4 address in a IPv6 data field

::ffff:c000:0280 == ::ffff:192.0.2.128 == 192.0.2.128

# IP address based protection 1 - Blacklists

- IP reputation based Spam block list for IPv6 are not there yet
  - difficult for vast IPv6 address space
  - Sender can utilize ‘nearly unlimited’ source addresses
  - Blacklisting of address ranges can lead to overblocking





# IP address based protection 2 - ACLs

- IPv4 based Access Control Lists (ACLs) only protect the IPv4 access
- Enable IPv6? → Review all your ACLs!

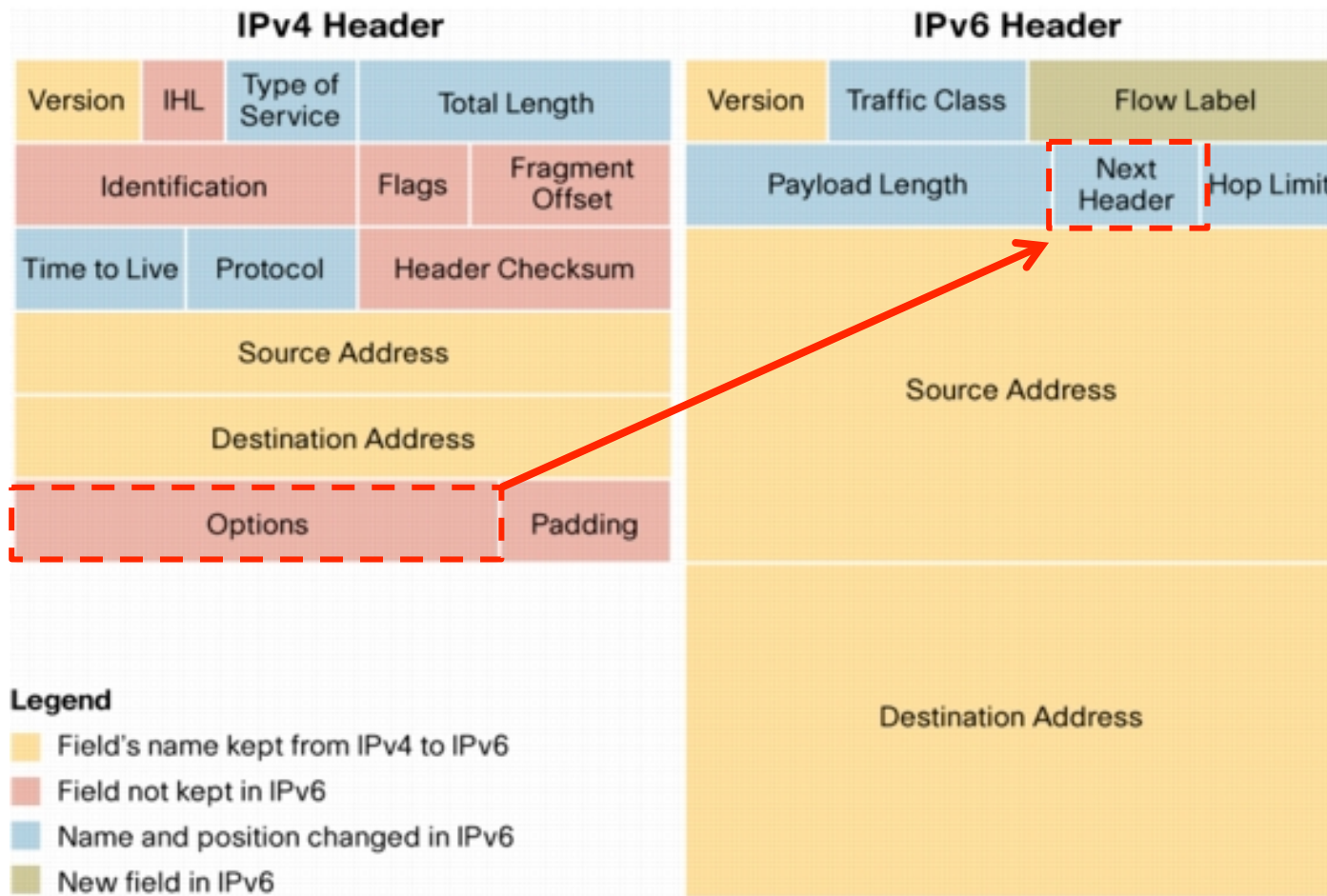
## Inventory

- Firewall Management Interface
- IDS Management Interface
- Router Management Interface
- Database Server
- Backup Database Server
- Power Station Control System
- ...

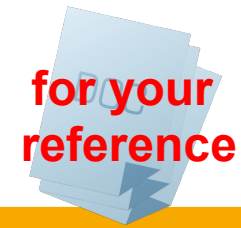
Both doors locked?



# Simplified format of the IP header fixed size → fast processing options go into Extension Header



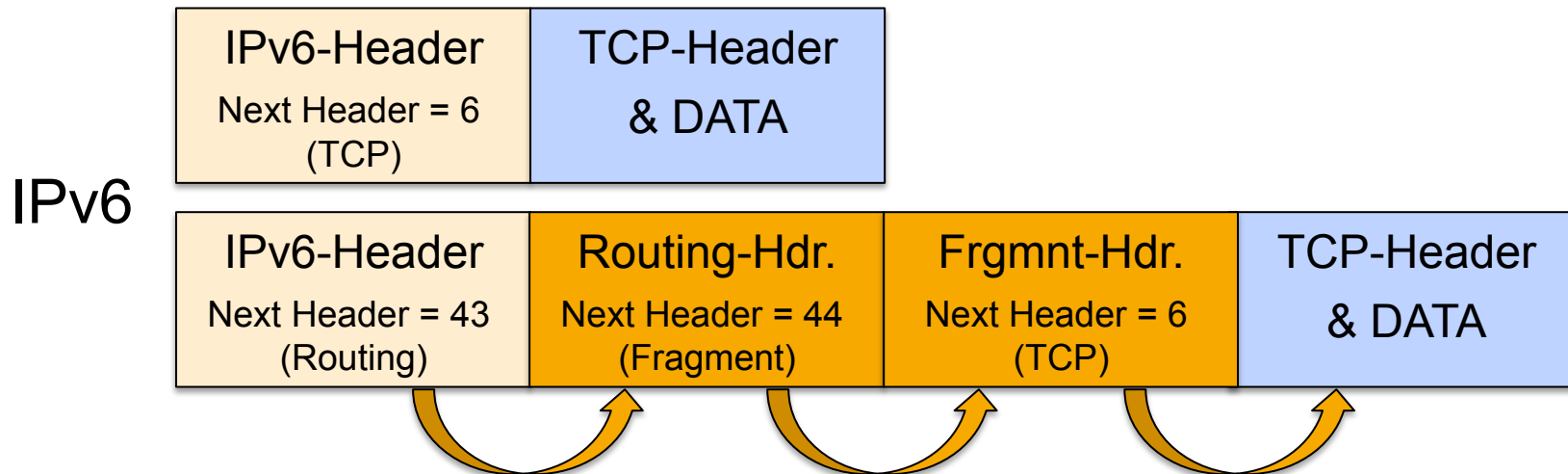
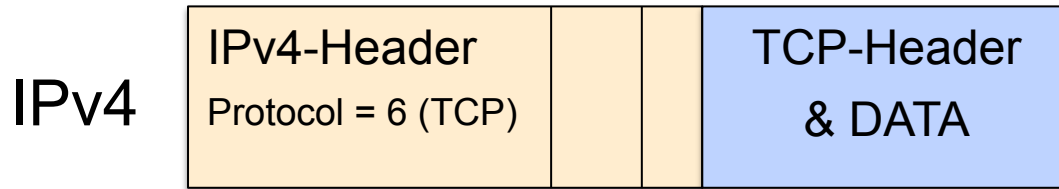
# Extension Header Examples



No.	Name	Functions	Remarks
0	Hop-by-Hop-Options	carries options for hops, e.g. Router Alert (for MLD, RSVP)	<b>must be examined by every hop on the path</b> Must be first EH, only one allowed per packet
60	Destination Options	carries options for destination (e.g. for Mobile IPv6)	<b>processed by destination node only</b>
43	Routing Header	Lists IPv6 nodes that must be "hopped" on the way to dest.	
44	Fragmentation Header	Fragmentation (at source)	only source can fragment, processed by destination node only

Other examples: 6:TCP, 17:UDP, 58:ICMPv6, 50/51: ESP/AH (IPSec)

# Extension Headers increase complexity



# Inspecting packets with EH is challenging...

- The number of EHs is **not limited**
- The number of options within an (Hop-by-Hop or Destination) Options Header is **not limited**
- There is **no defined order** of EHs (only a recommendation)
  - (Exception: Hop-by-Hop Options Header must be first and nonrecurring)
- EH have **different formats**





# According to RFC2460, Section 4 "IPv6 Specification"

- "In-between-Boxes" (such as Firewalls) are not intended to examine EHs...

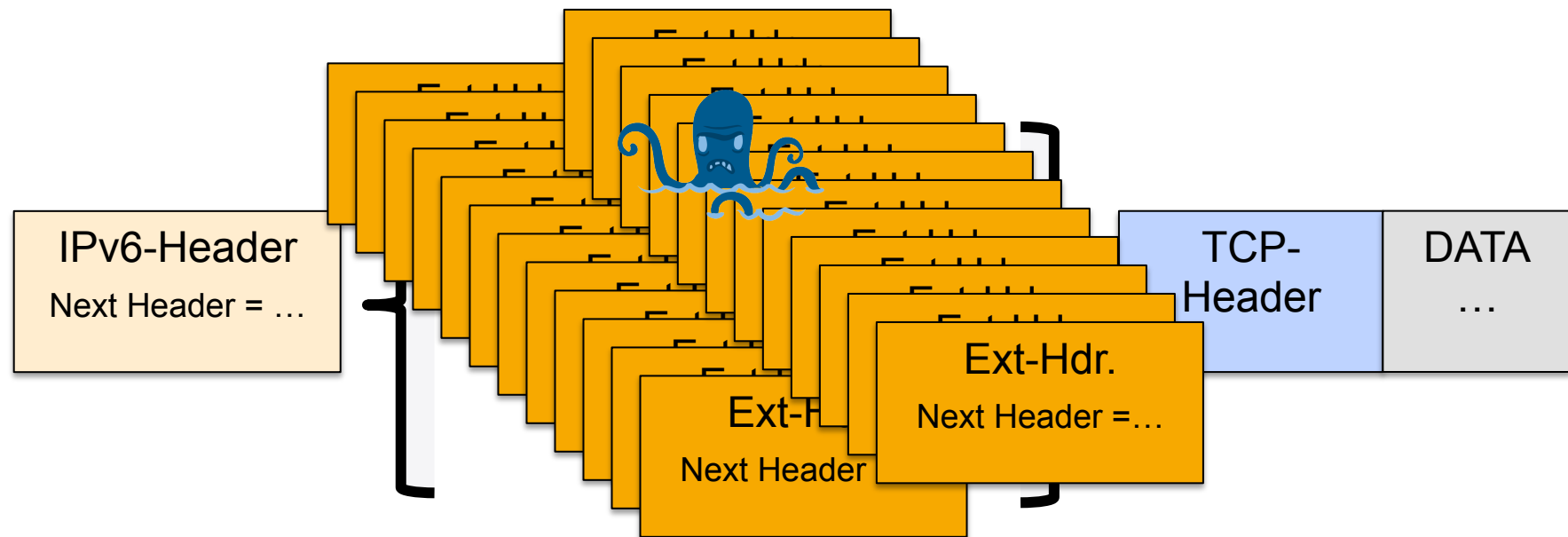
"With one exception, **extension headers are not examined or processed by any node along a packet's delivery path,** until the packet reaches the node."

- ...but the *destination node* must completely process all EHs

"**any order** and occurring **any number** of times in the same packet"

# Possible Threat: High Number of EHs

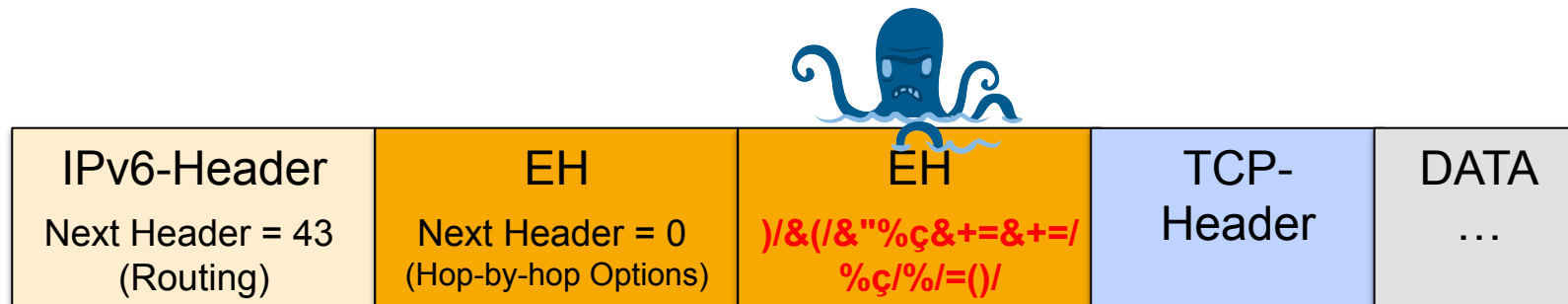
- An attacker could create packet with high number of EH
  - ➔ to try to evade FW / IPS / RA-Guard / other security
  - ➔ might crash or DOS the destination system



**Mitigation option:** Drop packets with more than x EHs

# Possible Threat: Manipulation of the EHs

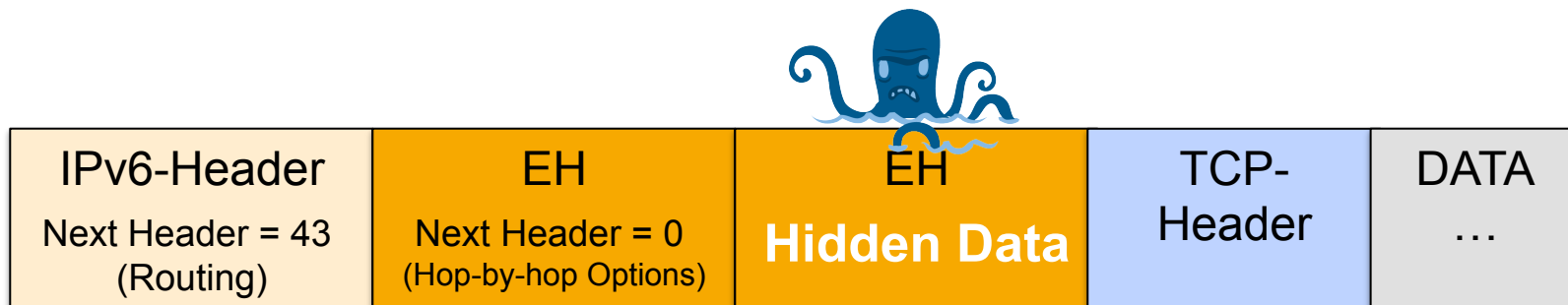
- An attacker could perform header manipulation to create attacks
    - Fuzzing (try everything – it's not limited)
    - add (many) unknown options to an EH, e.g. Hop-by-hop-Options
  - The Destination node / Server has to process crafted EHs
- ➔ Destination System might crash



**Mitigation option:** Perform sanity checks on EH (format / no. of options)

# Possible Threat: Covert Channel

- An attacker could use Extension Headers as a covert channel  
→ to exchange payload undiscovered



**Mitigation option:** Drop unknown EH

# Extension Headers *increeaaase* complexity





# To make it worse: Add fragmentation to it!



# Some examples from Blackhat 2014

## Tipping Point

1. First case: Evading TOS Tipping Point, Package 3.6.1.4036 by using two fragments and wrong next header values.

```
./chiron_scanner.py eth0 -d 2001:db8:1:1::1 -sS -p 80 -lfE 60 -nf 2 -l4_data  
"AAAAAAAAABBBBBBBBCCCCCCCCDDDDDDDDDEEEE" -ll 7,1 -lm 1,0 -lo 0,7 -lnh 60,6
```

## Snort

1. First case: Sending nine Fragment Extension headers in an atomic fragment

```
./chiron_scanner.py eth0 -d 2001:db8:1:1::1 -sn -luE 9x44
```

2. Second case: Sending nine various IPv6 Extension headers in an atomic fragment

```
./chiron_scanner.py eth0 -d 2001:db8:1:1::1 -sn -luE 0,60,43,44,5x60
```

## Suricata

1. First Case: Version 2.0.1: Fragment an IPv6 Extension header and send the fragments mis-ordered.

```
./chiron_scanner.py vboxnet0 -iL /root/IPv6_targets.txt -sS -p 80,22,445  
-l4_data "AAAAAAAA" -lfE 60 -seh 2 -nf 5 -lnh 60,60,60,60,60,60 -lo  
5,1,2,3,4,0 -ll 1,1,1,1,1,1 -lm 0,1,1,1,1,1
```

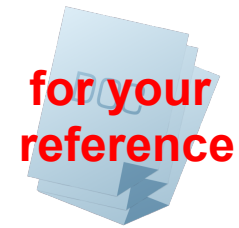


## Some examples from Blackhat 2014

- Blackhat-Paper: “Evasion of High-End IDPS Devices at the IPv6 Era”

<https://www.blackhat.com/docs/eu-14/materials/eu-14-Atlasis-Evasion-Of-High-End-IDPS-Devices-At-The-IPv6-Era-wp.pdf>

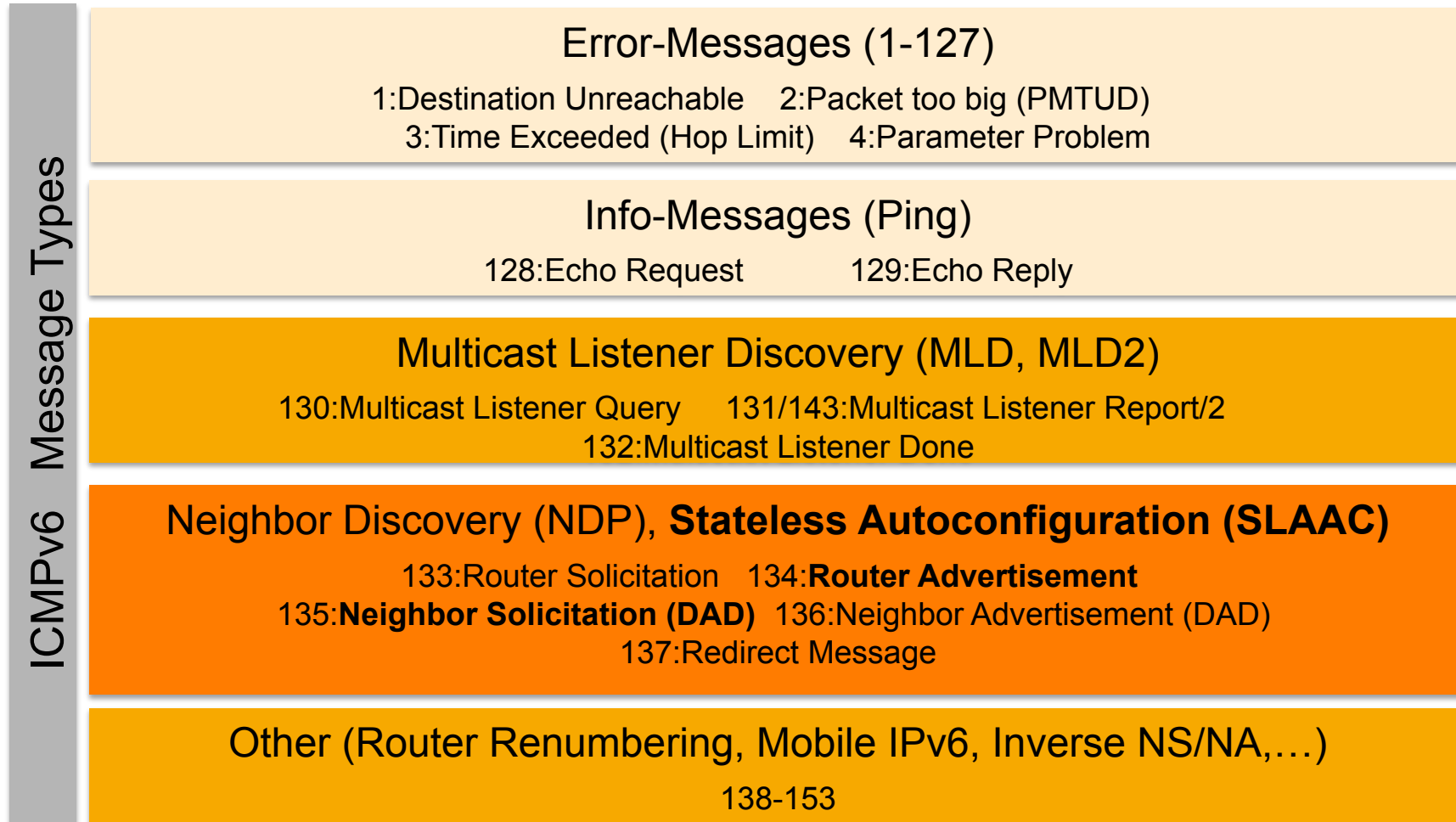
# Preventing Fragmentation Attacks



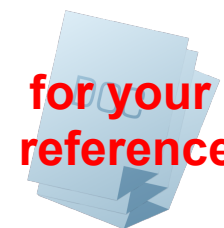
You can

- monitor the amount of fragmented packets
  - ➔ high increase might indicate attack
- block fragments which are below a certain size (if not the last one of a set [M-flag=0])
  - ➔ don't appear in proper communication
- look for Inspection capabilities of fragmented packets
  - e.g. Cisco: Virtual Fragment Inspection (VFR)
    - `ipv6 virtual-reassembly`

# ICMPv6 is much more complex than ICMP



# ICMPv6 filtering is more complex



- If you filter ICMPv6 completely you break IPv6
- Recommendations for Filtering ICMPv6:
  - RFC 4890, 38 pages
- Aim of the RFC:
  - **Allow** propagation of ICMPv6 messages needed to maintain functionality of the network
  - but**
  - **Drop** messages posing potential security risks

# New attacks with ICMPv6

- NDP
- SLAAC
- MLD

**→ Learn more in the Demo-Part**

# IPv6 Tunneling mechanisms can be misused and attacked...

**TEREDO**

**6in4**

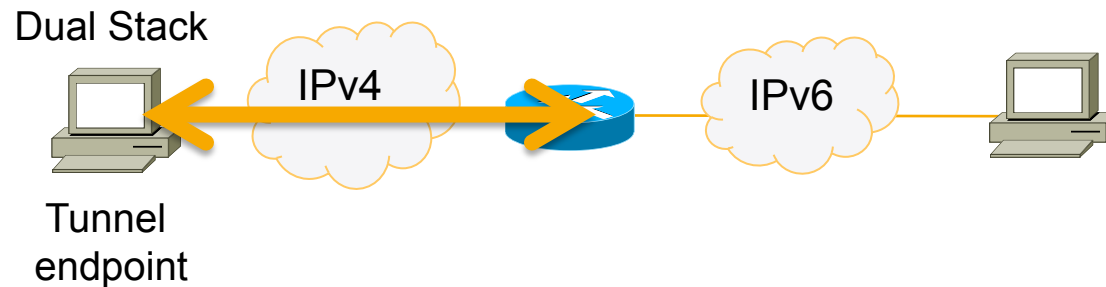
**6to4**

**6rd**

**ISATAP**

...different sorts of tunnels around

# Tunneling: transport of IPv6 packets across IPv4 infrastructure



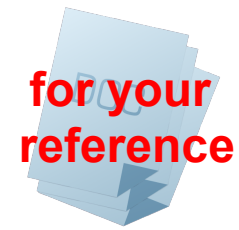
- Autoconfiguration: If MS client can't reach IPv6 resource it tries to establish a tunnel (ISATAP, 6to4, TEREDO)



# Some IPv6 tunneling characteristics

- Tunnel endpoints can configure **automatically**
- or deliberate (by a user/attacker) **and** unknowingly (for the operator)
- Tunnels can possibly **traverse your "Security devices"** (Firewall, NAT-GW)
- Tunnels can be used as **covert channels** or **backdoors**
- Tunnels use **remote Tunnel endpoints** (who operates it, can you trust them?)

# Detect IPv6 tunnels in network logs



Look inside logs / NetFlow records:

- IPv4 Protocol type 41 (ISATAP, 6to4 traffic)
- IPv4 to UDP 3544 (Teredo traffic)
- Traffic to 192.88.99.1 (6to4 anycast server)
- DNS server log: resolution of "ISATAP"

# Lower maturity than IPv4...

- **...in the Design/Specs**  
frequent new RFCs
- **...in the Implementations**  
Vendors have to deal with complexity  
and a moving target
- **... and often little Know-how**

➔ **Learn IPv6**

➔ **Test IPv6 functionality of your (security) devices**

# Example: "Remote system freeze thanks to Kaspersky Internet Security 2013"



Full Disclosure mailing list archives

By Date

By Thread

Search

## Remote system freeze thanks to Kaspersky Internet Security 2013

From: Marc Heuse <mh () mh-sec de>

Date: Mon, 04 Mar 2013 07:01:10 +0100

I usually do not write security advisories unless absolutely necessary.

This time I should, however I have neither the time, nor the desire to do so.  
But Kaspersky did not react, so ... quick and dirty:

Kaspersky Internet Security 2013 (and any other Kaspersky product which includes the firewall functionality) is susceptible to a remote system freeze.

As of the 3rd March 2013, the bug is still unfixed.

If IPv6 connectivity to a victim is possible (which is always the case on local networks), a fragmented packet with multiple but one large extension header leads to a complete freeze of the operating system. No log message or warning window is generated, nor is the system able to perform any task.

To test:

1. download the thc-ipv6 IPv6 protocol attack suite for Linux from [www.thc.org/thc-ipv6](http://www.thc.org/thc-ipv6)

2. compile the tools with "make"

3. run the following tool on the target:

```
firewall6 <interface> <target> <port> 19
```

where interface is the network interface (e.g. eth0)

target is the IPv6 address of the victim (e.g. ff02::1)

port is any tcp port, doesnt matter which (e.g. 80)

and 19 is the test case number.

The test case numbers 18, 19, 20 and 21 lead to a remote system freeze.

Solution: Remove the Kaspersky Anti-Virus NDIS 6 Filter from all network interfaces or uninstall the Kaspersky software until a fix is provided.

The bug was reported to Kaspersky first on the 21st January 2013, then reminded on the 14th February 2013.

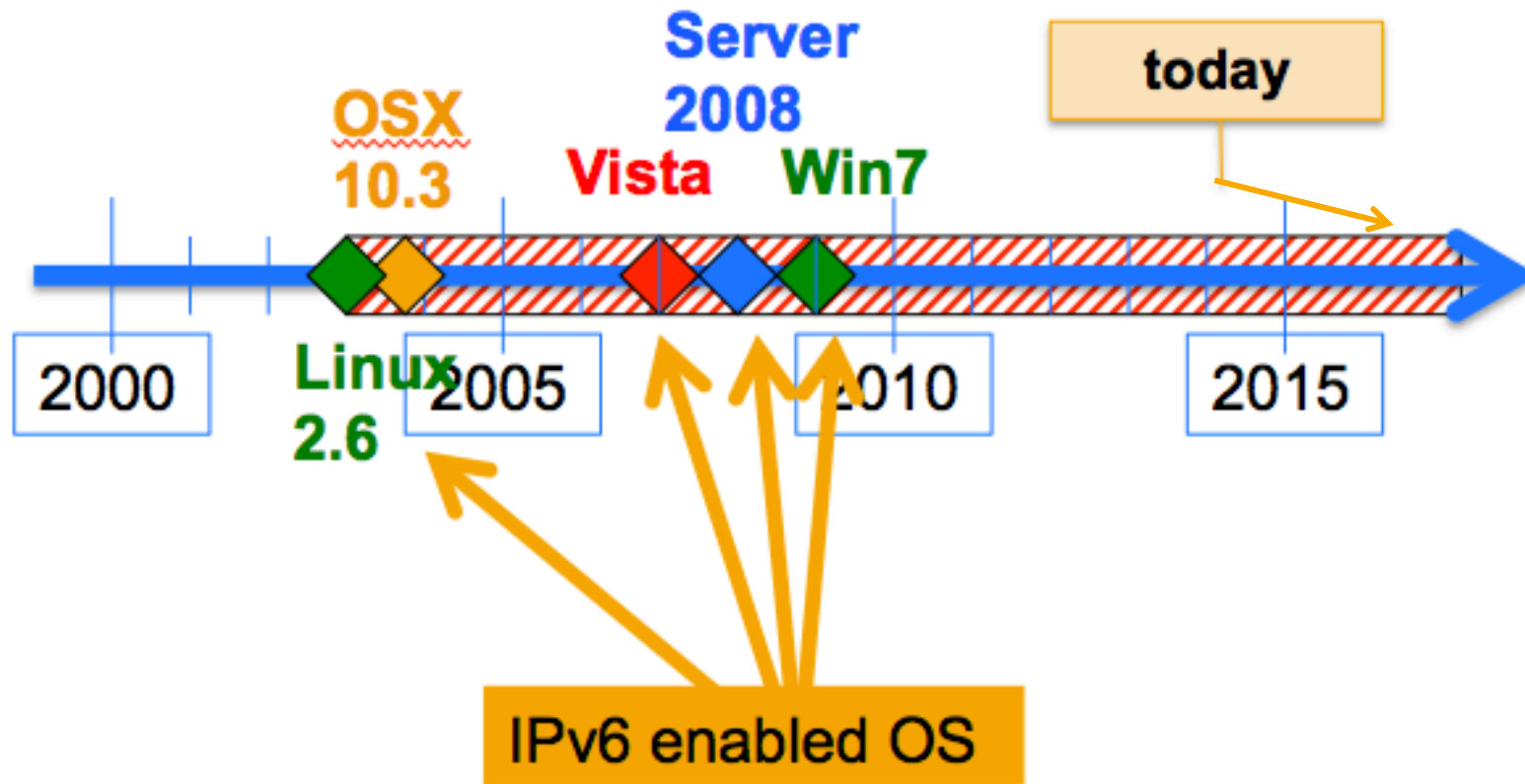
No feedback was given by Kaspersky, and the reminder contained a warning that without feedback the bug would be disclosed on this day. So here we are.

a fragmented packet  
with one large  
extension header leads  
to a complete freeze  
of the operating  
system...

# Latent Threat – IPv6 attacks in "IPv4-only" environment

- IPv6 is enabled on all common OSs and can be auto-configured ("SLAAC-Attack")
  - IPv6 address / Default Route to rogue Router
- Also tunnels might be enabled and can be auto-configured
  - and bypass your FW
- can be misused for DOS- and MITM-Attacks
- can be misused to bypass IPv4-ACLs

# Latent Threat – IPv6 attacks in "IPv4-only" environment



Q: Do you have these OSs in your network?

# Latent Threat – Questions to ask yourself

- Do you see IPv6 traffic on your network? (Monitoring)
- Are you sure your firewalls filter (tunneled) IPv6 traffic?
- Do you have enough knowledge about IPv6 and its specific attacks to detect them?
- Do you rely on IP-based ACLs – which are ineffective for IPv6?
- **<http://securityblog.switch.ch/2014/08/26/ipv6-insecurities-on-ipv4-only-networks/>**

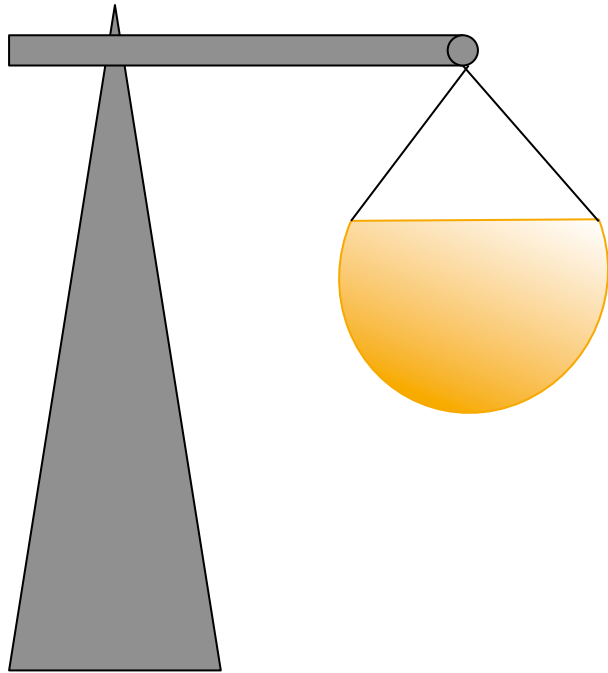


# Opportunities for improved IT-Security?

**Yes!**

- A chance to review the existing level of security
- Consolidation of the Network-Design
- Re-documentation! (remember: IPv4-ACLs)
- IPv6 Addressing plan – more or less Policy friendly
- Rethink NAT vs. real Security (operational cost)
- Preparation for future security features vs. maintaining legacy technology

# Bottom line: How IPv6 affects IT-Security



- Higher complexity (protocol and network)
- Lower maturity (especially security devices)
- Less Know-how / experience
- New / more Attack vectors
- Less visibility (Monitoring)
- Already active in "IPv4-only" net
- A lot of changes (also new opportunities to improve things)



Part 2:

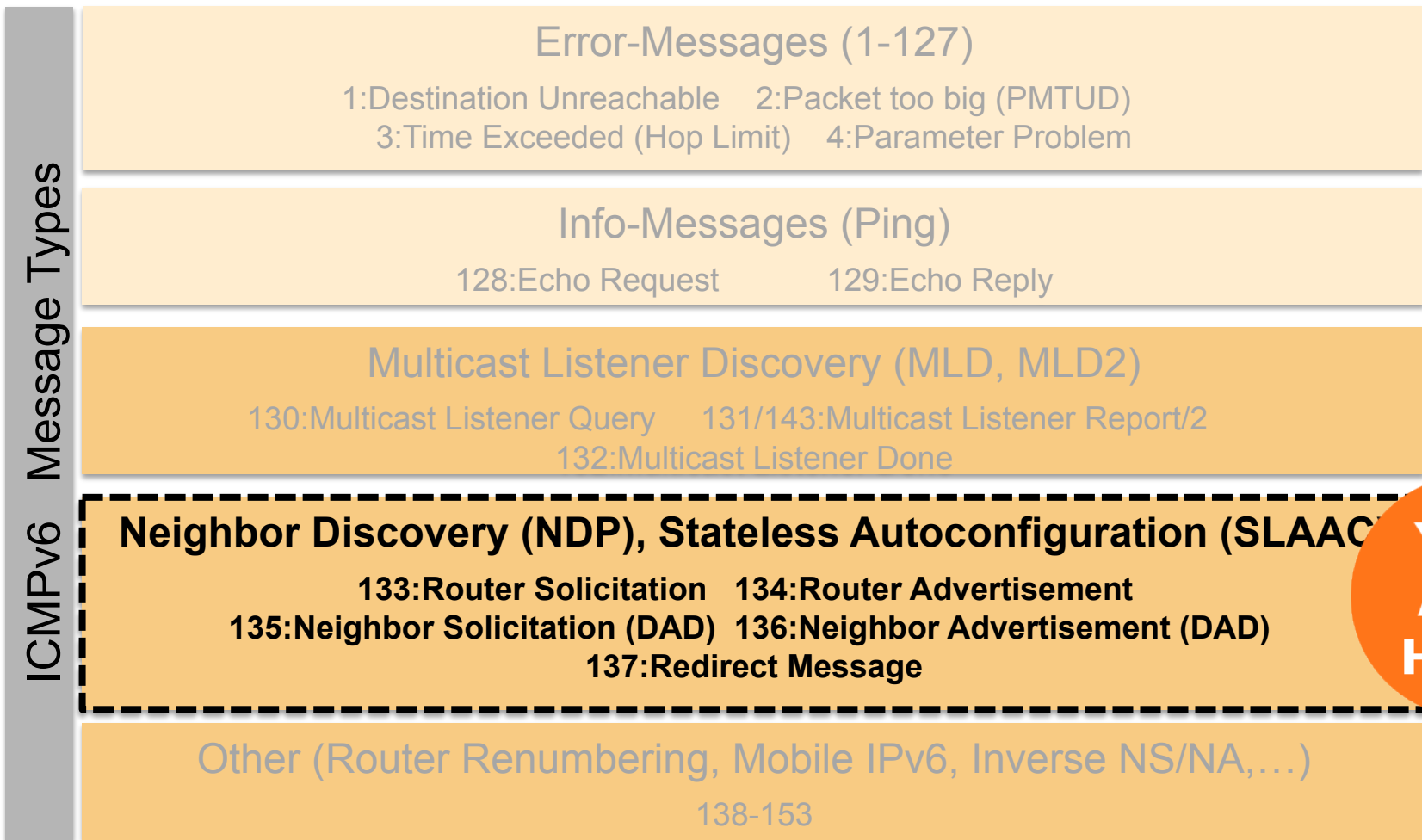
# Selected IPv6 attacks

# IPv6

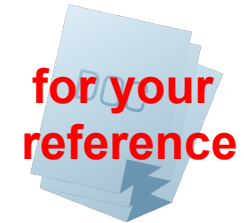


**Still some preparation needed:  
How Stateless Address Autoconfiguration  
works in IPv6**

# ICMPv6



# Neighbor Discovery Protocol consists of 5 ICMPv6 Message Types (133-137)



Router Solicitation

Host sends **RS** to request **RA** after activation of an interface

Router Advertisement

Routers send **RA** to advertise their presence (and parameters) - either periodically, or in response to a **RS** message

Neighbor Solicitation

**NS** requests the link-layer address of a target – and provides its link-layer address to the target

Neighbor Advertisement

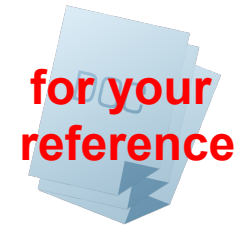
**NA** confirms the existence of a host or router and provides link-layer address

**DAD:** Host with new IP address sends **NS** from (::) to special multicast address\*. No response = it can use this IP **or NA** to Multicast = it will not use this IP (because it already exists on the network)

Redirect

Routers inform hosts of a better first hop for a destination

# Neighbor Discovery Protocol consists of 5 ICMPv6 Message Types (133-137)



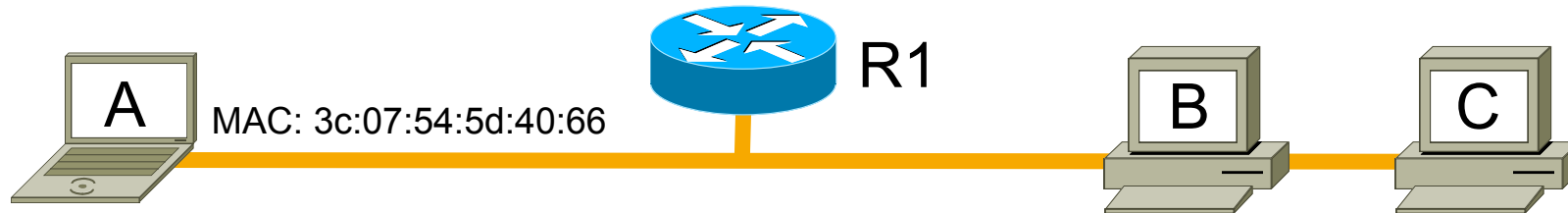
Multiple functions:

- Autoconfigure IP addresses (SLAAC)
- Find gateway routers (SLAAC)
- Detect duplicate addresses (DAD)
- Tell the node to use DHCPv6
- Discover other nodes on the link
- Determine link-layer addresses (Address Resolution)
- Maintain neighbor reachability information
- Redirects

# Stateless Address Autoconfiguration (SLAAC)



# SLAAC Step 1: configure link-local address



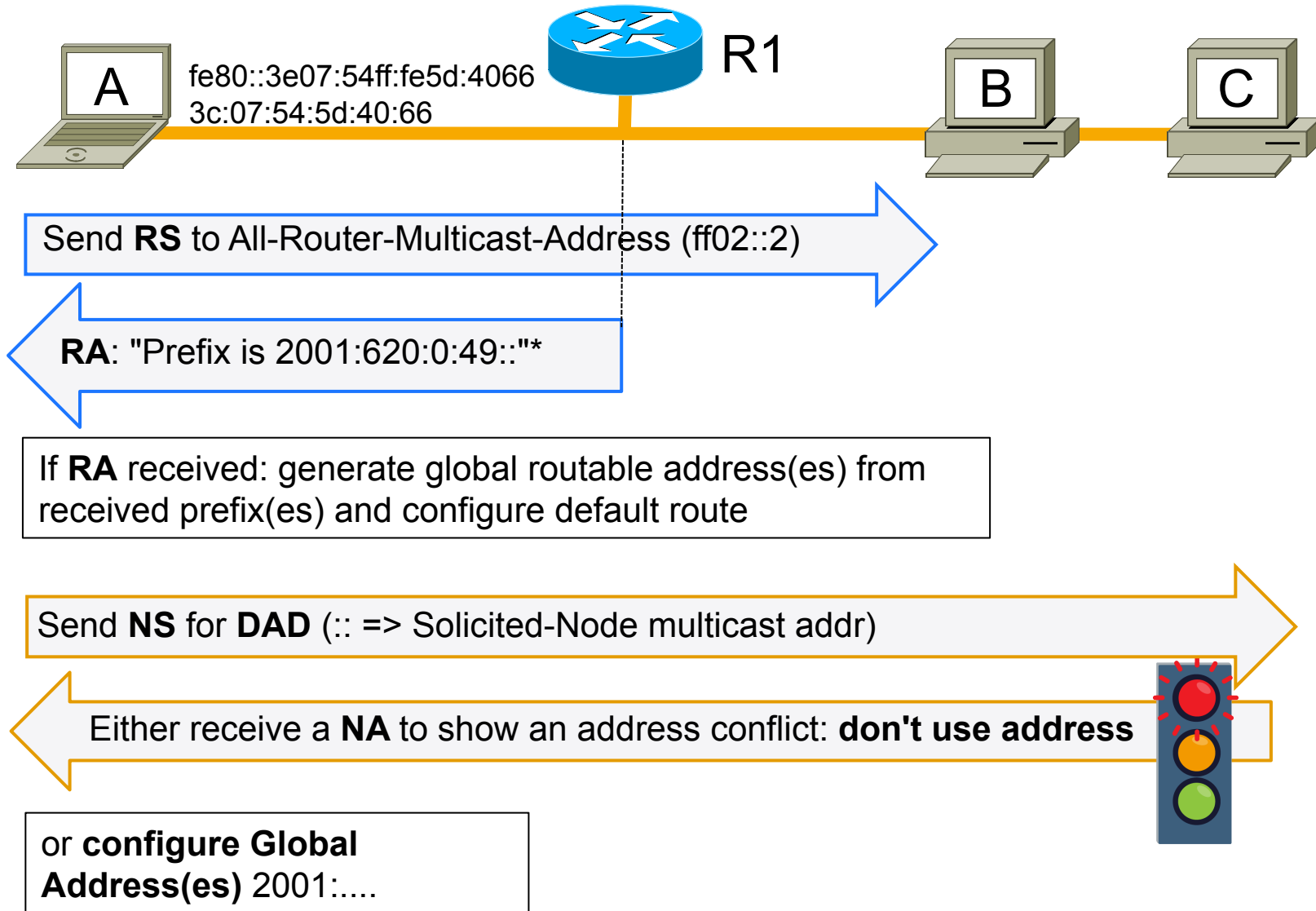
Generate a link local address (FE80), from MAC address  
*state: tentative*

Send **NS** for **DAD** (:: => Solicited-Node multicast addr)

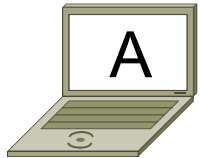
Either receive a **NA** to show an address conflict: **stop autoconfig**

or change state of link local address to: *preferred*  
**fe80::3e07:54ff:fe5d:4066**

# SLAAC Step 2: configure global addresses



# SLAAC successful:



eth0:

Link Layer Address: 3c:07:54:5d:40:66

Link Local Address: fe80::3e07:54ff:fe5d:4066

Global Address: 2001:620::49:3e07:54ff:fe5d:4066

Global Address: 2001:620::49:1c78:9b29:27c1:7564

- Default Router Address (implicitly learned from RA)
- Options (RDNSS,...)

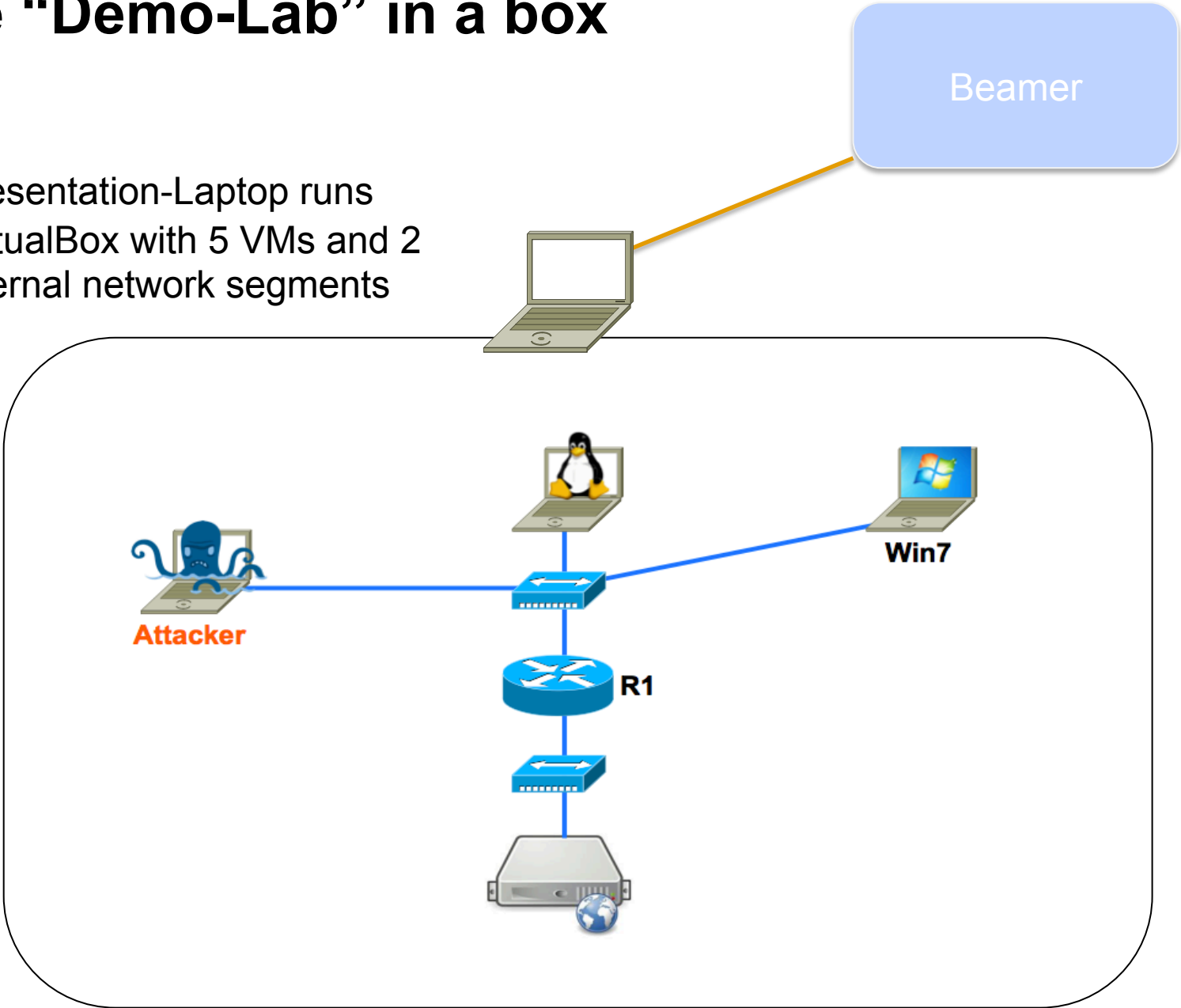
## IPv6 addresses don't live forever

- IPv6 addresses have count down timers (for link local = infinite)
- Regular RAs reset them
- Intended for Renumbering scenario

# Demo setup

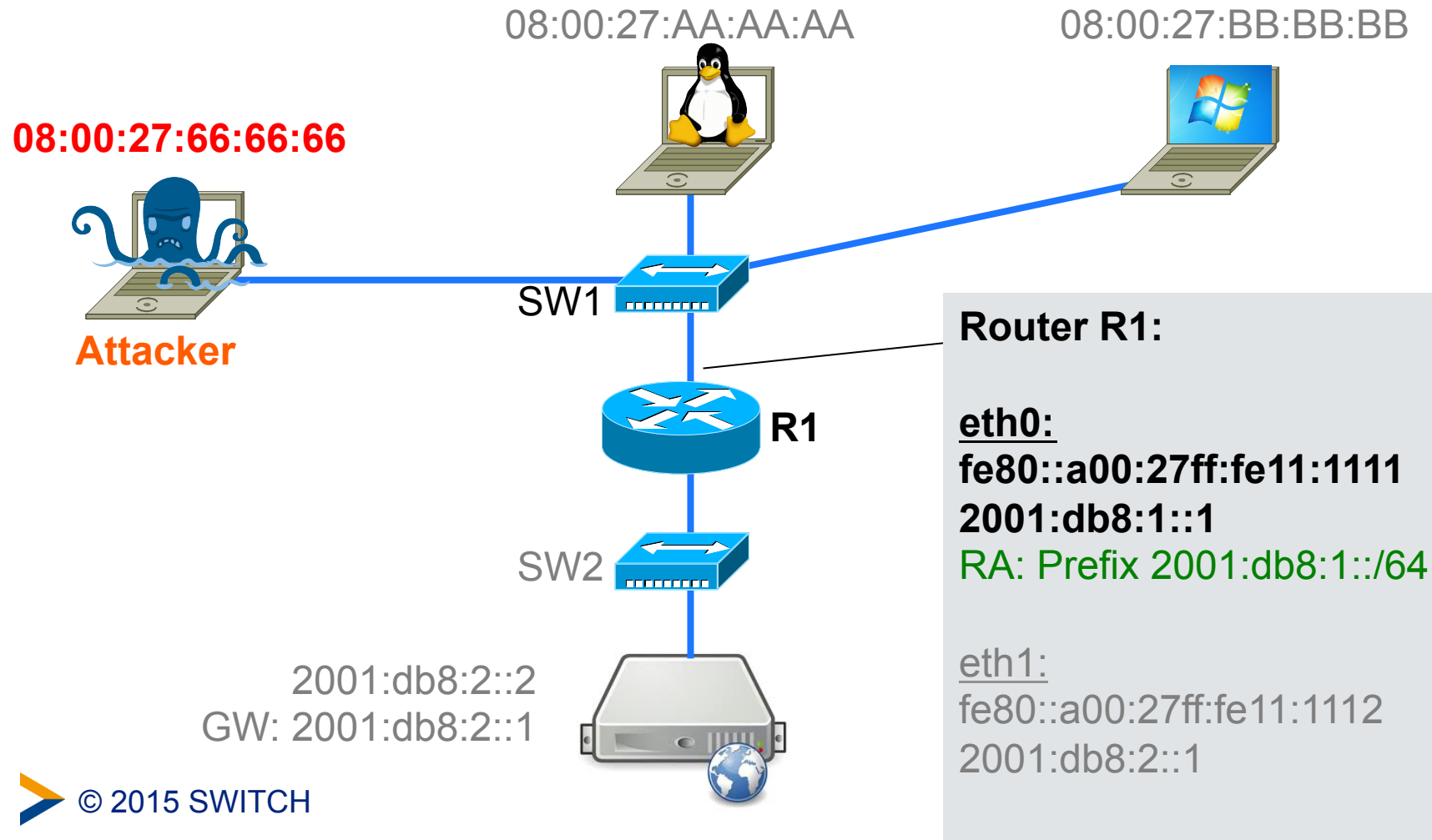
# The “Demo-Lab” in a box

Presentation-Laptop runs  
VirtualBox with 5 VMs and 2  
internal network segments



# Lab Configuration

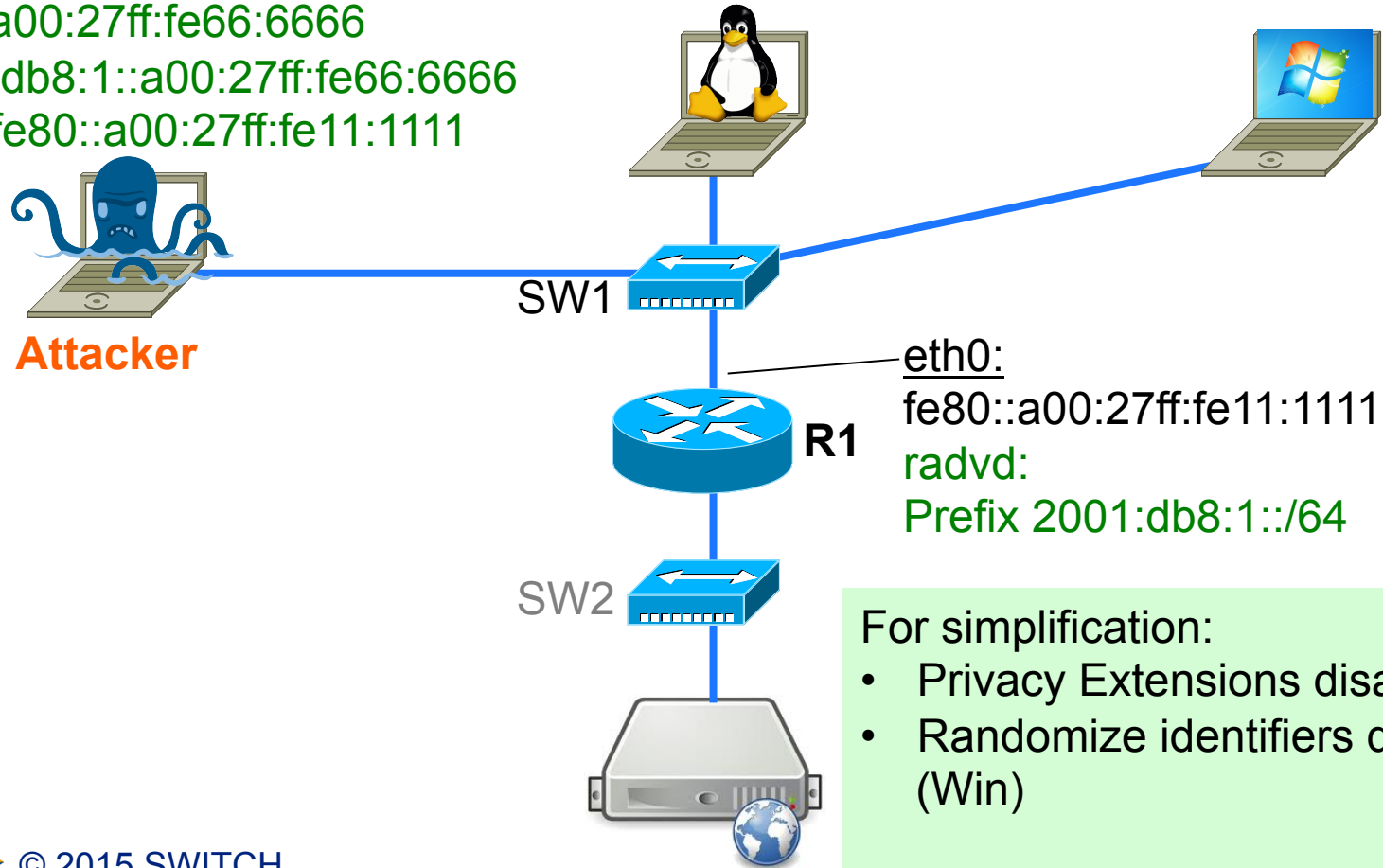
- 3 Clients
- 1 IPv6 Router
- 1 Webserver



# Lab Configuration after Autoconfiguration

08:00:27:AA:AA:AA	08:00:27:BB:BB:BB
fe80:a00:27ff:feaa:aaaa	fe80:a00:27ff:febb:bbbb
2001:db8:1::a00:27ff:feaa:aaaa	2001:db8:1::a00:27ff:febb:bbbb
GW: fe80::a00:27ff:fe11:1111	GW: fe80::a00:27ff:fe11:1111

08:00:27:66:66:66  
 fe80:a00:27ff:fe66:6666  
 2001:db8:1::a00:27ff:fe66:6666  
 GW: fe80::a00:27ff:fe11:1111



For simplification:

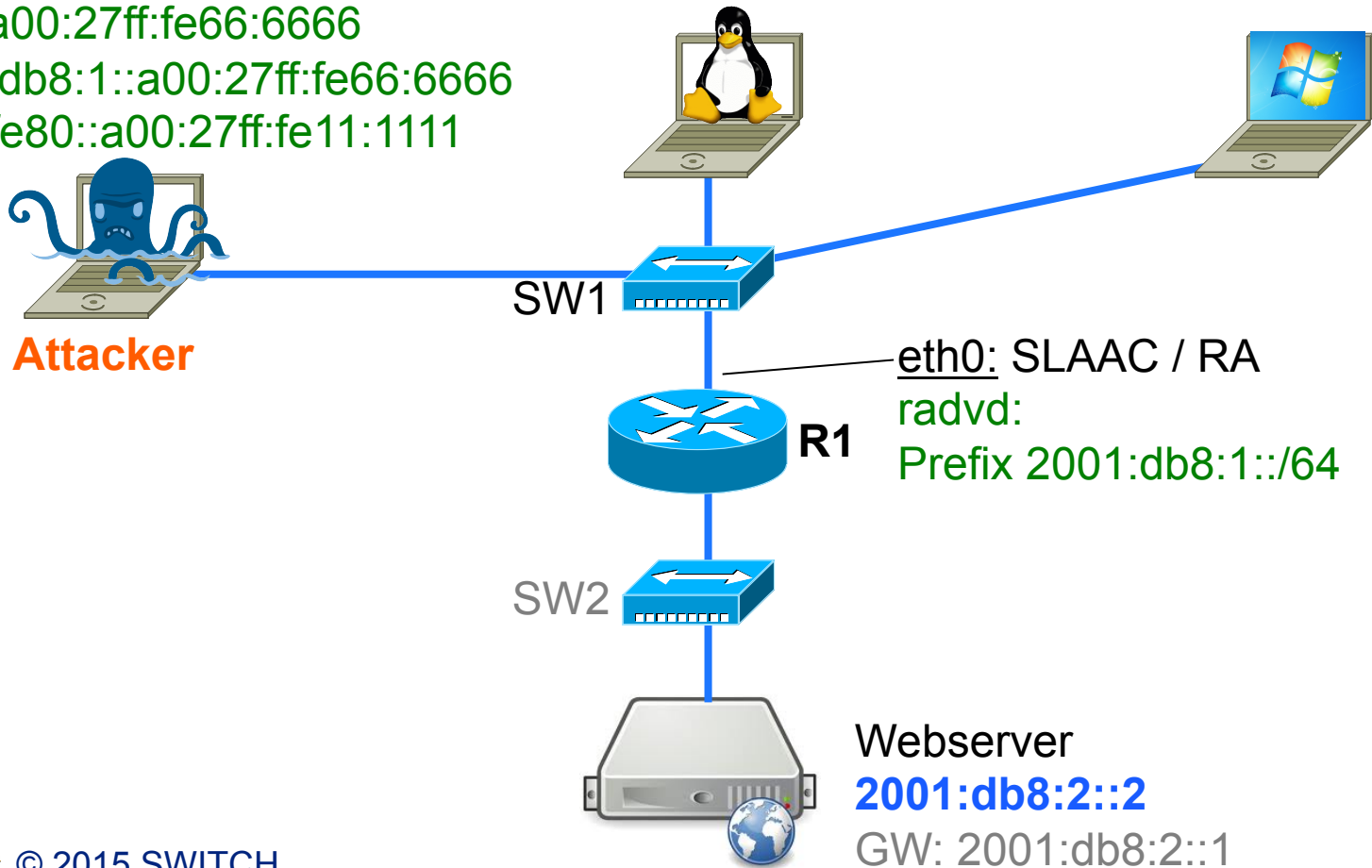
- Privacy Extensions disabled
- Randomize identifiers disabled (Win)



# Access Webserver: [http://\[2001:db8:2::2\]/](http://[2001:db8:2::2]/)

08:00:27:AA:AA:AA	08:00:27:BB:BB:BB
fe80:a00:27ff:feaa:aaaa	fe80:a00:27ff:febb:bbbb
2001:db8:1::a00:27ff:feaa:aaaa	2001:db8:1::a00:27ff:febb:bbbb
GW: fe80::a00:27ff:fe11:1111	GW: fe80::a00:27ff:fe11:1111

08:00:27:66:66:66  
 fe80:a00:27ff:fe66:6666  
 2001:db8:1::a00:27ff:fe66:6666  
 GW: fe80::a00:27ff:fe11:1111





**It's Demo time!**

**Selected IPv6 attacks**

**IPv6**

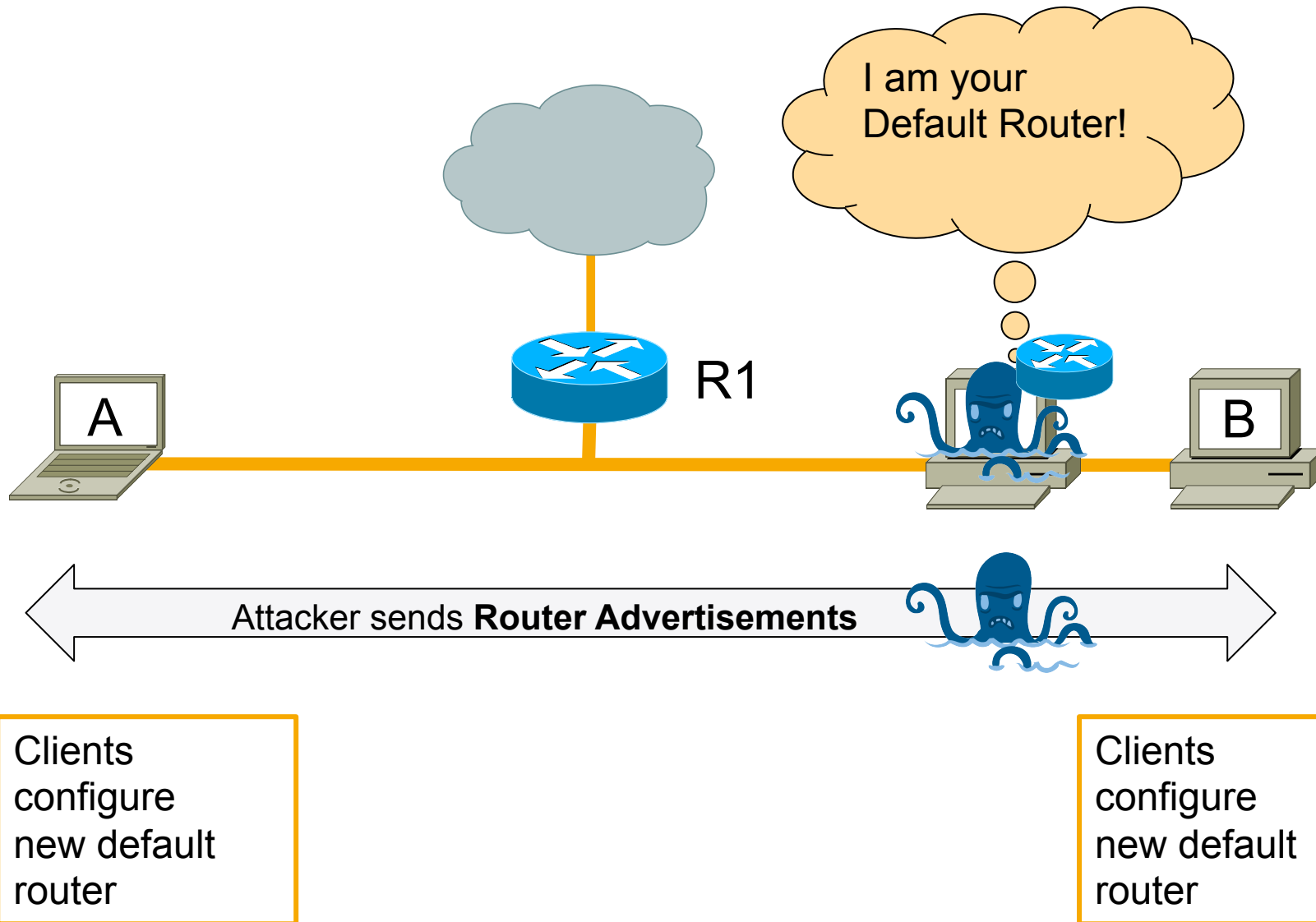


Demo 1:

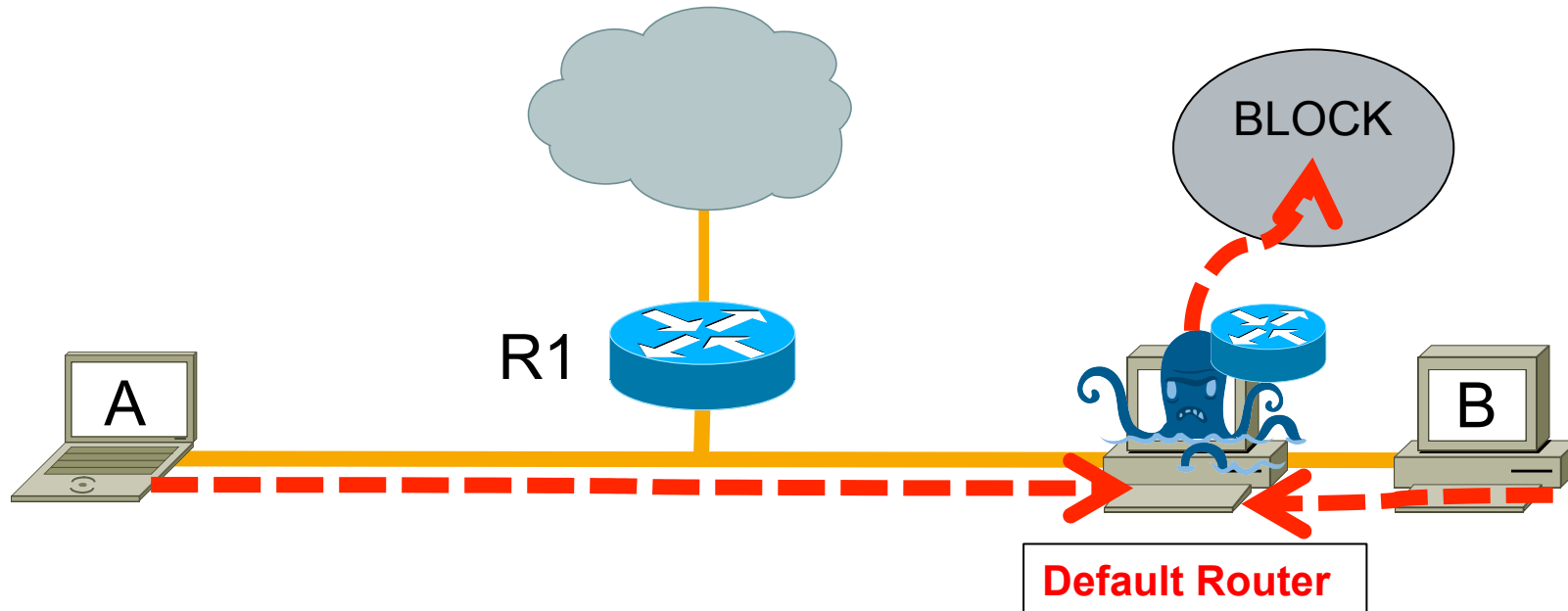
**Add a rogue Router**

**IPv6**

# Rogue RA Principle

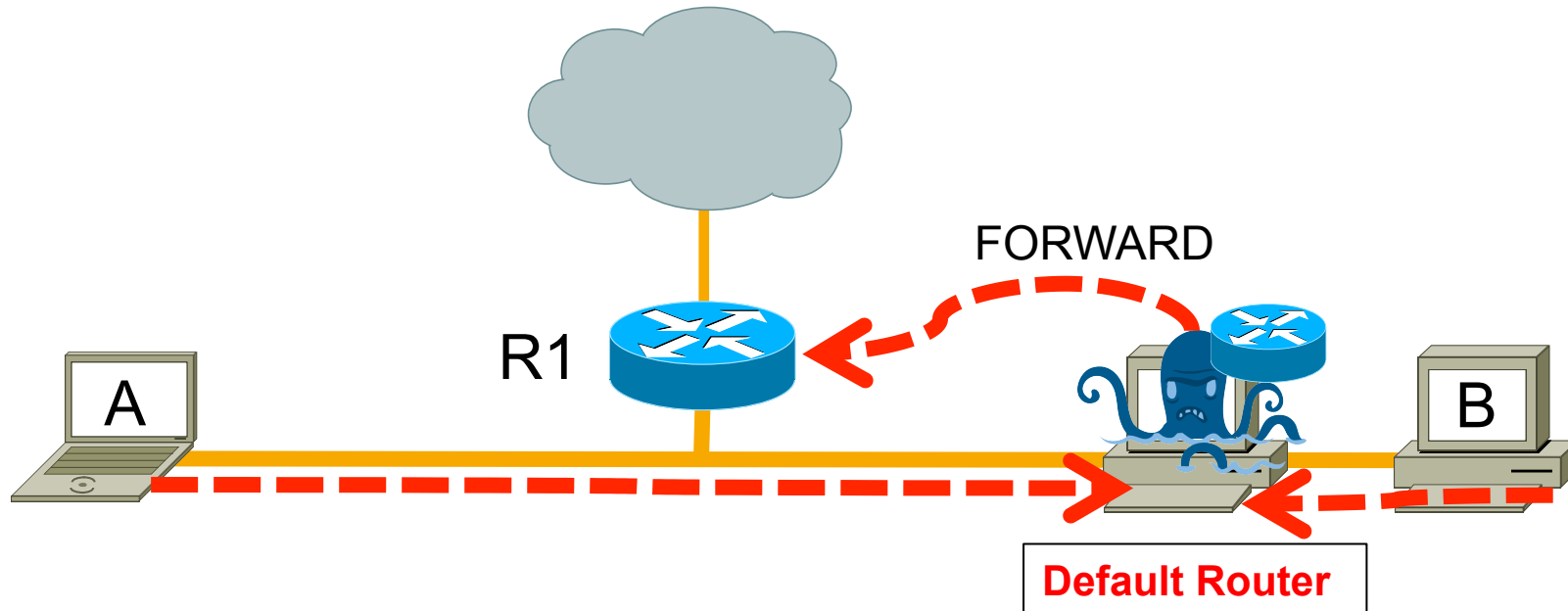


# Rogue RA – Denial of Service



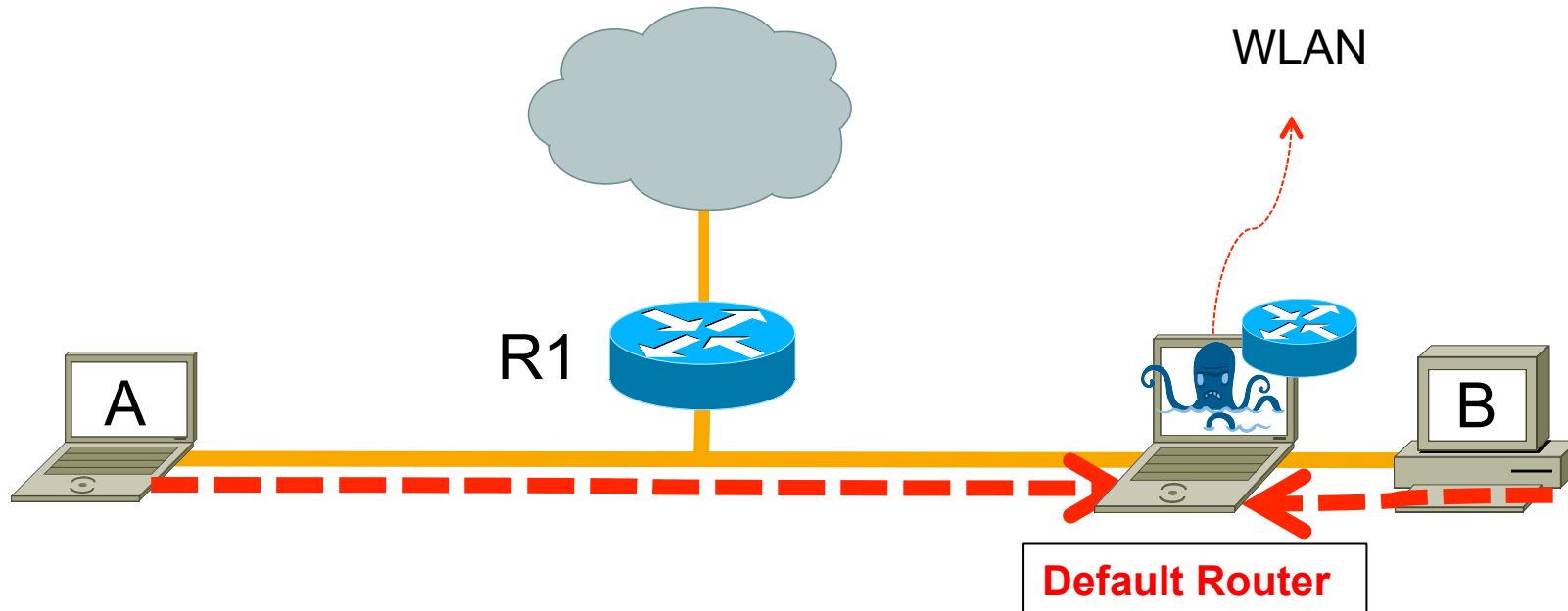
Attacker attracts traffic, ending up in a black hole

# Rogue RA – Man in the Middle Attack



Attacker can intercept, listen, modify unprotected data

# Rogue RA – Performance Issue



Rogue Router becomes a bottleneck  
Often not an attack but misconfigured client



# Rogue RA Attacking Tool



## fake\_router6 / fake\_router26

Announce yourself as a router and try to become the default router.

If a non-existing link-local or mac address is supplied, this results in a DOS.

**Syntax:** fake\_router26 [-E type] [-A network/prefix] [-R network/prefix] [-D dns-server] [-s sourceip] [-S sourcemac] [-ardl seconds] [-Tt ms] interface

### Options:

-A network/prefix	add autoconfiguration network (up to 16 times)
-a seconds	valid lifetime of prefix -A (defaults to 99999)
-R network/prefix	add a route entry (up to 16 times)
-r seconds	route entry lifetime of -R (defaults to 4096)
-D dns-server	specify a DNS server (up to 16 times)
-d seconds	dns entry lifetime of -D (defaults to 4096)
-M mtu	the MTU to send, defaults to the interface setting
-s sourceip	the source ip of the router, defaults to your link local
-S sourcemac	the source mac of the router, defaults to your interface
-l seconds	router lifetime (defaults to 2048)
-T ms	reachable timer (defaults to 0)
-t ms	retrans timer (defaults to 0)
-E type	Router Advertisement Guard Evasion option. Types:
	H      simple hop-by-hop header
	1      simple one-shot fragment. hdr. (can add multiple)
	D      insert a large destin. hdr. so that it fragments

Examples: -E H111, -E D

**Example: fake\_router6 eth1 2004::/48**

# Attack: Rogue IPv6 Router

08:00:27:AA:AA:AA

fe80:a00:27ff:feaa:aaaa

2001:db8:1::a00:27ff:feaa:aaaa

GW: fe80::a00:27ff:fe11:1111

GW: fe80::a00:27ff:fe66:6666

08:00:27:BB:BB:BB

fe80:a00:27ff:febb:bbbb

2001:db8:1::a00:27ff:febb:bbbb

GW: fe80::a00:27ff:fe11:1111

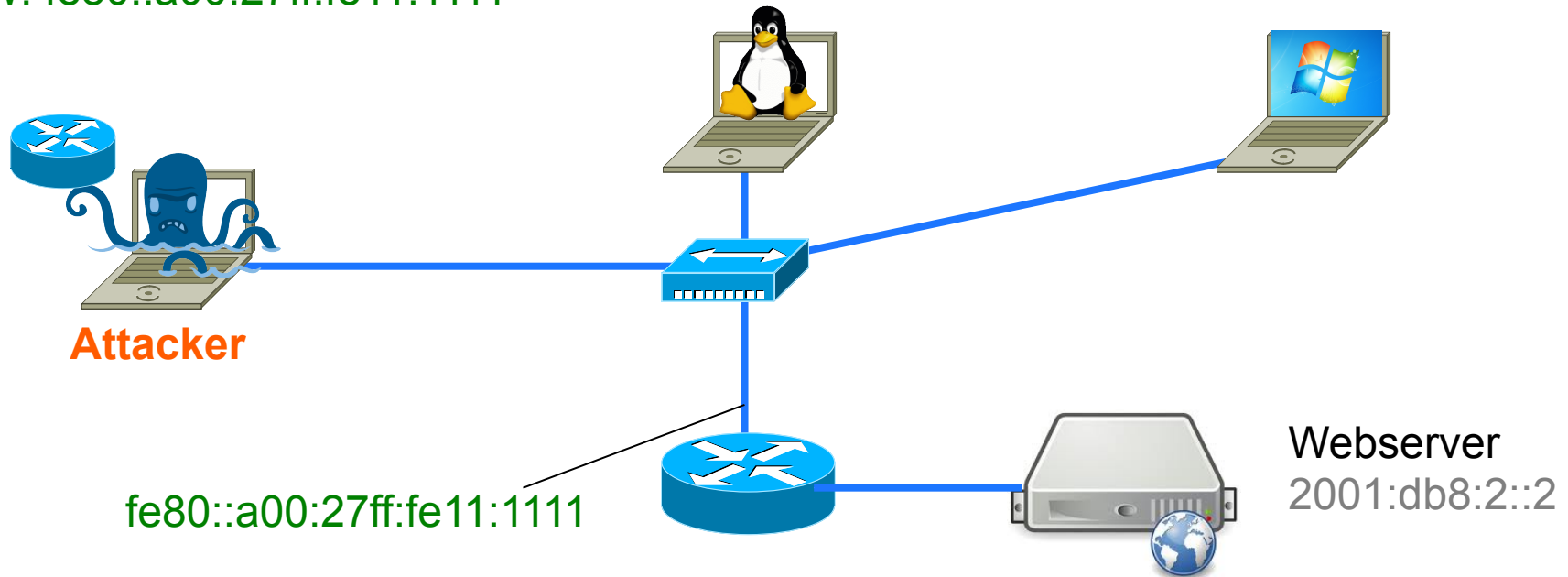
GW: fe80::a00:27ff:fe66:6666

08:00:27:66:66:66

fe80:a00:27ff:fe66:6666

2001:db8:1::a00:27ff:fe66:6666

GW: fe80::a00:27ff:fe11:1111



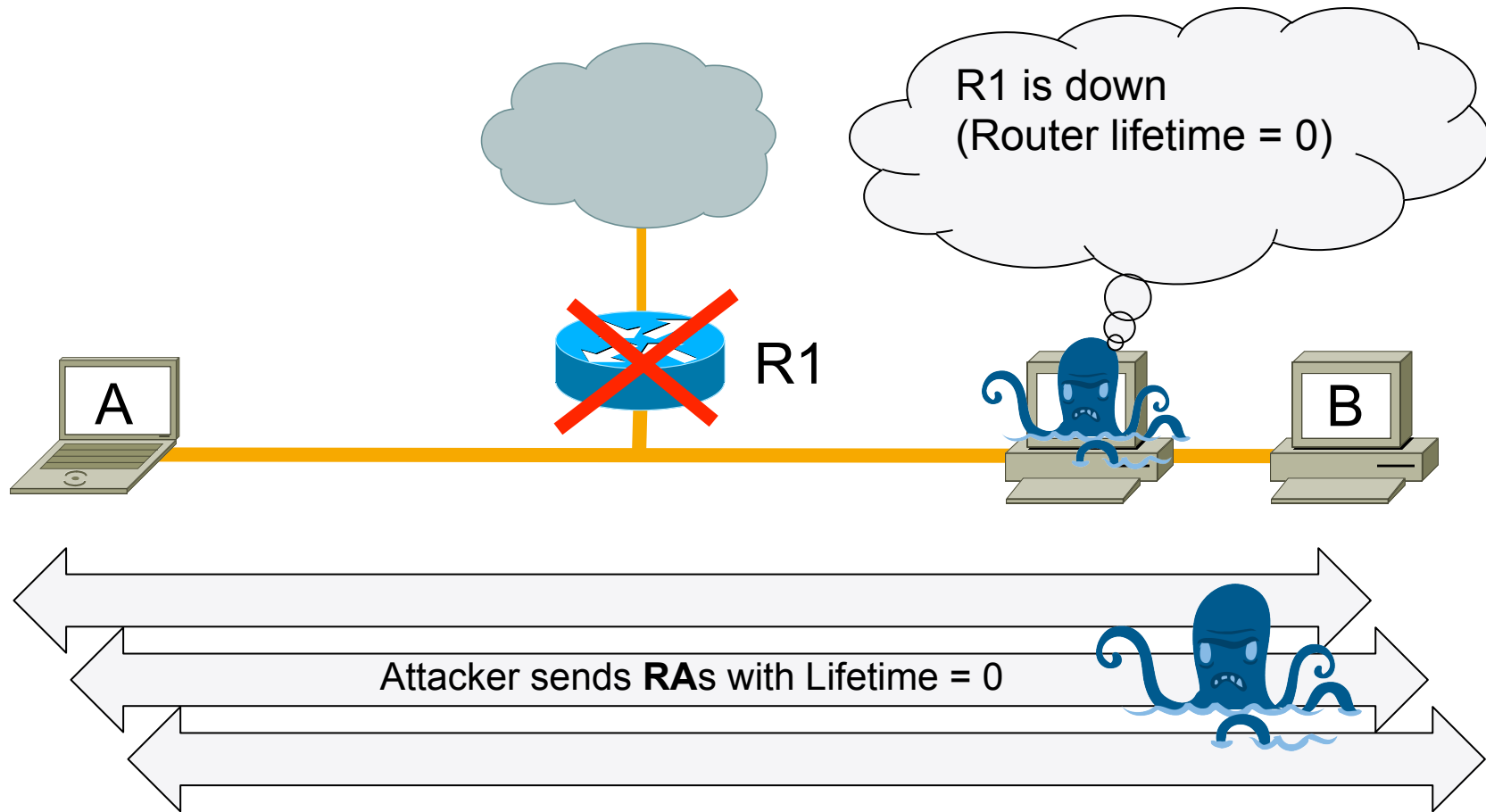


**Demo 2:**

**Delete legitimate Router**

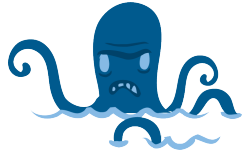
**IPv6**

# Router Lifetime 0 Attack



Remove legitimate router from routing table

# Router Lifetime 0 Attack



**kill\_router6**

Announce (to ff02:1) that a router is going down (RA with Router Lifetime 0) to delete it from the routing tables.

Using asterix '\*' as router-address, this tool will sniff the network for RAs and immediately send a kill packet.

Option -H adds hop-by-hop, -F fragmentation header and -D dst header.

**Syntax:** kill\_router6 [-HFD] interface router-address [srcmac [dstmac]]

**Example:** kill\_router6 eth1 '\*'

# MITM-Attack: rogue RA plus lifetime 0 clones

08:00:27:AA:AA:AA

fe80:a00:27ff:feaa:aaaa

2001:db8:1::a00:27ff:feaa:aaaa

~~GW: fe80::a00:27ff:fe11:1111~~

GW: fe80::a00:27ff:fe66:6666

08:00:27:BB:BB:BB

fe80:a00:27ff:febb:bbbb

2001:db8:1::a00:27ff:febb:bbbb

~~GW: fe80::a00:27ff:fe11:1111~~

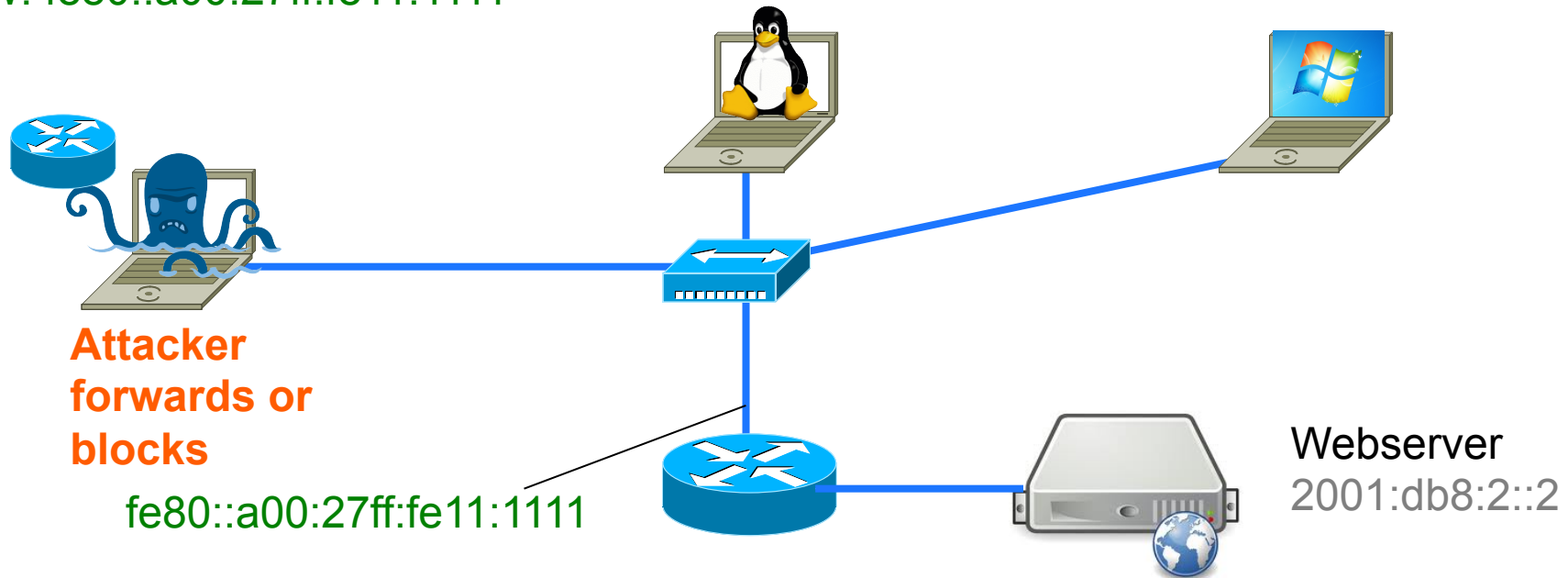
GW: fe80::a00:27ff:fe66:6666

08:00:27:66:66:66

fe80:a00:27ff:fe66:6666

2001:db8:1::a00:27ff:fe66:6666

GW: fe80::a00:27ff:fe11:1111





Demo 3:

**Duplicate Address Detection  
DOS**

**IPv6**

# What is DAD?

## Duplicate Address Detection, RFC 2462, Section 5.4

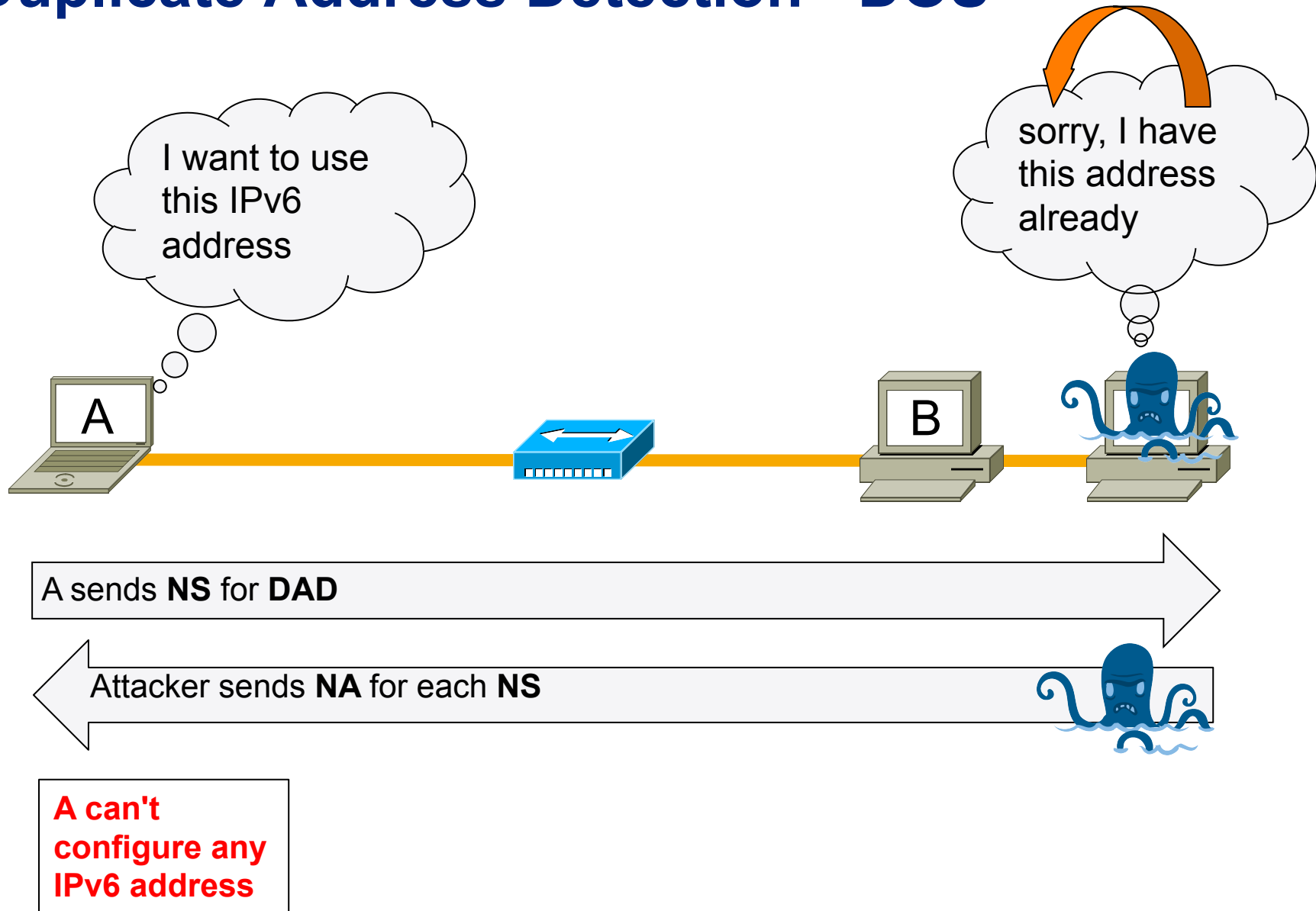
A mechanism assuring that two IPv6 nodes on the same link are not using the same address

(remember SLAAC slides at the beginning)

- DAD is performed on unicast addresses prior to assigning them to an interface
- DAD **must** take place on all unicast addresses, *regardless of whether they are obtained through stateful (DHCP), stateless or manual configuration*



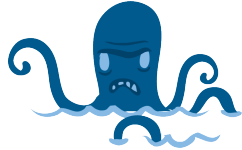
# Duplicate Address Detection - DOS



# Duplicate Address Detection - DOS

- Attacker replies to each DAD-NS
- Victim can't configure an IPv6 address at all
- **Works also if Autoconfiguration is disabled:** DAD is mandatory also for DHCPv6 or manually configured addresses!

# Duplicate Address Detection - DOS



## **dos-new-ip6**

This tool prevents new ipv6 interfaces to come up, by sending answers to duplicate ip6 checks (DAD). This results in a DOS for new ipv6 devices.

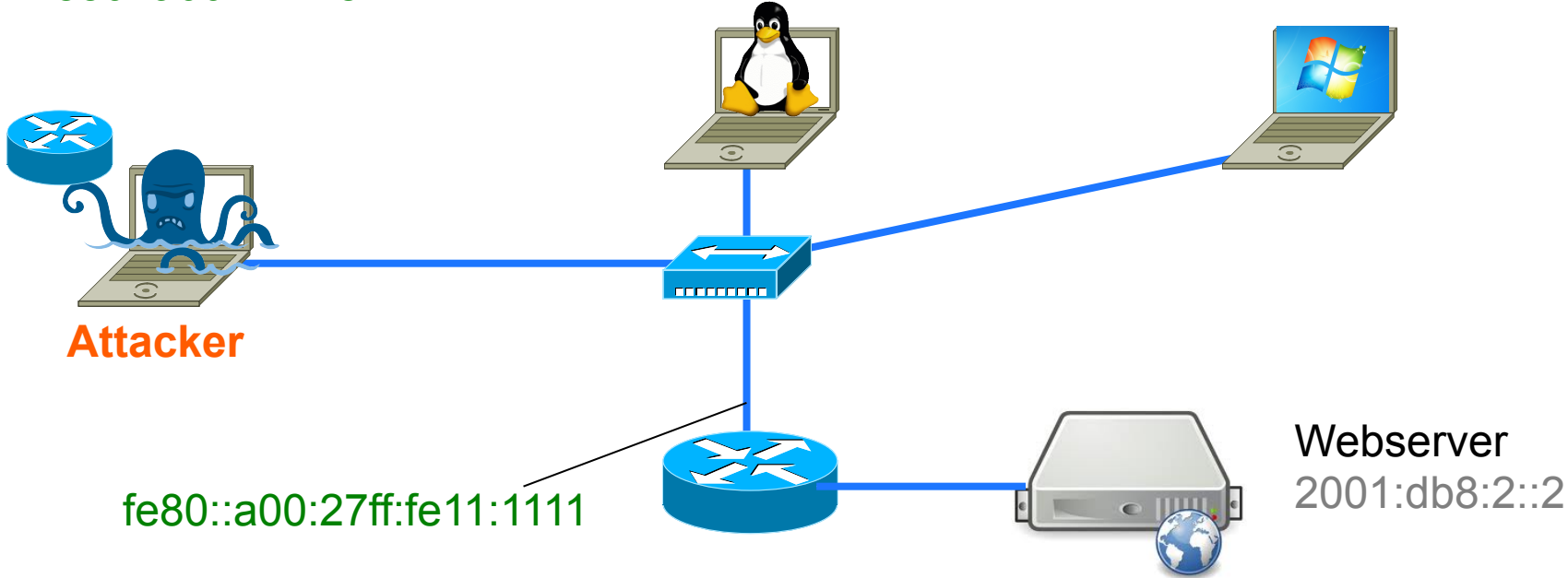
**Syntax:** dos-new-ip6 <interface>

# Attack: Duplicate Address Detection DOS

08:00:27:AA:AA:AA  
fe80:a00:27ff:feaa:aaaa

08:00:27:66:66:66  
fe80:a00:27ff:fe66:6666  
2001:db8:1::a00:27ff:fe66:6666  
GW: fe80::a00:27ff:fe11:1111

08:00:27:BB:BB:BB



# DAD DOS Mitigation

- NS/NA can't be blocked because it's used also for Address Resolution ("ARP")
- **But:** Many Switches can forward multicast packets only to the necessary ports
- feature is called "MLD snooping", check if it is enabled



Demo 4: **Add your  
addresses to the network**

**IPv6**

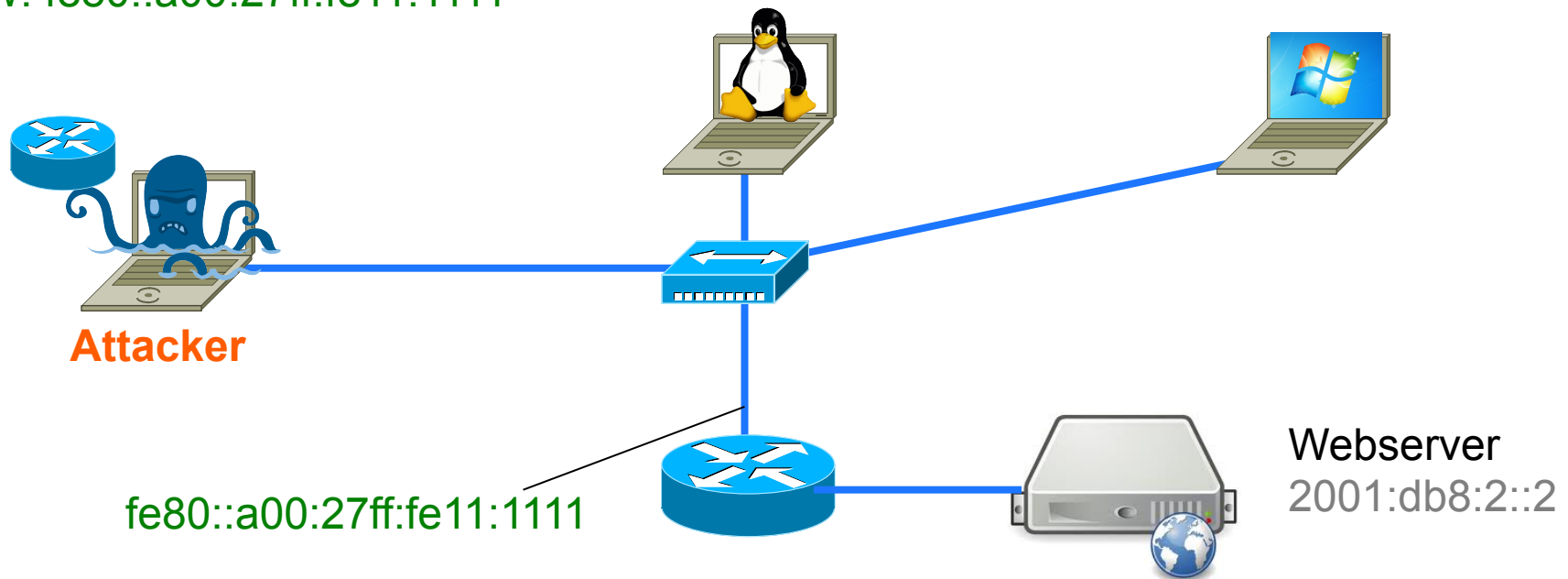
# Rogue Router configures new IP addresses in the network

```
Attack command:      fake_router6 eth0 1234::/64  
                    fake_router26 -A 5678::/64 eth0
```

# Attack: Add new addresses

08:00:27:AA:AA:AA	08:00:27:BB:BB:BB
fe80:a00:27ff:feaa:aaaa	fe80:a00:27ff:febb:bbbb
2001:db8:1::a00:27ff:feaa:aaaa	2001:db8:1::a00:27ff:febb:bbbb
dead:beef::a00:27ff:feaa:aaaa	dead:beef::a00:27ff:feaa:aaaa
GW: fe80::a00:27ff:fe11:1111	GW: fe80::a00:27ff:fe11:1111

08:00:27:66:66:66  
fe80:a00:27ff:fe66:6666  
2001:db8:1::a00:27ff:fe66:6666  
GW: fe80::a00:27ff:fe11:1111





## **This also works in an “IPv4 only” network**

IPv6-enabled hosts will configure IPv6 addresses and can then be attacked over IPv6 (second door)

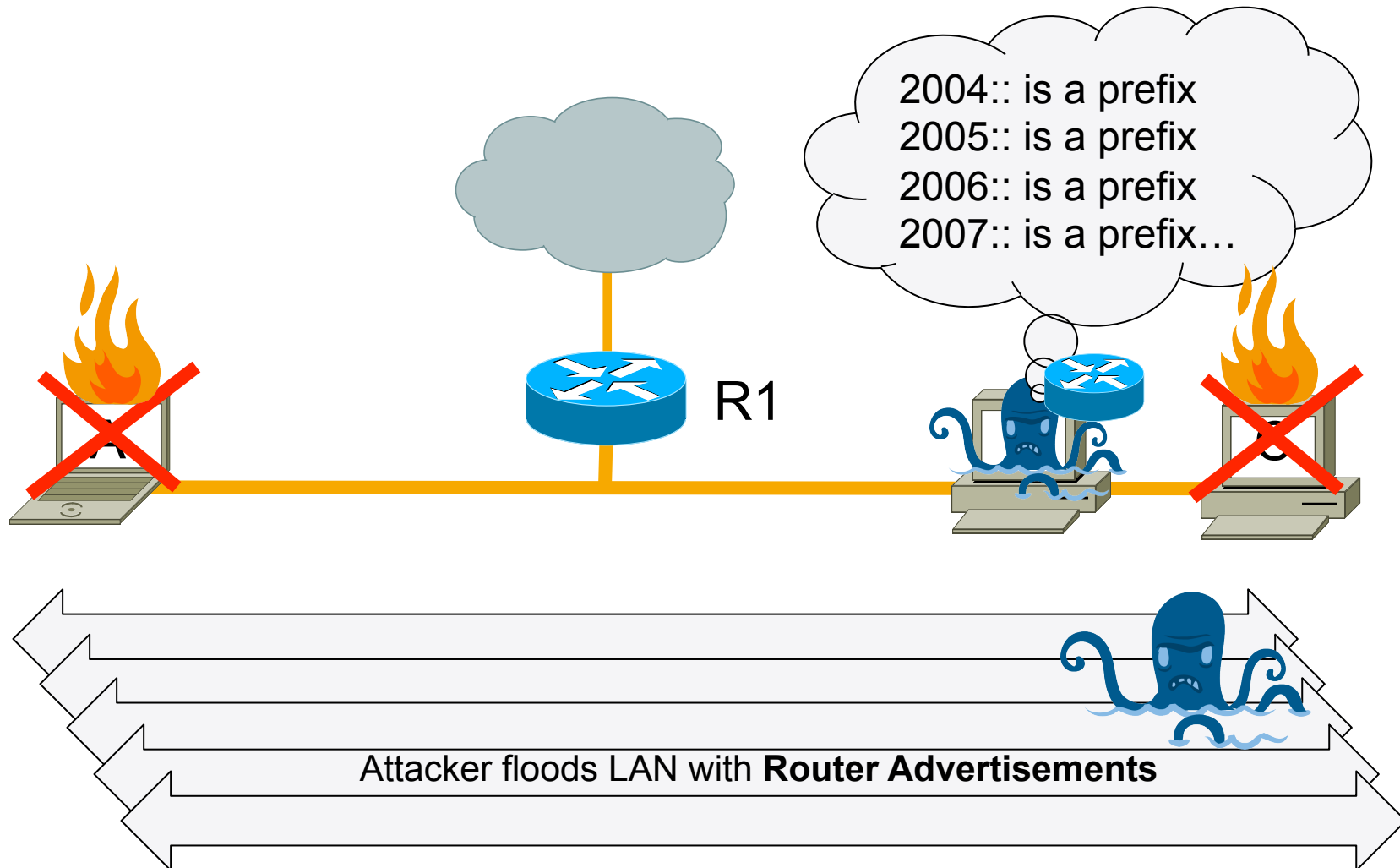


# Demo 5: **RA Flooding**

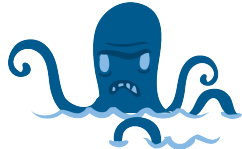
# IPv6



# Router Advertisement Flooding



# Router Advertisement Flooding



**flood\_router6, flood\_router26**

Flood the local network with router advertisements.

Each packet contains 17 prefix and route entries (only Version \_26)

-F/-D/-H add fragment/destination/hop-by-hop header to bypass RA guard security.

**Syntax:** flood\_router6 [-HFD] interface

**Example:** flood\_router6 eth0

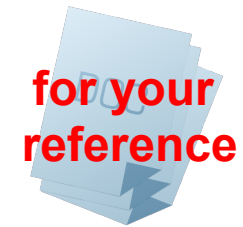


# Rogue RA Attack Conclusions



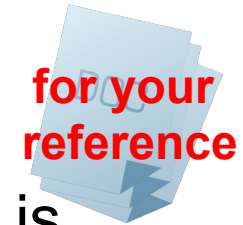
- Everybody on the local network can
  - add IPs, delete / change default router
  - DOS network
  - try a MITM attack
  - decrease Network-Performance
  - decrease System-Performance
  - crash Systems
  - Autoconf. IPv6 in IPv4-only network = open 2nd door

# Mitigation Approaches 1



- Disable IPv6 (hmmm...)
- Disable RA processing (but it's needed for DHCPv6, also)
- Filter on Switch: RA-Guard, Port-ACLs (can be bypassed using EH)
- Router Preference value on legitimate Router = High (works for misconfigured clients)
- Layer-2-Authentication IEEE 802.1X (heavyweight deployment)
- Host based filters configured to accept RAs only from valid Router addresses (works only in managed environment)

## Mitigation Approaches 2



- SEND - SEcure Neighbor Discovery (RFC 3971) is an approach to encrypt ND messages using public/private keys (not widely implemented)
- Deprecation Daemon: watch for incorrect RAs and then in turn send a deprecating RA with a router lifetime of zero (not for flooding)
- Partitioning, Microsegmentation or Host Isolation (Example: "Access Point Isolation Mode" in Cisco Wireless Routers)
- DHCPv6-only? No: RA informs about use of DHCPv6

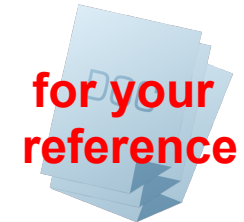


# *At least detect* rogue RAs & ND spoofing

for your  
reference

- With a generic **Intrusion Detection System**
  - signatures needed
  - decentralized sensors in all network segments needed
- With **NDPmon**
  - can monitor RAs, NAs, DAD-DOS
  - generates syslog-events and/or sends e-mails
  - free available at [ndpmon.sourceforge.net](http://ndpmon.sourceforge.net)
- Using **Deprecation Daemons**:
  - ramond, rafxid

# One size doesn't fit all.



Zone	Rogue RA Mitigation Measure	cost (+ o -)	feasibility	effect (+ o -)
Internal Network	Router-Preference=high / Monitor NDP Managed Switch (RAGuard, PACLs)	+/-	+	0/+
Internal Server-Zone	Router-Preference=high / Monitor NDP Disable RA processing	+	+	+
DMZ	Router-Preference=high / Monitor NDP Disable RA processing	+	+	+
Guestnet Wired	Router-Preference=high Managed Switch with RA Guard or Port ACLs	-	+	+
Guestnet Wireless	Router-Preference=high Partitioning	+/o	+	+



**Example for a remote attack**

**IPv6**

# Remote Neighbor Cache Exhaustion Attack

## Problem:

- Aggressive IPv6 address scanning consumes router resources
  - Big subnet, small neighbor cache table
  - neighbor cache is similar to IPv4 ARP entry  
(ip addr:phys. addr)
- ➔ A ping scan floods neighbor cache table (fast)

# Remote Neighbor Cache Exhaustion Attack

## Impact:

- Some routers break all interfaces
- Some routers break targeted interface
- At least legitimate entries are evicted from table

# Remote Neighbor Cache Exhaustion Attack

for your  
reference

## Mitigation:

- Ingress ACL allowing only valid destination and dropping the rest
- Maybe you have a built-in Rate limiter
- Cisco Feature: "IPv6 Destination Guard"
  - (is coming...)
- Workaround: Allocate /64, configure /120 (brakes SLAAC, maybe more)


## Some other Attacks:

- Multicast Listener Discovery DOS
  - Attacker messes with MLD messages
- Fragmentation Reassembly Time exceeded DOS
  - Attacker sends lot of fragmented packets with More-flag set
- Also well known attacks from IPv4 like
  - ICMP Redirect → ICMPv6 Redirect
  - ARP spoofing → Neighbor Cache spoofing

# Recommendations, Resources and Tools







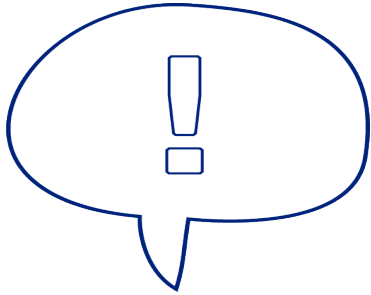
*"It's hard enough to deploy IPv6,  
let's deal with the Security stuff  
afterwards!"*



# 1. Secure existing Operations

- Do you have a **IPv6 Latent Threat** risk in your network?
- If yes take steps against it:

- Deactivate IPv6 or SLAAC where reasonable
- Filter tunnel traffic at the perimeter
- Update your monitoring (Rogue Router Advrts.)



## 2. Raise awareness at Management level

- Has IPv6 arrived on **the IT Management** Agenda?  
Priority – Resources – Budget
- Do you have an IPv6 **Integration Strategy**?
  - leverage existing life-cycles and projects
  - realistic, phased roadmap
  - Define a IPv6 Transition Manager
- Make sure **IT-Security** is involved!  
e.g. Security-Devices, Design decisions, NAT, Addressing plan, Security-Policy update



### 3. Build up Know-how

- Define a **Training Plan** - different people (roles) need different knowledge
- Build up a **Testing Lab** - to gain experiences & to test equipment
- Perform a **Pilot project** – which is not critical but also not only in the lab
- Learn from others – for example Local **IPv6 Councils**



## 4. Take into account the IPv6 readiness of your Security equipment

- Have an **Inventory** of your security equipment
- Define your IPv6 **Requirements**
- Do **Vendor Management** (IPv6-Roadmap?)
- Update **Purchasing Guidelines** and define a **Testplan**
- **Synchronise deployment** with security readiness!



## 5. Recognize and use opportunities

- **Start early** – avoid time pressure
- Leverage existing **Life cycles** of equipment
- Add IPv6 to the requirements of **existing projects**
- Prefer **step-by-step** approach (know dependencies)
- If indicated: use opportunity for a **network re-design**

# Recommended Resources

- S. Hogg/E.Vyncke: "IPv6-Security"

Cisco Press

- NIST - Guidelines for the Secure Deployment of IPv6

<http://csrc.nist.gov/publications/nistpubs/800-119/sp800-119.pdf>

- Mailing List ipv6hackers

<http://lists.si6networks.com/listinfo/ipv6hackers>

- IPv6 Security Whitepaper, Slides and Videos from Eric Vynce, Fernando Gont, Marc Heuse, Scott Hogg, Enno Rey, Antonios Atlasis

scan Internet with your preferred search engine

# Recommended IPv6 Security Tools

Tool suite	Description	Platform / License
<b>THC The Hacker Choice IPv6 Attack Toolkit</b> <i>Marc Heuse &amp; others</i>	<ul style="list-style-type: none"><li>• lots of small tools (<math>\approx 70</math>)</li><li>• poorly documented</li><li>• pioneer work</li><li>• C library available</li></ul>	<ul style="list-style-type: none"><li>• C</li><li>• Linux</li><li>• GNU/AGPL</li></ul>
<b>SI6 Networks</b> Security assessment and troubleshooting toolkit for IPv6 <i>Fernando Gont</i>	<ul style="list-style-type: none"><li>• a few comprehensive tools (<math>\approx 12</math>)</li><li>• lots of parameters</li><li>• well documented</li><li>• mature</li></ul>	<ul style="list-style-type: none"><li>• C</li><li>• Linux/xBSD/OS X</li><li>• GNU/GPL</li></ul>
<b>chiron</b> All-in-one IPv6 Penetration Testing Framework <i>Antonios Atlasis</i>	<ul style="list-style-type: none"><li>• Craft arbitrary IPv6 packets to test IDS/IPS evasion</li><li>• And other interesting tools</li></ul>	<ul style="list-style-type: none"><li>• Python/Scapy (modified)</li><li>• Linux</li><li>• GNU/GPL</li></ul>





# Thank you!



Find more here:

Blog: [securityblog.switch.ch](http://securityblog.switch.ch)

Twitter: [@switchcert](https://twitter.com/switchcert)

# Some backup / reference slides



# Differences between IPv4 and IPv6



Property	IPv4	IPv6
Address size and network size	32 bits, network size 8-30 bits	128 bits, network size 64 bits
Packet header size	20-60 bytes	40 bytes
Header-level extension	limited number of small IP options	unlimited number of IPv6 extension headers
Fragmentation	sender or any intermediate router allowed to fragment	only sender may fragment
Control protocols	mixture of non-IP (ARP), ICMP, and other protocols	all control protocols based on ICMPv6
Minimum allowed MTU	576 bytes	1280 bytes
Path MTU discovery	optional, not widely used	strongly recommended
Address assignment	usually one address per host	usually multiple addresses per interface
Address types	use of unicast, multicast, and broadcast address types	broadcast addressing no longer used, use of unicast, multicast and anycast address types
Address configuration	devices configured manually or with host configuration protocols like DHCP	devices configure themselves independently using stateless address autoconfiguration (SLAAC) or use DHCP

Source: NIST 800-119

# IPv6 Transition Mechanisms

- Transition Mechanisms enable communication during the transition phase between v4 and v6 (Co-Existence)
- Three approaches:
  - **Dual Stack** - IPv4 and IPv6 are running parallel
  - **Tunneling** - IPv6 over IPv4 (and IPv4 over IPv6)
  - **Translation** - IPv6 to IPv4 (and IPv4 to IPv6)

➔ Transition Mechanisms introduce new security threats and attack vectors to the network.

# Generating Interface ID from MAC using modified EUI-64 format



08:00:27:AA:AA:AA

Step 1: Insert FFFE to get 64 Bit

0800:27FF:FEAA:AAAA

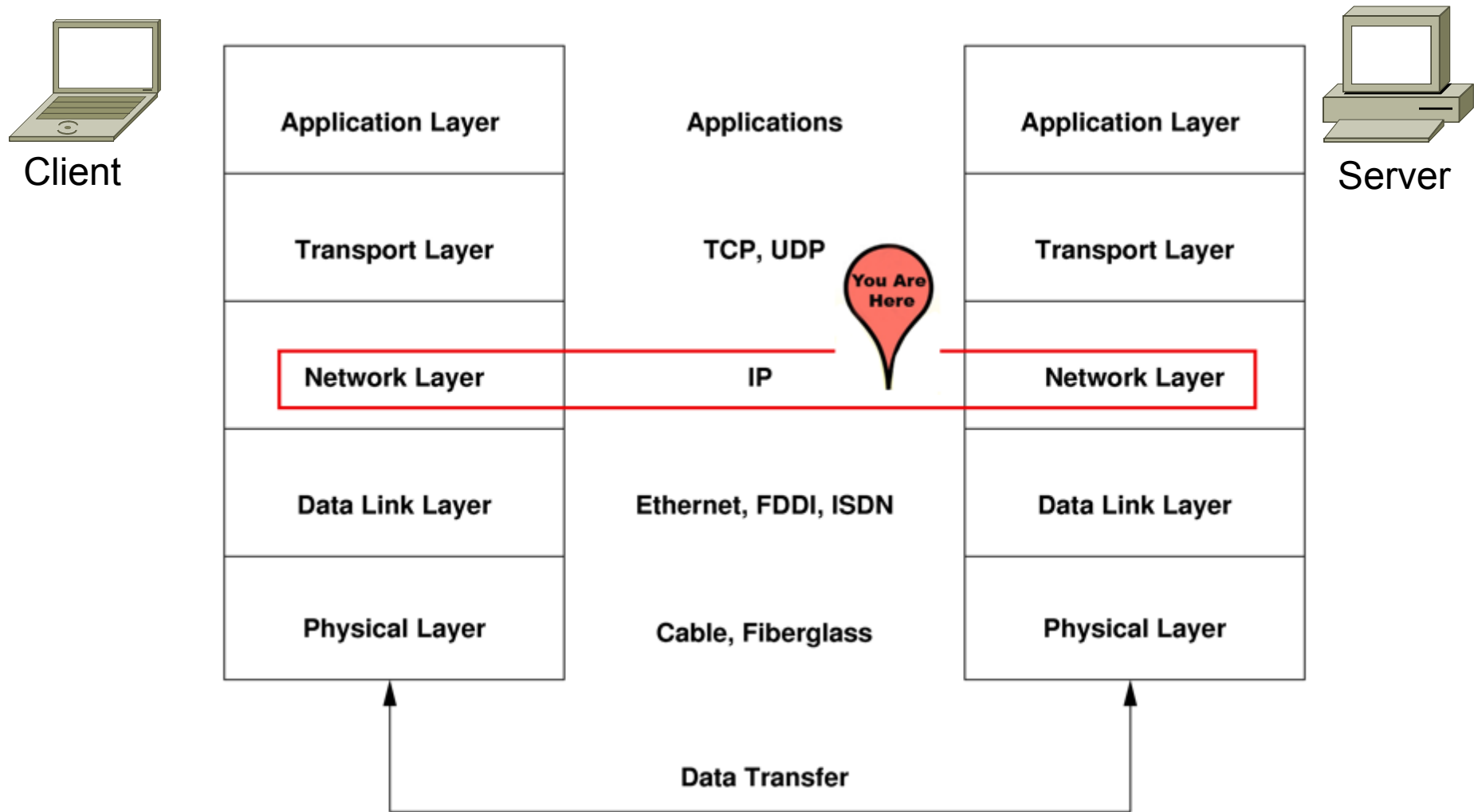
Step 2: Toggle Bit 7

0000 1000 = 08

0000 1010 = 0A

0A00:27FF:FEAA:AAAA

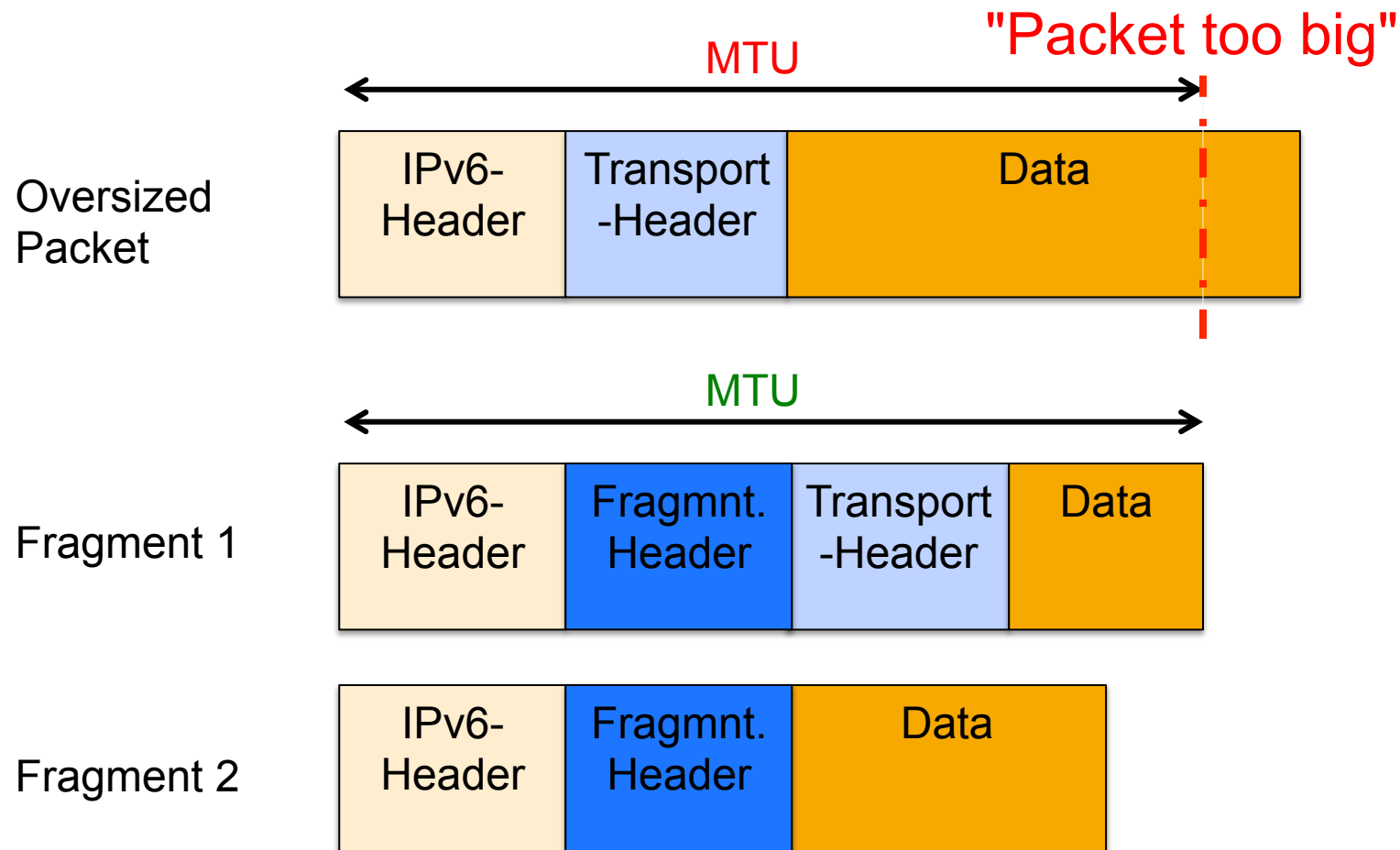
# Open Systems Interconnection model



# Requirements for (Security) Network Equipment - Some Resources

- RIPE: RIPE-554 "Requirements for IPv6 in ICT Equipment"
  - RIPE document that lists mandatory / optional RFCs for different types of equipment
  - Contains a proposed text for tenders / RFPs
  - <http://www.ripe.net/ripe/docs/current-ripe-documents/ripe-554>
- IPv6-Forum: IPv6 Ready Logo Program
  - Certification Program that covers basic IPv6 requirements and some advanced features, but it is not exhaustive.
  - <http://www.ipv6ready.org/>
- NIST/USGv6: IPv6 Profile and Testing Program
  - <http://www.antd.nist.gov/usgv6/>

# The sender can fragment IP datagrams into multiple packets and the IDS/IPS/Firewall/Receiver has to deal with it





# Some fragmentation attacks

- Attacker can try to **bypass filtering/detection** (IDS/IPS evasion technique)
  - by putting the attack into many small fragments
  - by combination of multiple extension headers and fragmentation so that layer 4 header is in 2<sup>nd</sup> fragment
- Attacker can **exploit weaknesses in the destination**
  - Overlapping fragments, nested fragments
- Attacker can **DOS destination**
  - send lots of incomplete fragment sets (M-flag 1 → more fragments)

# ICMPv6 Security Concerns (according to RFC 4890)

- **Denial-of-Service Attacks**
- **Probing** to identify topology and hosts
- **Redirection Attacks** using the Redirect message
- **Renumbering Attacks** (Renumbering messages are required to be authenticated with IPsec)
- **Covert conversation** through the payload of ICMPv6 error messages