IPv6 Security

SWITCH

Frank Herberg
frank.herberg@switch.ch

Berlin, 18 June 2015

Agenda

- **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, but…

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

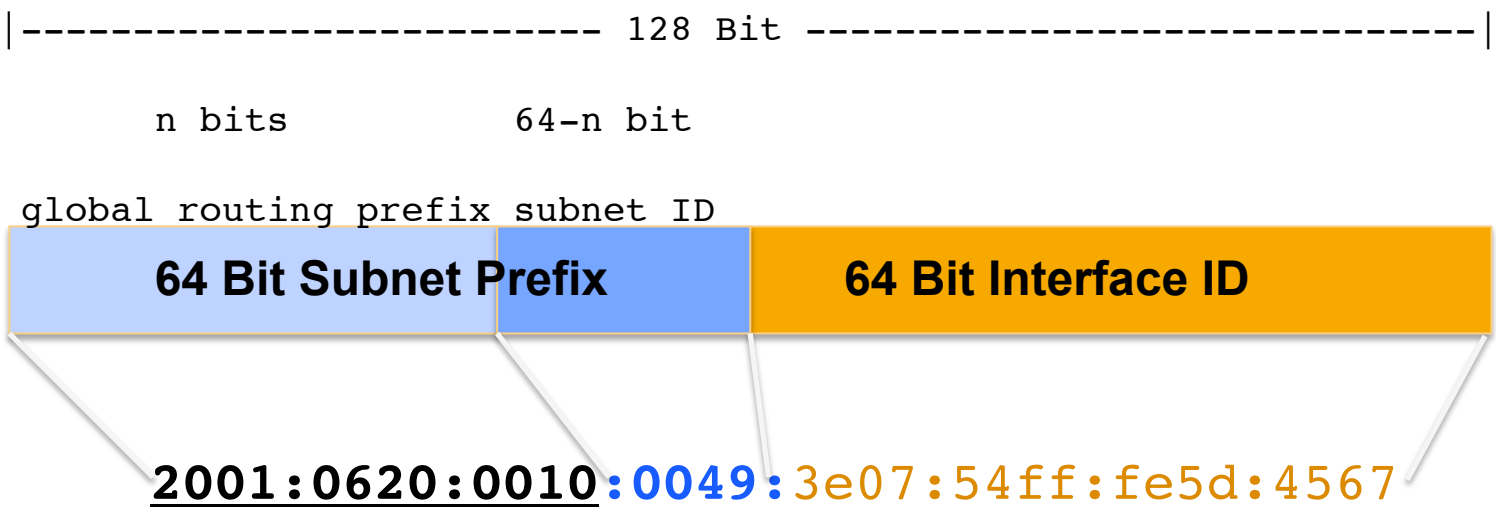
- **IANAs global pool of available IPv4 addresses was exhausted on 1 February, 2011**
- The five Regional Internet Registries each received one of the IANA's five reserved /8 blocks
- Policy: A LIR may **receive only 1,024 IPv4 addresses**, even if they can justify a larger allocation

…but the Internet is growing

- Mobile Devices, Internet of Things,...

- 128 bit IPv6 address space is 2^{96} x larger than IPv4's 32 bit

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`

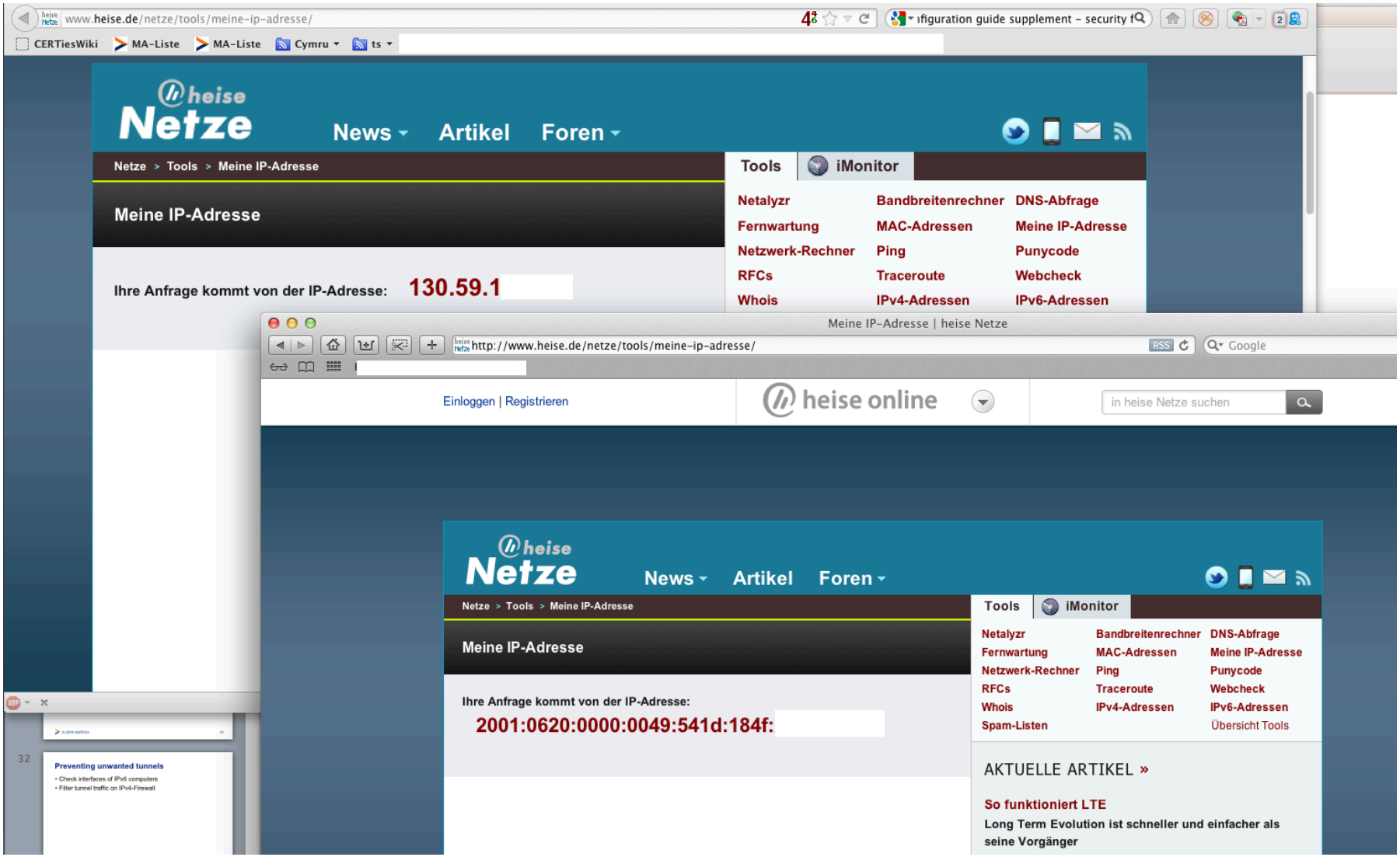
Part 1: Introduction to IPv6 Security

Multiple IPv6 addresses per interface (plus the IPv4 address)

IPv4	173.194.32.119
Link Local	fe80::3e07:54ff:fe5d:abcd
Global	2001:610::41:3e07:54ff:fe5d:abcd*
Privacy Extensions = random / temporary	
Global PE	2001:610::41:65d2:e7eb:d16b:a761**
Unique Local Address = ‘private’ IPv6 address	
ULA	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



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)

…IDS/IPS (attacks distributed over 4/6)

IPv6 address notation isn't unique

full form:
fe80:0000:0000:0000:0204:61ab:fe9d:f156

drop leading zeroes:
fe80:0:0:0:204:61ab:fe9d:f156

collapse multiple zeroes to '::':
fe80::204:61ab:fe9d:f156

dotted quad at the end:
fe80::204:61ab:254.157.241.86

IP address based protection 1 - Blacklists

- 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

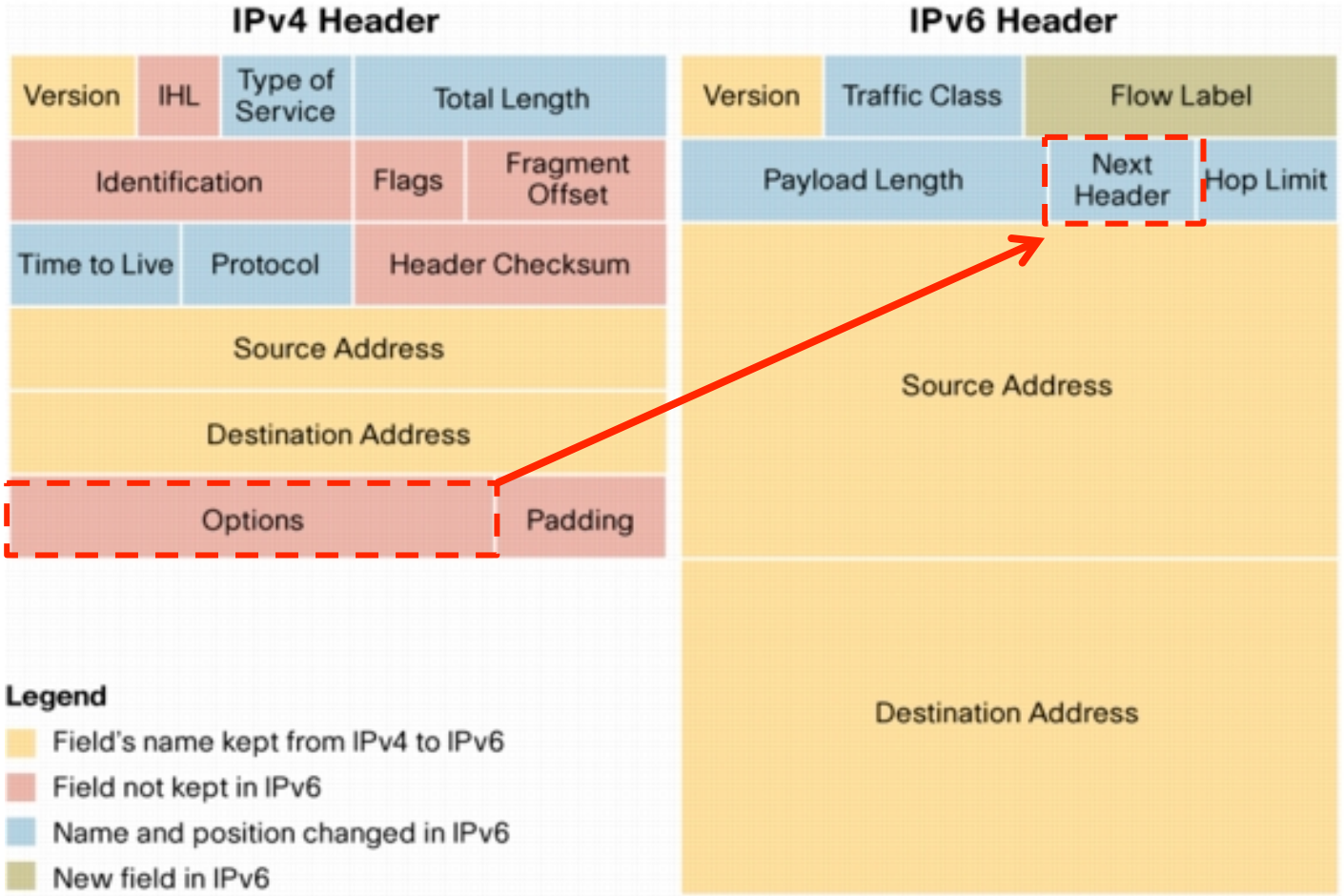
- IPv4 based Access Control Lists (ACLs) only protect the IPv4 access
- Enable IPv6? ➔ Review all your ACLs!

Checklist

- ☑ 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 (40 Byte)
options go into Extension Header

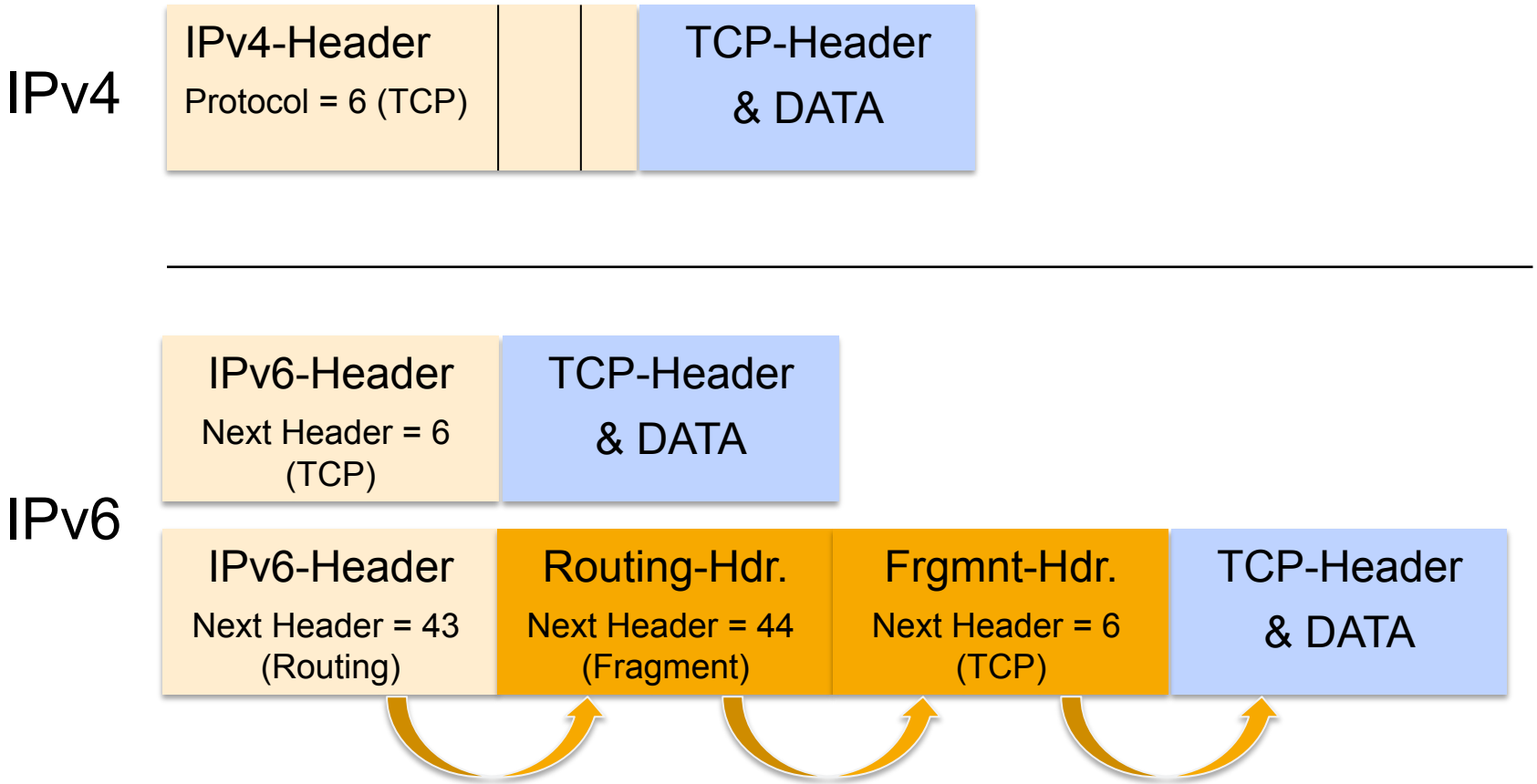


Extension Header Examples

No.	Name	Functions	Remarks
0	Hop-by-Hop-Options	carries options for hops, e.g. Router Alert (for MLD, RSVP)	**must be examined by every hop on the path** Must be first EH, only one allowed per packet
60	Destination Options	carries options for destination (e.g. for Mobile IPv6)	**processed by destination node only***
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

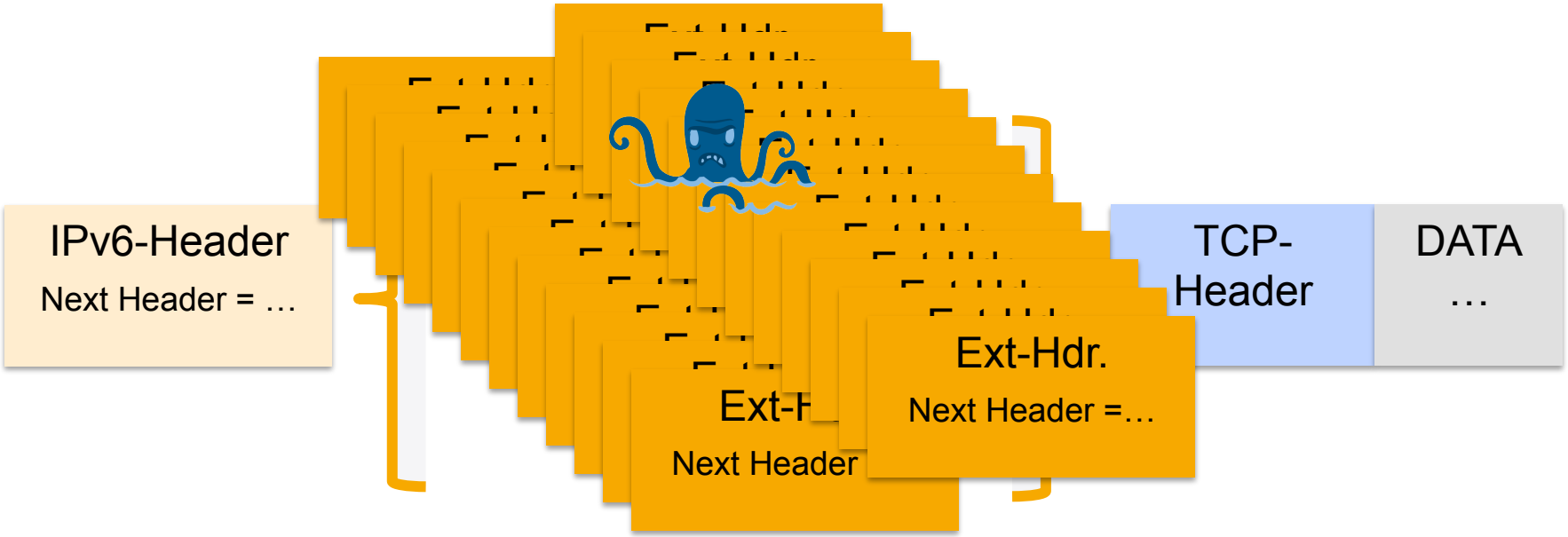
  "**any order** and occurring **any number** of times in the same packet"

Possible Threat: High Number of EHs

- An attacker could create packet with high number of EH

➔ to try to avoid FW / IPS

➔ might crash or DOS the destination system

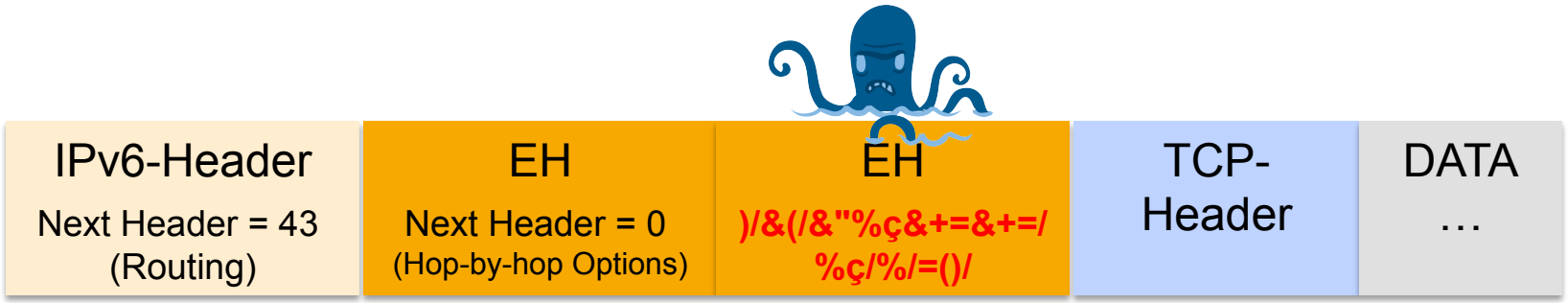


Mitigation option: Drop packets with more than x EHs

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 look into crafted EHs

➔Destination System might crash

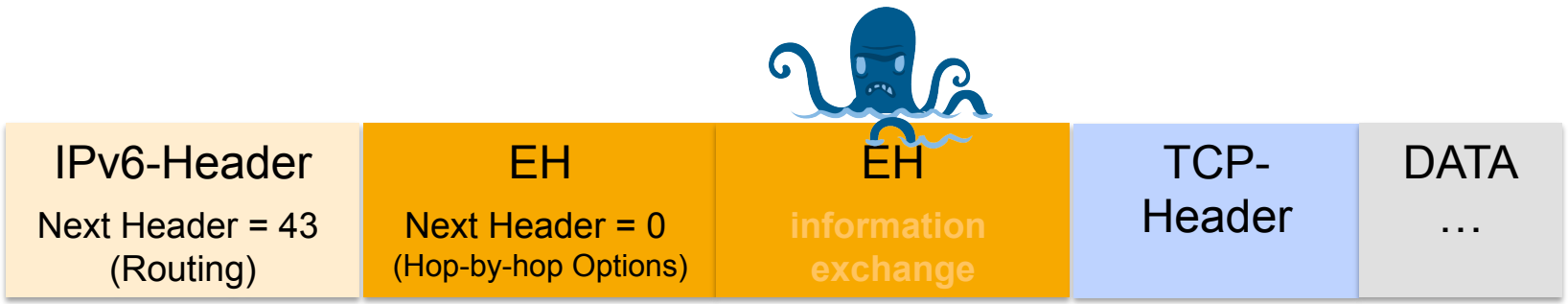


Mitigation option: Perform sanity checks on EH (format / no. of options)

Possible Threat: Covert Channel

- An attacker could use Extension Headers as a covert channel

➔to exchange payload undiscovered

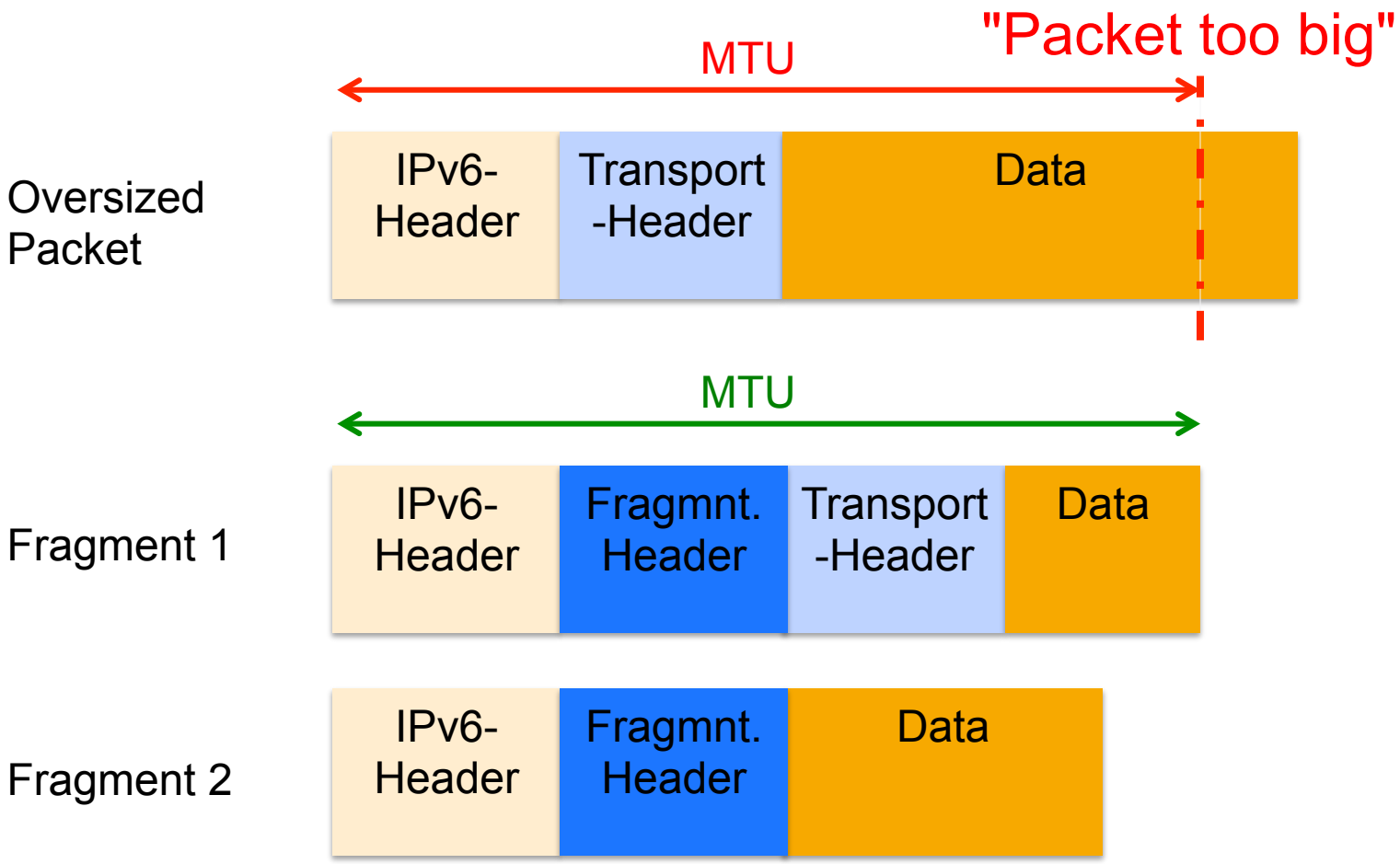


Mitigation option: Drop unknown EH

Extension Headers *increeeaaase* complexity

To make it worse: Add fragmentation to it!

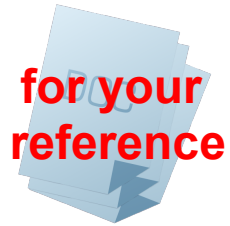
The sender can fragment IP datagrams into multiple packets and the IDS/IPS/Firewall/ Receiver has to deal with it



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 2nd 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)

Preventing Fragmentation Attacks



You can

- monitor the amount of fragmented packets

➔high increase might indicate attack

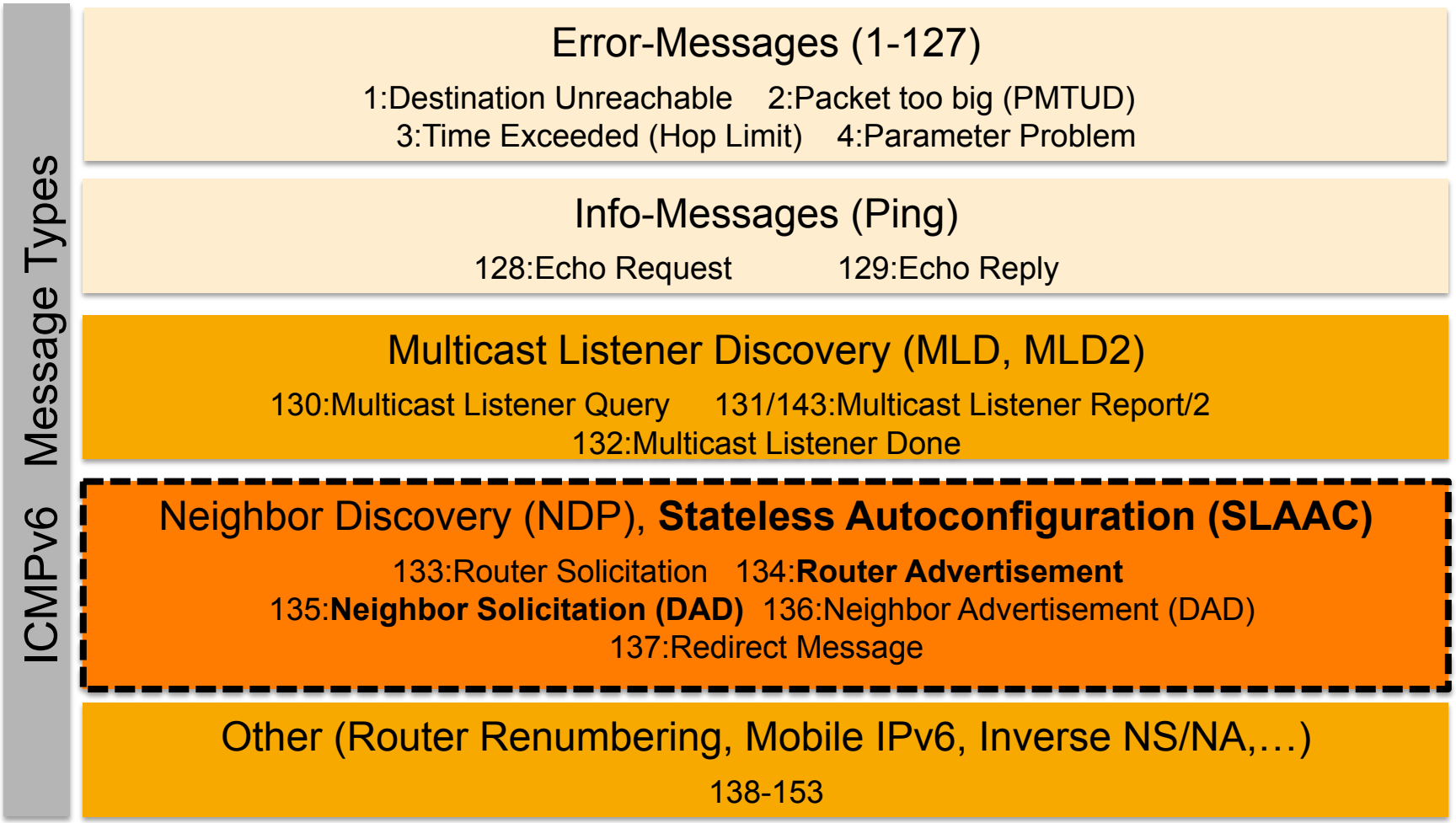
- block fragments which are below a certain size (if not the last one of a set [M-flag=0])

➔don't appear in proper communication

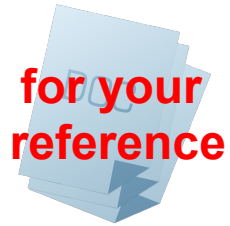
- look for Inspection capabilities of fragmented packets
  - e.g. Cisco: Virtual Fragment Inspection (VFR)

```
ipv6 virtual-reassemly
```

ICMPv6 is more complex



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

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

IPv6 Tunneling mechanisms can be misused and attacked…

TEREDO

6in4

6to4

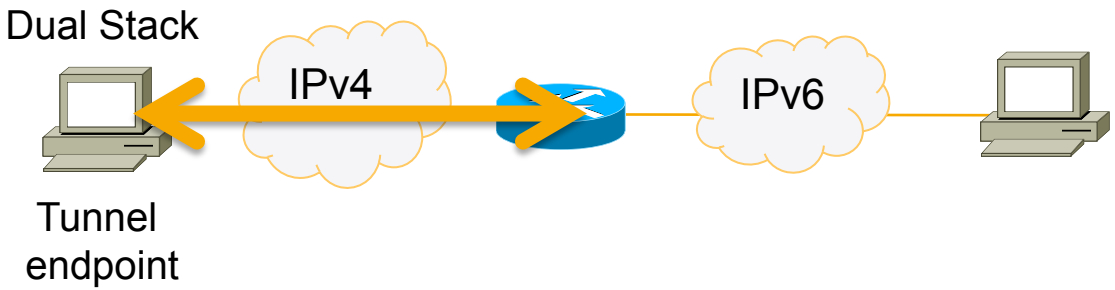
6rd

ISATAP

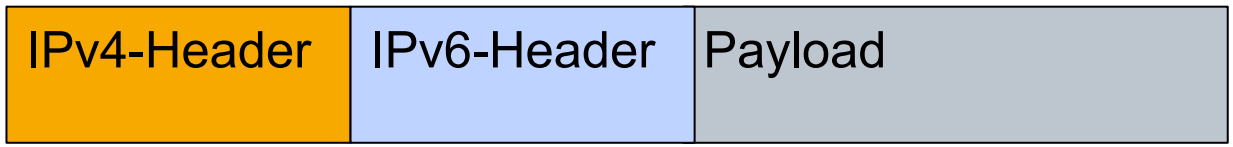
…different sorts of tunnels around

Tunneling: transport of IPv6 pakets across IPv4 infrastructure

Host-to-Site:



Site-to-Site:



Some Tunneling Characteristics

- Tunnel endpoints can be configured manually or **automatically**
- Tunnels can be configured deliberate or **unknowingly**
- 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** (can you trust them?)

Detect IPv6 tunnels in network logs

Look inside logs / NetFlow records:

- IPv4 Protocol 41 tunnel traffic (ISATAP, 6to4)
- IPv4 UDP 3544 tunnel traffic (Teredo)
- 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**

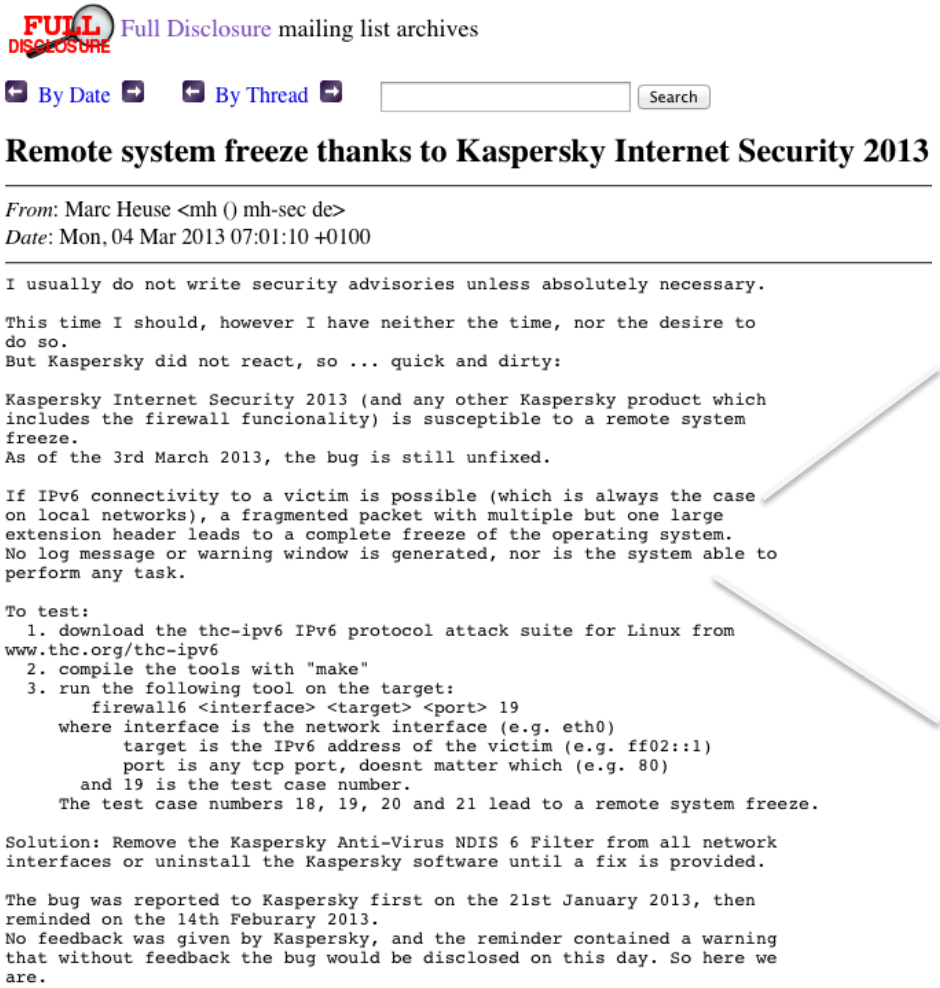
  Vendors have to deal with complexity and a moving target

- **… regarding Know-how**

  Often little or now Know-how

  And it needs time!

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 funcionality) 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
  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 Feburary 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
- Misconfigured clients can tie up your network

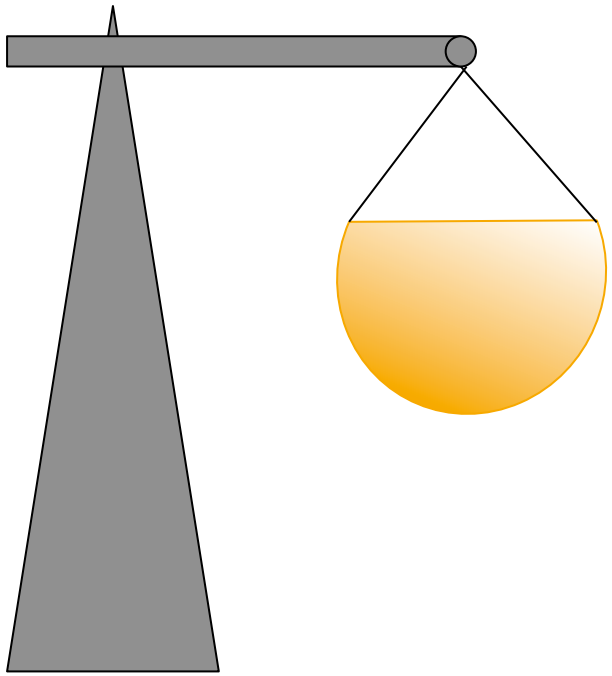
🡐 no IPv6 Monitoring / no IPv6 Knowledge

Opportunities for improved IT-Security?

Yes!

- Review the existing level of security
- Consolidation of the Network-Design / Re-documentation!
- IPv6 Addressing plan – more or less Policy friendly
- Rethink NAT vs. real Security (operational cost)
- Preparation for future security features vs. maintaining of legacy technology

Bottom line: How IPv6 affects IT-Security

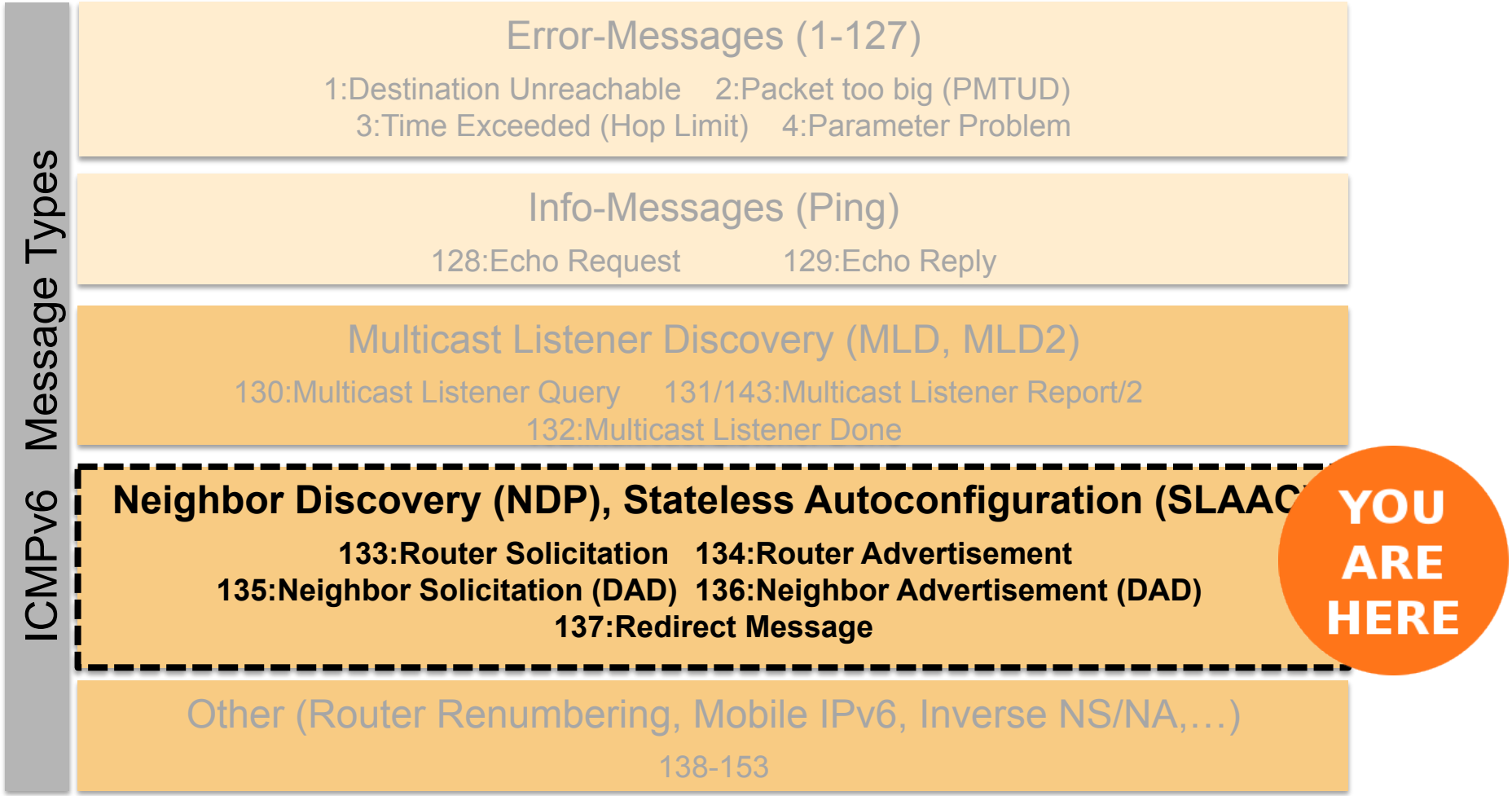


- Higher complexity (protocol and network)
- Lower maturity (especially security devices)
- Less Know-how / experience
- New / more Attack vectors
- Less visibility (Monitoring)
- Already active in "IPv4-only" net
- A lot of changes (also new opportunities to improve things)


Part 2:
Selected IPv6 attacks
IPv6

Some preparation needed:
How Address configuration works in IPv6

ICMPv6

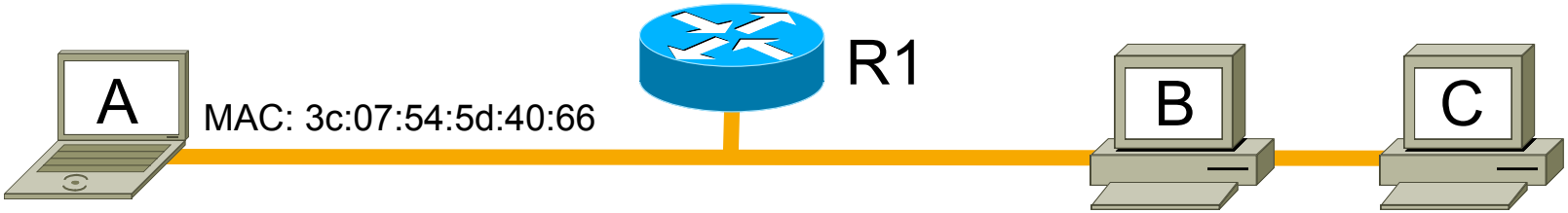


Neighbor Discovery Protocol consists of 5 ICMPv6 Message Types (133-137)

Multiple functions:

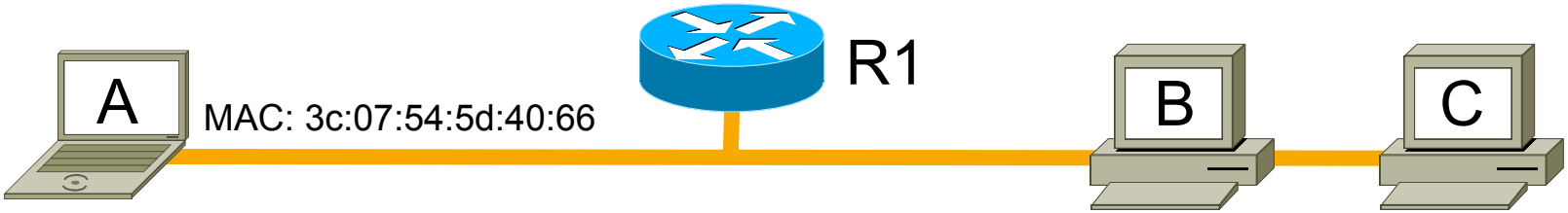
- Autoconfigure IP addresses (SLAAC)
- Find gateway routers (SLAAC)
- Detect duplicate addresses (DAD)
- Tell the node to use DHCPv6
- Discover other nodes on the link
- Determine link-layer addresses (Address Resolution)
- Maintain neighbor reachability information
- Redirects

Initial status: ‘A’ has a MAC address



Network interface comes up...

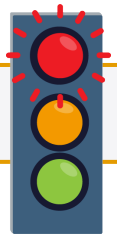
SLAAC Step 1: configure link-local address



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

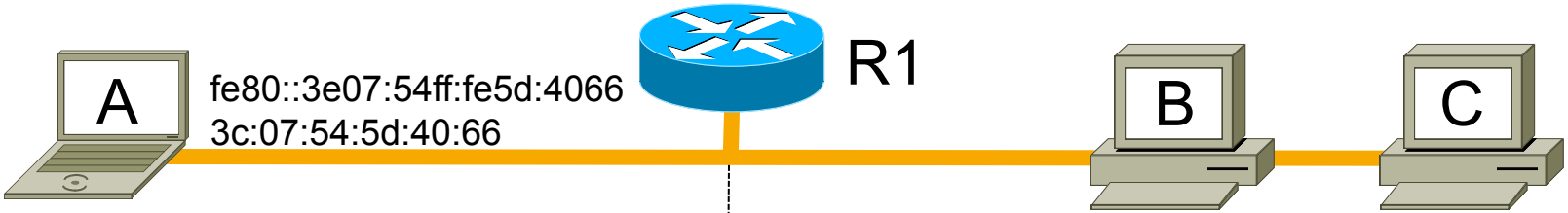
Send **NS** for **DAD** (:: => Solicited-Node multicast addr)

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



Send **RS** to All-Router-Multicast-Address (ff02::2)

RA: "Prefix is 2001:620:0:49::"*

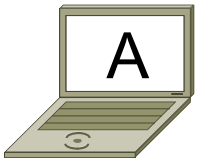
If **RA** received: generate global routable address(es) from received prefix(es) and configure default route

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

Either receive a **NA** to show an address conflict: **don't use address**

or **configure Global Address(es)** 2001:....

SLAAC successful:



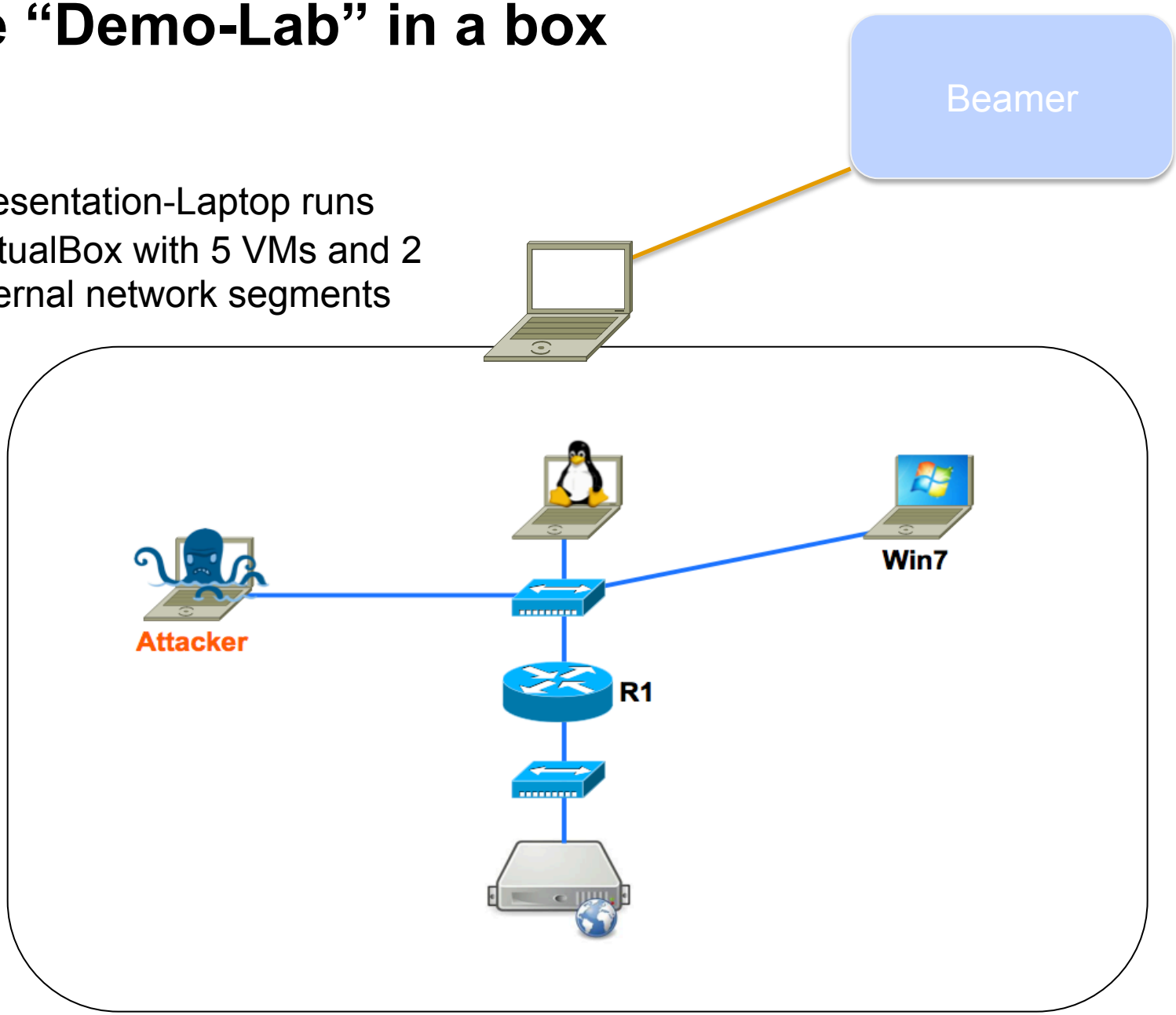
eth0:
Link Layer Address: 3c:07:54:5d:40:66
Link Local Address: fe80::3e07:54ff:fe5d:4066
Global Address: 2001:620::49:3e07:54ff:fe5d:4066
Global Address: 2001:620::49:1c78:9b29:27c1:7564

- Default Router Address
- Options (RDNSS,…)

Demo setup

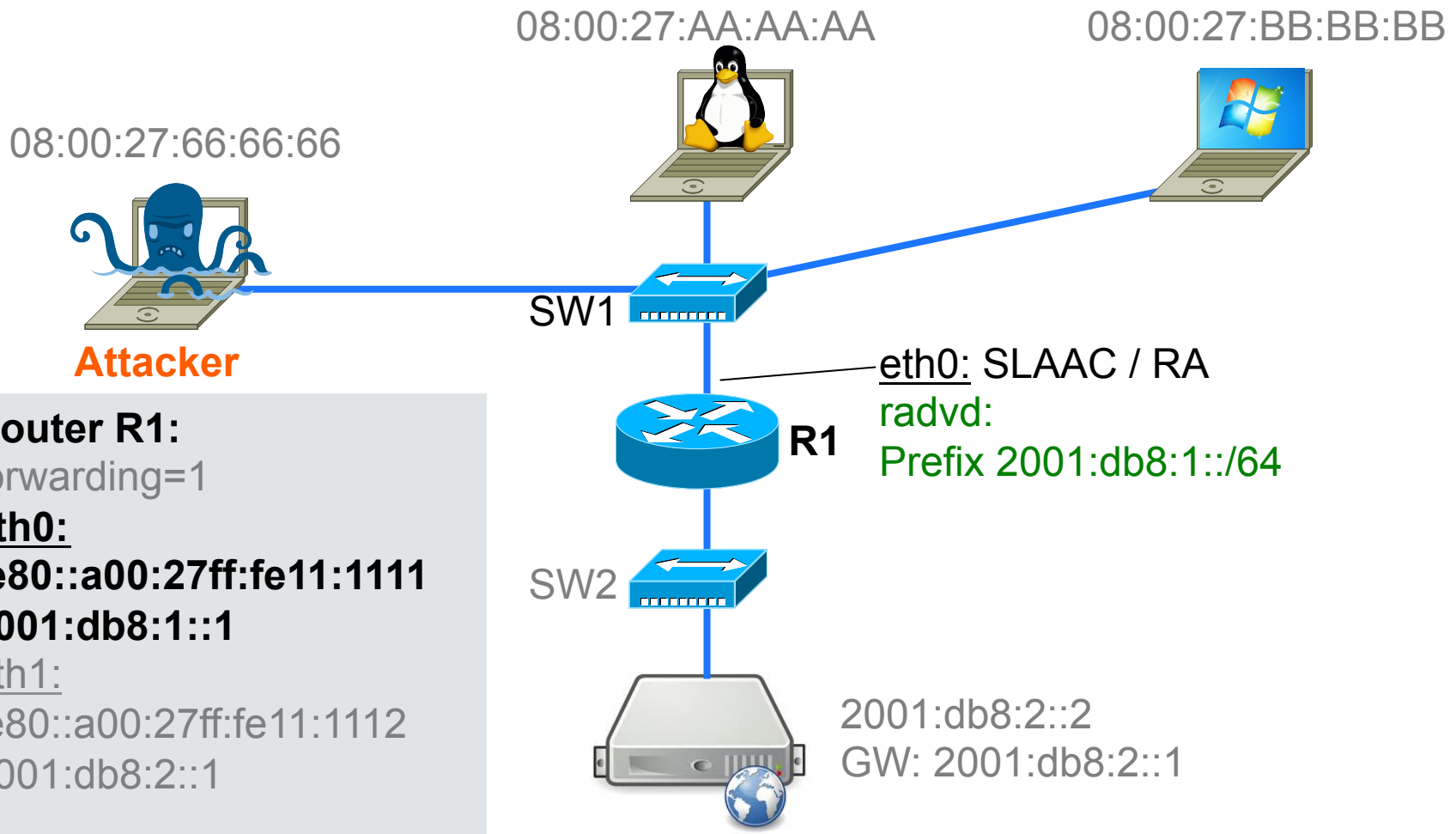
The “Demo-Lab” in a box

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



Lab Configuration

- 3 Clients
- 1 IPv6 Router
- 1 Webserver



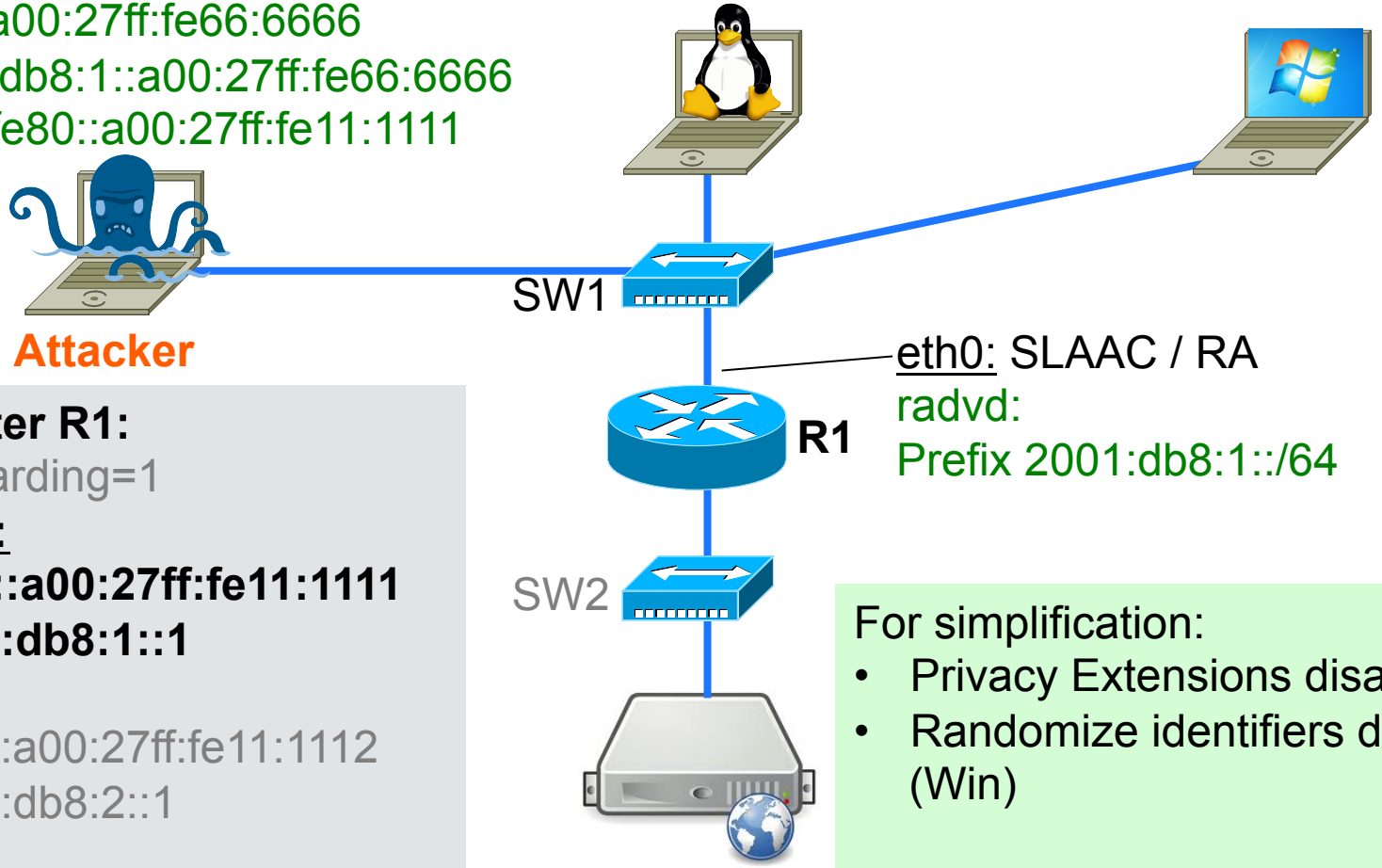
Router R1:
forwarding=1
eth0:
fe80::a00:27ff:fe11:1111
2001:db8:1::1
eth1:
fe80::a00:27ff:fe11:1112
2001:db8:2::1

Lab Configuration after Autoconfiguration

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

08:00:27:BB:BB:BB
fe80:a00:27ff:febb:bbbb
2001:db8:1::a00:27ff:febb:bbbb
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



eth0: SLAAC / RA
radvd:
Prefix 2001:db8:1::/64

Router R1:
forwarding=1
eth0:
fe80::a00:27ff:fe11:1111
2001:db8:1::1
eth1:
fe80::a00:27ff:fe11:1112
2001:db8:2::1

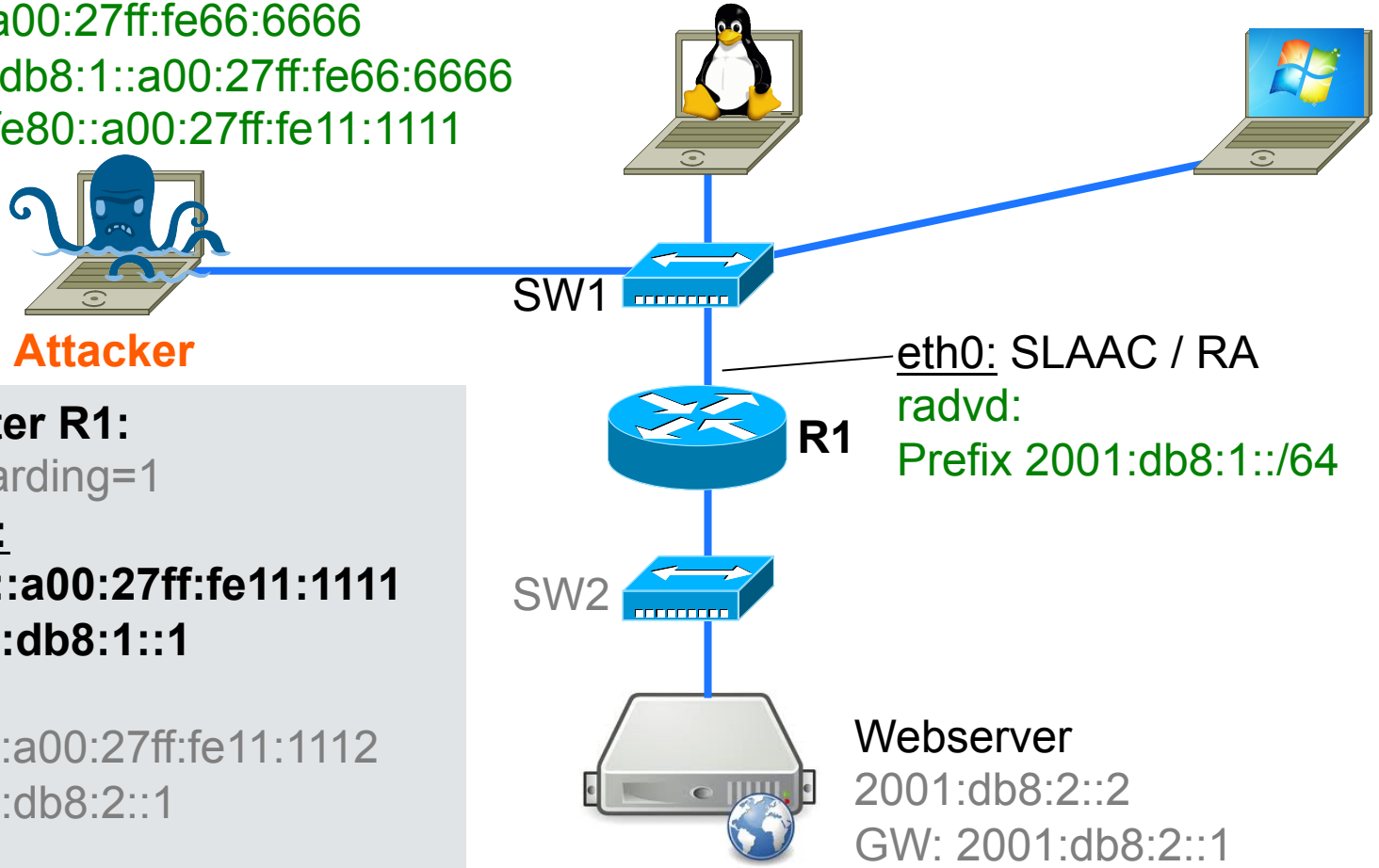
For simplification:
- Privacy Extensions disabled
- Randomize identifiers disabled (Win)

Access Webserver: http://[2001:db8:2::2]/

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

08:00:27:BB:BB:BB
fe80:a00:27ff:febb:bbbb
2001:db8:1::a00:27ff:febb:bbbb
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



Router R1:
forwarding=1
eth0:
fe80::a00:27ff:fe11:1111
2001:db8:1::1
eth1:
fe80::a00:27ff:fe11:1112
2001:db8:2::1


Its Demo time!
Selected IPv6 attacks
IPv6

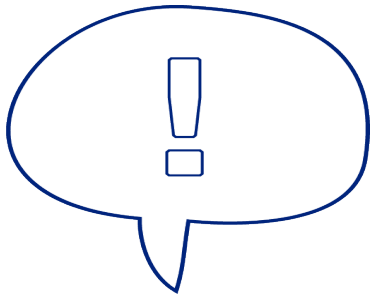
Recommendations, Resources and Tools

"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– Ressources – 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, Adressing 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

  not critical but also not only in the lab

- Inform and learn from others

  Swiss IPv6 Council,…

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**
- 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
- Improve not degrade IT-Security by means of collaboration

Suggested 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

  look up in Internet with your preferred search engine

Tool suite	Description	Platform / License
THC The Hacker Choice IPv6 Attack Toolkit *Marc Heuse & others*	• lots of small tools • thriftily documented • pioneer work • not stable or well tested	• C • Linux • GNU/AGPL
SI6 Networks Security assessment and troubleshooting toolkit for IPv6 *Fernando Gont*	• a few comprehensive tools • lots of parameters • well documented • mature	• C • Linux/xBSD/ OS X • GNU/GPL
FT6 Firewall Tester for IPv6 *Oliver Eggert*	• Client-Server-Application • GUI • carries a set of test cases • well documented • no further development	• Python/ Scapy/PyQt4 • Lin/Win/OS X • CC (BY-NC-SA)
chiron All-in-one IPv6 Penetration Testing Framework *Antonios Atlasis*	• ~~new kid on the block –~~ • Comprehensive useful tool set	• Python/ Scapy (modified) • Linux • GNU/GPL

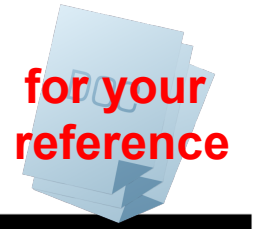
Q&A

Find more here:

Blog: securityblog.switch.ch

Twitter: @switchcert

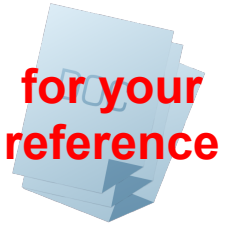
Differences between IPv4 and IPv6



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

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

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/