IPv6 Challenge

Fernando Gont

SI6 NETWORKS

AIS 2017 Hackathon
Nairobi, Kenya. May 27-28, 2017

About...

- Security Researcher and Consultant at SI6 Networks
- Published:
  - 20 IETF RFCs (9 on IPv6)
  - 10+ active IETF Internet-Drafts
- Author of the SI6 Networks' IPv6 toolkit
  - http://www.si6networks.com/tools/ipv6toolkit
- I have worked on security assessment of communication protocols for:
  - UK NISCC (National Infrastructure Security Co-ordination Centre)
  - UK CPNI (Centre for the Protection of National Infrastructure)
- More information at: http://www.gont.com.ar

Agenda

- Discuss two recent IETF RFCs (RFC6946 & RFC8021)
- Test their implementation
- Then:
  - Document the tests in an IETF Internet-Draft, **OR**,
  - Produce an implementation of such RFCs in open source OSes

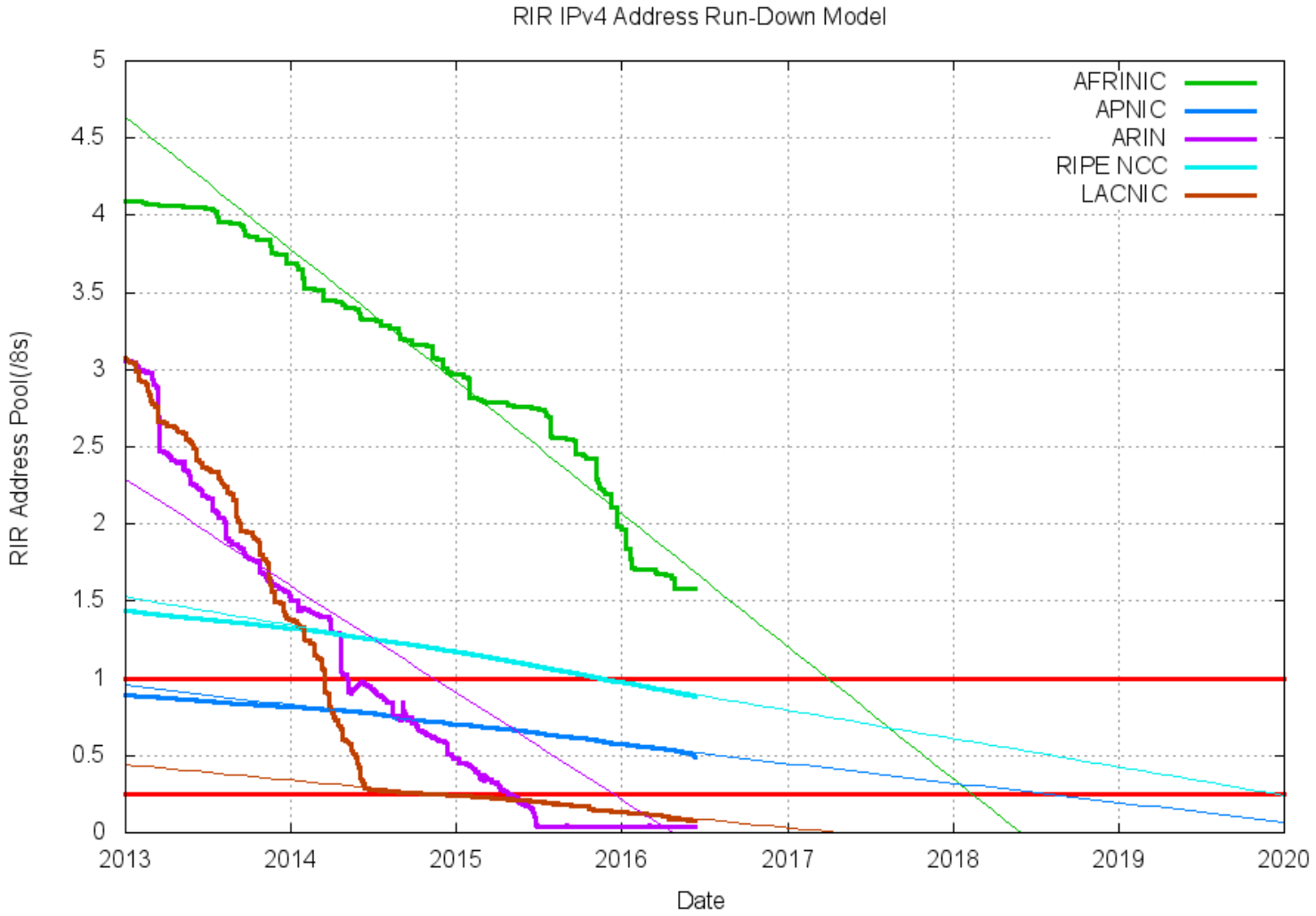
Brief introduction to IPv6

IPv4 address exhaustion

- The Internet relies on unique addresses for host communication
- More than 20 years ago it was already evident we'd eventually run out of IPv4 addresses
- Network Address Translators (NATs) have served as a stop-gap
- But nevertheless we're hitting IPv4 address exhaustion

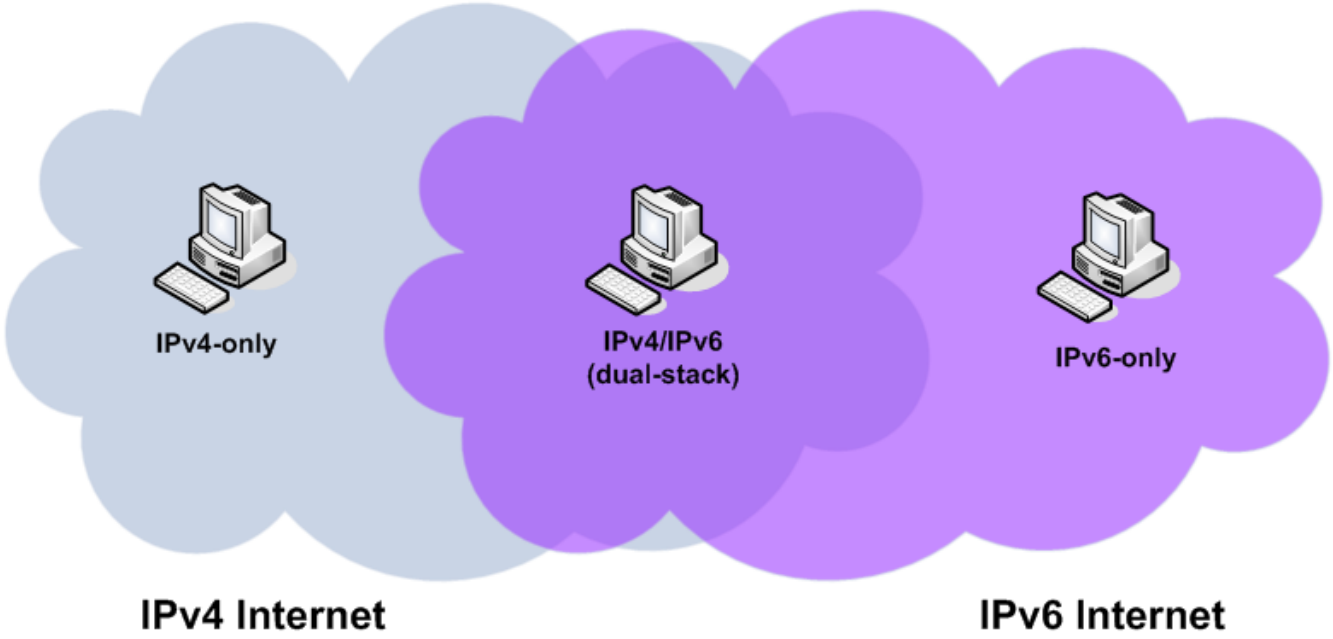
IPv4 address exhaustion (II)

- IPv4 address exhaustion, as predicted by Geoff Huston



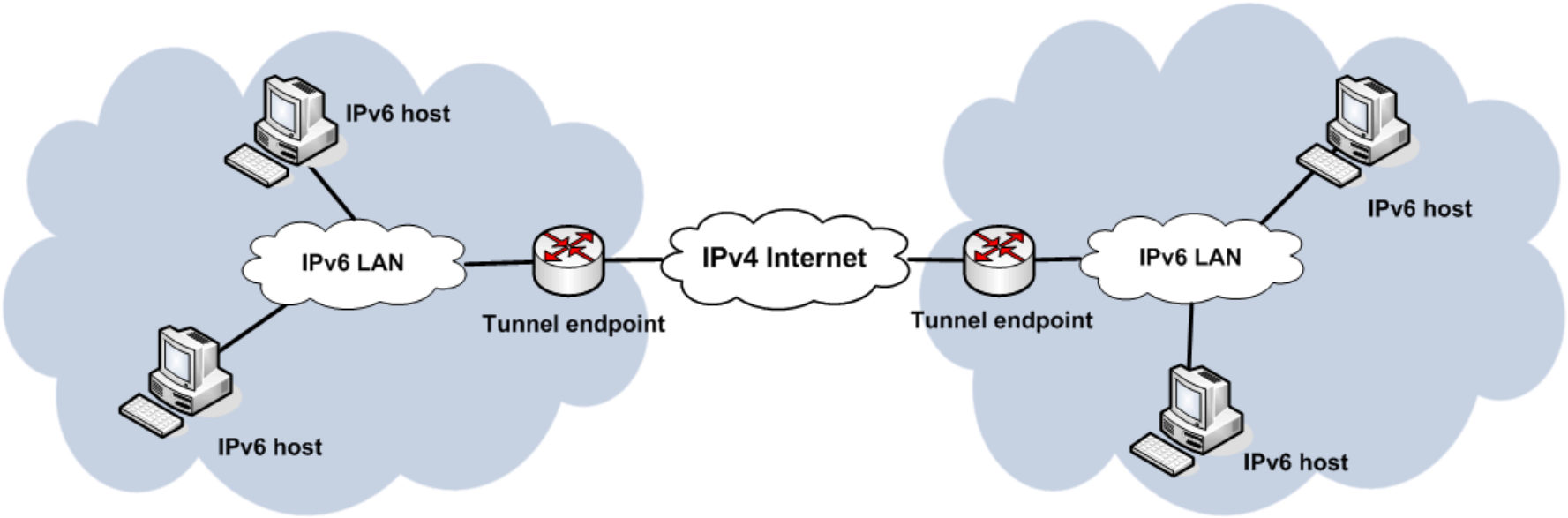
So... what is this “IPv6” thing about?

- It addresses the problem of IPv4 address exhaustion
- Employs 128-bit addresses (vs. IPv4's 32-bit addresses)
- Provides the same **service** as IPv4
- It is **not backwards-compatible with IPv4**



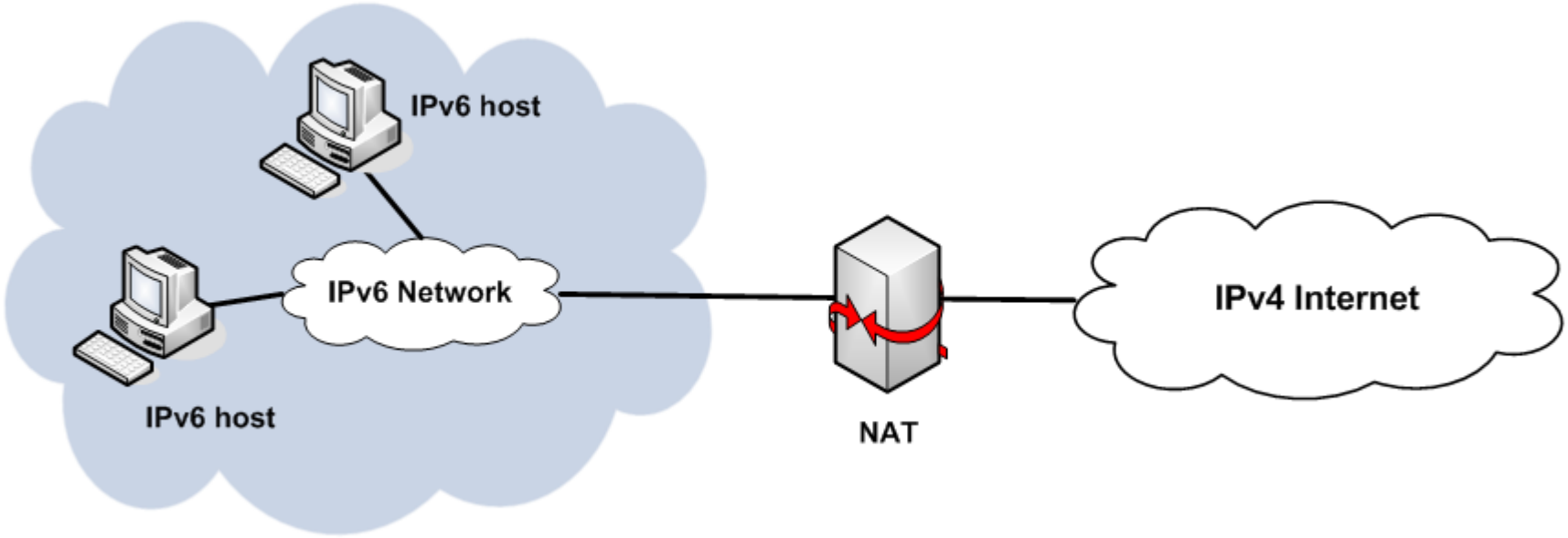
So... what is this “IPv6” thing about? (II)

- We can connect IPv6 islands across the IPv4 Internet with tunnels



So... what is this “IPv6” thing about? (III)

- We can interconnect IPv6-only hosts with IPv4-only hosts with “translators”



So... what is this “IPv6” thing about? (IV)

- For every domain name, the DNS can contain
  - A resource records (IPv4 addresses)
  - AAAA (Quad-A) resource records (IPv6 addresses)
- Host may query for A and/or AAAA resource records according different criteria
- Based on the available resource records, supported protocols, and local policy, IPv6 and/or IPv4 could be employed

Current state of affairs: Implementation

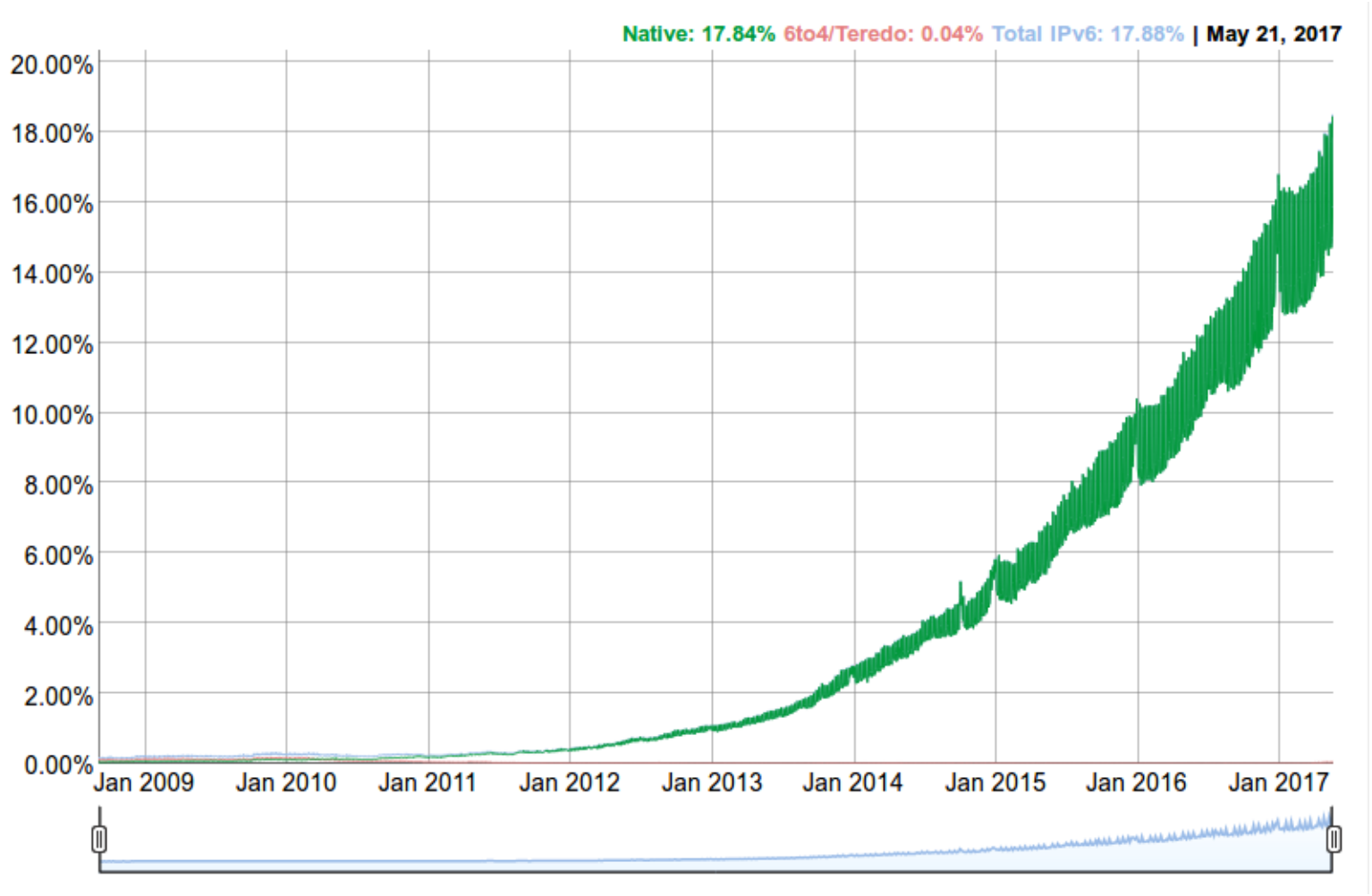
- General-purpose OSes have shipped with IPv6 support for a long time
  - part of your network is already running IPv6!
- Other devices may require updates or replacement:
  - CPE's
  - Firewalls
  - Routers
  - NIDSs
  - etc.

Current state of affairs: Deployment

- IPv6 was essentially **ignored for years**
- Many organizations have now started to take IPv6 more seriously, partly as a result of:
  - Exhaustion of the IANA IPv4 free pool
  - Imminent exhaustion of the address pool at the different RIRs
  - Awareness activities (“World IPv6 Day” & “World IPv6 Launch Day”)
  - Main content providers (Google, Facebook, Yahoo, etc.) have deployed IPv6 on their public-facing servers

Current state of affairs: Deployment (II)

- IPv6 usage as measured by Google:



SI6 NETWORKS

Current state of affairs: Deployment (III)

- IPv6 deployment per country
  - Visit: https://www.google.com/intl/en/ipv6/statistics.html

IPv6 tools

THC-IPv6 Toolkit: Introduction

- First and only IPv6 attack toolkit for many years
- Easy to use
  - Only minimal IPv6 knowledge required
- Features:
  - Only runs on Linux with Ethernet
  - Free software
  - Lacks of comprehensive documentation
- Available at: http://www.thc.org/thc-ipv6

SI6 Networks' IPv6 Toolkit

- Brief history:
  - Originally produced as part of a governmental project on IPv6 security
  - Maintenance and extension taken over by SI6 Networks
- Goals:
  - Security assessment and trouble-shooting of IPv6 networks and implementations
  - Clean, portable, and secure code
  - Good documentation

SI6 Networks' IPv6 Toolkit (II)

- Supported OSes:
  - Linux, FreeBSD, NetBSD, OpenBSD, OpenSolaris, and Mac OS
- License:
  - GPL (free software)
- Home:
  - http://www.si6networks.com/tools/ipv6toolkit
- Collaborative development:
  - https://www.github.com/fgont/ipv6toolkit.git

SI6 Networks' IPv6 Toolkit: Philosophy

IDEAS TOOLS IPV6 NETWORK

"*an interface between your ideas and an IPv6 network*"

SI6 Networks' IPv6 Toolkit: Tools

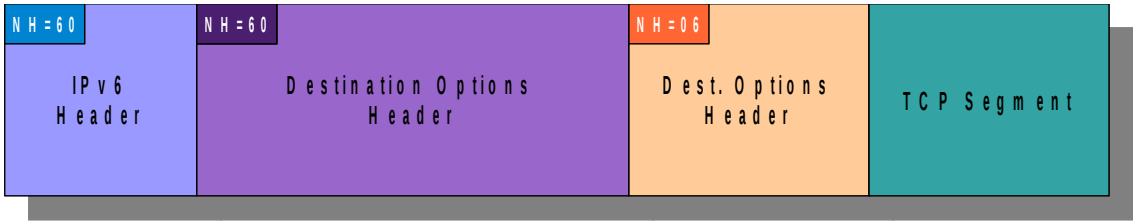
- ns6
- na6
- rs6
- ra6
- addr6
- path6
- rd6
- scan6
- frag6
- tcp6
- script6
- blackhole6
- icmp6
- ni6
- flow6
- jumbo6
- udp6

IPv6 Extension Headers

IPv6 Extension Headers
Overview

IPv6's Next Header field

- Identifies the header/protocol type following this header.
- IPv6 options are included in “extension headers”
  - These headers sit between the IPv6 header and the upper-layer protocol
  - There may be multiple instances, of multiple extension headers, each with multiple options
- Hence, IPv6 follow a “header chain” type structure. e.g.,



IPv6 Extension Headers

General implications of Extension Headers

Processing the IPv6 header chain

- Large number of headers/options may have a negative impact on performance
- Many routers can only look into a few dozen bytes into the packet
- It is harder to spot e.g. layer-4 information (if at all possible)

Fragmentation deemed as 'insecure'

- DoS vector:
  - Some are afraid about stateful-ness of IPv6 fragments
- Evasion:
  - It becomes harder (if at all possible) to implement ACLs
- Buggy implementations:
  - e.g. some boxes crash when a malformed fragment traverses it

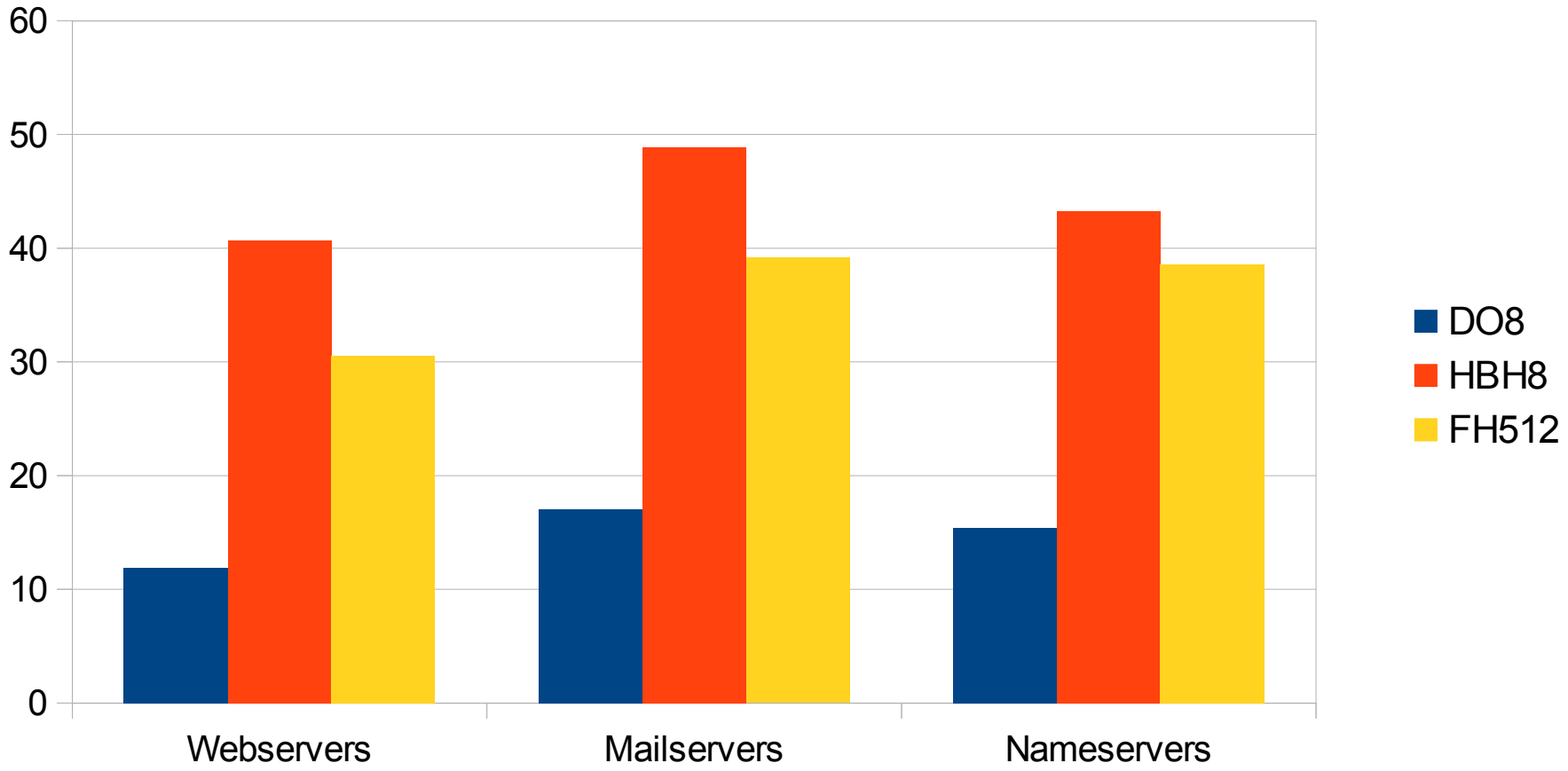
IPv6 Extension Headers
In The Real World

SI6 NETWORKS

IPv6 Fragmentation and EH reliability

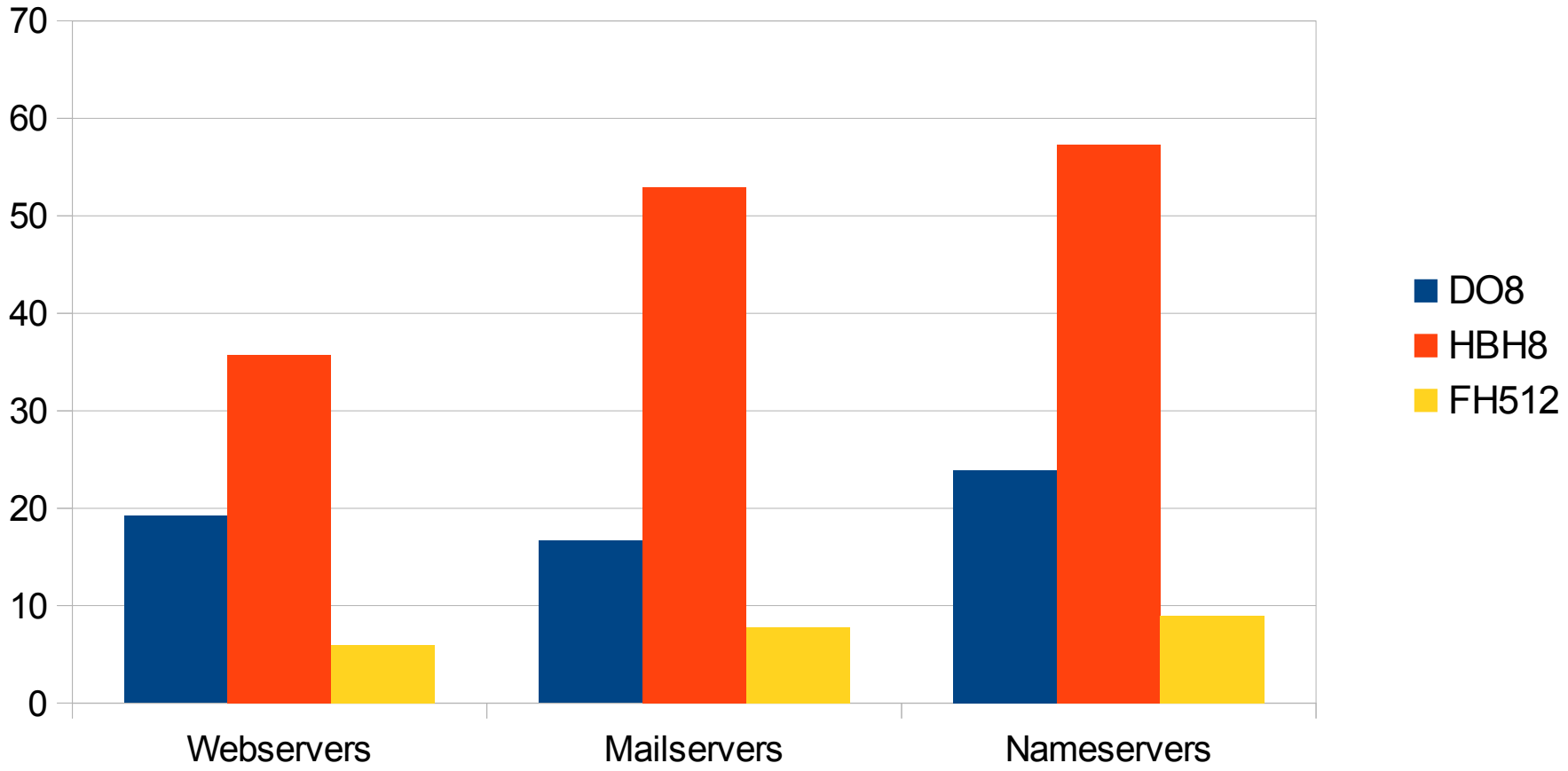
- Operators filter them, as a result of:
  - Perceived issues with IPv6 Fragmentation and EH
  - Almost no current dependence on them
- IPv6 Extension Headers result in unreliability

WIPv6LD dataset: Packet Drop rate

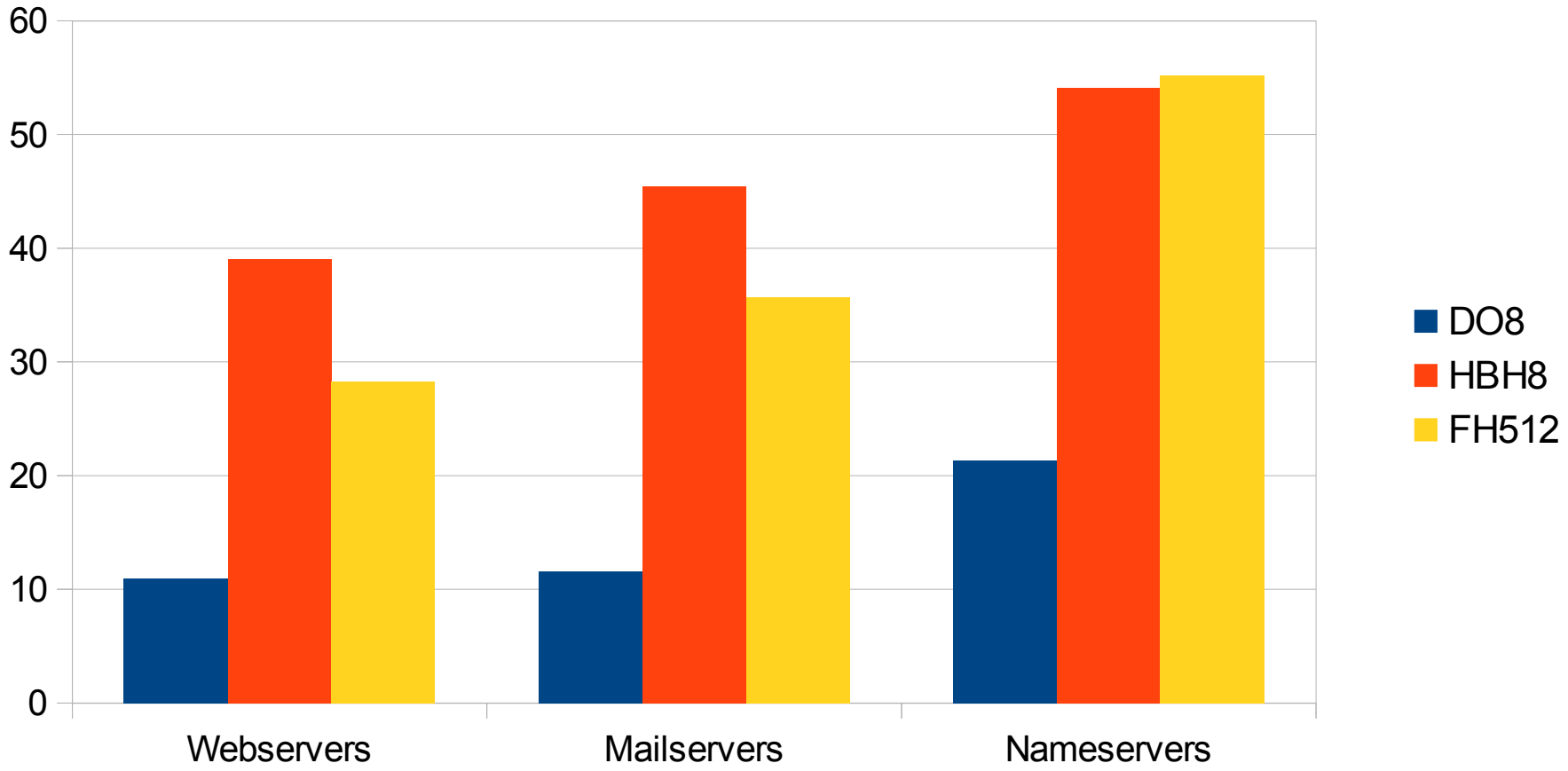


SI6 NETWORKS

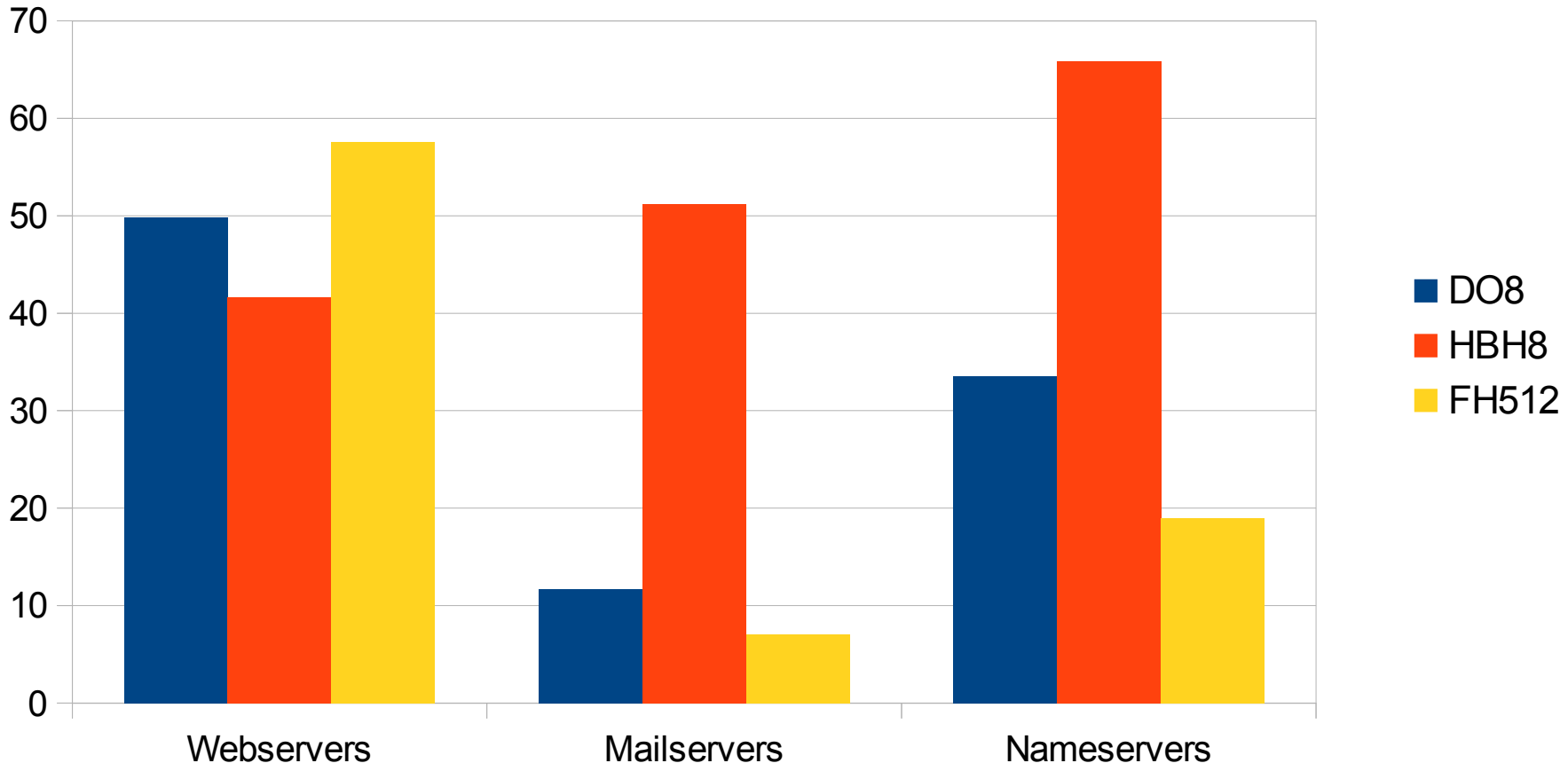
WIPv6LD dataset: Drops by diff. AS



Alexa dataset: Packet Drop rate



Alexa dataset: Drops by diff. AS



So... what does this all mean?

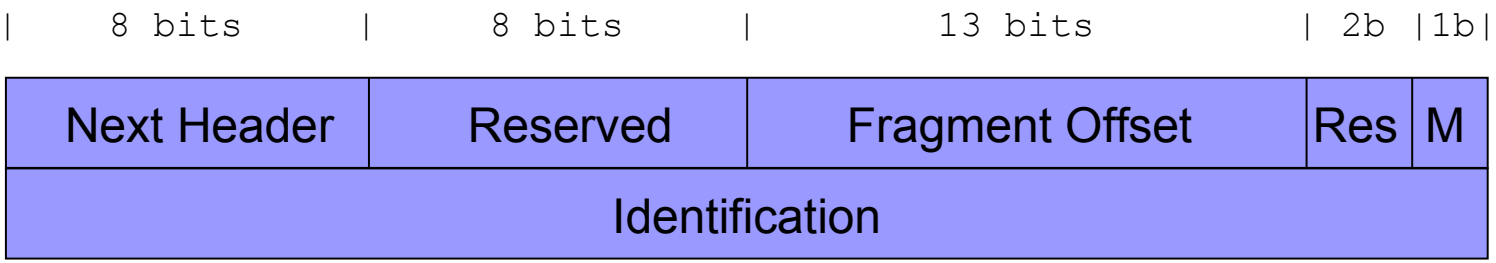
- Good luck with getting IPv6 EHs working in the Internet!
  - They are widely dropped
- IPv6 EHs “not that cool” for evasion, either
  - Chances are that you will not even hit your target

IPv6 Extension Headers

Fragment Header

IPv6 Fragmentation Overview

- IPv6 fragmentation performed only by hosts (never by routers)
- Fragmentation support implemented in “Fragmentation Header”



- Where:
  - Fragment Offset: Position of this fragment with respect to the start of the fragmentable part
  - M: “More Fragments”, as in IPv4
  - “Identification”: Identifies the packet (with Src IP and Dst IP)

Fragmentation: Example

- **`ping6 -s 1800 2004::1`**

```
PING 2004::1(2004::1) 1800 data bytes

1808 bytes from 2004::1: icmp_seq=1 ttl=64 time=0.973 ms

--- 2004::1 ping statistics ---

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 0.973/0.973/0.973/0.000 ms
```

- tcpdump output:

20:35:27.232273 IP6 2004::5e26:aff:fe33:7063 > 2004::1: **frag (0|1448)** ICMP6, echo request, seq 1, length 1448

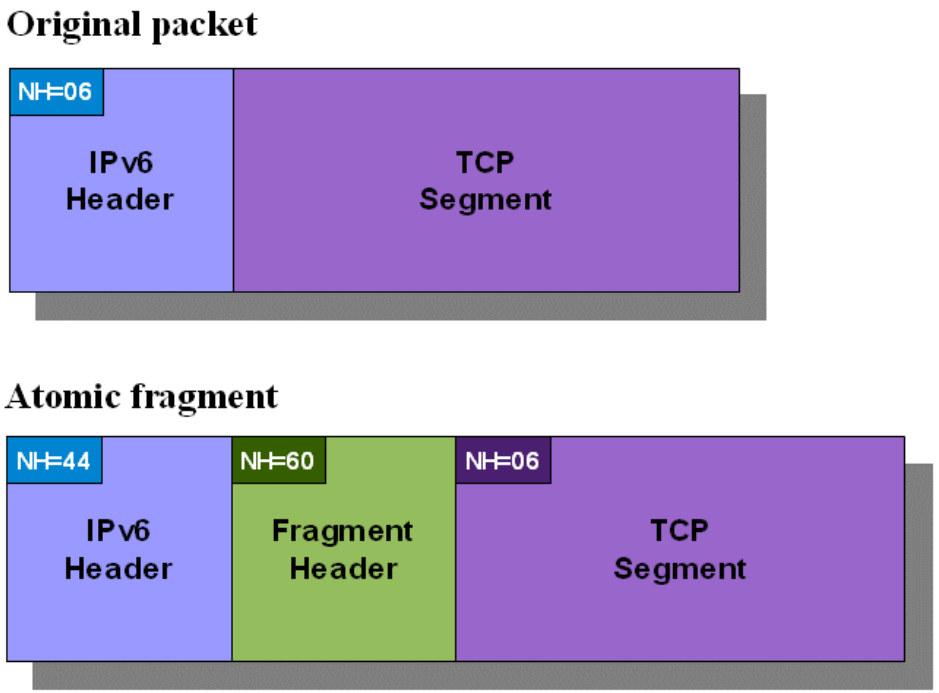
20:35:27.232314 IP6 2004::5e26:aff:fe33:7063 > 2004::1: **frag (1448|360)**

20:35:27.233133 IP6 2004::1 > 2004::5e26:aff:fe33:7063: **frag (0|1232)** ICMP6, echo reply, seq 1, length 1232

20:35:27.233187 IP6 2004::1 > 2004::5e26:aff:fe33:7063: **frag (1232|576)**

IPv6 “atomic” fragments

- ICMPv6 PTB < 1280 triggers inclusion of a FH in all packets to that destination (not actual fragmentation)
- Result: IPv6 atomic fragments (Frag. Offset=0, More Frag.=0)



Issues with IPv6 atomic fragments

- Some implementations mix “atomic fragments” with queued fragments
- Atomic fragments thus become subject of IPv6 fragmentation attacks
- How to leverage this issue:
  - Trigger atomic fragments with ICMPv6 PTB messages
  - Now perform IPv6 fragmentation-based attacks

Processing of IPv6 atomic fragments

- Atomic fragment do not need to be mixed with other fragments – they are **atomic**!
- Skipping the normal reassembly procedure eliminates fragmentation-based attacks for such traffic
- RFC 6946 improves the handling of IPv6 atomic fragments:
  - They are required to be processed as non-fragmented traffic

Assessing support for atomic fragments

- Check response to atomic fragments

```
# frag6 --frag-type atomic --frag-id 100 -d
fc00:1::1
```

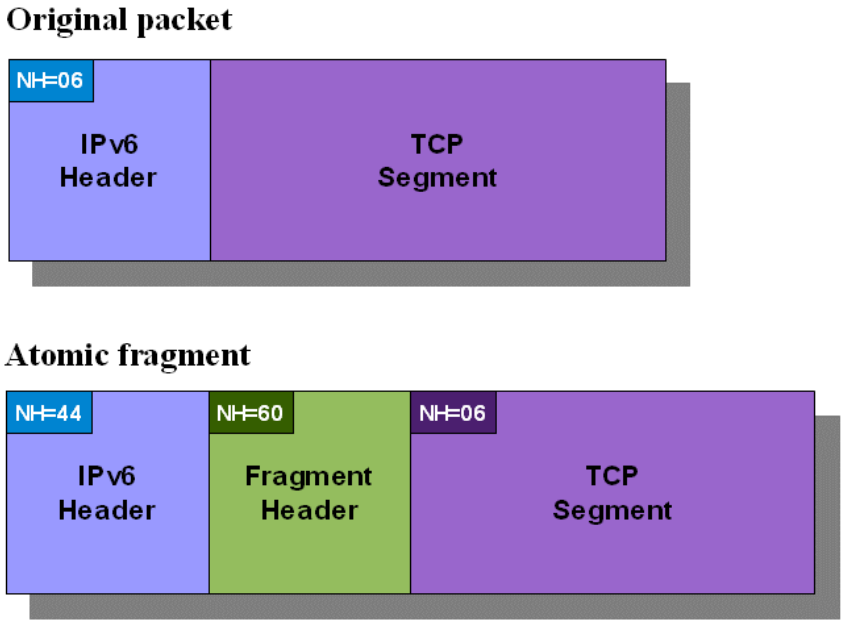
- Assess support for atomic fragments:

```
# frag6 --frag-type first --frag-id 100 -d
fc00:1::1

# frag6 --frag-type atomic --frag-id 100 -d
fc00:1::1
```

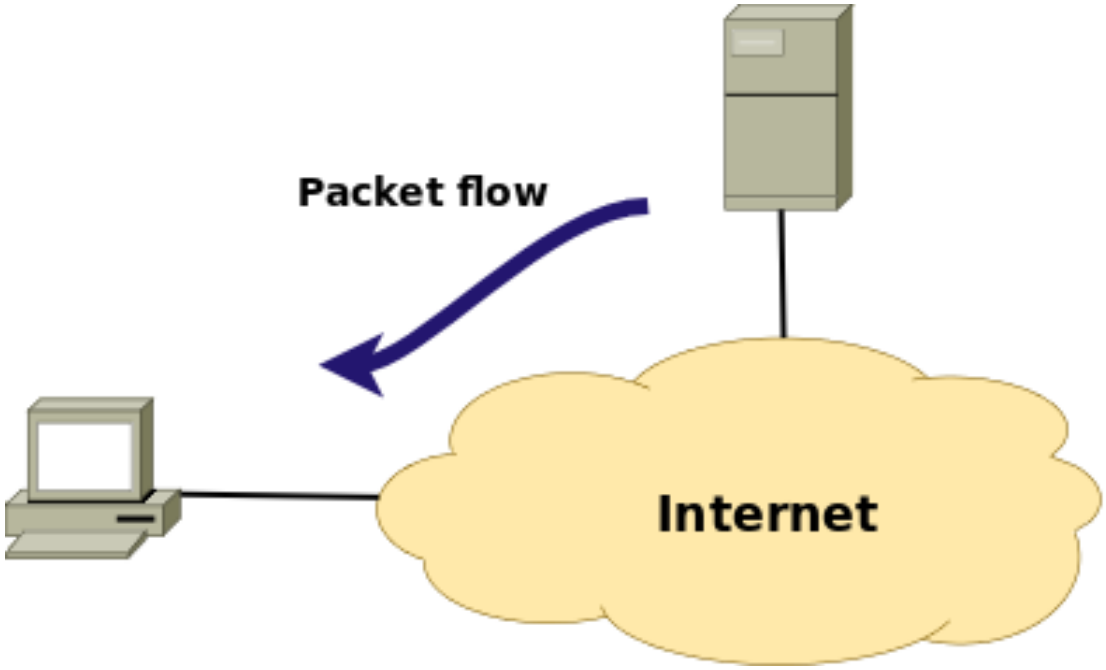
Generation of IPv6 atomic fragments

- If IPv6 frags are widely dropped...What if we triggered their generation?
  - Send an ICMPv6 PTB with an MTU<1280
  - The node will then generate IPv6 atomic fragments
  - Packets will get dropped



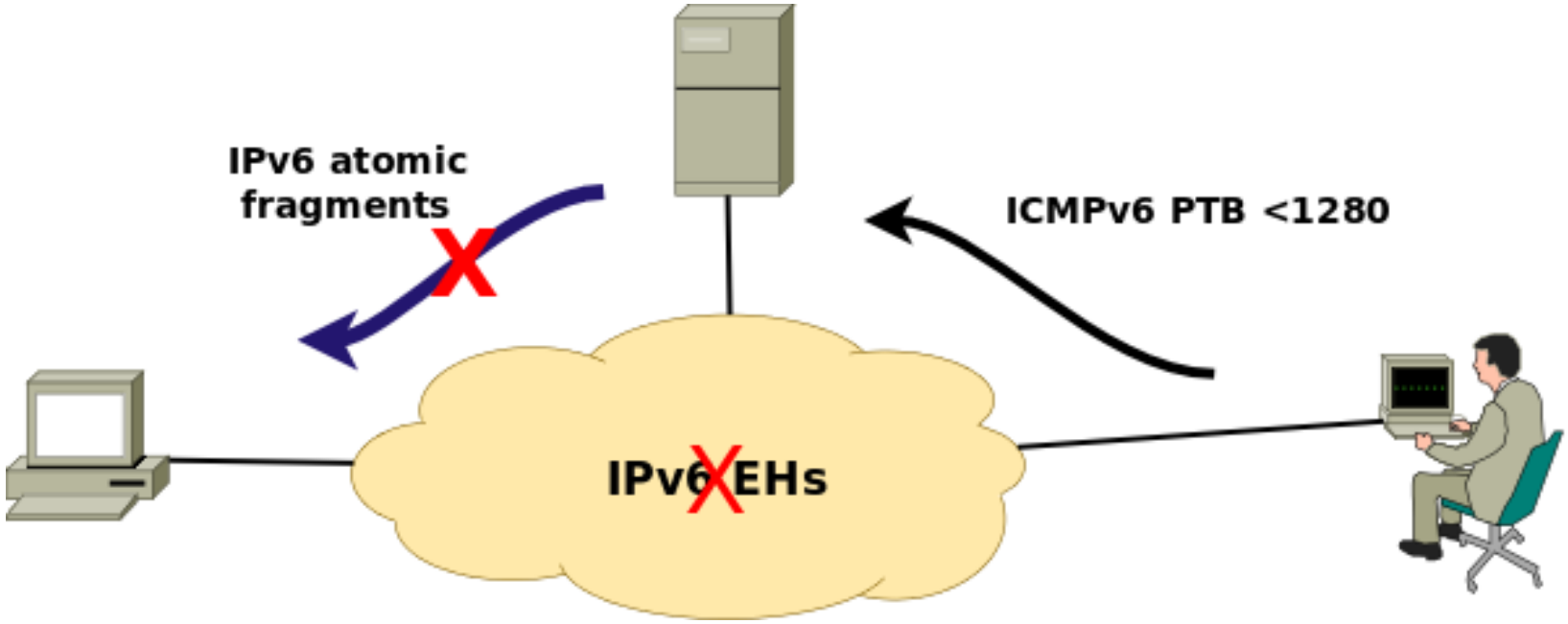
Attack Scenario #1

- Client communicates with a server



Attack Scenario #1 (II)

- Attacking client-server communications



Attack scenario #1 (II)

- Simple way to reproduce it:
  - Attack and client machine is the same one
  - So we attack our own “connections”
- Attack:
  - Test IPv6 connetivity:

```
telnet 2001:4f8:1:10:0:1991:8:25 80
```

  - Send an ICMPv6 PTB < 1280 to trigger atomic fragments

```
sudo icmp6  --icmp6-packet-too-big -d
2001:4f8:1:10:0:1991:8:25 --peer-addr
2001:5c0:1000:a::a37 --mtu 1000 -o 80 -v
```

  - Test IPv6 connectivity again:

```
telnet 2001:4f8:1:10:0:1991:8:25 80
```

Generation of IPv6 atomic fragments

- RFC8021
  - Discusses the rationale for deprecating the generation of IPv6 atomic fragments in the upcoming revision of RFC2460
  - i.e. Hosts are not required to generated them in response to ICMPv6 PTB<1280

IPv6 Challenge

SI6 NETWORKS

Testing

- Installl the SI6 Networks IPv6 toolkit
- Test:
  - Processing of IPv6 atomic fragments (RFC6946)
  - Generation of IPv6 atomic fragments (RFC8021)
- Then:
  - Group #1: Document the testing process in an IETF Internet-Draft
  - Group #2: Implement support of such RFCs in open source OSes

Thanks!

Fernando Gont

fgont@si6networks.com

IPv6 Hackers mailing-list

http://www.si6networks.com/community/

www.si6networks.com