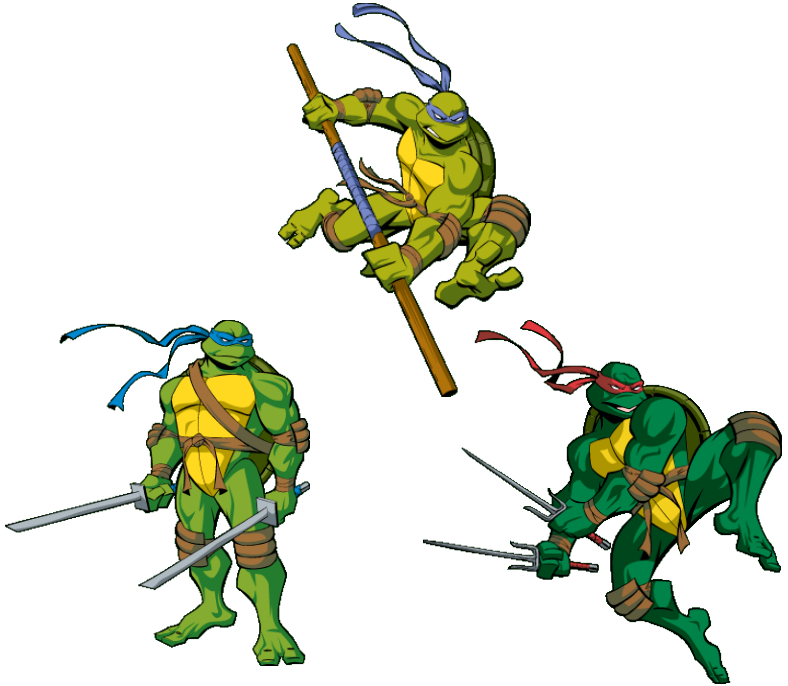
Evasion of High-End IDPS Devices at the IPv6 Era

Antonios Atlasis
secfu.net
aatlasis@secfu.net

Enno Rey
ERNW GmbH
erey@ernw.de

Rafael Schaefer
ERNW GmbH
rschaefer@ernw.de

Who We Are



- Enno Rey
  - Old school network security guy. Back in 2001 founder of ERNW & still proudly running the team.
- Antonios Atlasis
  - IT Security enthusiast.

.

- Rafael Schaefer
  - Young researcher @ ERNW

Outline of the Presentation

- Introduction
  - IPv6 is here
  - What IPv6 brings with it:
    The Extension Headers
- Problem Statement. Describe the Mess
- Tested IDPS devices:
  - Suricata
  - Tipping Point
  - Sourcefire
  - Snort
- Mitigation & Conclusions

ERNW
providing security.

Disclaimer

- This is the talk/version we gave at Black Hat Europe in October 2014.
- In the interim newer releases of the stuff we tested are available, so some of the techniques might no longer work.
  - See, for example: http://www.insinuator.net/2015/01/how-to-configure-snort-to-stop-ipv6-evasion-attacks/

What an IPv6 Datagrams Looks Like...

The IPv6 Extension Headers

- Currently defined:
  - Hop-by-Hop Options [RFC2460]
  - Routing [RFC2460]
  - Fragment [RFC2460]
  - Destination Options [RFC2460]
  - Authentication [RFC4302]
  - Encapsulating Security Payload [RFC4303]
  - MIPv6, [RFC6275] (Mobility Support in IPv6)
  - HIP, [RFC5201] (Host Identity Protocol)
  - shim6, [RFC5533] (Level 3 Multihoming Shim Protocol for IPv6)
- There is a RECOMMENDED order.
- All (but the Destination Options header) SHOULD occur at most once.
- How a device should react **if NOT** ?

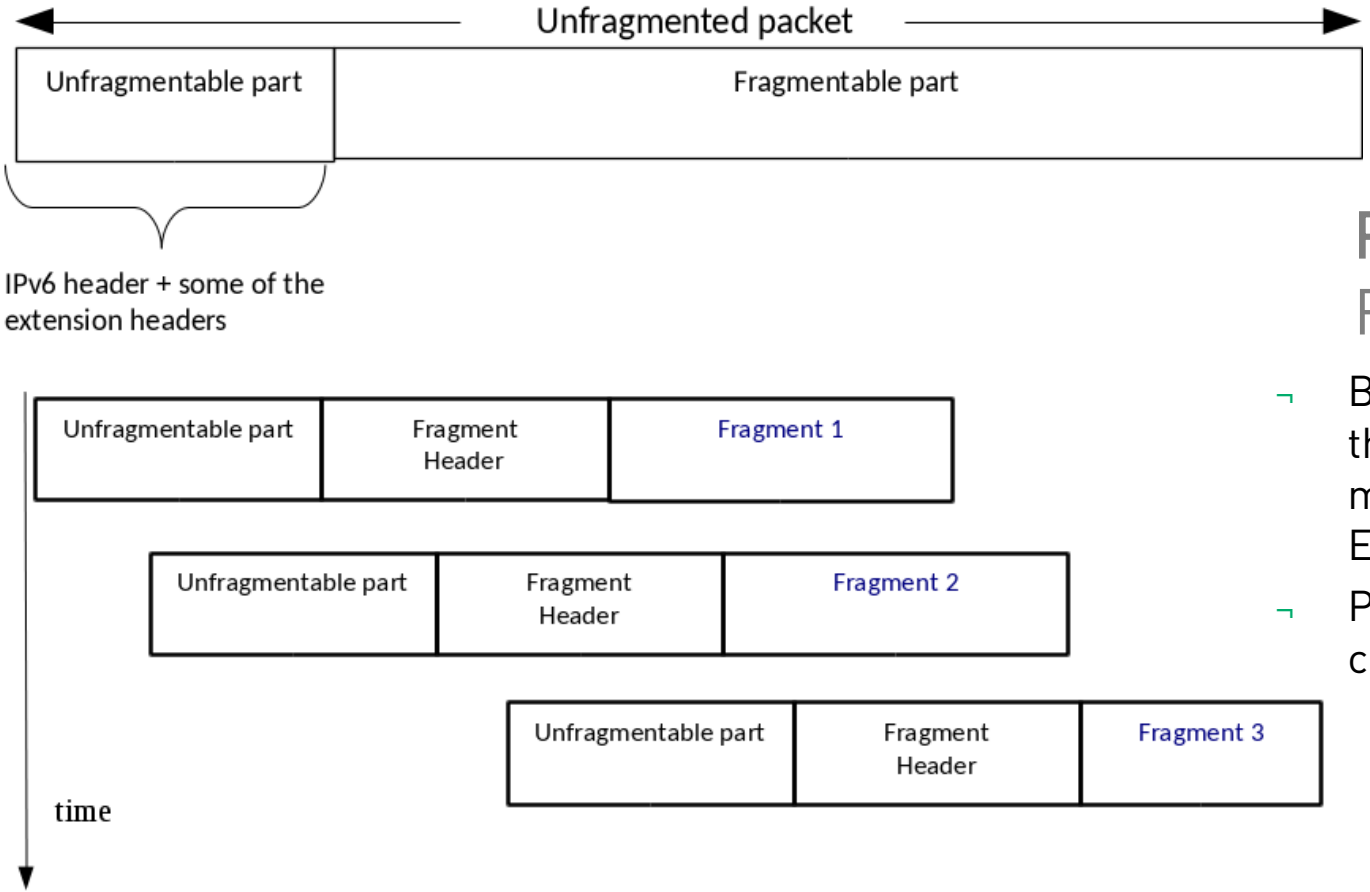
Transmission & Processing of IPv6 Ext. Hdrs

- RFC 7045. Any forwarding node along an IPv6 packet's path:
  - should forward the packet regardless of any extension headers that are present.
  - MUST recognize and deal appropriately with all standard IPv6 extension header types.
  - SHOULD NOT discard packets containing unrecognised extension headers.

Problem 1: Too Many Things to Vary

- Variable types
- Variable sizes
- Variable order
- Variable number of occurrences of each one.
- Variable fields

IPv6 = f(v,w,x,y,z,)



Problem 2: Fragmentation

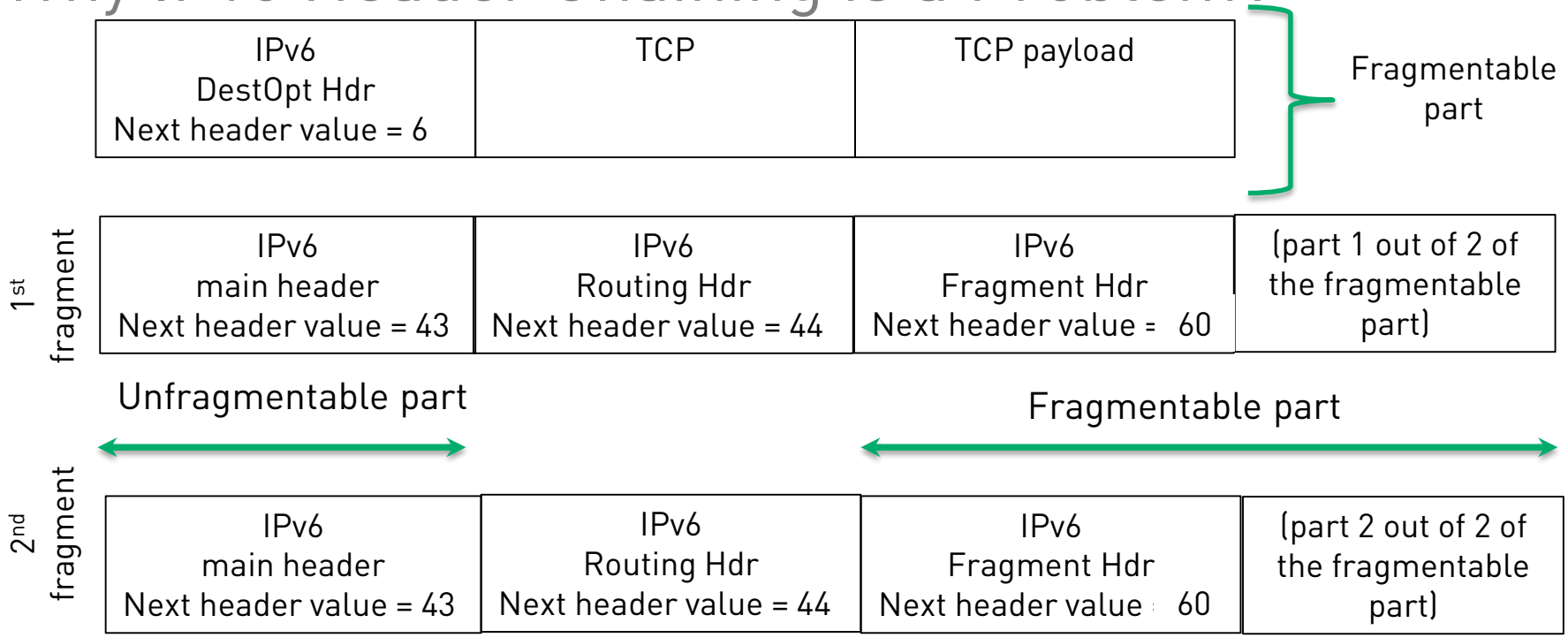
- Both the *Fragmentable* and the *Unfragmentable* parts may contain any IPv6 Extension headers.
- Problem 1 becomes more complicated.

Problem 3: How IPv6 Extension Headers are Chained?

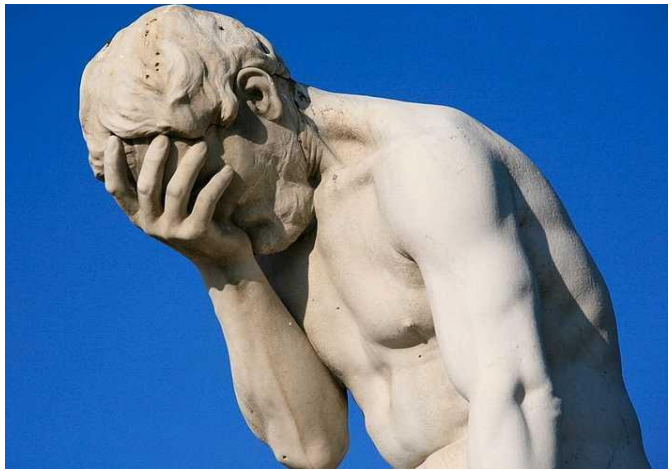
IPv6 header	IPv6 Routing Extension header	IPv6 Destination Options header	TCP header + payload ...
Next Header Value = 43	Next Header Value = 60	Next Header Value = 6	

- Next header fields:
  - Contained in IPv6 headers, identify the type of header immediately following the current one.
  - Some use the same values as the IPv4 Protocol field.

Why IPv6 Header Chaining is a Problem?



To sum up the Mess in IPv6



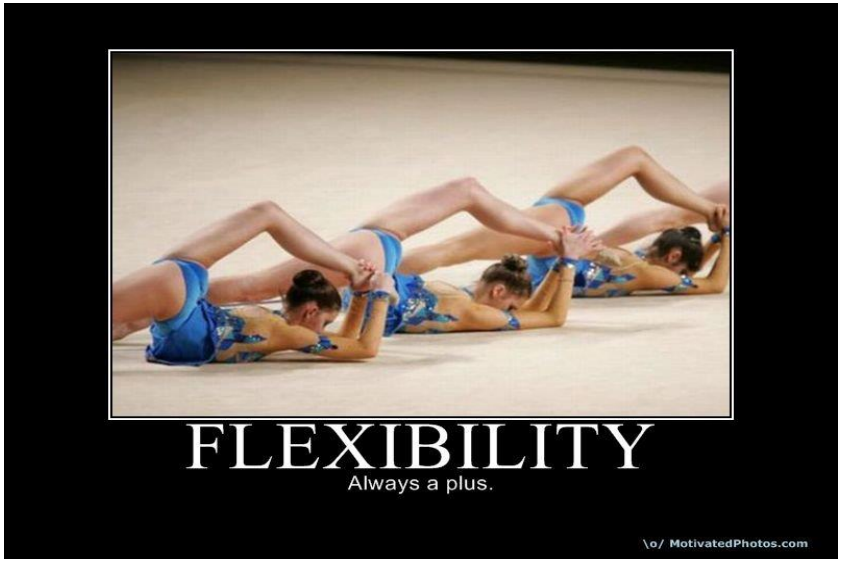
- Vary:
  - The types of the IPv6 Extension headers
  - The order of the IPv6 Extension headers
  - The number of their occurrences.
  - Their size.
  - Their fields.
  - The Next Header values of the IPv6 Fragment Extension headers in each fragment.
  - Fragmentation (where to split the datagram)
- And combine them.

Did You Notice?

- When designing/writing IPv6 protocols & parsers they didn‘t pay too much attention to #LANGSEC.
- Please visit www.langsec.org.

We May Have a Fundamental Problem Here…

- There is too much flexibility and freedom…
- Which is usually inverse proportional to security :-)
- And it can potentially lead to a complete *cha0s*…

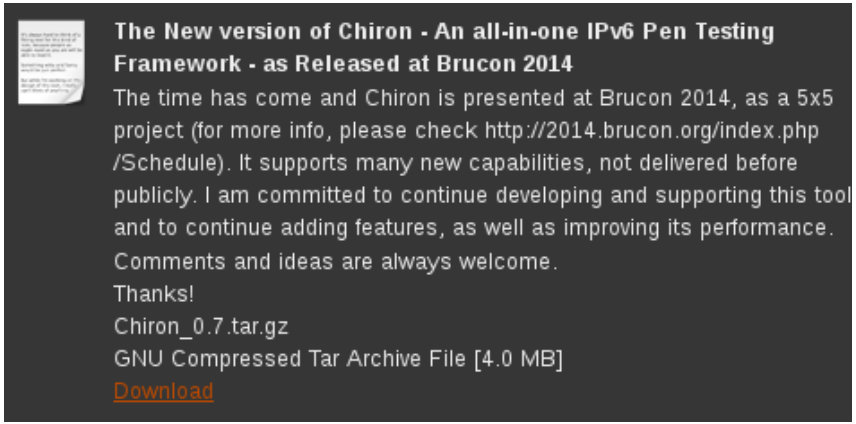


So, What Can Possibly Go Wrong?

- Detection Signatures, e.g. used by IDPS rules, etc. are based on blacklisting traffic.

- What if we confuse their parsers by abusing IPv6 Extension headers in an unusual / unexpected way?

All this is not just a theory



The New version of Chiron - An all-in-one IPv6 Pen Testing Framework - as Released at Brucon 2014

The time has come and Chiron is presented at Brucon 2014, as a 5x5 project (for more info, please check http://2014.brucon.org/index.php /Schedule). It supports many new capabilities, not delivered before publicly. I am committed to continue developing and supporting this tool and to continue adding features, as well as improving its performance. Comments and ideas are always welcome.

Thanks!

Chiron_0.7.tar.gz

GNU Compressed Tar Archive File [4.0 MB]

Download

- You can reproduce all the results that we shall demonstrate using *Chiron*
- It can be downloaded from: http://www.secfu.net/tools-scripts/

Chiron – Sample

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
"AAAAAAAABBBBBBBBCCCCCCCCDDDDDDDDEEEE" -ll 7,1 -lm 1,0 -lo 0,7 -lnh 60,6
```

2. Second case: Evading TOS version 3.6.2.4109 and TOS version 3.7.0.4228 and Digital Vaccine 3.2.0.8560 by repeating the second fragment

```
./chiron_scanner.py eth0 -d 2001:db8:1:1::1 -sS -p 80 -lfE 60 -nf 3 -l4_data
"AAAAAAAABBBBBBBBCCCCCCCCDDDDDDDDEEEE" -ll 7,1,1 -lm 1,0,0 -lo 0,7,7 -lnh
60,60,6
```

or,

```
./chiron_scanner.py eth0 -d 2001:db8:1:1::1 -sS -p 80 -lfE 60 -nf 3 -l4_data
"AAAAAAAABBBBBBBBCCCCCCCCDDDDDDDDEEEE" -ll 7,1,1 -lm 1,0,0 -lo 0,7,7 -lnh
60,6,6
```

Snort

From: https://www.ernw.de/download/eu-14-Atlasis-Rey-Schaefer-briefings-Evasion-of-HighEnd-IPS-Devices-wp.pdf

ERNW
providing security.

Our Tests at a Glance

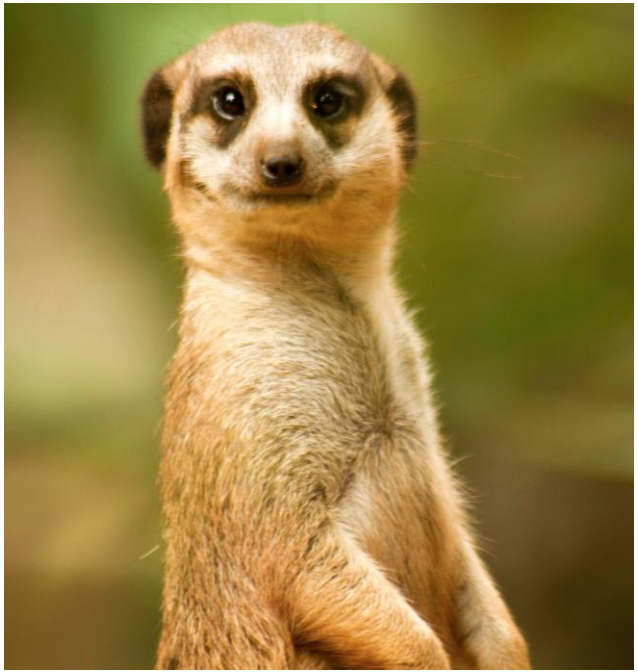
- Four (4) IDPS (two open-source, two high-end commercial ones).
- At least twelve (12) different evasion techniques, in total.
- All of them 0-days at the time of the finding.
- All of them were reported (disclosed responsibly).
- Most of them were patched, either promptly or not that promptly ☺.
- Some guys were too busy though, so two of the products still suffer from 0-days IPv6 evasion techniques.

Demos

- I didn't bring devices (on short notice and, more importantly, I flew in... ever went through airport security with some IDPSs in your bag? ;-).
- Still, some demos here:

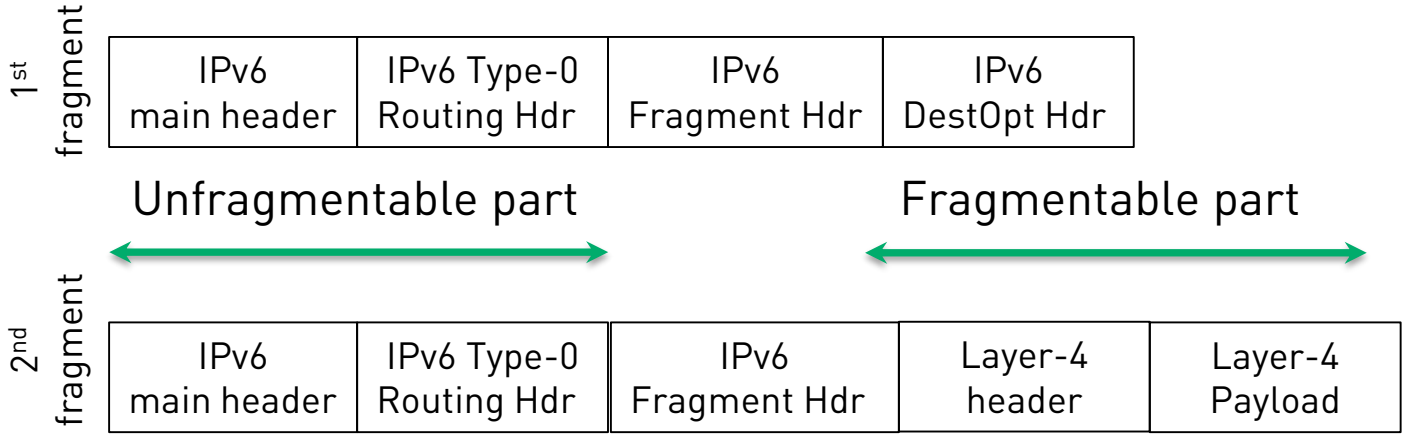
  https://www.youtube.com/watch?v=avMeYIaU8DA&feature=youtu.be&t=2640

Evading Suricata



- Versions 2.0.1, 2.0.2 and 2.0.3 were evaded one by one by using various techniques.
- All of them can be found in the white paper (see above) and can be reproduced by using *Chiron*.

Evading Suricata 2.0.3



Note: Other combinations of Extension Headers can also work (your ...homework)

Time for Action

- Demo against Suricata 2.0.3

Suricata Developers in Each Reported Case Reacted really Fast

Suricata 2.0.4 Available!

The OISF development team is pleased to announce Suricata 2.0.4. This release fixes a number of important issues in the 2.0 series.

This update fixes a bug in the SSH parser, where a malformed banner could lead to evasion of SSH rules and missing log entries. In some cases it may also lead to a crash. Bug discovered and reported by Steffen Bauch.

Additionally, this release also addresses a new IPv6 issue that can lead to evasion. Bug discovered by Rafael Schaefer working with ERNW GmbH.

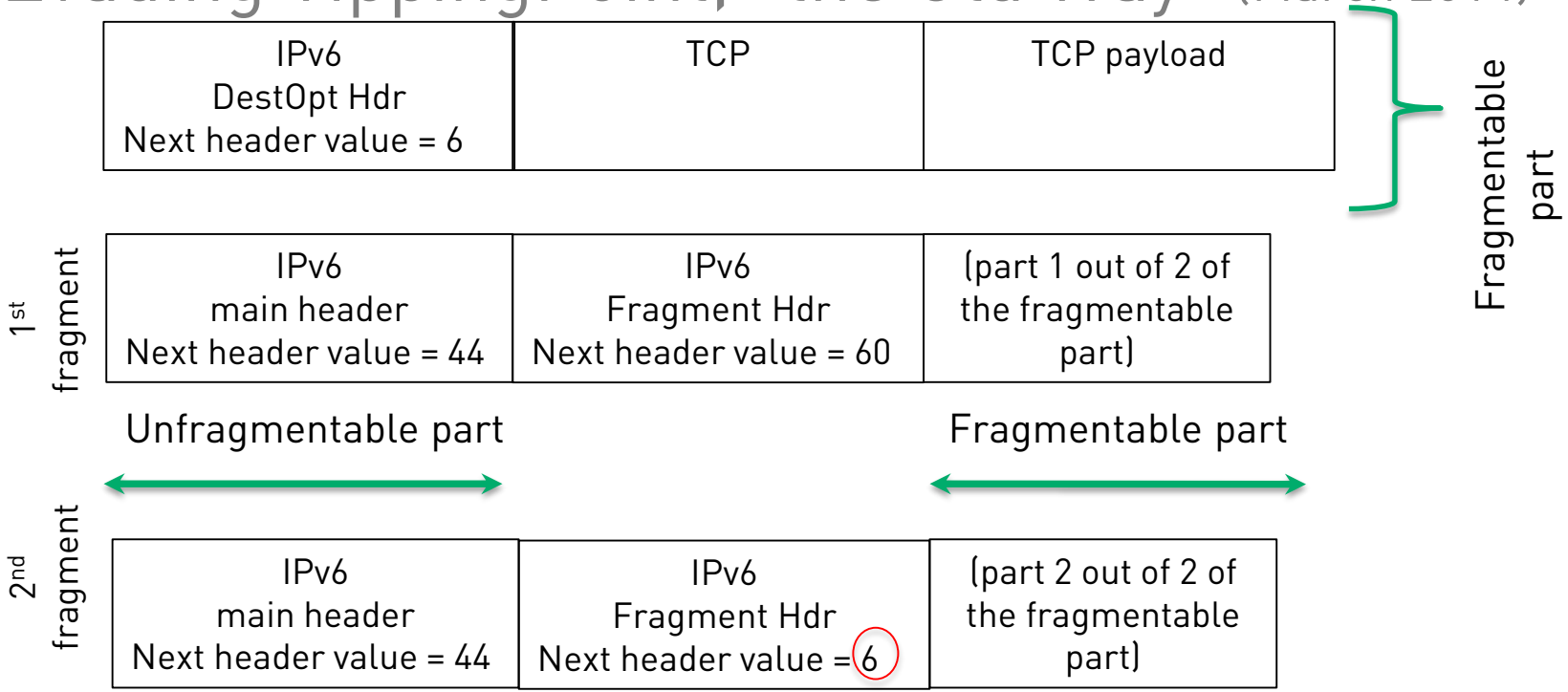
Download

Get the new release here: http://www.openinfosecfoundation.org/download/suricata-2.0.4.tar.gz

Changes

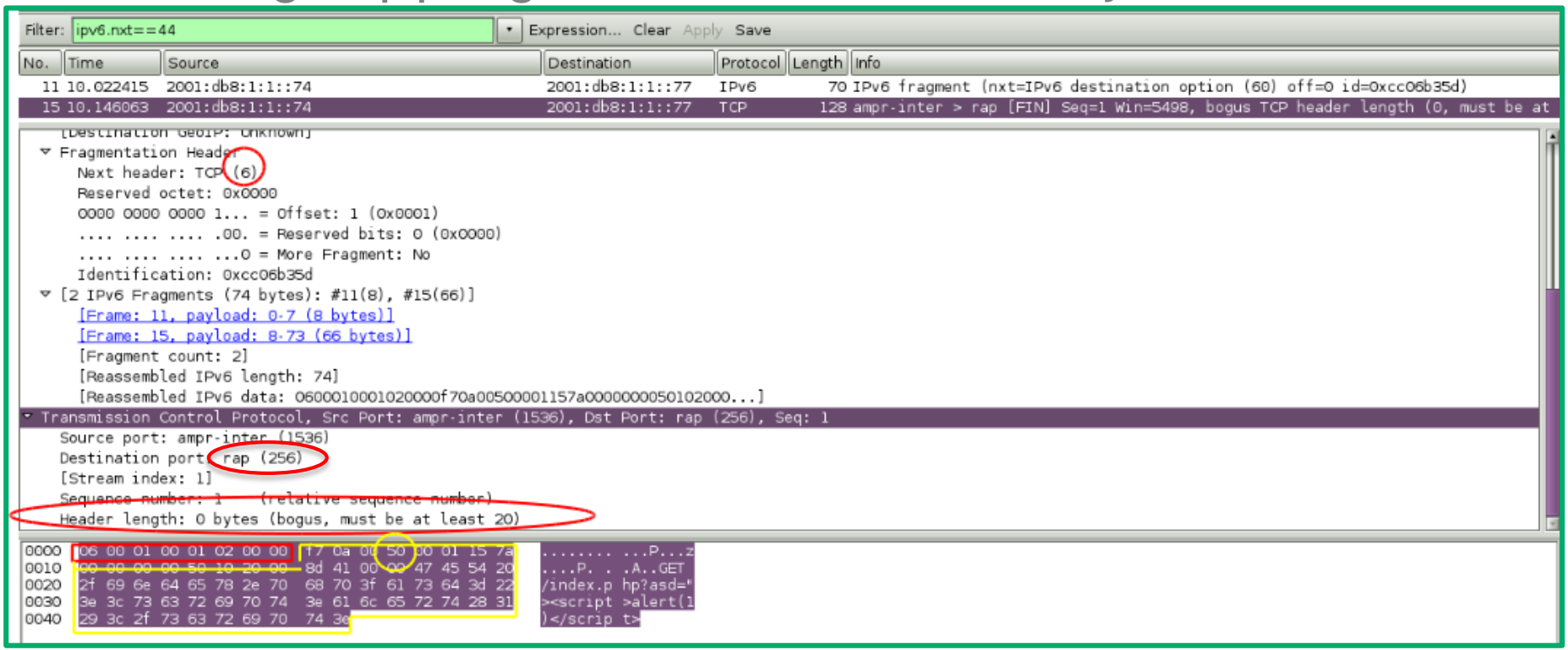
- Bug #1276: ipv6 defrag issue with routing headers
- Bug #1278: ssh banner parser issue
- Bug #1254: sig parsing crash on malformed rev keyword
- Bug #1267: issue with ipv6 logging
- Bug #1273: Lua – http.request_line not working
- Bug #1284: AF_PACKET IPS mode not logging drops and stream inline issue

Evading TippingPoint, "the Old Way" (March 2014)



Note: Layer-4 header can be in the 1st fragment and the attack still works

Evading TippingPoint, “The Old Way”



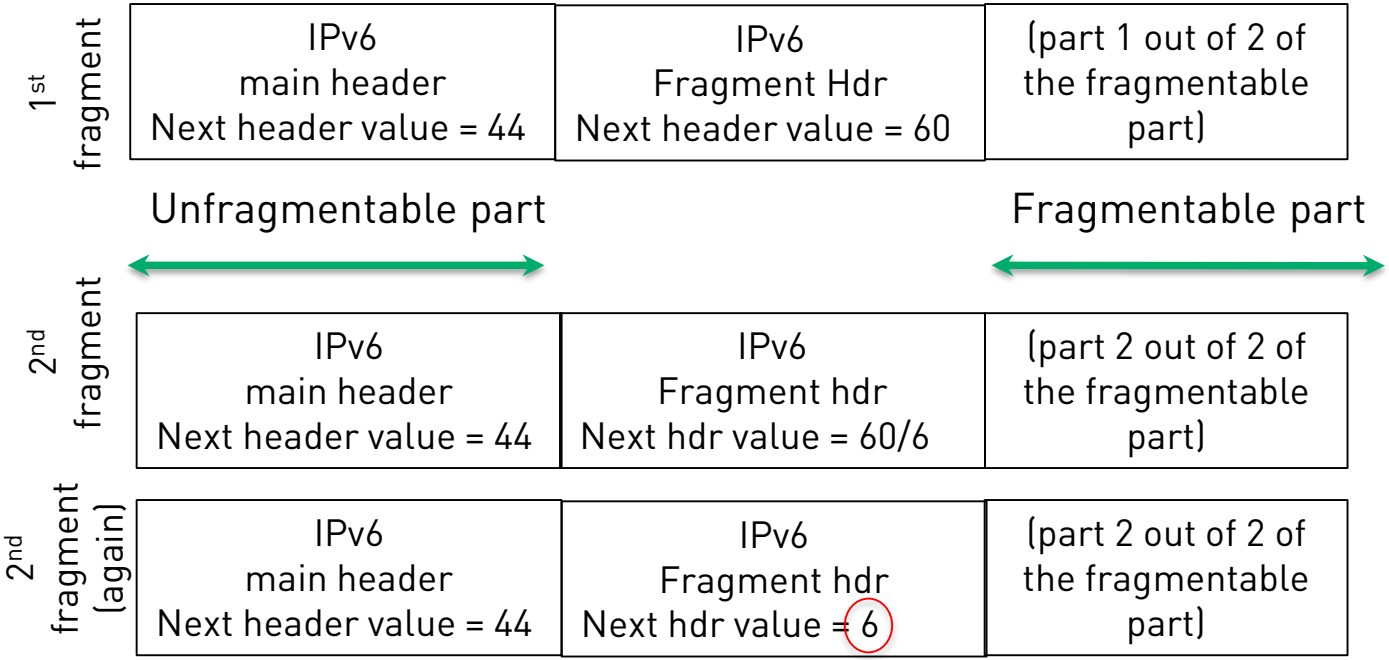
That First One Was Patched...

But Again We Had a New One ;-)

Model Number	110
Serial Number	U110C-50F
TOS Version	3.6.2.4109
Digital Vaccine	3.2.0.8565

- Configured to:
  - Operate inline at Layer 2.
  - Block any HTTP traffic.
  - Additional XSS rules (to test attacks at the payload too).

Evading TippingPoint, after First Patching



Note: Layer-4 header can be in the 1st fragment and the attack still works

Time for some more …Action

- Evading TippingPoint 3.6.2 demonstration

Snort / Sourcefire

- Quite similar situations, as expected.
- Still, the commercial device suffers from a 0-day evasion technique that the latest open-source version does not!

The Chronicle of the Communication

- We first contacted the Snort devs on17th of June.
  - "Please, send us the pcap files"
  - We did; no news since then…
- Reported a Sourcefire issue in Sep 14, and Sep 25, etc., including pcap files.
  - A kind of "don't waste my time" approach.
  - "Please, contact the customer support…"

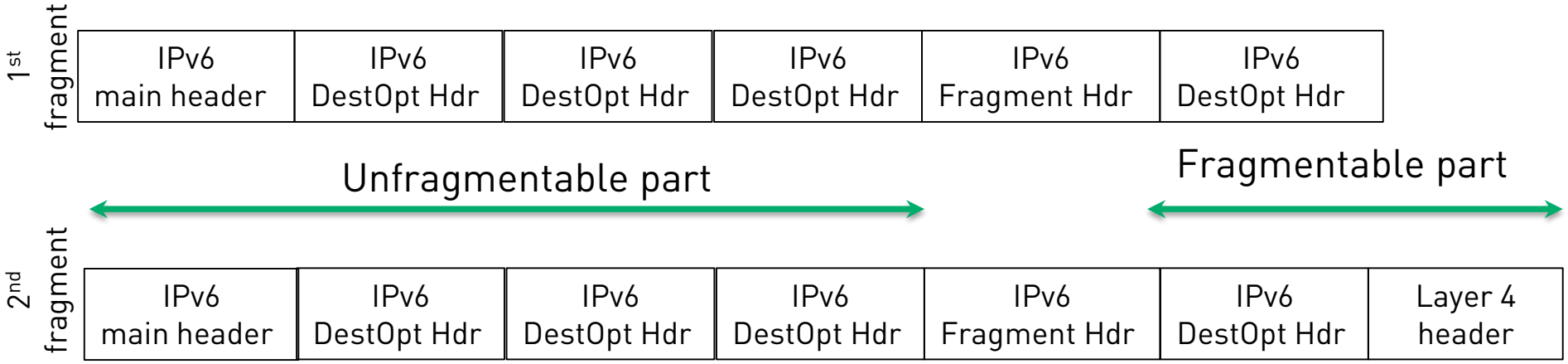
Fair enough!

- Time for a full disclosure!
- Live demos for both.

Evading Sourcefire

- Sourcefire, Model 3D7020 (81) Version 5.2.0.3 (Build 48).
- Preproc decoder rules were enabled:
  - GID 116 family and specifically, SID 458 (IPV6_BAD_FRAG_PKT), 272 and 273 are enabled.

Evading Sourcefire

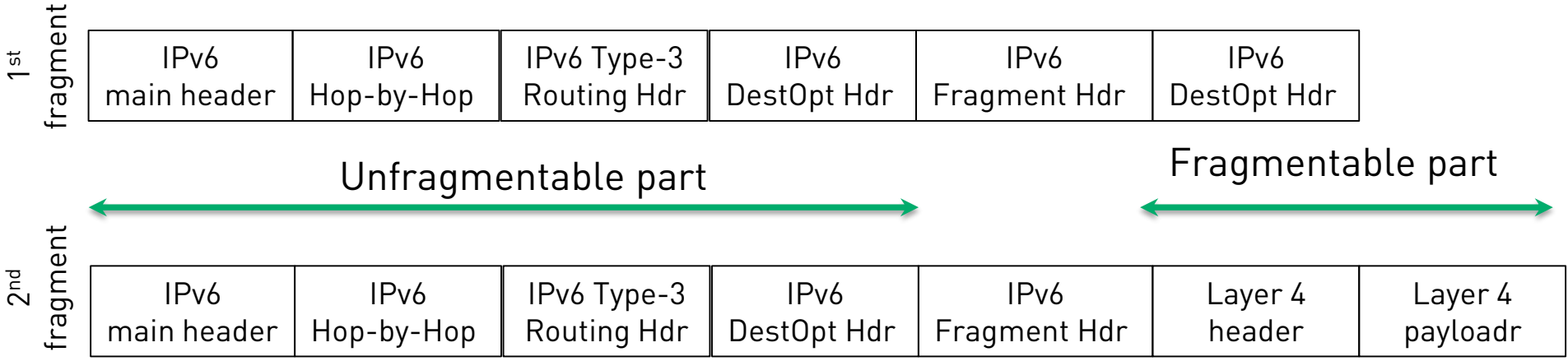


Note: Next header values for Fragment Extension headers: The correct ones (60)

Evading Snort

- Latest Snort version, 2.9.6.2
- Preproc decoder rules are enabled:
  - GID 116 family and specifically, SID 458 (IPV6_BAD_FRAG_PKT), 272 and 273 are enabled.

Evading Snort



Note: Next header values for Fragment Extension headers: the correct ones (60)

“Culture” Mitigations

- RFCs should strictly define the exact legitimate usage.
  - “Loose” specifications result in ambiguities and so they introduce potential attack vectors.
  - Functionality and flexibility are definitely good things, but security is non-negotiable.
- Make fully-compliant IPv6 products and test them thoroughly.

Technical Mitigations

- Implementation of RFC 7112.
  - An intermediate system (e.g., router or firewall) that receives an IPv6 First Fragment that does not include the entire IPv6 Header Chain MAY discard that packet.
  - Still, not a panacea…
- For the time being:
  - Configure your devices to drop IPv6 extension headers not used in your environment. OR
  - At least sanitize traffic before the IDPS.

This Is how a Certain Vendors Interprets This

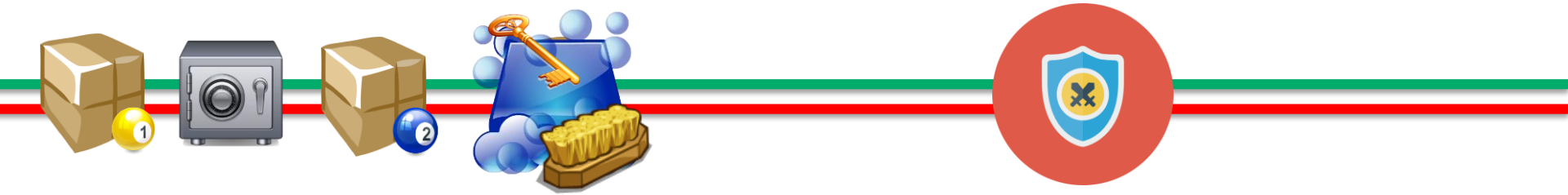
From sk39374

- How to handle IPv6 Extension Headers

  By default, Check Point Security Gateway drops all extension headers, except fragmentation. This can be adjusted by editing the `allowed_ipv6_extension_headers` section of `$FWDIR/lib/table.deffile` on the Security Management Server.

  Furthermore, as of R75.40 there is an option to block type zero even if Routing header is allowed. It is configurable via a kernel parameter `fw6_allow_rh_type_zero`. The default of 0 means it is always blocked. If the value is set to 1, then the action is according to `allowed_ipv6_extension_headers`.

In Case You still Want to Use an IDPS …



- you MUST (header-wise) scrub the traffic before entering the IDPS.

The Most Important “Take Away”

- These are just some of the IPv6 “grey areas”. Other may also exist.
  - Hint: MLD comes to mind… [see other talk later]
- IPv6 security awareness.
  - Test it and use it, in your lab.
  - You will have to do it, sooner or later, anyway…

There's never enough time...

THANK YOU...

...for yours!

Tool & Slides:
https://www.insinuator.net
http://www.secfu.net/tools-scripts/

Questions?

- You can reach us at:
  - aatlasis@secfu.net, www.secfu.net
  - erey@ernw.de, www.insinuator.net
- Follow us at:
  - @AntoniosAtlasis
  - @Enno_Insinuator

Guys, we would love to see you in Heidelberg!

March, 14-18 2016
Heidelberg, Germany
Make the world a safer place.

FOLLOW THE NEWS
@WEareTROOPERS




MORE INFO, EXTENSIVE ARCHIVE & REGISTRATION @ WWW.TROOPERS.DE