IPv6 Security Summit

Troopers 16

The Impact of Extension Headers on IPv6 Access Control Lists

Real Life Use Cases

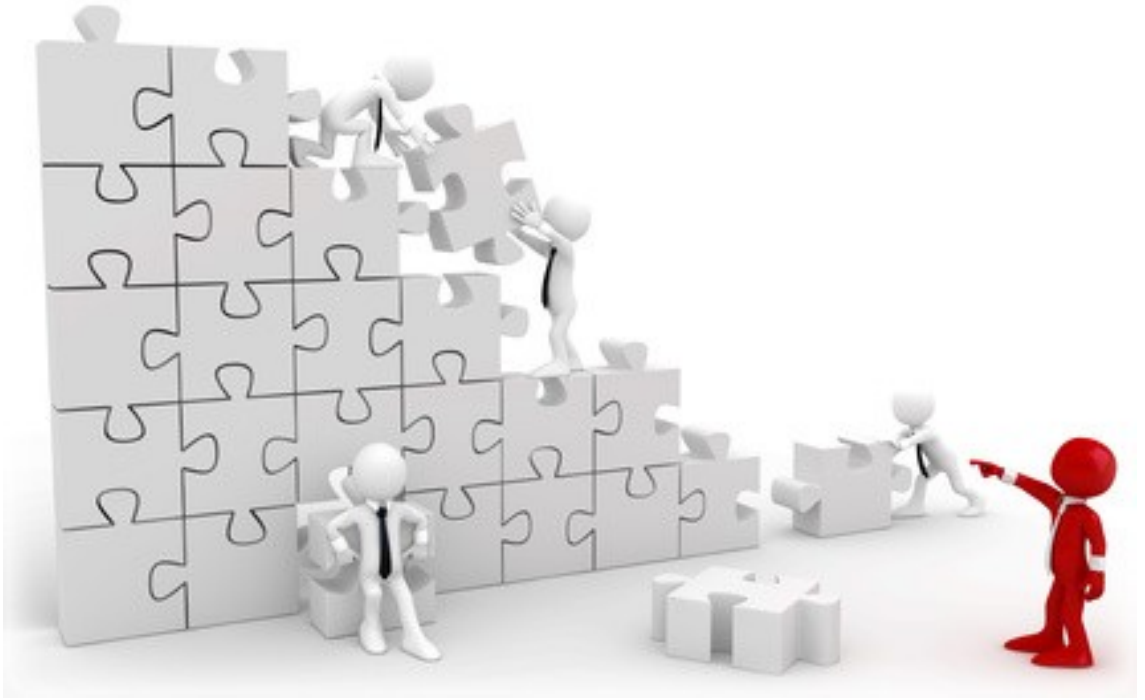
Antonios Atlasis

Heidelberg, 14 March 2016

Who Am I?

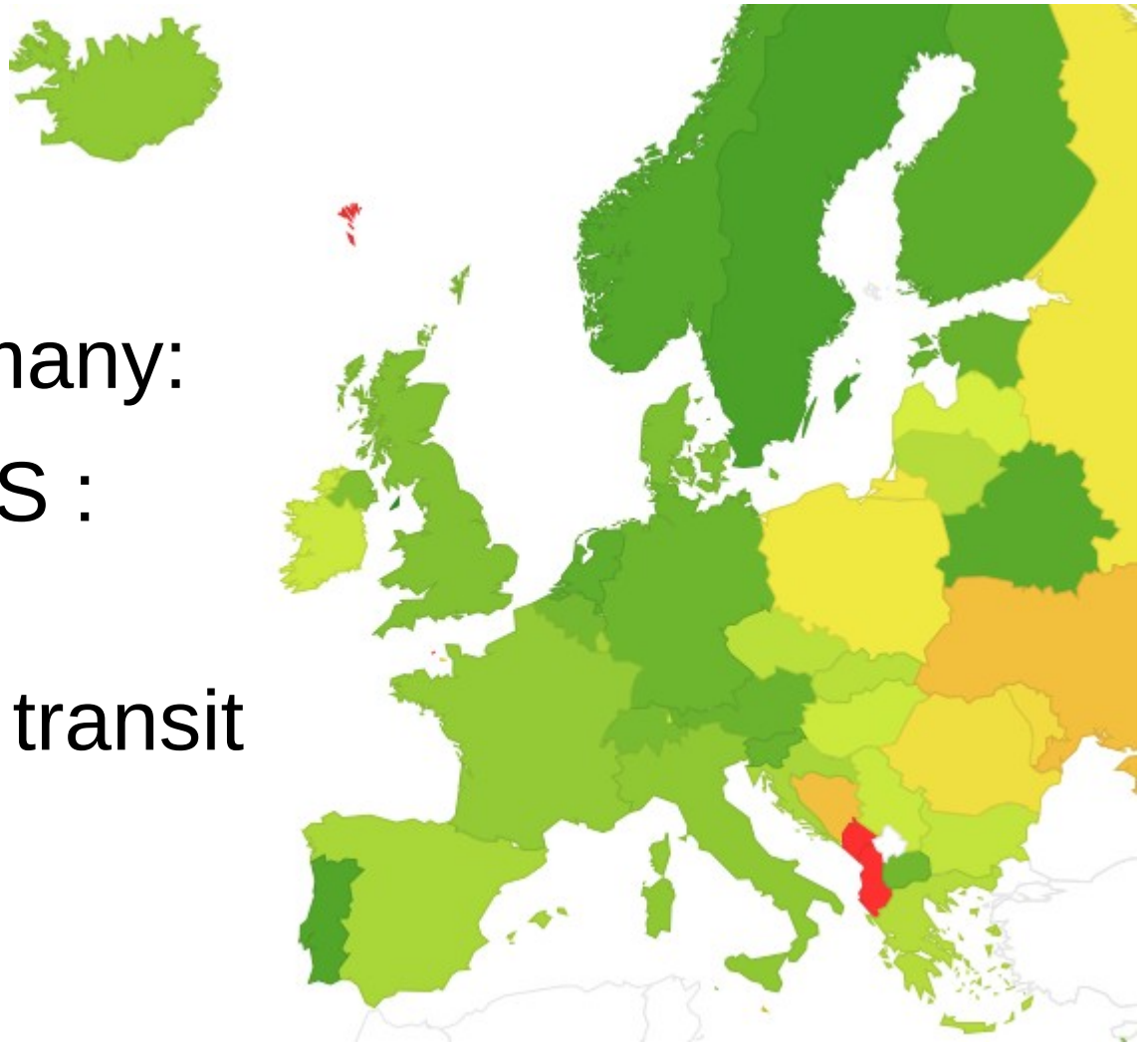
- IT Security researcher with a special interest in IPv6 (in)securities.
- Several related findings, discovered vulnerabilities, and talks in various IT Security conferences.
- Author of *Chiron*.
- Twitter: *@AntoniosAtlasis*

Some Background Information



aatlasis@secfu.net

Cisco Labs Measurements



IPv6 AS in Germany:

- IPv6 transit AS : 81.07%
- IPv6 enabled transit AS : 92.08%

(as of February 2016)

aatlasis@secfu.net

“Protecting Your Core”

Infrastructure Protection

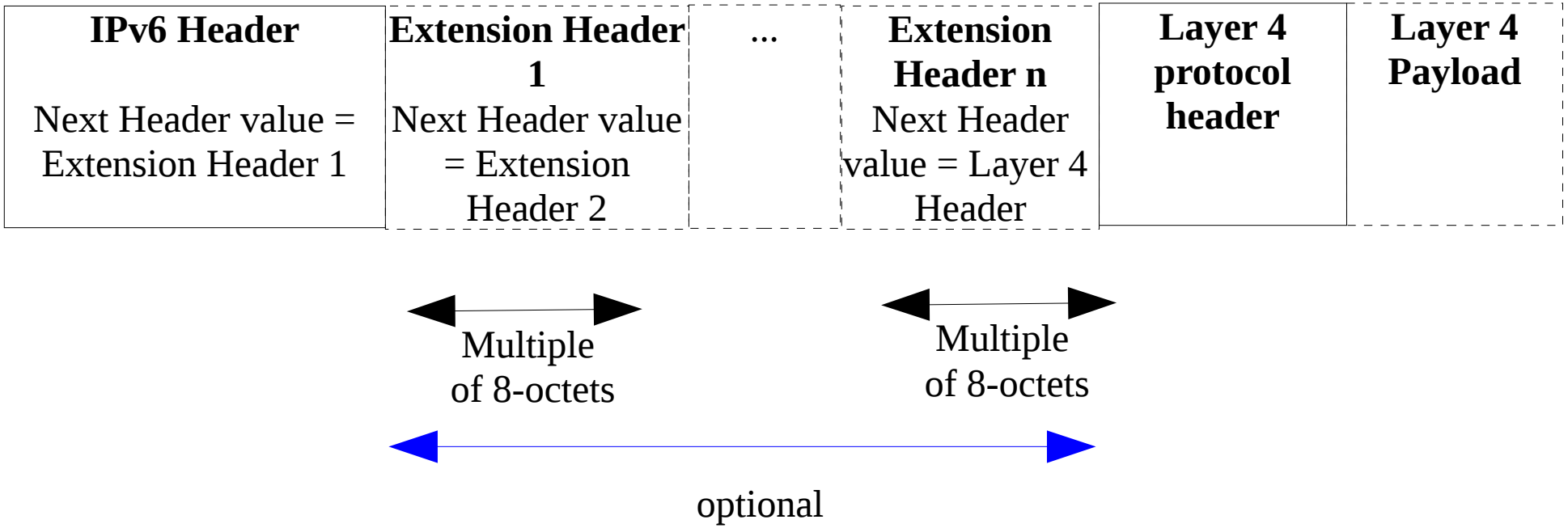
Background

In an effort to protect routers from various risks—both accidental and malicious—infrastructure protection ACLs should be deployed at network ingress points. These IPv4 and IPv6 ACLs deny access from external sources to all infrastructure addresses, such as router interfaces. At the same time, the ACLs permit routine transit traffic to flow uninterrupted and provide basic RFC 1918 , RFC 3330 , and anti-spoof filtering.

Source: http://www.cisco.com/c/en/us/support/docs/ip/access-lists/43920-iacl.html

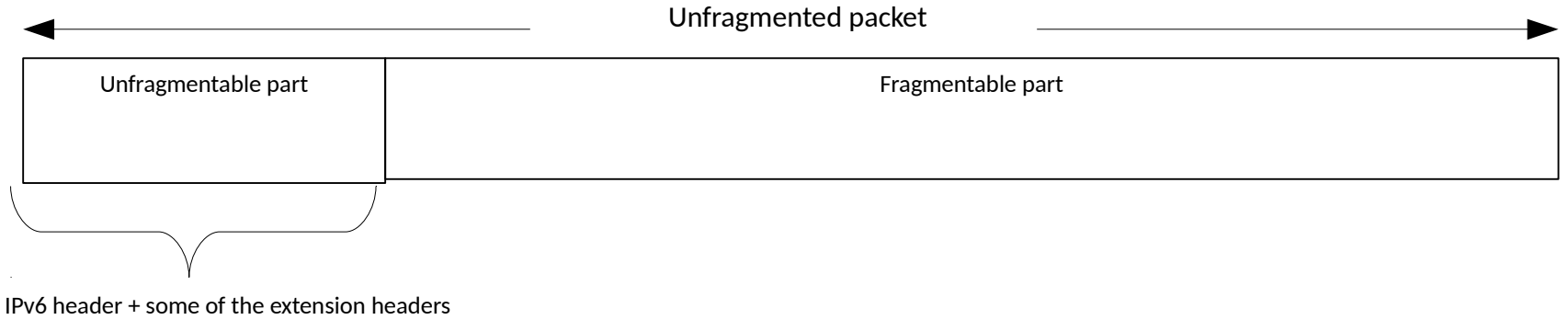
aatlasis@secfu.net

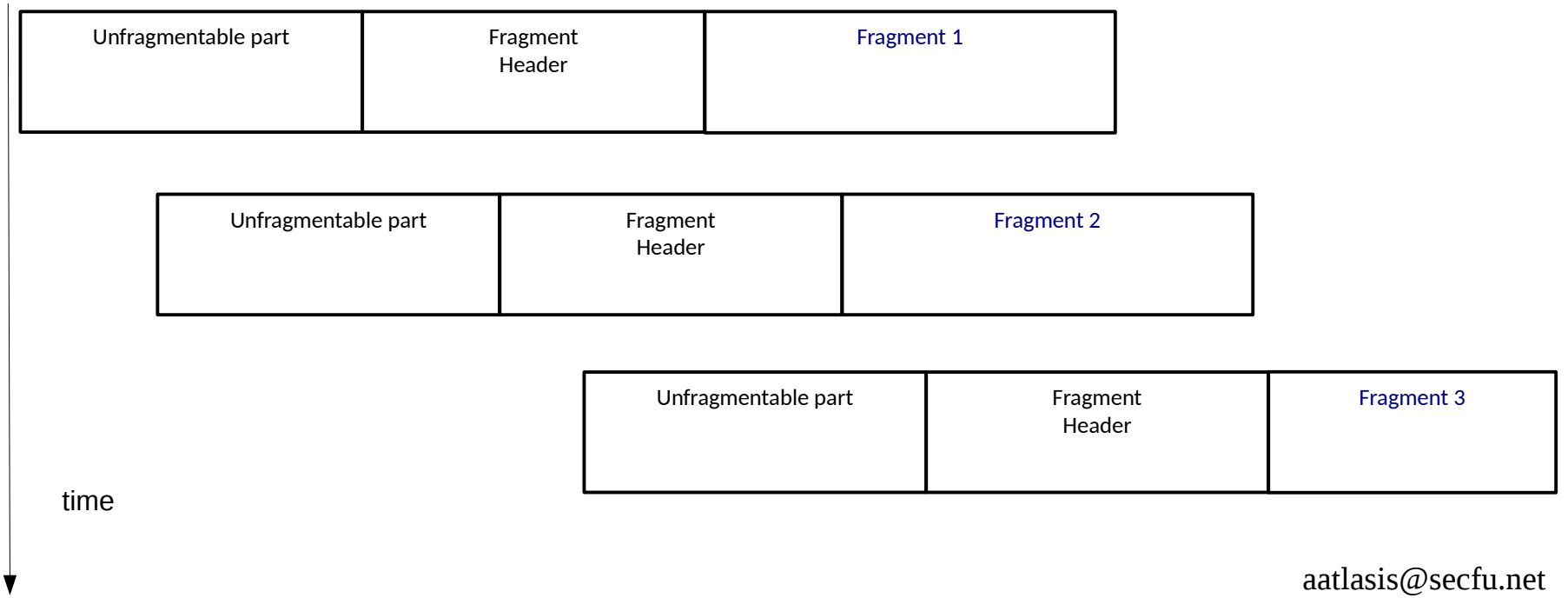
Structure of an IPv6 Datagram



aatlasis@secfu.net

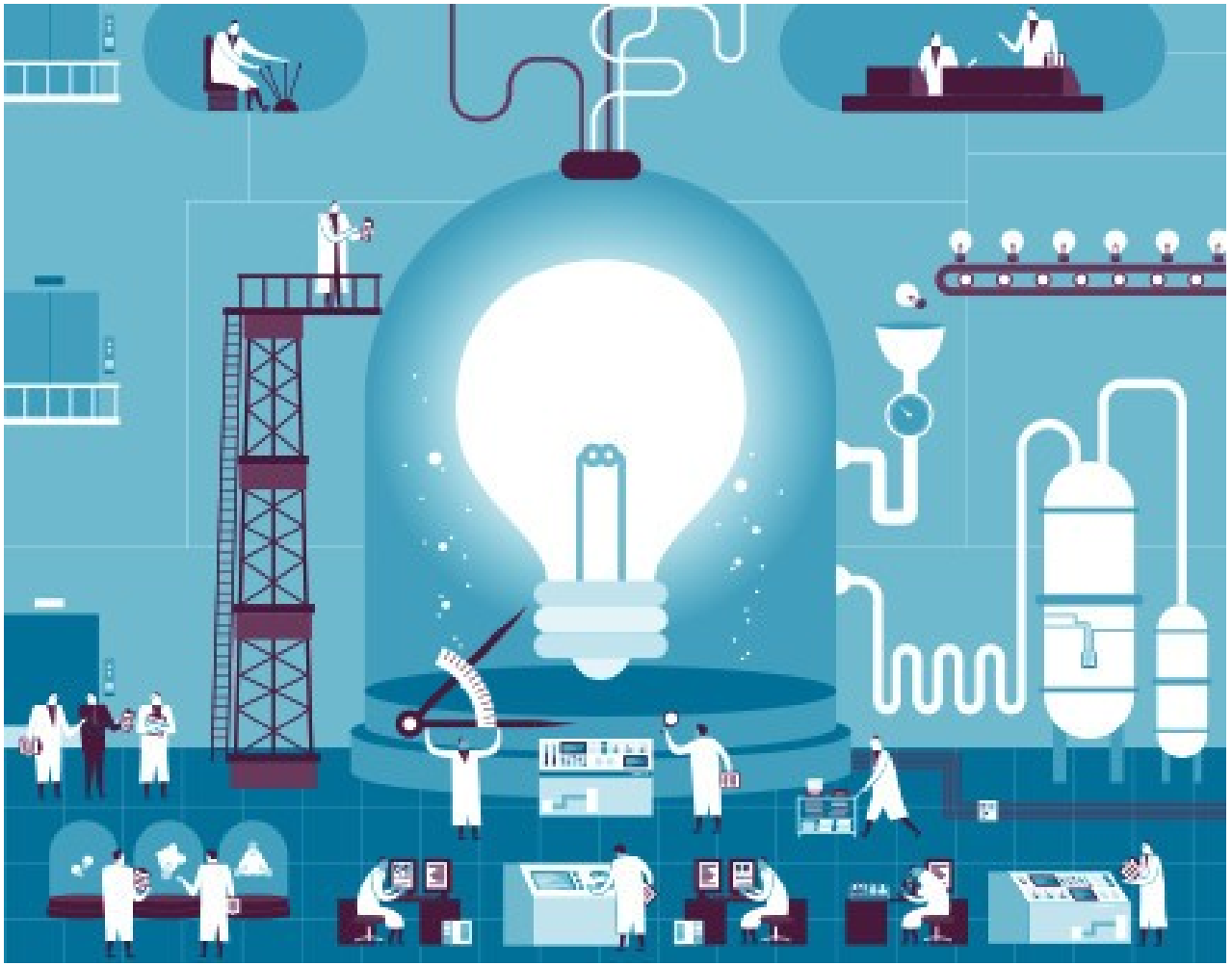
An Example of an IPv6 Fragmentation





aatlasis@secfu.net

Testing Environment



aatlasis@secfu.net

Tested Devices

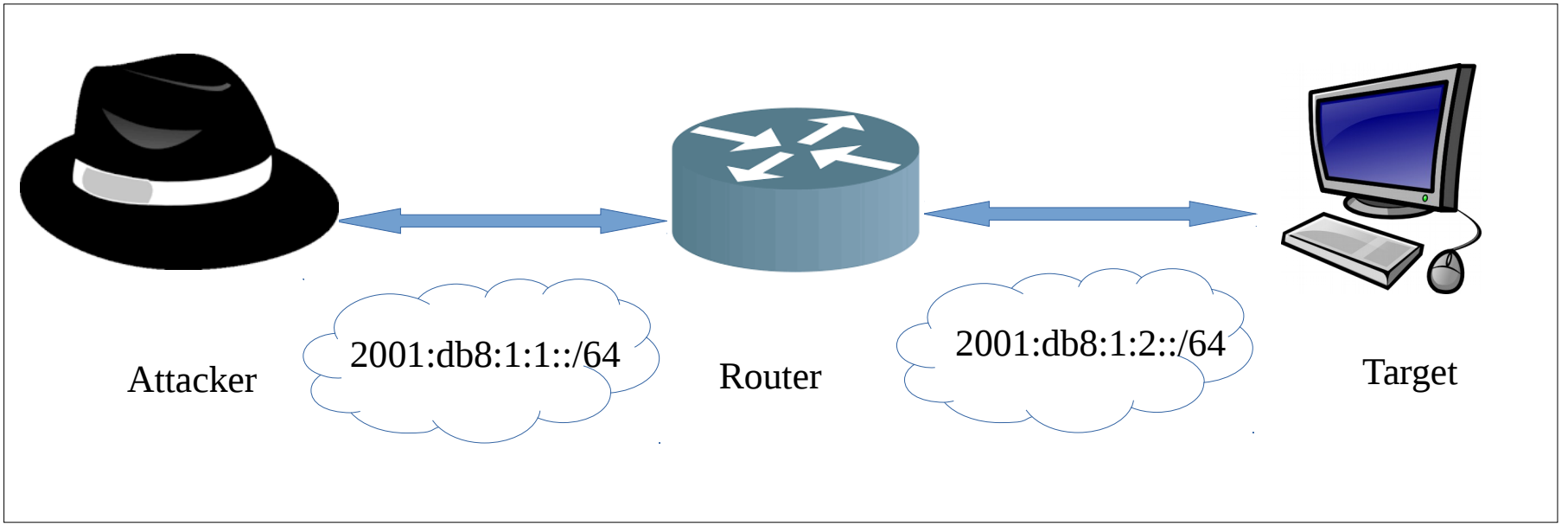
- Cisco:
  - Cisco CISCO1921/K9 (revision 1.0), C1900 Software (C1900-UNIVERSALK9-M), Version 15.4(3)M, REL)
- Hewlett-Packard:
  - HP A5800 JC100A layer-3 switch
- Alcatel
  - TimOS

But the (root cause of the) problem is (almost) vendor neutral.

- Implementation and mitigation techniques may differ.

aatlasis@secfu.net

Lab Set-Up



aatlasis@secfu.net

Use Case A

aatlasis@secfu.net

The Need for Device Management

- Devices need to be managed, many times even remotely.
- Some services (e.g. SSH) need to be open for administration purposes.
- ACLs are used to “protect” them (block their access from the “wild”).

IPv6 ACL Example

The IPv6 access-list must be applied as an extended, named access-list.

```
!--- Configure the access-list.

ipv6 access-list iacl

!--- Deny your space as source from entering your AS. !--- Deploy only at the AS edge.

 deny ipv6 YOUR_CIDR_BLOCK_IPV6 any

!--- Permit multiprotocol BGP.

 permit tcp host bgp_peer_ipv6 host router_ipv6 eq bgp
 permit tcp host bgp_peer_ipv6 eq bgp host router_ipv6

!--- Deny access to internal infrastructure addresses.

deny ipv6 any INTERNAL_INFRASTRUCTURE_ADDRESSES_IPV6

!--- Permit transit traffic.

permit ipv6 any any
```

This entry ensures that all IP protocols are permitted through the core and that customers can continue to run applications without issues

Source: http://www.cisco.com/c/en/us/support/docs/ip/access-lists/43920-iacl.html

aatlasis@secfu.net

Use-Case A: SSH is Blocked and a “Default Allow” Rule is Used

Router#show ipv6 access-list

IPv6 access list protect_infrastructure
deny tcp any any eq 22 sequence 10
permit ipv6 any any sequence 20

The attacker’s goal is to reach the target’s SSH port as well as the SSH port of the router itself.

aatlasis@secfu.net

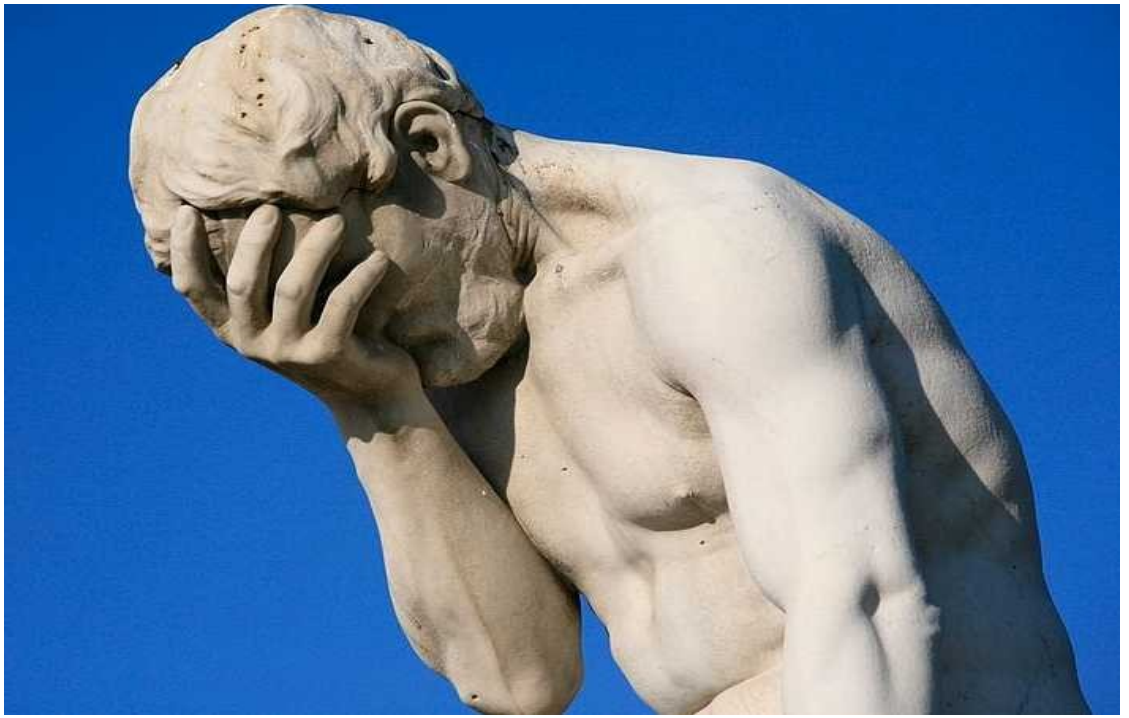
Use Case A(1): Evasion of the ACL at Cisco Devices

- Two Fragments:
  - One Extension Header in the fragmentable part of the 1st fragment.
  - Layer-4 Header in the 2nd fragment.
- Wait, this is not new…
  - But did you know that this Extension Header can be anything (even a Type-0 Routing Header) except from a Hop-by-Hop Header.

aatlasis@secfu.net

Use Case A(2): Evasion of the ACL at HP Device

- Three IPv6 Extension Headers (any) in a row.
- NO FRAGMENTATION IS NEEDED.



aatlasis@secfu.net

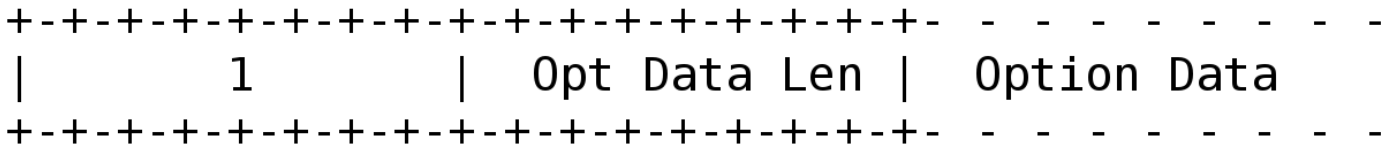
Use Case A(3): Evasion of the ACL at Alcatel Device

- We had to try harder:
  - Six (6) IPv6 Extension Headers (e.g. six Destination Option Headers or different ones) in an UNFRAGMENTED IPv6 datagram.
  - One (1) Extension Header (e.g. a Destination Option Header) and split the datagram in two fragments.

Use Case B

aatlasis@secfu.net

PadN option (alignment requirement: none)



The PadN option is used to insert two or more octets of padding into the Options area of a header. For N octets of padding, the Opt Data Len field contains the value N-2, and the Option Data consists of N-2 zero-valued octets.

Appendix B contains formatting guidelines for designing new options.

4.3 Hop-by-Hop Options Header

The Hop-by-Hop Options header is used to carry optional information that must be examined by every node along a packet's delivery path.

Use-Case B: A HbH Header is Allowed and “Default Deny” Rule

IPv6 access list myrule2

permit hbh any any (1 match) sequence 10
deny tcp any any eq 22 (1 match) sequence 20

The goal of the attacker is to reach any service (like SSH) which is nevertheless blocked by the default deny rule.

aatlasis@secfu.net

Use Case B: Evasion of the ACL when HbH is Allowed

- ALL tested devices:
  - Simply add a Hop-by-Hop header
  - Fragmentation is Optional.
- Similar results can be obtained if a different Extension Header is Allowed.

aatlasis@secfu.net

Use Case C

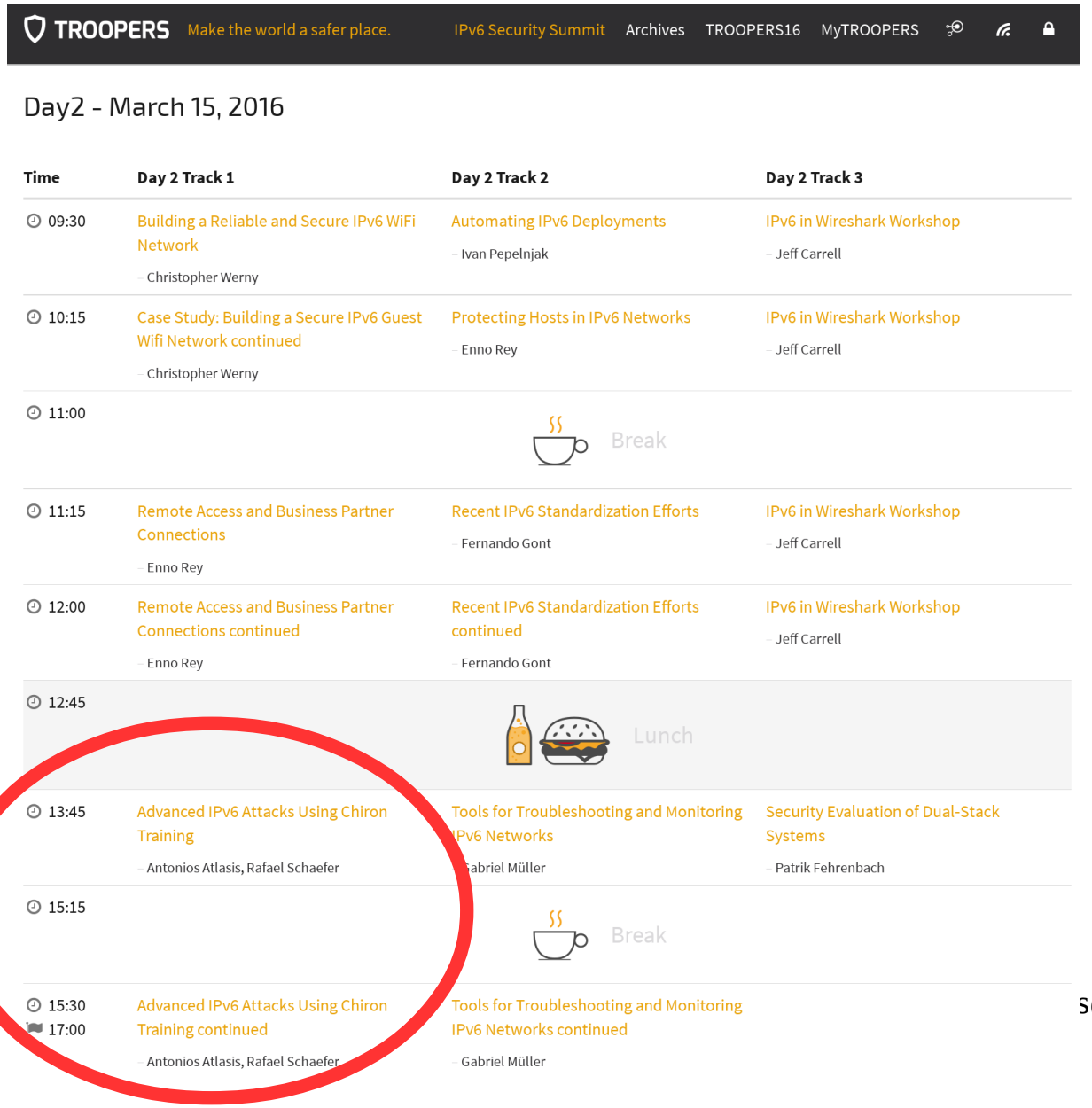
aatlasis@secfu.net

Allow Fragmentation and Block ALL the rest

- We assume that an ISP must support and provide fragmentation capabilities to its customers.
  - We need it, right?
- Spare me the details:
  - Alcatel:
    - Any TCP port number at the target can be reached if the datagram is simply split in two fragments (without adding any Extension Header).
    - *This technique can also be used against other ports or protocols which are explicitly blocked.*

aatlasis@secfu.net

How To Reproduce The Discussed Attacks



TROOPERS Make the world a safer place. IPv6 Security Summit Archives TROOPERS16 MyTROOPERS

Day2 - March 15, 2016

Time	Day 2 Track 1	Day 2 Track 2	Day 2 Track 3
09:30	Building a Reliable and Secure IPv6 WiFi Network Christopher Werny	Automating IPv6 Deployments Ivan Pepelnjak	IPv6 in Wireshark Workshop Jeff Carrell
10:15	Case Study: Building a Secure IPv6 Guest Wifi Network continued Christopher Werny	Protecting Hosts in IPv6 Networks Enno Rey	IPv6 in Wireshark Workshop Jeff Carrell
11:00		Break	
11:15	Remote Access and Business Partner Connections Enno Rey	Recent IPv6 Standardization Efforts Fernando Gont	IPv6 in Wireshark Workshop Jeff Carrell
12:00	Remote Access and Business Partner Connections continued Enno Rey	Recent IPv6 Standardization Efforts continued Fernando Gont	IPv6 in Wireshark Workshop Jeff Carrell
12:45		Lunch	
13:45	Advanced IPv6 Attacks Using Chiron Training Antonios Atlasis, Rafael Schaefer	Tools for Troubleshooting and Monitoring IPv6 Networks Gabriel Müller	Security Evaluation of Dual-Stack Systems Patrik Fehrenbach
15:15		Break	
15:30 17:00	Advanced IPv6 Attacks Using Chiron Training continued Antonios Atlasis, Rafael Schaefer	Tools for Troubleshooting and Monitoring IPv6 Networks continued Gabriel Müller	

s@secfu.net

Mitigation Efforts

aatlasis@secfu.net

Some (Desperate) Attempts

- Simply didn't work:
  - Blocking No Next Headers
  - Use of Cpm Hw Filters

aatlasis@secfu.net

5. Updates to RFC 2460

When a host fragments an IPv6 datagram, it MUST include the entire IPv6 Header Chain in the First Fragment.

A host that receives a First Fragment that does not satisfy the above-stated requirement SHOULD discard the packet and SHOULD send an ICMPv6 error message to the source address of the offending packet (subject to the rules for ICMPv6 errors specified in [RFC4443]). However, for backwards compatibility, implementations MAY include a configuration option that allows such fragments to be accepted.

Likewise, an intermediate system (e.g., router or firewall) that receives an IPv6 First Fragment that does not satisfy the above-stated requirement MAY discard that packet, and it MAY send an ICMPv6 error message to the source address of the offending packet (subject to the rules for ICMPv6 error messages specified in [RFC4443]). Intermediate systems having this capability SHOULD support configuration (e.g., enable/disable) of whether or not such packets are dropped by the intermediate system.

Cisco “Undetermined Transport”

- It is not a “panacea”.

"undetermined-transport" keyword support on various platforms

The access list above warrants some more explanation.

Some platforms may not support acl keyword "undetermined-transport". In that case they may either reject the command altogether, act erratically on such ACLs, or refuse to accept the ACL on the interface, like in the following example:

```
IPv6_FHS(config-if)#ipv6 traffic-filter nofrags in
% This ACL contains following unsupported entries.
% Remove those entries and try again.
    deny ipv6 any FE80::/64 undetermined-transport sequence 20
% This ACL can not be attached to the interface.
IPv6_FHS(config-if)#
```

Source: http://docwiki.cisco.com/wiki/FHS#.22undetermined-transport.22_keyword_support_on_various_platforms

aatlasis@secfu.net

An Alternative to Undetermined Transport

In this case there is still a way to filter "undetermined transport" at the expense of a larger configuration. In this case we must simply apply the logic of "double negatives": instead of denying undetermined transport, we will permit all transports we can determine (the result will be the same!)

The modified configuration and access-list will look like this:

```
!
interface GigabitEthernet1/0/1
 ipv6 traffic-filter nofrags2 in
!
ipv6 access-list nofrags2
 !!!! Uncomment if using the legacy OS vulnerable to overlapping fragments
 ! deny ipv6 any FE80::/64 fragments
 permit 1 any any
 permit 2 any any
 permit 3 any any
 permit 4 any any
 permit 5 any any
 permit tcp any any
 permit 7 any any
 permit 8 any any
 permit 9 any any
 permit 10 any any
 permit 11 any any
 permit 12 any any
 permit 13 any any
 permit 14 any any
 permit 15 any any
 permit 16 any any
 permit udp any any
 permit 18 any any
```

Is this feasible?

Source: http://docwiki.cisco.com/wiki/FHS#.22undetermined-transport.22_keyword_support_on_various_platforms

Going one Step Further!

- Block explicitly unneeded IPv6 Extension Headers:
  - In the Cisco world:

    deny 43 any any

    deny 60 any any

    etc.
- Do not accept fragmented packets:
  - In the Cisco world:

    deny ipv6 any any fragments

aatlasis@secfu.net

What Else Could Work (for handling Use Case B)?

permit tcp any any eq www sequence 10
permit tcp any any eq www hbh sequence 20

In the above example, we do not allow hbh on its own.

It cannot be evaded, but it creates a few problems.

1. Combinations must be repeated for all the services that we want to allow, as well as for all the corresponding Extension headers.

2. False alarms are triggered:

e.g. if we add a Destination Options Header and fragment it in two fragments, these are blocked even when we try to reach the www service.

aatlasis@secfu.net

Conclusions

- There is no silver bullet to protect infrastructure IPv6 routers from ACL evasion attacks.
  - Root cause: Combination of the core network routers and IPv6 “flexibility”
- RFC 7112 certainly to the right direction.
- Vendors' implementation issues makes matter worse.
- The same debate is raised again and again (blocking or not of IPv6 Extension Headers and/or fragmentation).

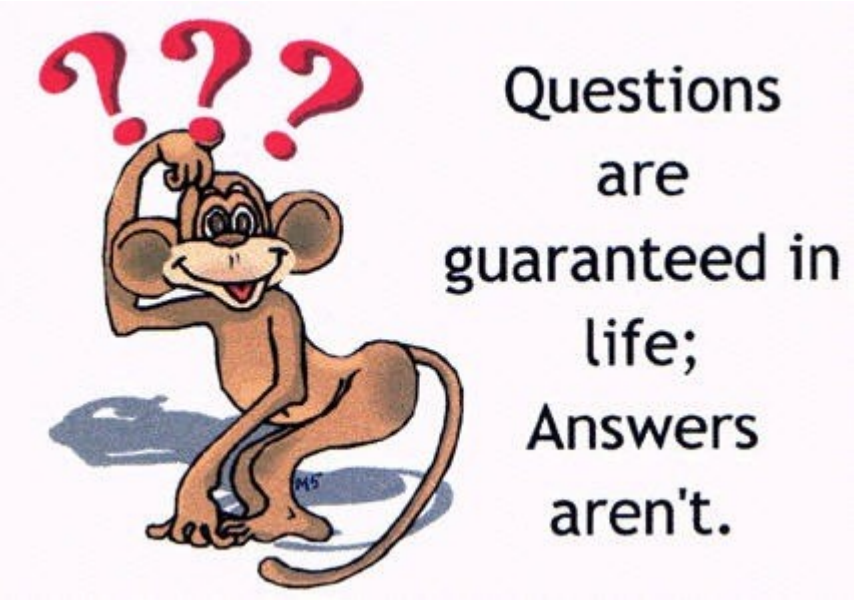
aatlasis@secfu.net

Our Take Away

- Don't take anything for granted in the IPv6 world.
  - Things has changed
  - Including the protection measures that we need to take for the Core networks…
- Test, test, and test :-)
  - *Chiron* can become your friend

aatlasis@secfu.net

Questions?



aatlasis@secfu.net