

IPv6

Testing IPv6 Security Devices



Swiss IPv6 Council
Christoph Weber

Version 1.0 / 24.02.2014

about me

- Christoph Weber
First Hack was more the 30 years ago.
- worked nine years for a large ISP in Switzerland for the development team data center, network and security
 - integration IPv6 in the data center environment
 - IPv4 + IPv6 Security
 - IPv4 old world routing / switching
- Now working as security analyst and engineer in a security operation center.

WARNING !

- Do it in your test environment, especially if you want to keep your job!
- ALL information's are for internal and testing purpose only !
- we are NOT responsible for any abuse use of this information's !
- maybe it is against your local law!
- It may crash perhaps "your" network or server!

agenda

- IPv6 security requirements
- Security threats
- Test case
- Test environment
- Tools and some practical tests
- Results
- Conclusion
- Q&A (at the end of the 2nd presentation)

Types of Security Devices Testing

- Performance testing (not covered)
 - New session/sec
 - Speed with 10000 rules
 - Delay / Jitter
- Usability (not covered)
 - Administration
 - Rule upload
 - Easy to use / handling
- **Security (this presentation)**
 - **Filtering options**
 - **Detection**
 - **IPv6 self protection**



Live sample

Is this a IPv6 Security Problem ?

- Log entry:

```
3411206; 4Feb2014;  
6:03:03;aaa.bbb.ccc.4;log;accept;inbound;Lan1;;VP  
N-1 & FireWall-1;300;{81CBF2C9-3D89-4C85-A0C5-  
E58D7ED842A4};;SIT;xxx.yyy.zzz.132;;192.88.99.1;;  
41;iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii
```

aaa.bbb.ccc.4 is the IP of the Firewall
xxx.yyy.zzz.132 is a public IPv4 in the DMZ

- Traffic outgoing to 192.88.99.1

IDS/IPS Says

Record Details

Previous Next Copy Switch Colors

IPv6 In IPv4 Tunneling
IPv6 Protocol Violation

Confidence Level: Medium-High
Severity: High

Log Info	
Product	IPS Software Blade
Date	4Feb2014
Time	9:33:13
Number	7513003
Type	Log
Origin	[REDACTED]

Traffic	
Source	[REDACTED]
Destination	192.88.99.1
Service	---
Protocol	ipv6
Interface	Exp1-1
Source Port	---

Policy	
Policy Name	Default [REDACTED] New_7-6-10
Policy Date	Thu Jan 16 12:11:37 2014
Policy Management	[REDACTED]
IPS Profile	Inactive

General Event Information	
Action	Drop
Protection Name	IPv6 In IPv4 Tunneling
Attack	IPv6 Protocol Violation
Attack Information	IPv6 in IPv4 tunneling
CVE List	
Severity	High
Confidence Level	Medium-High
Performance Impact	Medium
Protection Type	Application Control
Follow Up	Followed
	Open Protection... Add Exception... Go To Advisory...

Attack Information	
Resource	---
Reject ID	---
Reason	---

More	
Source	[REDACTED]
Rule	300
Rule UID	{81CBF2C9-3D89-4C85-A0C5-E58D7ED842A4}
Current Rule Number	300-Default [REDACTED] New_7-6-10
Industry Reference	None
Information	Update Version: 633120785

Abort Close

Customer says

- Customer response to the demand about IPv6 in IPv4 tunneling traffic:

```
Grüezi Herr [REDACTED]  
  
seitens Informatik sind keine Clients oder Server mit  
IPv6 konfiguriert. Da der Verkehr scheinbar von innen  
nach aussen geht, ist es aus unserer Sicht nicht  
kritisch.  
  
Bitte teilen Sie uns mit, wenn Sie es als notwendig  
ansehen, einen Filter zu setzen.  
  
Freundliche Grüsse  
[REDACTED]
```

Answer

- YES, it is a security problem !
- RFC 3068

An Anycast Prefix for 6to4 Relay Routers

.

2.4 6to4 Relay anycast address An IPv4 address used to reach the nearest 6to4 Relay Router, as defined in this memo. The address corresponds to host number 1 in the 6to4 Relay anycast prefix, 192.88.99.1.

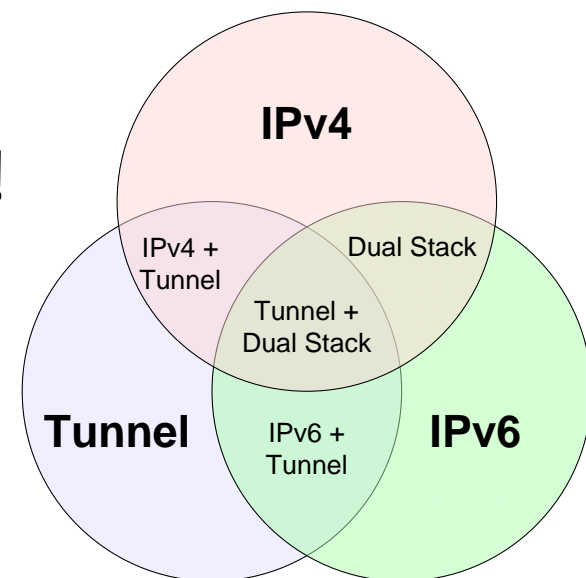
- RFC 7123
- Security Implications of IPv6 on IPv4 Networks



Security requirements

IPv6 Security requirement

- Mostly heard from the customer:
 - the same „Security level“ as in IPv4 !
 - NO additional equipment
 - NO additional resources
- But:
 - is this still enough ?
 - new and more attacking vectors !
 - new requirements for IPv4 !



IPv6 Security dreams and wishes

- Security devices can handle IPv6 and IPv4 and any IPv4 in IPv6 and IPv6 in IPv4 and
- Filter extension headers
- Filter any protocols
- Filter possibility for «any» fields in the packet
- Allows only „good“ packets
- Self defending
- intelligent devices
- ...

never ending list

Firewalling IPv6 packet options

- Link layer (L2) type verification (Ethertype 0x86DD) and version (6) matching
- Filtering of traffic class (filtering unwanted data channel)
 - remove unwanted QOS Flags (zeroing)
 - match if not equal zero
- Filtering of flow label (filtering unwanted data channel)
 - eliminate unwanted flowlabels
 - match if not equal Zero
- Filtering of payload length
- Filtering for “hop limit” field
 - for some neighbor discovery and autoaddress packets (=255)



Firewall rules IPv6 packet options

- Next header filtering
 - any type of next header (256 Types)
 - a max amount of next header
 - a defined order of next headers
- On each option header type
 - matching of any header type specific fields (different on each option header type)

IPv6 header	TCP header + data
Next header = TCP	

IPv6 header	Routing header	TCP header + data
Next header = Routing	Next header = TCP	

IPv6 header	Routing header	Fragment header	Fragment of TCP header + data
Next header = Routing	Next header = Fragment	Next header = TCP	

← 4 octets →	
Option Type	Opt Data Len
Address Version	
Holding Time	
Timestamp	
Identifier (16 octets)	
Address (16 octets)	
Authentication Data (64 octets) (RSA digital signature)	

(b) Mobility option for control

IDS / IPS dreams

- IPv6 packet anomaly detection
- Deep packet inspection in IPv6 and all kind of tunnels (6in6, 4in6, 6in4, 4in4)
- Reassembling of fragmented IPv6 streams
- One box and ruleset for ALL

Spam / Antivirus / DDoS

- Same SPAM functionality like in IPv4
- Antivirus function in IPv6 and IPv4
- DDoS protection for both IPv4/IPv6

- For all
 - Correlation of IPv4 and IPv6 attacks
 - a configuration for both stacks

Many other dreams, but..

- Are all these dreams really necessary ?
- Possible ?
- Manageable ?
- Useful ?
- Make they sense ?
- How big is the speed / performance impact
- is this only my dream ?

draft-gont-opsec-ipv6-firewall-reqs-00

- New IETF Draft from Fernando Gont Requirements for IPv6 Firewalls

5. IPv6-Specific Features

REQ SPC-1:

MUST be able to filter ICMPv6 [[RFC4443](#)] traffic at a message type/code granularity.

REQ SPC-2:

MUST be able to block IPv6 packets that employ a Routing Header (both at the granularity of Extension Header Type and Routing Header Type).

REQ SPC-3:

MUST be able to detect IPv6 tunnels such as SIT, 6to4, 6in4, ISATAP and Teredo (please see [[RFC7123](#)], and must be able to selectively block or allow them for specific sources, destinations, routes or interfaces.

REQ SPC-4:

MUST be able to filter ICMPv6 traffic at a message type/code granularity.

REQ SPC-5:

MUST be able to validate IPv6 Neighbor Discovery [[RFC4861](#)] packets (RS, RA, NS, NA, Redirect) according to [[I-D.ietf-opsec-ipv6-nd-security](#)].

REQ SPC-6:

MUST be able to statefully match ICMPv6 errors to TCP [[RFC0793](#)], UDP [[RFC0768](#)], and ICMPv6 [[RFC4443](#)] communication instances.

REQ SPC-7:

MUST be able to find the upper-layer protocol in an IPv6 header chain (see [[RFC7112](#)]).



IPv6 Testcenters

The «official» ones

NIAP

- <http://www.techopedia.com/definition/24850/national-information-assurance-partnership-niap>

National Information Assurance Partnership (NIAP)



Definition - What does *National Information Assurance Partnership (NIAP)* mean?

The National Information Assurance Partnership (NIAP) is a U.S. government initiative that looks at products in the information technology (IT) realm and ensures that they adhere to international standards. Adhering to standards is highly desirable in today's technological world. NIAP was created as a partnership between the National Institute of Standards and Technology (NIST) and the National Security Agency (NSA) to ensure that products related to technology are conforming to certain standards.

NIAP is also a Common Criteria Evaluation and Validation Scheme (CCEVS) validation body that is managed by the NSA. The purpose of the CCEVS is to create a national program for evaluating IT products against what is called the international common criteria for information technology security evaluation. There are also labs for IT product security testing.



Techopedia explains *National Information Assurance Partnership (NIAP)*

The CCEVS is responsible for looking at security evaluations conducted by the Common Criteria Testing Laboratories (CCTLs), which are approved by CCEVS, and issuing common criteria certificates for those products. When an IT product receives the certificate and the validation report accompanying it, this indicates the product received an evaluation at a laboratory accredited using the common evaluation methodology to conform to the common criteria.

In addition, CCEVS keeps a list of all products that have received evaluations and validations in a validated products list. Therefore, if someone is interested in finding out whether a product has been evaluated and received a certificate, they could simply look on NIAP's CCEVS website under the validated products list page.

Posted by: [Cory Janssen](#)

The screenshot shows the NIAP website header with the text "National Information Assurance Partnership Common Criteria Evaluation & Validation Scheme". Below the header is a navigation menu with links for NIAP Home, Evolution, CCEVS Big Picture, Announcements, CCEVS Products, Documents & Guidance, Useful Links, Protection Profiles, Contact Us, and NIAP Community. The main content area is titled "Product Compliant List" and contains text about the requirements of the NIAP program and where applicable, the requirements of the Federal Information Processing Standard (FIPS) Cryptographic validation program(s). The text mentions that products on the PCL are evaluated and accredited at licensed/approved evaluation facilities for conformance to the Common Criteria for IT Security Evaluation (ISO Standard 15408). U.S. Customers (designated approving authorities, authorizing officials, integrators, etc.) may treat these mutually-recognized evaluation results as complying with the Committee on National Security Systems Policy (CNSSP) 11 National Policy Governing the Acquisition of Information Assurance (IA) and IA-Enabled Information Technology (IT) Products - dated June 2013 (https://www.cnss.gov/policies.html). The text also mentions that NIAP embraces the CCRA Management Committee Vision Statement for the future direction of the application of the CC and the CCRA. We are transitioning to an evaluation paradigm that produces achievable, testable, and repeatable evaluation results. Achieving success in the CCRA evolution requires a transition to Protection Profile compliance and a move away from EALs. This strengthens evaluations by focusing on technology specific, tailored assurance requirements. The list of products in the following portion of the list (not those listed in the VPL) are evaluated under the new Protection Profile paradigm against a NIAP-approved PP. This includes a collection of assurance activities tailored to the technology with no EAL assigned - hence the conformance claim is "PP".

On the right side of the screenshot, there is a search bar labeled "SEARCH CCEVS" and a section titled "Announcements" with several links: MDFFP Annex Published, Trusted Platform Module (TPM) Call for Participants!, NDPP Errata Published, Successful Completion of Gossamer Accreditation Evaluation, and NDPP Errata Published.

USGV6

- <http://www-x.antd.nist.gov/usgv6/index.html>

Information Technology Laboratory
Advanced Network Technologies Division

NIST
National Institute of Standards and Technology

NIST Home > IITL > Advanced Network Technologies Division > USGV6

Links

1. USGV6 Home
 - o Intro&News
 - o USGV6 Profile
 - o USGV6 FAQ
2. For Purchasers:
 - o Buyers Guide
 - o Tested Products
 - o User's Guide
 - o SDOC
 - o NRT v1.0
 - o Deployment Test Spec
3. For Suppliers:
 - o User's Guide
 - o SDOC
 - o Labs and Accreditors
4. For Test Labs:
 - o Test Method Validation
 - o Test Methods
 - o Test Specifications
 - o Forward Tests
 - o Labs and Accreditors
 - o Interlab Comparisons
 - o Test Labs Feedback
5. For Operators:
 - o Secure v6 Deployment
 - o Deploy Test Spec
 - o Deploy Monitor
 - o Test your IPv6
 - o Berkeley Netalzyr
6. For Everybody:
 - o Intro & News

Top News Item

Following changes to DHCP and IKEv2 test suites, the SDOC template has been amended, and a new version issued as SDOC Version 1.9.

USGV6: A Technical Infrastructure to Assist IPv6 Adoption

OMB Memorandum M-05-22 directed the National Institute of Standards and Technology (NIST) to develop the technical infrastructure (standards and testing) necessary to support wide scale adoption of IPv6 in the US Government (USG). In response NIST developed a technical standards profile to assist acquisition of IPv6 capabilities in Hosts, Routers, and Network Protection Devices. The Host and Router profile includes a forward looking set of RFCs published by the Internet Engineering Task Force (IETF), encompassing basic IPv6 functionality, and specific requirements and key optional capabilities for routing, security, multicasting, mobility, network management, and quality of service. The Network Protection Device profile contains a NIST established set of capability requirements for IPv6-aware firewalls and intrusion detection systems.

In addition to the profiles, a testing program has been established to enable products to be tested for compliance with the profile by accredited laboratories.

The profile is embodied as a set of recommendations of NIST, and the test program supports open and voluntary involvement. While neither contribution from NIST was developed to embody policy directly, it is envisioned that these components can provide a technical infrastructure upon which other policies and plans can be based. For example Federal Acquisition Regulations and specific OMB USG IPv6 Adoption Directives have been based upon the USGV6 Profile and Testing System.

This web site provides information on the USGV6 Profile and Testing Program. The menu on the left highlights the pages relevant to IPv6 product Purchasers, Suppliers who develop and sell products, Test laboratories who perform conformance, interoperability and network protection testing. Please note that this website does NOT provide an "approved products list", though it does reference the tested products pages of the accredited laboratories. Product suppliers are directed to provide a Supplier's Declaration of Conformity (SDOC) for their tested products, and Purchasers to express their requirements using the vocabulary of the profile and summarized in USGV6 capabilities requirements as described in the USGV6 Profile.

This website describes the testing infrastructure for products seeking compliance with the USGV6 Profile. The menu on the left highlights the pages relevant to IPv6 product Purchasers, Suppliers who develop and sell products, Test laboratories who perform conformance, interoperability and network protection testing. Please note that this website does NOT provide an "approved products list", though it does reference the tested products pages of the accredited laboratories. Product suppliers are directed to provide a Supplier's Declaration of Conformity (SDOC) for their tested products, and Purchasers to express their requirements using the vocabulary of the profile and summarized in USGV6 capabilities requirements as described in the



Security threats

Know the Attacks

Security threats

- Define the IPv6 security threats
- Classify the threats
- Sort threats by relevance, impact, ... related in your environment
- Watch for NEW upcoming threats

- Know the OLD IPv4 threats

Overview

Threats table

Overview Threats

		Case / Impact	Global	Local	Single Device Impact	IPv6 Service Impact	DDoS Local Link	DDoS Global	DDoS	Device Table Impact	CPU Impact	Routing Impact Local	Routing Impact Global	MITM	Split Horizon	Reconnaissance
Nr	Threats															
1	Router Advertisement DoS Attacks					x	x		x	x	x	x	x			
2	Local DoS Amplifier					x	x		x							
3	ICMP Redirect-Spoofing											x		x		
4	ICMPv6 Renumbering spoofing					x	x		x			x		x		
5	Neighbor Advertisements DoS Attack ?????					x	x		x	x	x					
6	Neighbor Cache table overload					x	x		x	x	x					
7	Multicast DNS table overload					x	x		x	x				x		
8	SEcure Neighbor Discovery (SEND) DoS Attack					x	x		x		x					
9	MTU attack (ICMPv6 Too big)					x		x	x							

Threats

- Sample: „ICMPv6 packet too big tunneling“

Titel	ICPMv6 packet too big tunnelling / flooding	T	
Description	Angriffsszenario: Da für eine richtige Funktionsweise von IPv6 und der MTU auf allen Nodes/Firewall/Router ICMPv6 Too Big (Type 2) erlaubt sein muss, kann diese Art von ICMPv6 Messages dazu verwendet werden trotz Firewall und anderen Devices, einen Tunnel von Intern nach Extern oder Umgekehrt aufzubauen, oder mit diesen ICMPv6 Paketen die Netze über Firewalls/Filterdevices hinweg zu fluten.		
	Auswirkungen: Bypass von Firewall / Security Filtern / unbekannte Kommunikation via ICMPv6 Too Big Tunnels ICMPv6 flooding ins interne Netz		
	Lösung Security Devices , die nur ICMPv6 Antwort Pakete durchlassen, für die sie auch einen Verbindungsaufbau Versuch (SYN – Packet) dazu haben. Eine Art Statefull Tabelle.		
	Links:		
Referenz	RFC 4443		



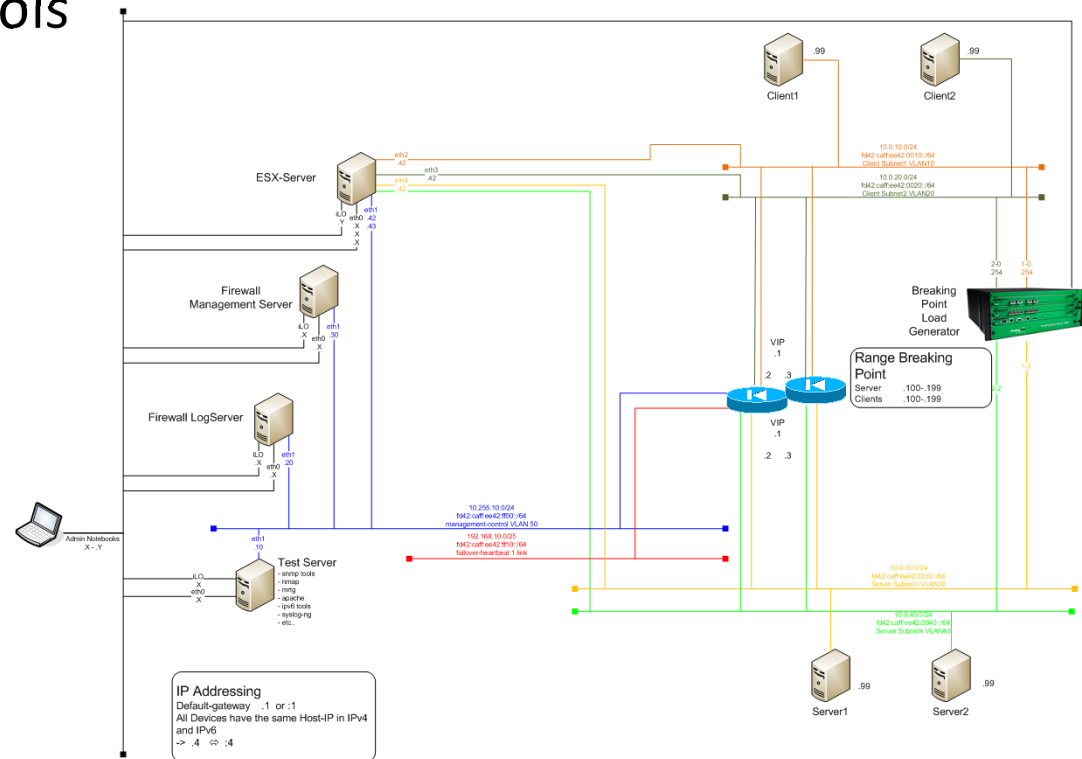
Test environment & case

Testlab & test scenarios

Define test cases / test environment

- for any security threat it is necessary to create a test case.
- Build test environment -> based on your the requirements
- Determine the test tools

Lab Overview



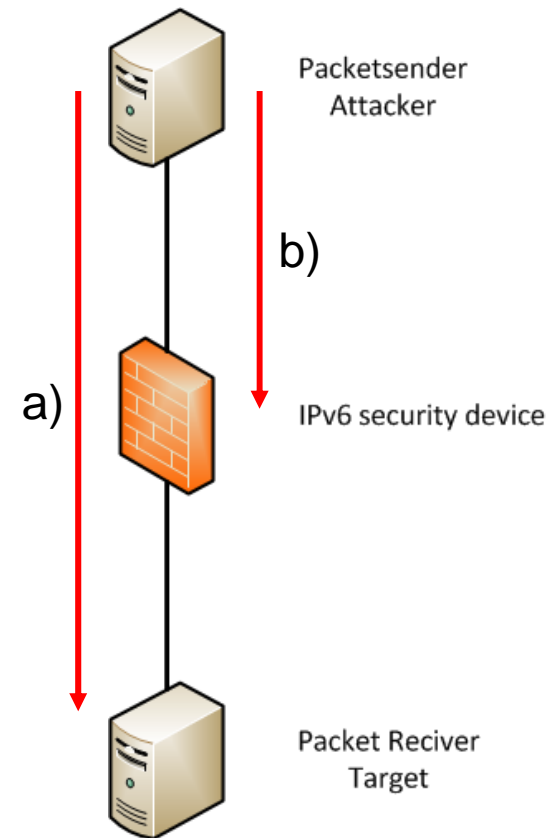
Define test case

Basic setup

- Traffic sniffing on the wire. attacker's side, and on target side.

2 Types of tests

- a) Function of the security device
- b) Attacking the security device



Define Testcase

- Write test case for each security threat with all sub cases.
- Define test case very detailed and clearly, for a clear testing and comprehensibility
- recycle test cases

IPv6 ICMPv6 Packet too Big filtering / Bypass / Flooding		IP6-ICMP6-004
Priority	2	
Description	<ul style="list-style-type: none"> • ICMPv6 Packet too Big Type 2 Code 0 	
Device	Security Device State less / State full	
Setup	<p>Create Firewall Object with ICMPv6 Type 2 Code 0 and apply the Object to the Security Rule</p> <p>Rule Ext → Int PERMIT Source_IPv6_External/64 TO Dest_IPv6_Internal/64 Object "Packet_too_Big" log</p> <p>Rule Int → Ext PERMIT Dest_IPv6_Internal/64 TO Source_IPv6_External/64 Object "Packet_too_Big" log</p> <p>Send ICMPv6 Packet from Internal to External with</p> <ul style="list-style-type: none"> - Type 2 Code 0 → PASS / FAIL if Devices knows Status - Type 2 Code 1 to Code 255 → FAIL <p>Test all Code (0 to 255) with different Data in the ICMP Packet</p> <ul style="list-style-type: none"> - Date is from a SYN Packet - Random Data <p>Do all the tests with only the Packet too Big PERMIT Rule, and a DENY ANY ANY Rule</p>	
Procedure	<p>a) Create ICMPv6 Object with Type 2 Code 0</p> <p>b) Create Firewall Rule</p> <p>Rule Ext → Int PERMIT Source_IPv6_External/64 TO Dest_IPv6_Internal/64 Object "Packet_too_Big" log</p> <p>Rule Int → Ext PERMIT Dest_IPv6_Internal/64 TO Source_IPv6_External/64 Object "Packet_too_Big" log</p> <p>Apply to the Security Device</p> <p>Create on the Client Workstation with Scapy the Packet ICMPv6 Type 2 Code 0 to Type 2 Code 255 See with tcpdump / Wireshark if the packet is seen on the Server interface</p>	

Write down the results

- Results must be documented !!
- Required information
 - Device type, serial number, software version
 - Date / Time / Tester
 - Results / capture-files / screenshots /
all info / references to external documents
 - Results and summary (PASS/FAIL/Part. PASS)
 - Overall Status / Next Steps



Test case sample

Samples form the real live

Cisco ACL's

- IPv6 is not IPv4 !
- Know the difference between IPv4 and IPv6
- Watch for CPU impact and rule length

implicit deny rule

Difference between

- IPv4

```
deny ip any any
```

- IPv6

```
permit icmp any any nd-na
```

```
permit icmp any any nd-ns
```

```
deny ip any any
```

implicit deny Rule

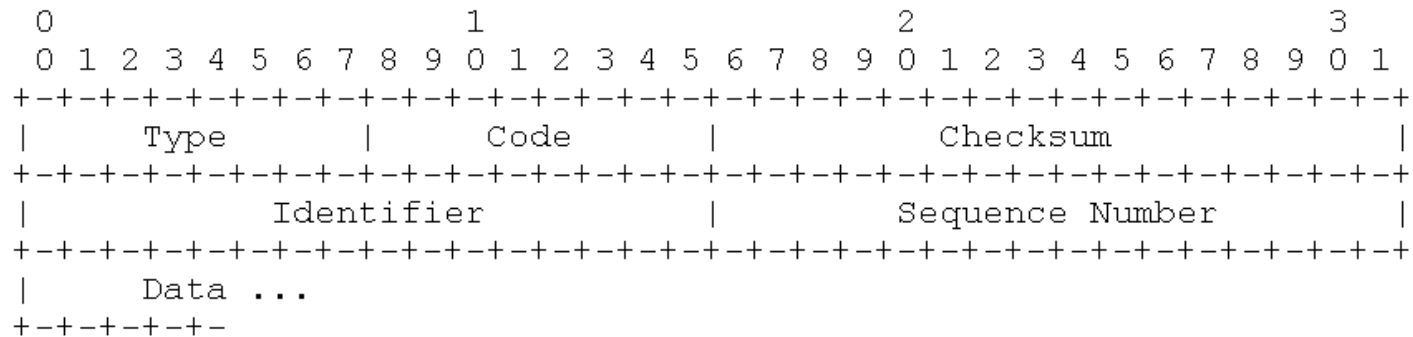
- Main question: filter all the same ?
 - a) `permit icmp any any echo-request`
 - b) `permit icmp any any 128`
 - c) `permit icmp any any 128 0`

- a) and b) are the same
 - They don't filter on the code level
- c) Allows only type 128 code 0

RFC 4443

4. ICMPv6 Informational Messages

4.1. Echo Request Message



IPv6 Fields:

Destination Address

Any legal IPv6 address.

ICMPv6 Fields:

Type 128

Code 0

Identifier An identifier to aid in matching Echo Replies to this Echo Request. May be zero.

ACL for the lab

Test with differed ACL are required

Version “echo-request”

```
IPv6 access-list ICMP-TEST-IN
permit icmp any any nd-na
permit icmp any any nd-ns
permit icmp any any echo-request
deny ipv6 any any log
```

Version “Type / Code”

```
IPv6 access-list ICMP-TEST-IN
permit icmp any any nd-na
permit icmp any any nd-ns
permit icmp any any 128 0
deny ipv6 any any log
```

Impact / Solution

- What have we done in the IPv4 ruleset ?
Mostly filtering «ICMP echo-request»
- On some Cisco devices, huge impact to the CPU, if filtering “code” options
Example: Cisco 6500
- do your best, but do it !

Firewall config (Sample Fortinet)

- Predefined objects ? „ALL_ICMP6“

The screenshot shows the Fortinet Firewall configuration interface. On the left, a navigation tree is visible with 'Firewall Objects' selected, and 'Service' is highlighted. The main configuration area shows the following fields:

Name	ALL_ICMP6
Comments	Write a comment... 0/255
Show in Service List	<input checked="" type="checkbox"/>
Category	General
Protocol Type	ICMP6
Type	
Code	

- Read the documentations and/or ask the vendor, what each field means.
- One of the questions is: what is, if one field is empty ? Example „Code“ (here it means „ALL“)


The screenshot shows the Fortinet Firewall configuration interface for a specific service. The fields are filled as follows:

Name	ICMPv6 Packet too Big (Type2/Code0)
Comments	Write a comment... 0/255
Show in Service List	<input checked="" type="checkbox"/>
Category	General
Protocol Type	ICMP6
Type	2
Code	0

Sample „packet to big“

- Create packet and send (manual way)

```
root@net-bear:~/ipv6# scapy
INFO: Can't import python gnuplot wrapper . Won't be able to plot.
Welcome to Scapy (2.2.0-dev)
>>> P=IPv6()
>>> I=ICMPv6EchoRequest()
>>> P.src="fd42:caff:ee42::1000"
>>> P.dst="fd42:c0d0:e0f0::1000"
>>> I.code=0
>>> S=(P/I)
>>> S.show2()
###[ IPv6 ]###
  version= 6L
  tc= 0L
  fl= 0L
  plen= 8
  nh= ICMPv6
  hlim= 64
  src= fd42:caff:ee42::1000
  dst= fd42:c0d0:e0f0::1000
###[ ICMPv6 Echo Request ]###
  type= Echo Request
  code= 0
  cksum= 0xa33
  id= 0x0
  seq= 0x0
  data= ''
>>> send(S)
.
Sent 1 packets.
```

The background of the terminal window features the Kali Linux logo, a stylized dragon, and the text "KALI LINUX" in a large, outlined font. Below the logo, the slogan "The quieter you become, the more you are able to hear." is visible.

Tools

- **SCAPY** (Use: release 2.2.0 DEV)
Python tool for easy creating single packet
<http://www.secdev.org/projects/scapy/>
- **THC-Tools**
IPv6 Attacking tools
<https://www.thc.org/thc-ipv6/>
- **IPv6 Toolkit**
Tool for testing IPv6
<http://www.si6networks.com/tools/ipv6toolkit/>
- **ft6**
Tool for IPv6 Firewall testing
<http://www.idsv6.de/en/index.html>
- **ostinato**
packet crafter/traffic generator
<http://code.google.com/p/ostinato/>

THC IPv6 tool

THE TOOLS

=====

The THC IPV6 ATTACK TOOLKIT comes already with lots of effective attacking tools:

- parasite6: ICMPv6 neighbor solicitation/advertisement spoofer, puts you as man-in-the-middle, same as ARP mitm (and parasite)
- alive6: an effective alive scanning, which will detect all systems listening to this address
- dnsdict6: parallized DNS IPv6 dictionary bruteforcer
- fake_router6: announce yourself as a router on the network, with the highest priority
- redir6: redirect traffic to you intelligently (man-in-the-middle) with a clever ICMPv6 redirect spoofer
- toobig6: mtu decreaser with the same intelligence as redir6
- detect-new-ip6: detect new IPv6 devices which join the network, you can run a script to automatically scan these systems etc.
- dos-new-ip6: detect new IPv6 devices and tell them that their chosen IP collides on the network (DOS).
- trace6: very fast traceroute6 with supports ICMP6 echo request and TCP-SYN
- flood_router6: flood a target with random router advertisements
- flood_advertise6: flood a target with random neighbor advertisements
- fuzz_ip6: fuzzer for IPv6
- implementation6: performs various implementation checks on IPv6
- implementation6d: listen daemon for implementation6 to check behind a FW
- fake_mld6: announce yourself in a multicast group of your choice on the net
- fake_mld26: same but for MLDv2
- fake_mldrouter6: fake MLD router messages
- fake_mip6: steal a mobile IP to yours if IPSEC is not needed for authentication
- fake_advertiser6: announce yourself on the network
- smurf6: local smurfer
- rsmurf6: remote smurfer, known to work only against linux at the moment
- exploit6: known IPv6 vulnerabilities to test against a target
- denial6: a collection of denial-of-service tests againsts a target
- thcping6: sends a hand crafted ping6 packet
- sendpees6: a tool by willdamn@gmail.com, which generates a neighbor solicitation requests with a lot of CGAs (crypto stuff ;-)) to keep the CPU busy. nice.

and about 25 more tools for you to discover :-)

Just run the tools without options and they will give you help and show the command line options.

THC-Tool firewall6

```
root@blubberli:/home/trilobit/software/thc/thc-ipv6-2.5# ./firewall6 eth0 fd42:caff:ee42:: 80
Starting firewall6: mode TCP against fd42:caff:ee42::1 port 80
Run a sniffer behind the firewall to see what passes through
```

```
Test 1: plain sending          TCP-SYN-ACK received
Test 2: plain sending with data TCP-SYN-ACK received
Test 3: IPv4 ethernet type     FAILED - no reply
Test 4: hop-by-hop hdr (ignore option) FAILED - no reply
Test 5: dst hdr (ignore option) FAILED - no reply
Test 6: hop-by-hop hdr router alert FAILED - no reply
Test 7: 3x dst hdr (ignore option) FAILED - no reply
Test 8: 130x dst hdr (ignore option) FAILED - no reply
Test 9: atomic fragment       FAILED - no reply
Test 10: 2x atomic fragment (same id) FAILED - no reply
Test 11: 2x atomic fragment (diff id) FAILED - no reply
Test 12: 3x atomic fragment (same id) FAILED - no reply
Test 13: 3x atomic fragment (diff id) FAILED - no reply
Test 14: 130x atomic fragment (same id) FAILED - no reply
Test 15: 130x atomic fragment (diff id) FAILED - no reply
Test 16: 260x atomic fragment (same id) FAILED - no reply
Test 17: 260x atomic fragment (diff id) FAILED - no reply
Test 18: 2kb dst hdr          FAILED - no reply
Test 19: 2kb dst + dst hdr    FAILED - no reply
Test 20: 32x 2kb dst hdr      FAILED - no reply
```

More than 38 different tests.
Unclear, what each test exactly
does (you must look at the
code)

Flooding IPv6 advertise

```
root@ipv6-craft:/home/trilobit/software/thc/thc-ipv6-2.1# ./flood_advertise6 bond0
Starting to flood network with neighbor advertisements on bond0 (Press Control-C to
end, a dot is printed for every 100 packet):
```

```
.....
.....
.....
.....^C
```

MX240 LOG (Active)

```
Jan  9 14:06:14 lab-zb0303-rt-mx240-2-re0 l2ald[1549]: L2ALD_MAC_LIMIT_REACHED_IFBD:
Limit on learned MAC addresses reached for ae2.10\__VPLS-VLAN-10-LDP__ flags [0x
6b] state [0x      0]; current count is 1024
```

```
Jan  9 14:06:18 lab-zb0303-rt-mx240-2-re0 jddosd[1570]: DDOS_PROTOCOL_VIOLATION_SET:
Protocol MLP:packets is violated at fpc 1 for 1 times, started at 2012-11-20
21:03:31 CET, last seen at 2012-11-20 21:03:31 CET
```

```
Jan  9 14:06:23 lab-zb0303-rt-mx240-2-re0 jddosd[1570]: DDOS_PROTOCOL_VIOLATION_SET:
Protocol NDPv6:aggregate is violated at fpc 1 for 1 times, started at 2012-11-20
21:03:31 CET, last seen at 2012-11-20 21:03:31 CET
```

```
Jan  9 14:06:56 lab-zb0303-rt-mx240-2-re0 l2ald[1549]: L2ALD_MAC_LIMIT_RESET_IF:
Resumed adding MAC addresses learned by ae2.10\__VPLS-VLAN-10-LDP__ flags [0x 6b]
state [0x      0]; current count is 1023
```

After 5 Min

```
Jan  9 14:11:18 lab-zb0303-rt-mx240-2-re0 jddosd[1570]: DDOS_PROTOCOL_VIOLATION_CLEAR:
Protocol MLP:packets has returned to normal. Violated at fpc 1 for 1 times, from
2012-11-20 21:03:31 CET to 2012-11-20 21:03:31 CET
```

```
Jan  9 14:11:23 lab-zb0303-rt-mx240-2-re0 jddosd[1570]: DDOS_PROTOCOL_VIOLATION_CLEAR:
Protocol NDPv6:aggregate has returned to normal. Violated at fpc 1 for 1 times,
from 2012-11-20 21:03:31 CET to 2012-11-20 21:03:31 CET
```

Solicitatie flooding

```
root@ipv6-craft:/home/trilobit/software/thc/thc-ipv6-2.1#  
  fake_solicitatie6 bond0 3ffe:10:1:10::1  
Starting solicitation of 3ffe:10:1:10::1 (Press Control-C to end)  
^C
```

Target Device:

```
root@lab-zb0303-rt-mx240-2-re0> show ipv6 neighbors
```

IPv6 Address	Linklayer Address	State	Exp Rtr	Secure	Interface
3ffe:10:10:14::1	00:10:db:ff:10:01	stale	1182	yes no	ae3.3010
3ffe:10:10:114::1	00:10:db:ff:10:01	delay	0	yes no	ae3.3011
3ffe:10:11:14::1	00:10:db:ff:10:01	reachable	0	yes no	ae3.3020
3ffe:10:11:114::1	00:10:db:ff:10:01	stale	1194	yes no	ae3.3021
fe80::211:22ff:fe33:4455	00:11:22:33:44:55	stale	424	no no	ae2.10
fe80::211:22ff:fe33:4488	00:11:22:33:44:88	stale	1198	no no	lsi.1048823
fe80::218:ff:fe00:b0ec	00:18:00:00:b0:ec	stale	1094	no no	ae2.10
fe80::218:ff:fe01:c660	00:18:00:01:c6:60	stale	1072	no no	ae2.10
fe80::218:ff:fe03:7859	00:18:00:03:78:59	stale	1095	no no	ae2.10
fe80::218:ff:fe04:6255	00:18:00:04:62:55	stale	1095	no no	ae2.10
fe80::218:ff:fe04:691b	00:18:00:04:69:1b	stale	1074	no no	ae2.10
fe80::218:ff:fe04:74a3	00:18:00:04:74:a3	stale	1073	no no	ae2.10
fe80::218:ff:fe06:982f	00:18:00:06:98:2f	stale	1094	no no	ae2.10

.

```
<------8<----->
```

Neighbor flooding

- **System CPU nearly 100%**

```
root@lab-zb0303-rt-mx240-2-re0> show system processes summary
last pid: 33300;  load averages:  1.39,  1.88,  1.40  up 55+03:15:28   15:10:43
148 processes: 3 running, 130 sleeping, 15 waiting
Mem: 662M Active, 92M Inact, 267M Wired, 951M Cache, 214M Buf, 5169M Free
Swap: 8192M Total, 8192M Free
  PID USERNAME   THR PRI NICE   SIZE   RES STATE   TIME  WCPU COMMAND
 1134 root          1  132   0  4928K  3112K RUN      9:36 92.94% eventd
   11 root          1  171  52     0K    16K RUN    1271.7 0.73% idle
```

- **Logfile of the MX**

Log Entries on the MX240

```
Jan  9 15:15:14 lab-zb0303-rt-mx240-2-re0 /kernel: Nexthop index allocation
failed: regular index space exhausted
Jan  9 15:15:14 lab-zb0303-rt-mx240-2-re0 fpc1 tsec_receive: .le1, failed to
allocate packet buffer
Jan  9 15:15:15 lab-zb0303-rt-mx240-2-re0 last message repeated 10 times
```

Neighbor Flooding

- Routing Entries

```
{master}[edit]
root@lab-zb0303-rt-mx240-2-re0# run show ipv6 neighbors | count
Count: 523053 lines
```

```
root@lab-zb0303-rt-mx240-2-re0> show route forwarding-table vpn L3VPN1 summary
Routing table: L3VPN1.inet
Internet:
    user:          2 routes
    perm:          5 routes
    intf:          8 routes
    dest:          13 routes
Routing table: L3VPN1.iso
ISO:
    perm:          1 routes
Routing table: L3VPN1.inet6
Internet6:
    user:          3 routes
    perm:          4 routes
    intf:          26 routes
    dest:          523056 routes
{master}
```

Neighbor flooding

Jan 12 14:48:01 lab-zb0305-rt-mx80-1-re0 rpd[1278]: bgp_hold_timeout:3967: NOTIFICATION sent to 10.100.100.4 (Internal AS 65000): code 4 (Hold Timer Expired Error), Reason: holdtime expired for 10.100.100.4 (Internal AS 65000), socket buffer sndcc: 91 rcvcc: 0 TCP state: 4, snd_una: 3569673048 snd_nxt: 3569673139 snd_wnd: 16384 rcv_nxt: 3958755043 rcv_adv: 3958771427, hold timer out 90s, hold timer remain 0s

Jan 12 14:48:55 lab-zb0305-rt-mx80-1-re0 rpd[1278]: bgp_pp_timeout: peer 3ffe:10:11:116::1+52922 (proto) timed out waiting for OPEN

Jan 12 14:48:55 lab-zb0305-rt-mx80-1-re0 rpd[1278]: bgp_pp_timeout:5572: NOTIFICATION sent to 3ffe:10:11:116::1+52922 (proto): code 4 (Hold Timer Expired Error), socket buffer sndcc: 0 rcvcc: 0 TCP state: 4, snd_una: 2890450339 snd_nxt: 2890450339 snd_wnd: 16384 rcv_nxt: 2714040868 rcv_adv: 2714057252

Jan 12 14:48:56 lab-zb0305-rt-mx80-1-re0 rpd[1278]: bgp_hold_timeout:3967: NOTIFICATION sent to 10.100.100.1 (Internal AS 65000): code 4 (Hold Timer Expired Error), Reason: holdtime expired for 10.100.100.1 (Internal AS 65000), socket buffer sndcc: 91 rcvcc: 0 TCP state: 4, snd_una: 1101029588 snd_nxt: 1101029679 snd_wnd: 16384 rcv_nxt: 3350882185 rcv_adv: 3350898569, hold timer out 90s, hold timer remain 0s

Jan 12 14:49:00 lab-zb0305-rt-mx80-1-re0 jddosd[1361]: DDOS_PROTOCOL_VIOLATION_CLEAR: Protocol MLP:packets has returned to normal. Violated at fpc 0 for 3 times, from 2013-01-12 14:44:00 CET to 2013-01-12 14:44:00 CET

Jan 12 14:49:01 lab-zb0305-rt-mx80-1-re0 rpd[1278]: bgp_hold_timeout:3967: NOTIFICATION sent to 10.10.116.1 (External AS 65001): code 4 (Hold Timer Expired Error), Reason: holdtime expired for 10.10.116.1 (External AS 65001), socket buffer sndcc: 162 rcvcc: 0 TCP state: 4, snd_una: 2798335501 snd_nxt: 2798335644 snd_wnd: 16384 rcv_nxt: 477463463 rcv_adv: 477479847, hold timer out 90s, hold timer remain 0s

Jan 12 14:49:01 lab-zb0305-rt-mx80-1-re0 bfdd[1259]: BFDD_TRAP_SHOP_STATE_DOWN: local discriminator: 25, new state: down, interface: irb.3021, peer addr: 10.10.116.1

Jan 12 14:53:25 lab-zb0305-rt-mx80-1-re0 eventd[1068]: SYSTEM_ABNORMAL_SHUTDOWN: System abnormally shut down

Jan 12 14:53:25 lab-zb0305-rt-mx80-1-re0 eventd[1068]: SYSTEM_OPERATIONAL: System is operational

Jan 12 14:53:25 lab-zb0305-rt-mx80-1-re0 savecore: writing core to vmcore.1

Jan 12 14:53:25 lab-zb0305-rt-mx80-1-re0 /kernel: platform_early_bootinit: MX-PPC Series Early Boot Initialization

Jan 12 14:53:25 lab-zb0305-rt-mx80-1-re0 /kernel: mxppc_set_re_type: hw.board.type is MX80

Jan 12 14:53:25 lab-zb0305-rt-mx80-1-re0 /kernel: mxppc_set_re_type: REtype:78, model:mx80, model:MX80, i2cid:2447

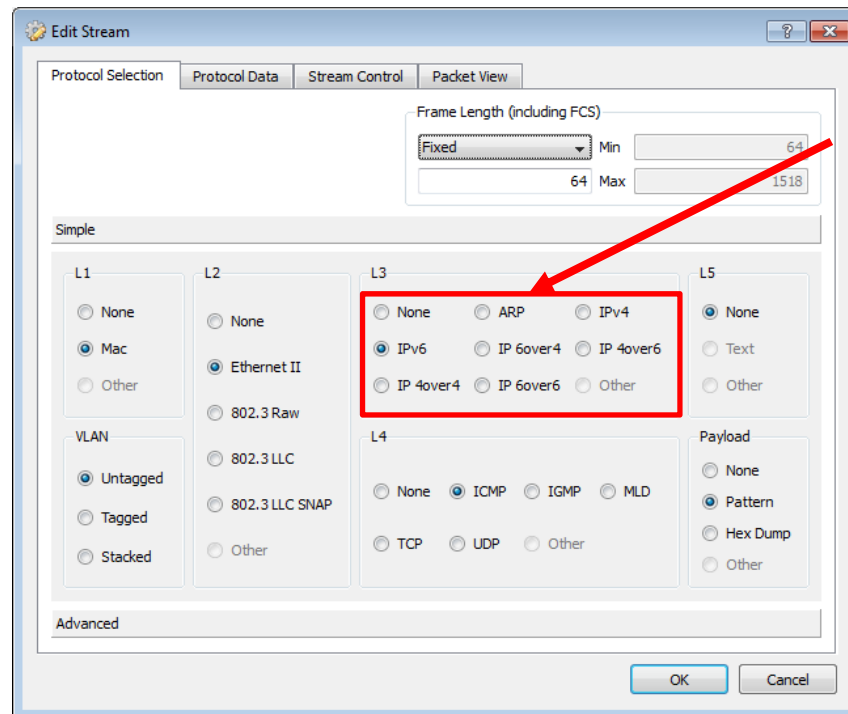
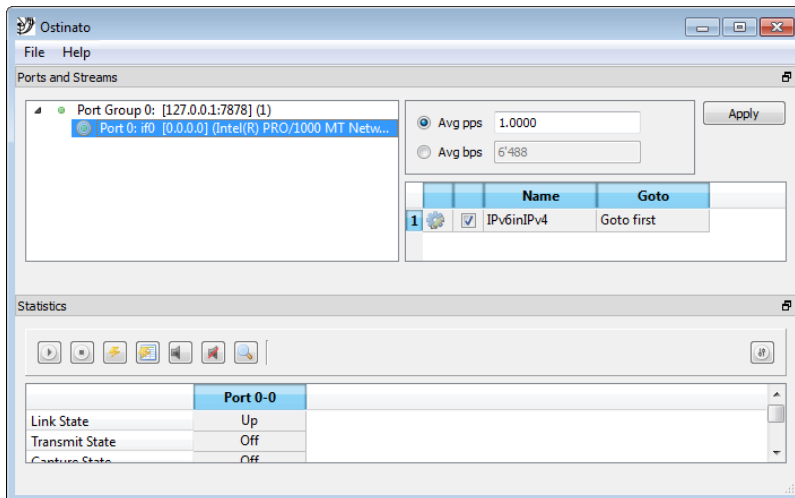
Jan 12 14:53:25 lab-zb0305-rt-mx80-1-re0 /kernel: WDOG initialized

Jan 12 14:53:25 lab-zb0305-rt-mx80-1-re0 /kernel: Copyright (c) 1996-2012, Juniper Networks, Inc.

Jan 12 14:53:25 lab-zb0305-rt-mx80-1-re0 /kernel: All rights reserved.

ostinato

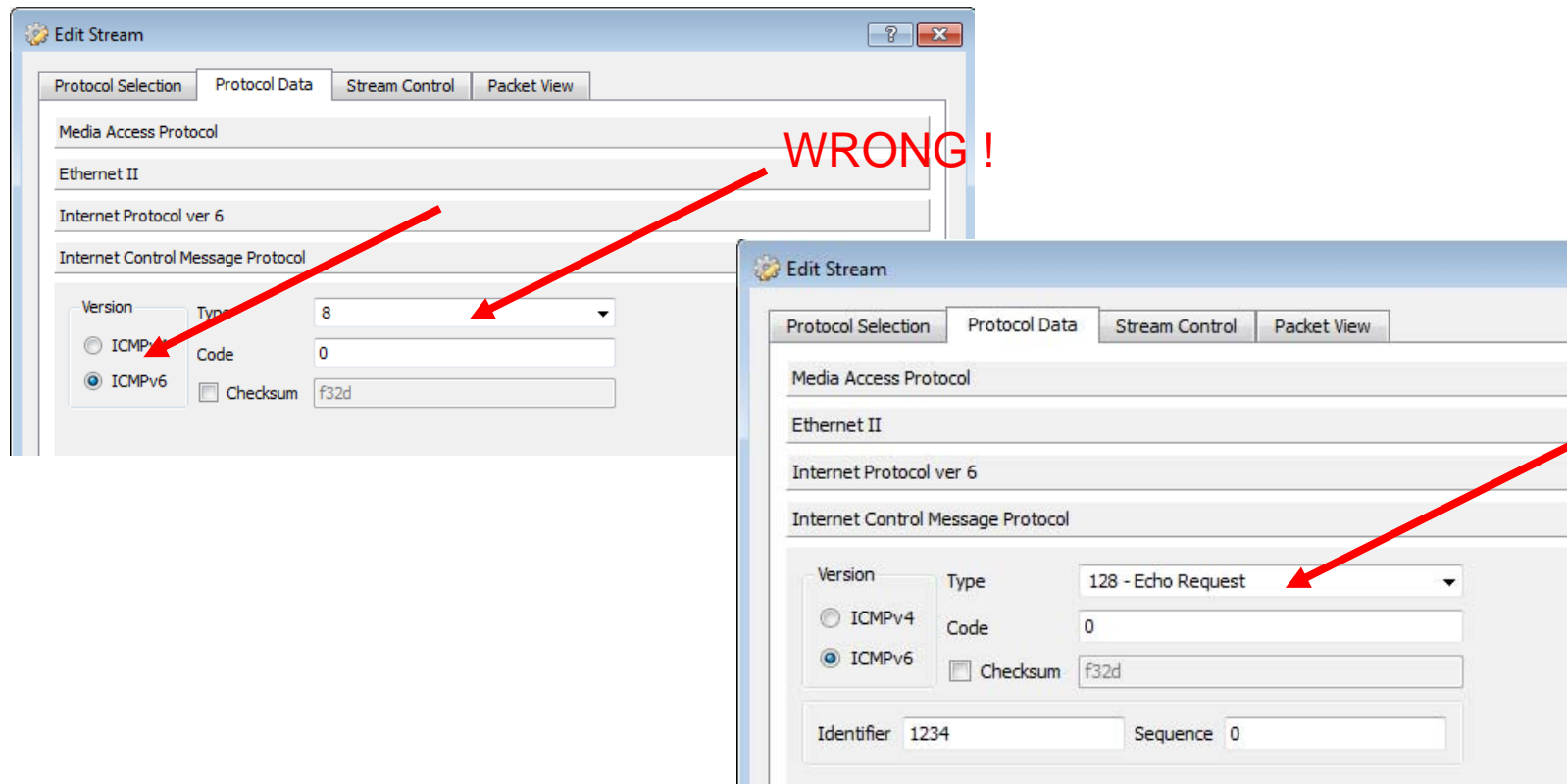
- Open source tool packet crafter/traffic generator (Windows/Linux/OS X/BSD)



ostinato

Sample: IPv6/ICMP

be careful with default values



IPv6 Security devices now and in the future

- They support IPv6
 - > but on different levels
- Vendors are working on it.
 - > Some very hard, others.....
- Request features
 - > ask for new implementations
- Interact with the vendors
 - > tell him your ideas
- Find and know the limits of your security device !






questions ?



christoph.weber@swisscom.com

Tools

**Security warning and disclaimer:
using this tools it's maybe against your local law or company policy !**

Function	Tools
Scanning/Surveillance:	halfscan6, nmap, Scan6, Strobe 
Covert Channel/Backdoor:	relay6, 6tunnel, nt6tunnel, netcat6, VoodooNet, etc.
Port Bouncing:	relay6, nt6tunnel, ncat, and asybo
Denial of Service (DOS):	6tunneldos, 6To4DDos, Imps6-tools
Packet-Level attack toolkits:	isic6, spak6, THC-6, IPv6-Tools 
Packet-Crafting:	scapy, sendIP, Packit, Spack, OSTINATO 
IRC Zombies/Bots:	Eggdrop, Supybot, etc.
Sniffer:	snort, tcpdump, snoop, wireshark, tshark etc. 
Firewall Testing	ft6
Pen Testing Tool:	Metasploit 

terminology

- **Node:** Device that implements IPv6
- **Router:** Node that forwards IPv6 Packets
- **Host:** Any Node, that isn't a router
- **Upper Layer:** Protocol layer above ipv6
- **Link:** Medium or communication Facility over which nodes can communicate at the link layer
- **Neighbors:** Nodes attached on the same link
- **Interface:** A Node's attachment to a link
- **Address:** IPv6 Layer identification for an interface
- **Packet:** IPv6 header + payload
- **Link MTU:** Link Maximum Transmission Unit
- **Path MTU:** Maximum link MTU of all links in a path between source and destination node's