the hacker's choice

presents:

Attacking the IPv6 Protocol Suite

van Hauser, THC

vh@thc.org

http://www.thc.org

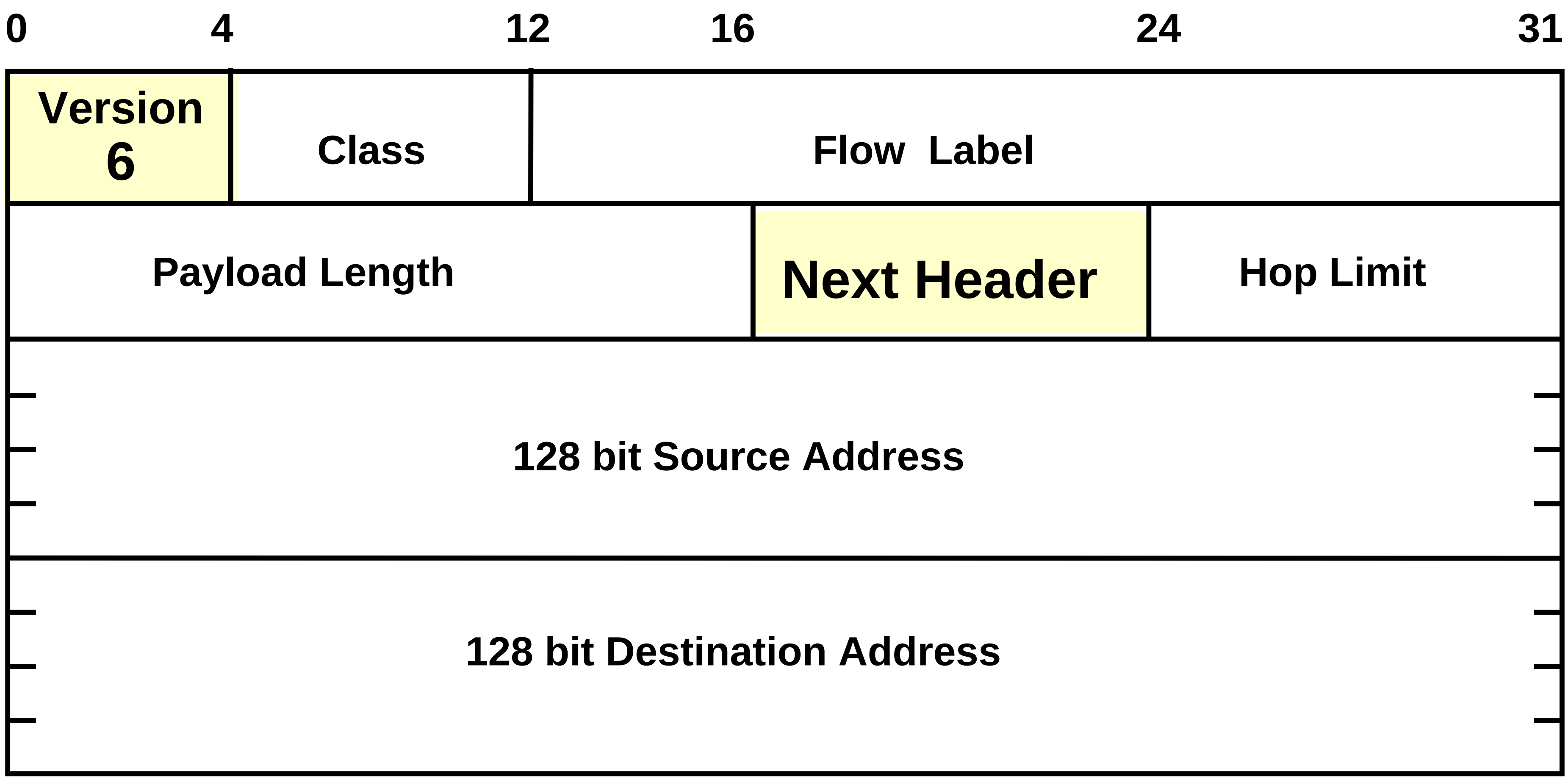
Contents

1. Very fast and short Introduction to IPv6
2. The all new THC IPV6 Attack Suite
3. Security relevant changes in IPv4<>IPv6 and Security Vulnerabilities in IPv6
4. Implementation Vulnerabilities in IPv6 so far
5. New Research & Future

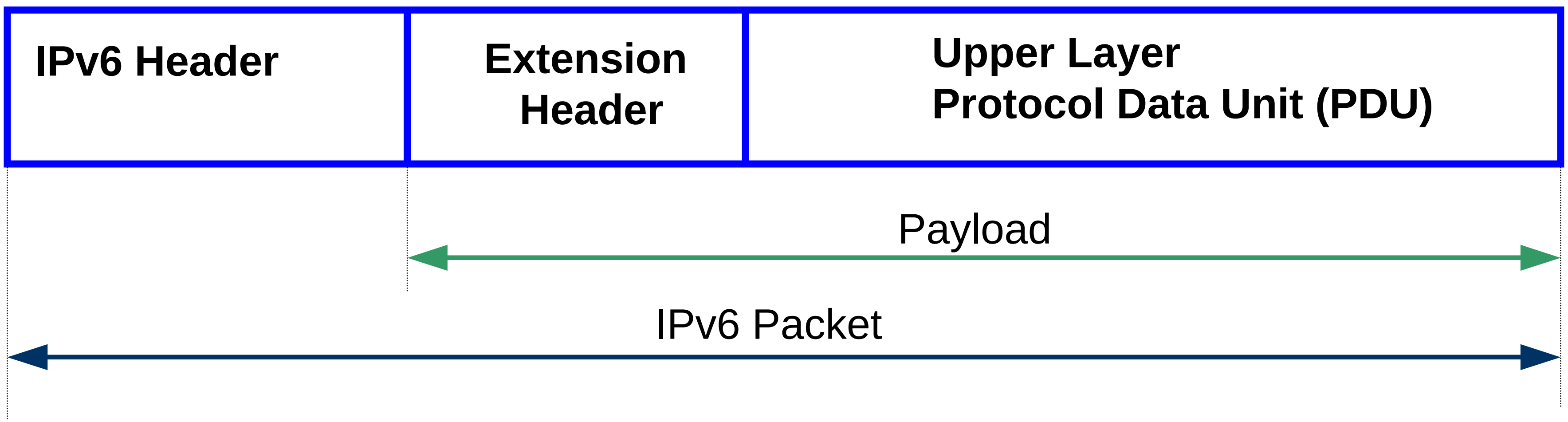
Very short and fast Introduction to IPv6

- Goals of IPv6:
  - Enough IP addresses for the next decades
    - 2^{128}=340.282.366.920.938.463.463.374.607.431.768.211.456
  - Autoconfiguration of IP addresses and networking
  - Hierarchical address structure
    - Reduces operational costs
  - Integrated security features

IPv6 Header Structure



IPv6 Layer Structure



IPv6 Header ≡ **40 Bytes**

Upper Layer PDU ≤ 65535 Bytes

Upper Layer PDU > 65535 Bytes = Jumbo Payload

IPv6 Header Structure

Examples for Extension Headers: Hop-by-Hop = 0; UDP = 17; Encapsulated Header = 41; RSVP = 46; IPSEC (Encapsulating Security Payload = 50; Authentication Header = 51;) ICMPv6 = 58; No Next Header = 59; Destination Options = 60; OSPFv3 = 98

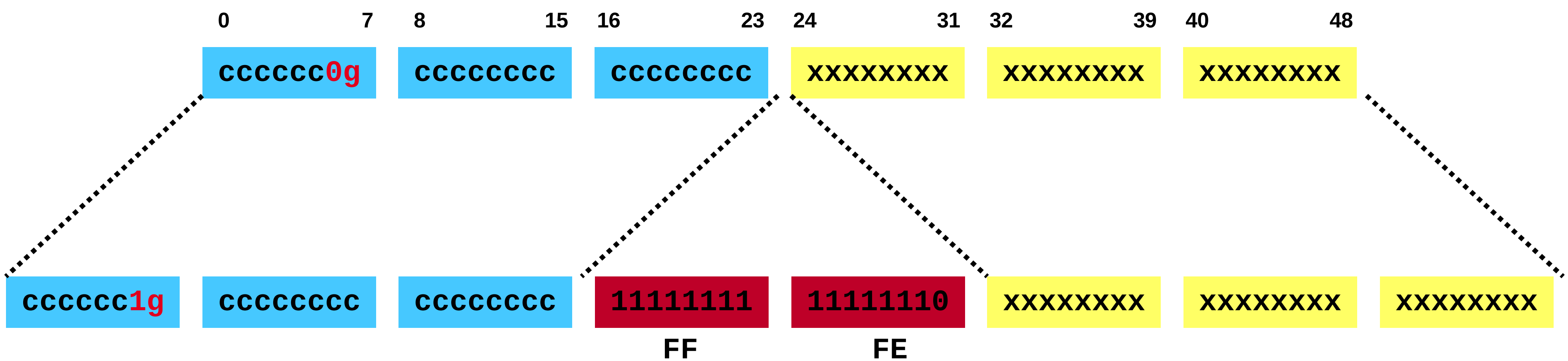
IPv6 Header **Next Header = 6**	**TCP Header**	**Application Data**

IPv6 Header **Next Header = 43**	**Routing Header** **Next Header = 6**	**TCP Header**	**Application Data**

IPv6 Header **Next Header = 43**	**Routing Header** **Next Header = 44**	**Fragment Header** **Next Header = 6**	**TCP Header**	**Data Fragment**

IPv6 Interface Identifier (EUI-64 Format) Mapping

IEEE 802 MAC Adresse



IPv6 Interface Identifier im EUI-64 Format

EUI: Extended Unique Identifier

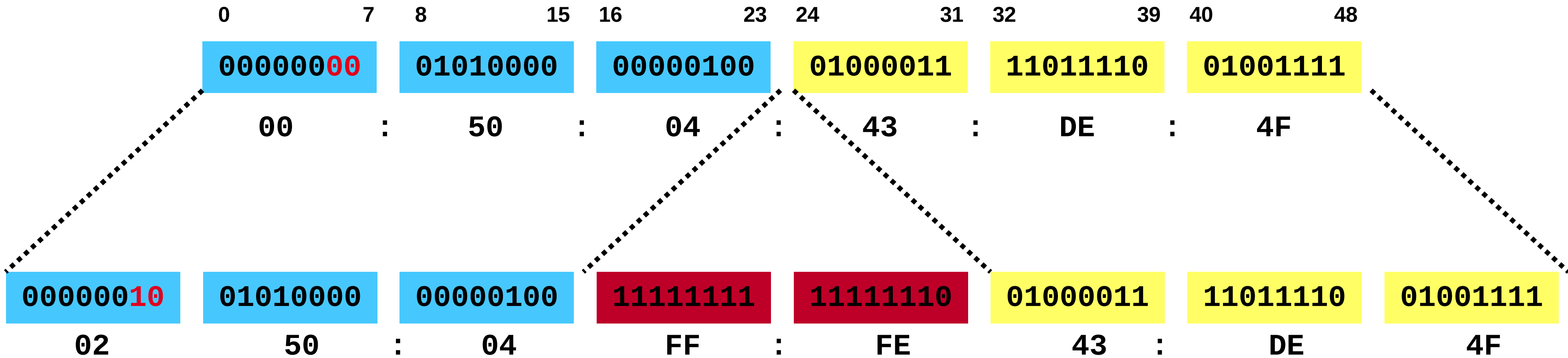
c = company id

x = extension identifier

g = Individual/Group (G): 0 - unicast   1 - multicast

Example

```
# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:50:04:43:DE:4F
          inet addr:10.2.1.1  Bcast:10.2.1.255  Mask:255.255.255.0
          inet6 addr: 3ffe:ffff:100:f101:250:4ff:fe43:de4f/64 Scope:Global
          inet6 addr: fe80::250:4ff:fe43:de4f/64 Scope:Link
          ...
```



Blackhat usage of IPv6 today

Backdoor deployment

- Enable IPv6 (6to4)
- Run Backdoor on IPv6
- No chance to detect by port scanning
- Hard to analyze if backdoor traffic is detected

Inter-Communication

- Establishing of IPv6 interconnections (via 6to4) for warez exchange, IRC and bouncing

Availability of Hacker Tools so far …

The following Hacker tools exist:

- Port Scanning: nmap, halfscan6, …
- Port Bouncers: relay6, 6tunnel, nt6tunnel, asybo, …
- Denial-of-Service (connection flooding): 6tunneldos
- Packet fun: isic6, libnet (partially implemented only)

No IPv6 specific attack tools exist so far!

This will change when IPv6 deployment is wider

… but you do not want to wait, right?

The THC IPV6 Attack Suite

- THC has developed an easy-to-use IPv6 packet factory library
- Numerous IPv6 protocol exploits tools can be coded in just 5-10 lines
- Lots of powerful protocol exploits already included

- Caveat of current code state:
  - Linux 2.6.x only
  - Little Endian
  - 32-Bit
  - Ethernet

The THC IPV6 Attack Suite – The Tools

- PARSITE6
  - ICMP Neighbor Spoofer for Man-In-The-Middle attacks
- DOS-NEW-IPV6
  - Denial any new IPv6 system access on the LAN (DAD Spoofing)
- REDIR6
  - Redirect traffic to your system on a LAN
- FAKE_ROUTER6
  - Fake a router, implant routes, become the default router, …
- SMURF6
  - Local Smurf Tool (attack you own LAN)
- RSMURF6
  - Remote Smurf Tool (attack a remote LAN)

The THC IPV6 Attack Suite – The Tools

- TOOBIG6
  - Reduce the MTU of a target
- Alive6
  - Find all local IPv6 systems, checks for aliveness of remote systems
- Protocol Implementation Tester:
  - Fragmentation + Routing Header
  - Mass Headers
  - Invalid Pointers
  - …

The THC IPV6 Attack Suite – Implementation Example

- Implementation is simple!
- Sending an ICMP6 Echo Request:
  - pkt = thc_create_ipv6(interface, PREFER_GLOBAL, &pkt_len, *src6*, ***target6***, 0, 0, 0, 0, 0);
  - thc_add_icmp6(pkt, &pkt_len, ICMP6_PINGREQUEST, 0, 0xdeadbeef, NULL, 0, 0);
  - thc_generate_and_send_pkt(interface, NULL, NULL, pkt, &pkt_len);
- Target6 will answer with an ICMP6 Echo Reply

Security relevant changes from IPv4 to IPv6

- Executive Summary:
  - IPv6 and IPv4 security is quite similar
  - Basic mechanisms are the same
  - Application layers are unaffected
  - IPv6 includes IPSec but currently not used
  - IPSec would not prevent attacks on application level in Internet applications

Overview of security relevant changes

1. Protocol Changes
2. Reconnaissance
3. Local Attacks: ARP, DHCP
4. Smurfing (Traffic Amplification)
5. Routing & Fragmentation Attacks
6. IPv4 and IPv6 coexistence

1. Protocol Changes

- A few IP header content and options were removed:
  - No IP ID field
    - Nice uptime check not possible anymore ☹
  - No IP Record Route Option
    - No traceroute alternative anymore ☹
- No Broadcast addresses exist
- Multicast addresses can not be destined from remote
  - This is a big problem for remote alive scanning!

2. Reconnaissance IPv4

Network size in a subnet usually 2^8 = 256

Usual attack methodology:

1. Ping sweeps to a target remote class C (takes 5-30 seconds)
2. Port scans to an alive host
3. Vulnerability test to active ports

Wide range of tools available

- Nmap
- Amap
- Nessus
- ...

2. Reconnaissance IPv6 (1/2)

Network size increased to 2^64 *(varies)* in a subnet

- 18.446.744.073.709.551.616 possible hosts in a subnet
- Ping sweeps will consume too much time
  - Brute force: ***500 millions years***
  - Being clever + technology advances: still some months
- Public servers need to be in the public DNS
- All hosts need to be in a private DNS for admin purposes

>> DNS Servers will become primary <<
>> sources of information – and primary targets <<

2. Reconnaissance IPv6 (2/2)

- New opportunities are standardized multicast addresses to identify key servers within the local Network (routers, DHCP, Time, etc.)
- Local multicasts will ensure that one compromised host can find all other hosts in a subnet
- Techniques to a single host remain the same (port scan, attacking active ports, exploitation, etc.)
- Remote alive scans (ping scans) on networks will become impossible

2. Reconnaissance with the THC-IPV6 Attack Toolkit

- alive6 – for local/remote unicast targets, and local multicast addresses
  - Sends three different type of packets:
    - ICMP6 Echo Request
    - IP6 packet with unknown header
    - IP6 packet with unknown hop-by-hop option
    - IP6 fragment (first fragment)
  - One-shot fragementation + routing header option:
    - Sends all packets in one fragments and a routing header for a router in the target network
    - Will only work if the target router allows routing header entries to multicast addresses – requires bad implementation!

3. DHCP IPv4

- DHCP uses broadcast messages
- Rouge device can respond instead of a legal one
- Feed the host with new DNS and routing information in order to perform “Man in the middle” Attacks

3. ARP IPv4

- ARP uses layer 2 broadcast to perform the IP > MAC lookup on the local network
- Attackers will respond in order to perform “Man in the middle” Attacks

3. ARP/DHCP IPv6

- No security added to both protocol variations
- ICMPv6 Stateless auto configuration = DHCP light
- ICMP6 Neighbor Discovery and Neighbor Solicitation = ARP replacement
- Duplicate Address Detection based on NS allows DoS against a host by responding to requests

3. ICMPv6 Stateless Auto-Configuration

1. RS:
ICMP Type = 133
Src = ::
Dst = FF02::2
query= please send RA

***fake_router6*:**
Sets any IP as default router ☺

2. RA:
ICMP Type = 134
Src = Router Link-local Address
Dst = FF02::1
Data= options, prefix, lifetime, *autoconfig* flag

Routers send periodic as well as soliticated Router Advertisements (RA) to the all-nodes multicast address FF02::1

Clients configure their routing tables and network prefix from advertisements. => Like a DHCP-light in IPv4

But anyone can send Router Advertisements! => *fake_router6*

3. ICMPv6 Neighbor Discovery

1. NS:
ICMP Type = 135
Src = **A**
Dst = All-Nodes Mulitcast Address
query= Who-has IP **B**?

***parasite6*:**
Answer to every NS, claim to be every system on the LAN ☺

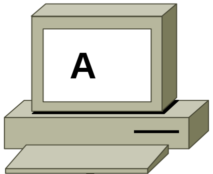
2. NA:
ICMP Type = 136
Src = **B**
Dst = **A**
Data= Link Layer Address

If A needs the MAC address of B, it sends an ICMP6 Neighbor Discovery to the All-Nodes multicast address

B sees the request and responds to A with its MAC address => Like ARP in IPv4

But everybody can respond to the request… => *parasite6*

3. ICMPv6 Duplicate Address Detection (DAD)



1. ND

1. NS:
ICMP Type = 135
Src = :: (unspecified)
Dst = All-Nodes Mulitcast Address
query= Who-has IP **A**?

***dos-new-ipv6*:**
Answer to every NS, claim to be every system on the LAN ☺

2.
No reply if nobody owns the IP address.

If A sets a new IP address, it makes the Duplicate Address Detection check, to see if anybody owns the address already.

Anybody can respond to the DAD checks… => *dos-new-ipv6* prevents new systems on the LAN

4. Smurf IPv4

- Sending a packet to a broadcast address with spoofed source will force response to on single target, e.g. with ICMP echo request/reply
- Traffic amplification
- DoS for target link

4. Smurf IPv6

- No broadcast addresses
- Replaced with various multicast addresses
- RFC 2463 states that no ICMP response should be sent when destination was a multicast address. However, exceptions are made.
  - Cisco Security Research got it all wrong ☺
- Exploitable?
  - Locally: YES!
  - Remote: Depends on Implementation of Routing Headers, Fragmentation etc.

4. Smurfing IPv6 with the THC-IPV6 Attack Toolkit

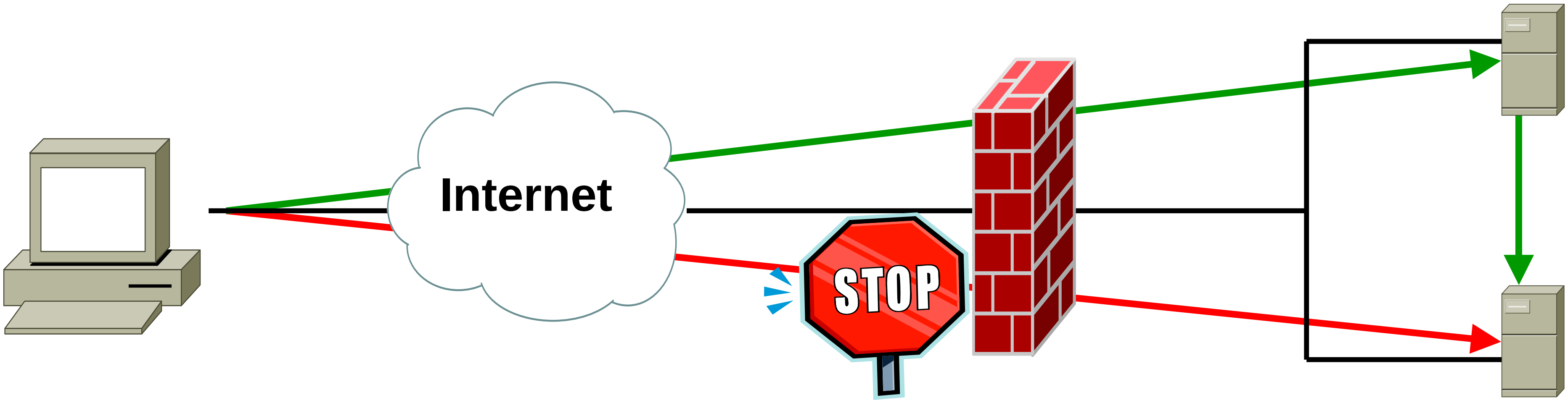
- smurf6 – for local initiated smurfs
  - Source is target, destination is local multicast address
  - Generates lots of local traffic that is sent to source
- rsmurf6 – reverse smurf, exploits mis-implementations (e.g. Linux)
  - Source is all-nodes multicast address (*255.255.255.255 in IPv6 speak*), destination is target
  - If target has mis-implemented IPv6 (e.g. linux), it responds with Echo Reply to the all-nodes multicast address, generating lots of traffic
  - In the local LAN, 1 packet in a network with 100 Linux servers generated 10000 processed packets altogether!

5. Routing Protocols

- Most Routing protocols provide their own security mechanisms
- This does not change with IPv6
- With the exception of OSPFv3, which has ***no*** security properties and relies on IPSEC usage

5. Routing Header Manipulation

Routing header attack (like IPv4 Source Routing)

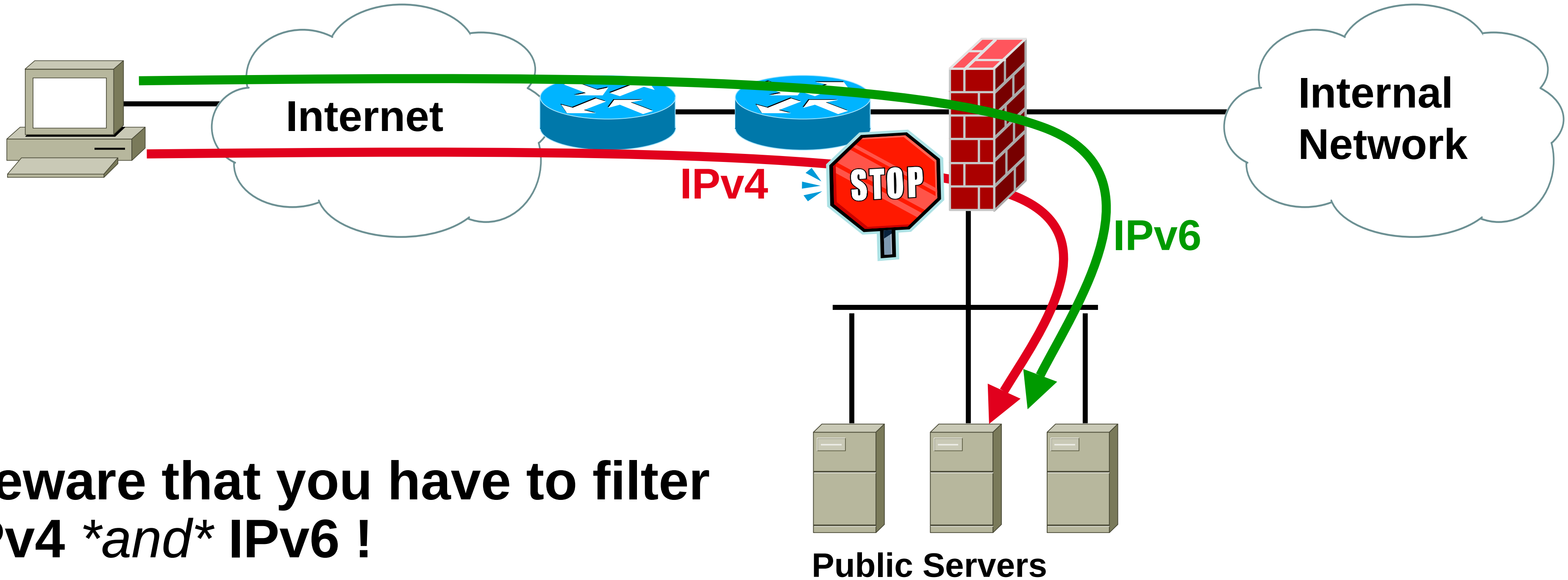


Use alive6-remote for checking if routing headers are allowed to target

5. Fragmentation

- Fragmentation is performed by source, not routers; reassembling performed by destination only
- Routers in path will not be able to drop packets with routing header if fragmentation comes first and routing header afterwards, after reassembling.

6. Dual stack attack



Beware that you have to filter IPv4 *and* IPv6 !

Implementation Vulnerabilities in IPv6 so far

- Python 03/2004 (when compiled without IPv6)
  - Crash when sending DNS replies with IPv6 addresses
- Ethereal 03/2004
  - Parsing bug, remote exploitable
- Apache 09/2004
  - URI parsing bug, remote crash, maybe exploitable
- Exim (MTA) 01/2005
  - Buffer overflow, local privileges escalation
- Cisco IOS 01/2005
  - Remote crash when receiving several malformed packets
- Postfix 02/2005
  - Allows spamming if a IPv6 config file is not present
- Linux Kernel 02/2005
  - Length validation bug, remote crash, maybe exploitable

Research and Implementation Tests

Tested: Linux 2.6, Windows XP SP2, Cisco IOS 12, FreeBSD 5.3

1. **Responding to packets to multicast destinations (Echo Request)**
   - Vulnerable: Linux, FreeBSD
2. **Responding to packets to multicast destinations (Invalid Header Options)**
   - Vulnerable: ***ALL***
3. **Responding to packets from multicast address sources**
   - Vulnerable: Linux
4. **Routing header to multicast address**
   - Vulnerable: none
5. **Fragmentation and following Routing Header**
   - Vulnerable: ***ALL***
6. **One-Shot Fragmentation**
   - Vulnerable: ***ALL***

Upcoming IPv6 Security Research from THC

- Multicast Fun
  - Global Multicast FF:0E exploitation
  - MLD v3 research
- IPv4 <> IPv6 co-existance solutions
  - Security weaknesses in Tunneling

Upcoming IPv6 Threats and Chances

1. **Specific attack tool development for IPv6**
   - No special difference to existing IPv4 attack tools
2. **Worms**
   - TCP/IP Worms (e.g. Slammer types) will die out
   - E-Mail Worms will stay
   - Messenger and P2P Worms will come
3. **DNS Server will become primary targets**
4. **Attacks will move to attack Clients from compromised servers in a LAN**
5. **When IPSEC is widely deployed, certificate stealing will be primary security concern**

Conclusion Internet Security with IPv6

So far no known new risks with IPv6, but some security improvements against IPv4:

- Alive-Scanning and TCP/IP Worming very hard
- IP Record Route Option removed, no uptime check
- Easier network filtering and attack tracing

Introduction of IPSEC will not make IPv6 secure, but will make attack tracing easy, and sniffing + Man-in-the-Middle very difficult

Some implications unclear yet, research needed

Questions?

?

Have fun!

Thank you very much!

Download Location:

http://www.thc.org/members/vh/thc-ipv6-0.2.tar.gz