
# “IPv6 Security ”
# 6NET, Zagreb, May ‘03

**Eric Marin**

**EMEA Senior Consulting Engineer**

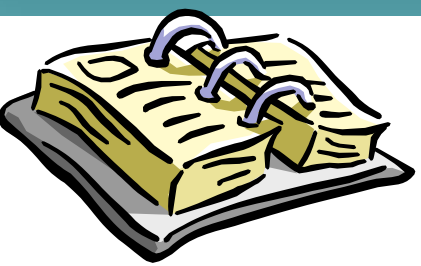
**Emarin@cisco.com**

# « But, we have IPsec for securing IPv6 !»

**Heard many times !**

# Topics





- Are some IPv4 security issues resolved with IPv6 ?
- Filtering IPv6
- Fragmentation
- Conclusion

# IPv6 Security



- All implementations required to support authentication and encryption headers (AH and ESP of IPsec)
- Authentication separate from encryption for use in situations where encryption is prohibited or prohibitively expensive
- Key distribution protocols are under development (independent of IP v4/v6)
- Support for manual key configuration required

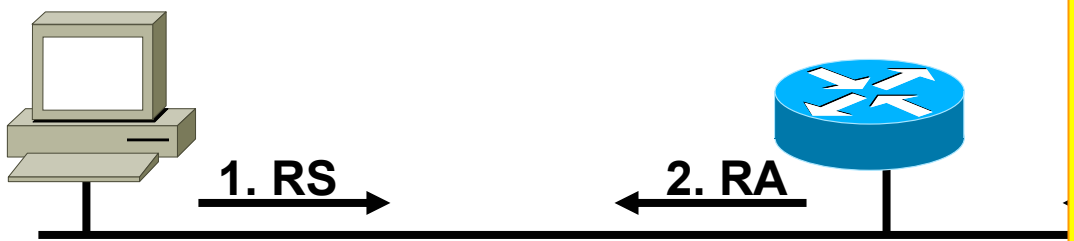
# IPv6 Security Exposures…



- **Autoconfiguration**
  - *– stateless configuration and discovery, contradicting requirements with security*
- **ICMPv6 protected by IPsec**
  - *– security bootstrap problem*
- **DAD**
  - *– duplicate address detection mechanism*

# Stateless autoconfiguration






**1. RS:**

- ICMP Type = 133
- Src = ::
- Dst = All-Routers multicast Address
- query= please send RA

**2. RA:**

- ICMP Type = 134
- Src = Router Link
- Dst = All-nodes multicast address
- Data= options, prefix, lifetime, autoconfig flag

**ICMP w/o IPsec AH⇔ gives exactly same level of security as ARP for IPv4 (none)**

**Bootstrap security problem!**

**Potential solution: 802.1X on L2.**

***Router solicitation are sent by booting nodes to request RAs for configuring the interfaces.***

# Neighbor Discovery - Neighbor Solicitation




ICMP type = 135
Src = A
Dst = Solicited-node multicast of B
Data = link-layer address of A
Query = what is your link address?

ICMP type = 136
Src = B
Dst = A
Data = link-layer address of B

A and B can now exchange packets on this link

**Security mechanisms built into discovery protocol ⇔ None.**

**Bootstrap security problem!**

**Potential solution: 802.1X on L2.**

# DAD (Duplicate Address Detection)




ICMP type = 135
Src = 0 (::)
Dst = Solicited-node multicast of *A*
Data = link-layer address of A
Query = what is your link address?

**From RFC 2462:**

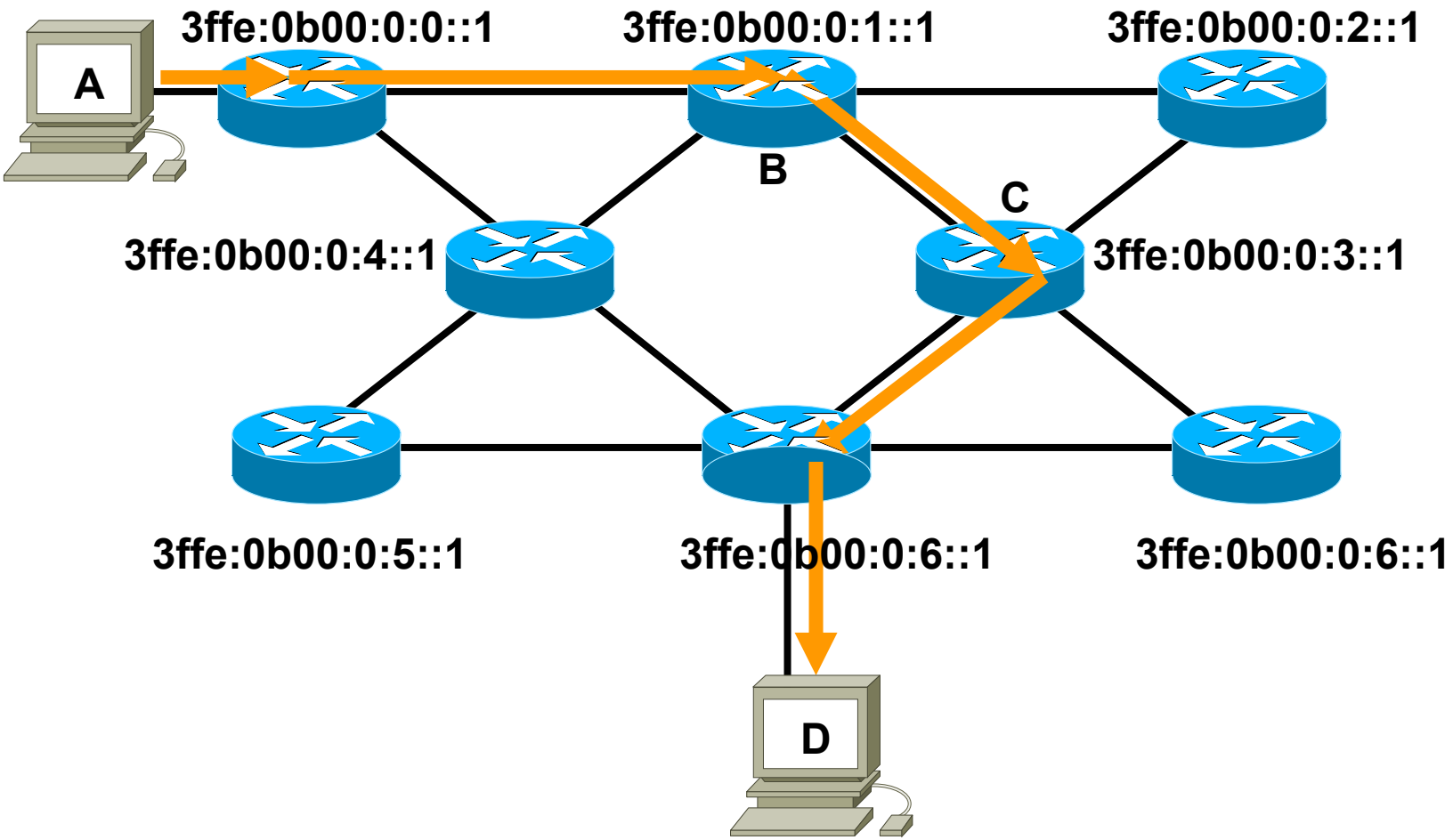
**« If a duplicate @ is discovered … the address *cannot* be assigned to the interface…»**

**⇔ What if: Use MAC@ of the node you want to DoS and fabricate its IPv6 @**

- **Duplicate Address Detection (DAD) uses neighbor solicitation to verify the existence of an address to be configured.**

# IPv6 Routing Header like Loose Source Routing ?

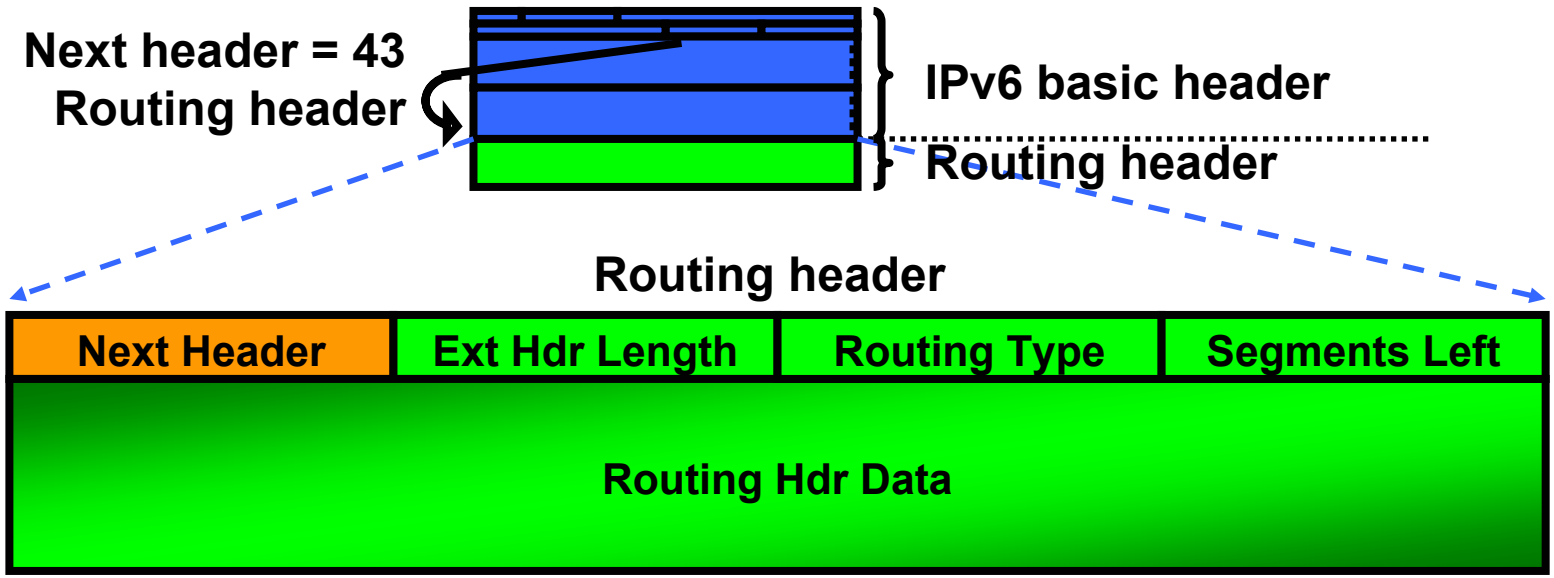





- **Routing type 0: Routers list = 3ffe:0b00:0:1::1, 3ffe:0b00:0:3::1**

# IPv6 Routing Header





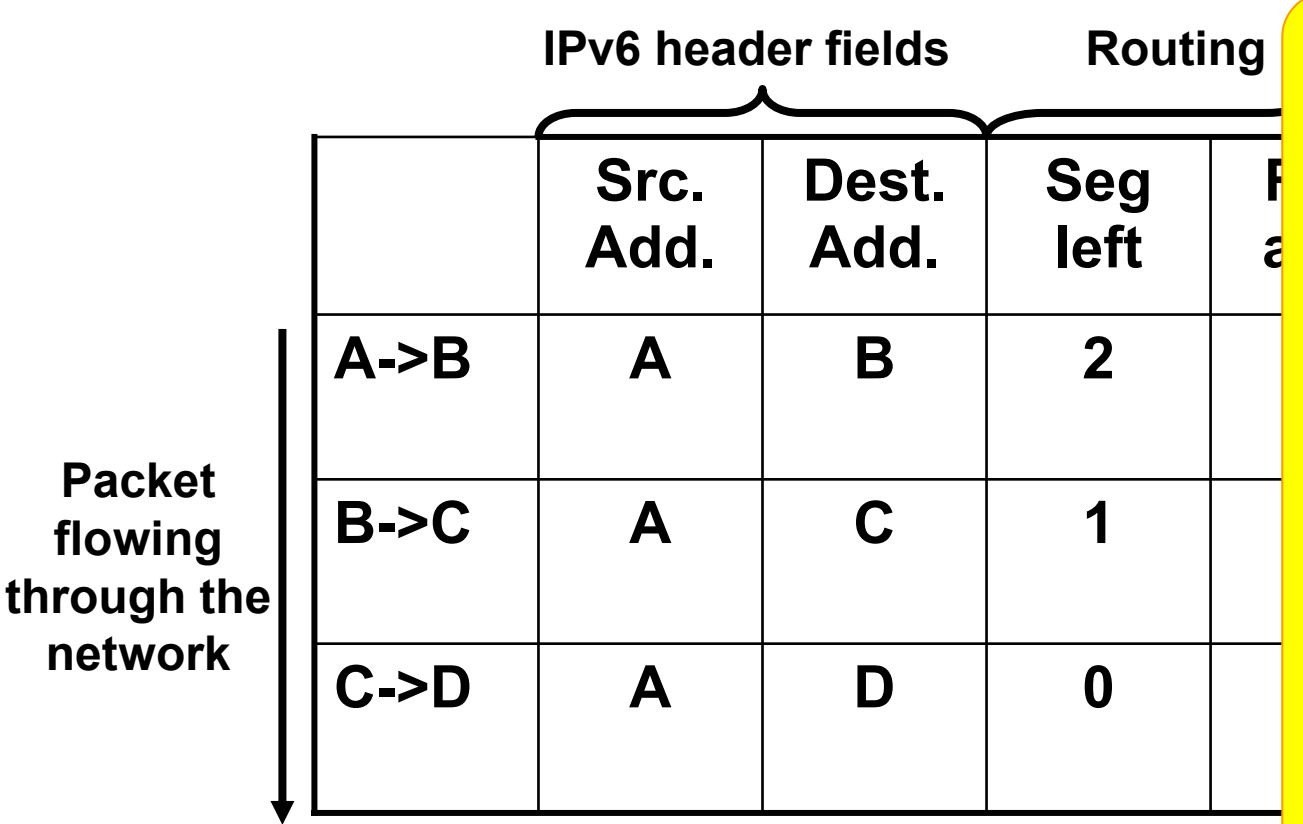

- Routing header is:

  An extension header.

  Processed by the listed intermediate routers.

# IPv6 Routing Header (cont.)






**Routing header IPv6 ⇔ Source routing in IPv4**

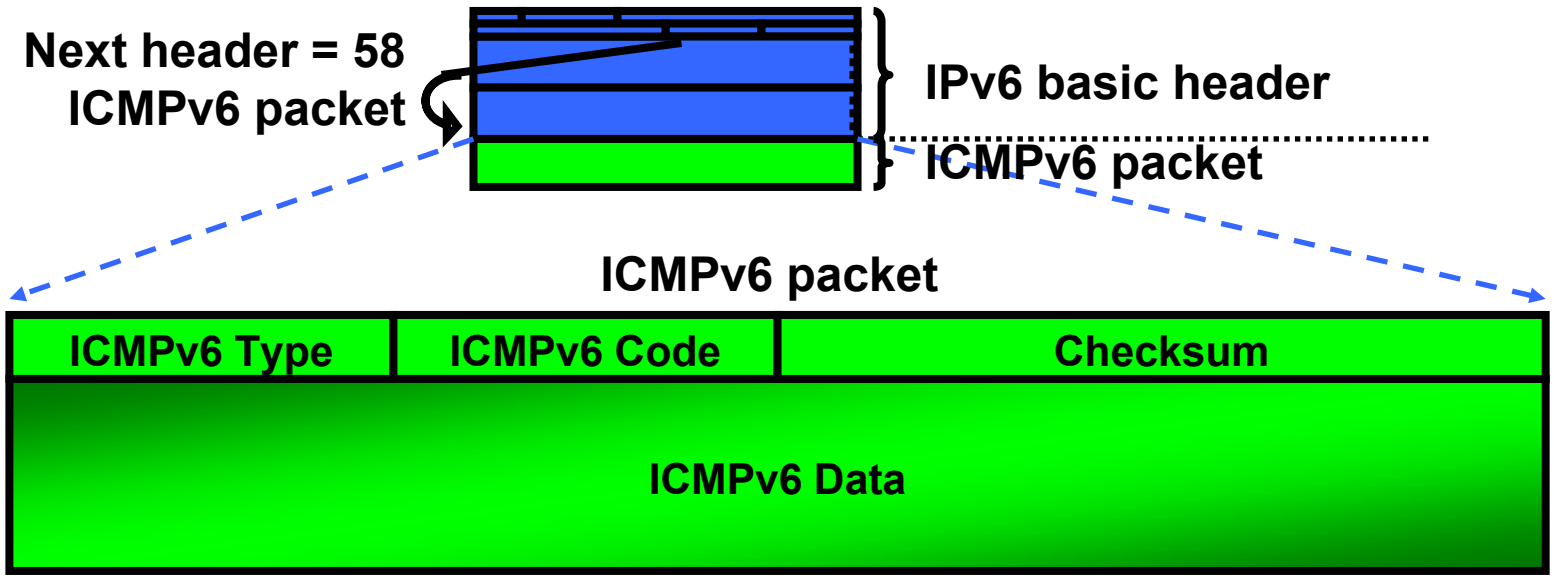
**« Cannot be turned off (like** ’no ip source-route’ **in IPv4) cause it is REQUIRED for mobile IPv6 !»**

**Solution: Use extended ACL (if mobile IPv6 not required)**

draft-savola-ipv6-rh-ha-security-03.txt

# ICMPv6






- **ICMPv6 is similar to IPv4:**
  - **Provides diagnostic and error messages**
  - **Is used for path MTU discovery**
  - **Runs *on top* of IPv6!**
  - **ARP-like security !**

# Renumbering




Router Advertisment (RA) relay sole on IPsec AH security…

RA packet definitions:

ICMP Type = 134

Src = Router Link-local Address

Dst = All-nodes multicast address

Data= 2 prefixes:

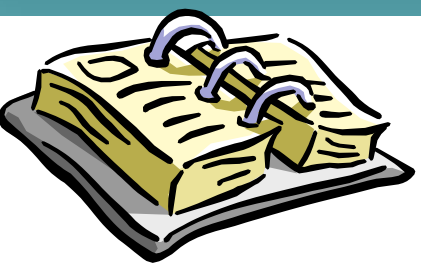
Current prefix (to be deprecated) with short lifetime

New prefix (to be used) with normal lifetime

- Renumbering is done by modifying the RA to announce the old prefix with a short lifetime and the new prefix.

# Topics





- IPv6 short introduction
- Are some IPv4 security issues resolved with IPv6 ?
- Filtering IPv6
- Fragmentation
- Conclusion

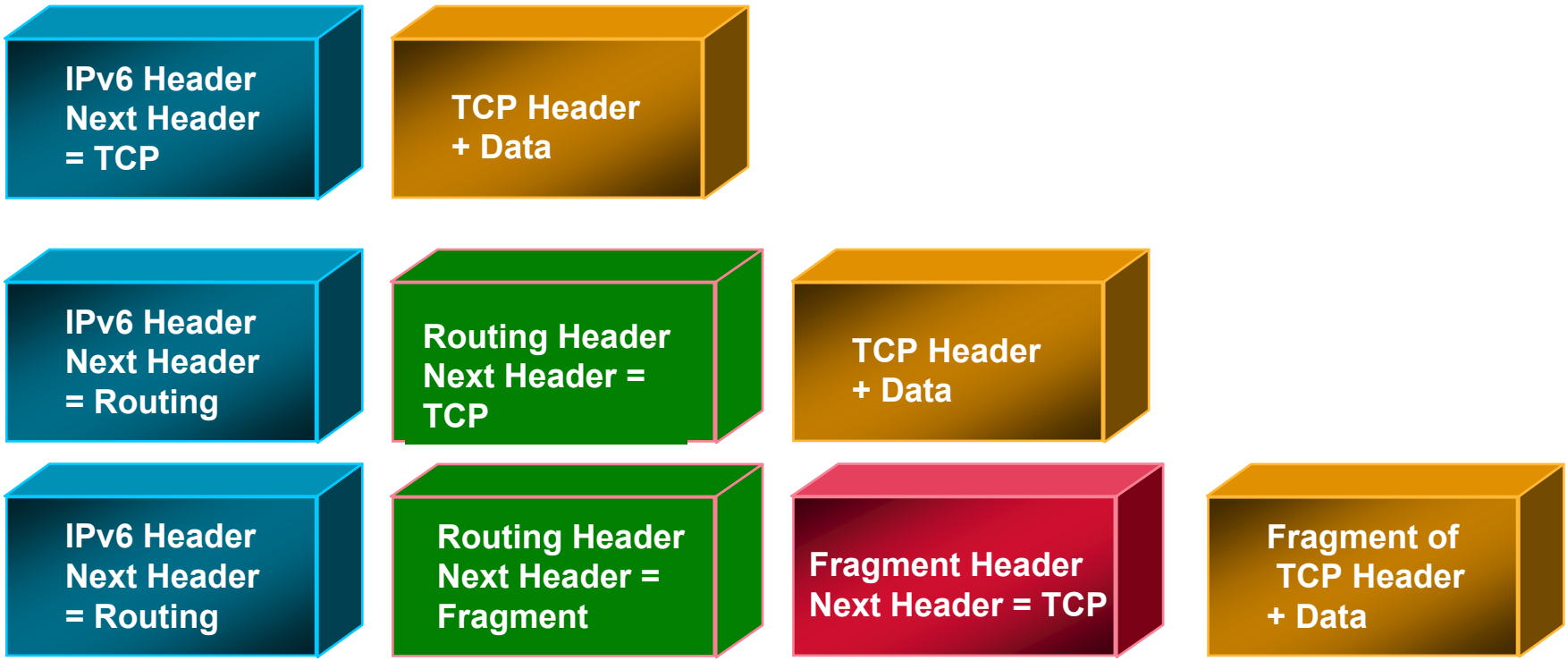
# Cisco IOS IPv6 ACL



- **IPv6 Access Control Lists**
  - **12.2(2)T Simple ACL, ONLY matching src and dest**
  - **12.2(8)T Extended ACL support**
- **IPv6 and IPv4 ACL functionality**
  - **Implicit deny any any as final rule in each ACL.**
  - **A reference to an empty ACL will permit any any.**
  - **ACLs are NEVER applied to self-originated traffic.**

# IPv6 Header Options (RFC 2460)






- **Processed only by node identified in IPv6 Destination Address field => much lower overhead than IPv4 options**
  - **exception: Hop-by-Hop Options header**
- **Eliminated IPv4's 40-octet limit on options**
  - **in IPv6, limit is total packet size, or Path MTU in some cases**

# Filtering Extension Headers



- IPv6 headers and optional extensions need to be scanned to access the upper layer protocols (UPL)
- May require searching through several extensions headers
  - Routing
  - AH (no special handling)
  - ESP (no special handling)
  - Fragmentation
  - Payload compression (no special handling)

# IPv6 Extended Access Control Lists



- Upper Layers : ICMP (next header 58), TCP (6), UDP (17), SCTP (132) – Could filter on any next header value (0-255)
- ICMPv6 code and type
- syn, ack, fin, psh, urg, rst and established (ack && rst)
- L4 port numbers
- Traffic class (only 6 bits/8) = DSCP
- Flow Label (0-0xFFFFF)
- IPv6 header options (Fragments, Routing, ...)

# IPv6 ACL Implicit Rules



- **Implicit permit for enable neighbor discovery**

  **The following implicit rules exist at the end of each IPv6 ACL to allow ICMPv6 neighbour discovery:**

  **permit icmp any any nd-na**
  **permit icmp any any nd-ns**
  **deny ipv6 any any**

# Issues with ACL filtering



- Filtering 2827 becomes difficult
- ACL more difficult to apply and deploy in a consistent manner
- Multiple addresses per node
- Renumbering : it means that for a certain lifetime 2 addresses are coexisting on the node.

# IPv6 ACL Reflexive : Stateful filtering



- **Reflect**

  A reflexive ACL is created dynamically, when traffic matches a permit entry containing the **reflect** keyword.

  The reflexive ACL mirrors the permit entry and times out (by default after 3 mins), unless further traffic matches the entry (or a FIN is detected for TCP traffic).

  Reflexive ACLs can be applied to TCP, UDP, SCTP and ICMPv6.

- **Evaluate**

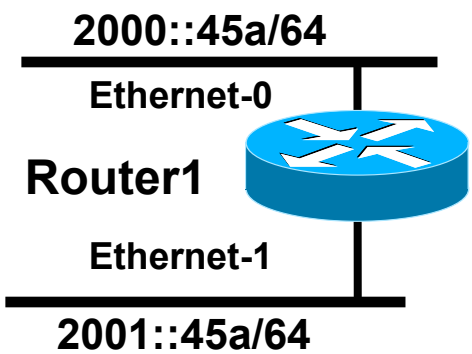
  Apply the packet against a reflexive ACL.

  Multiple evaluate statements are allowed per ACL.

  The implicit deny any any rule does not apply at the end of a reflexive ACL; matching continues after the evaluate in this case.

# IPv6 Reflexive ACL



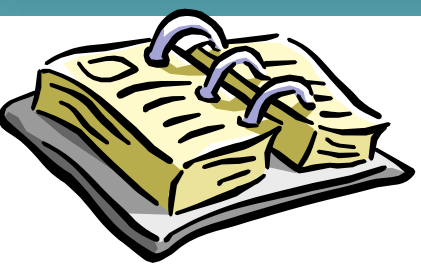



**Allow www traffic via a Reflexive ACL, based on time of day**

```
Router1#
interface ethernet-0
  ipv6 address 2000::45a/64
  ipv6 traffic-filter In in
  ipv6 traffic-filter Out out

interface ethernet-1
  ipv6 address 2001::45a/64

ipv6 access-list In
  permit tcp host 2000::1 eq www host 2001::2 time-range
tim reflect myp
   permit icmp any any router-solicitation

ipv6 access-list Out
   evaluate myp
   evaluate another

time-range tim
  periodic daily 16:00 to 21:00
```

# Topics

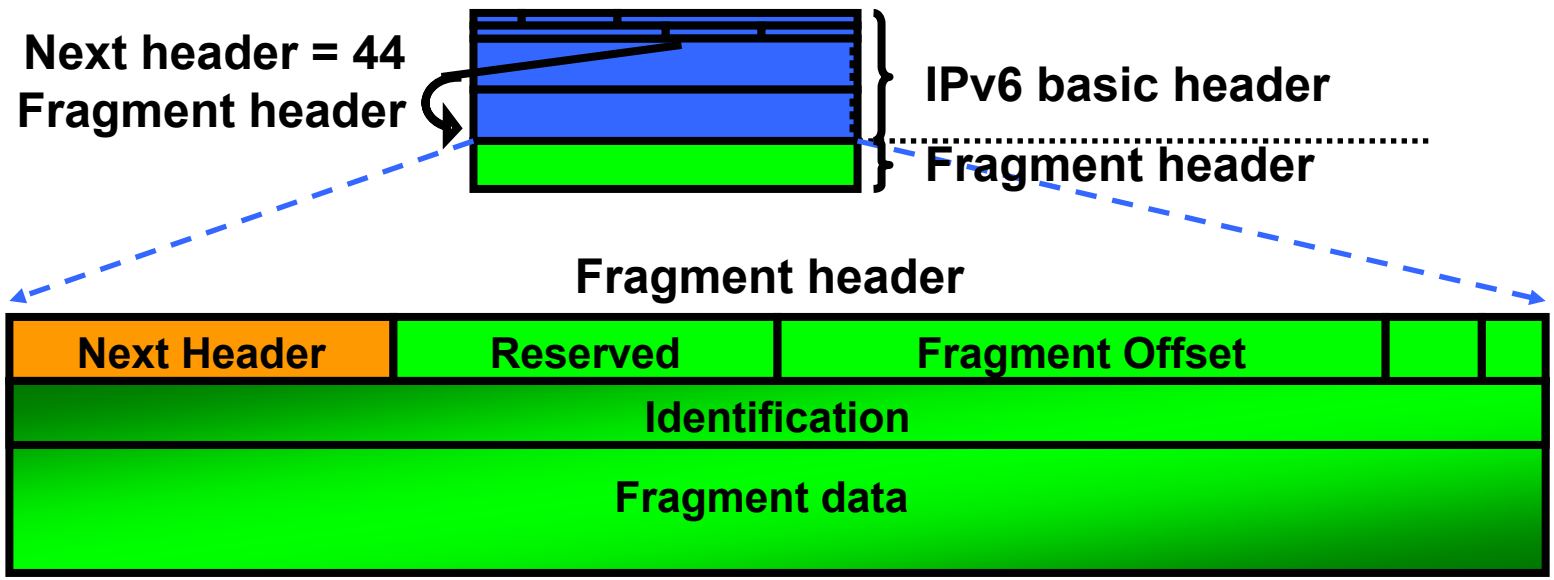




- IPv6 short introduction
- Are some IPv4 security issues resolved with IPv6 ?
- Filtering IPv6
- Fragmentation
- Conclusion

# Fragment Header - IPv6






- In IPv6 fragmentation is done *ONLY* by the end system
- Reassembly done by end system like in IPv4

# Fragmentation handling in IPv4



- In IPv4, you can use the « fragment » keyword for an extended ACL
- The only packets that will match are those that have fragment offset != 0, that is, non-first fragments.
- For IPv4 we know the protocol and fragments flags and offset from the IP header, so we can easily calculate if enough of the ULP is within the first fragment (likely)
- First fragments and non-fragmented packets go through the normal "extract L4 info" process
- Is used against DoS mainly

# Fragmentation issues in IPv6



- For IPv6, we must traverse the Next Headers before reaching the fragment header to extract the flags and offset.
- Then, we may need to traverse further NHs before reaching the ULP and then check if enough of the ULP header is within the first fragment.
- This makes matching against the first fragment non-deterministic : tcp/udp/icmp might not be there.

# « fragment » in IPv6 ACLs

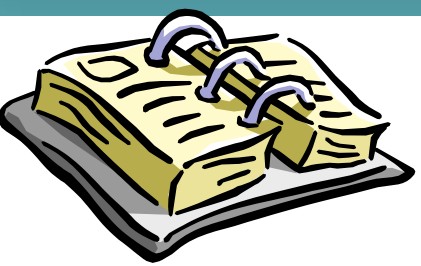


- For IPv6, the « fragment » keyword matches non-initial fragments (same as IPv4) AND the first fragment if the protocol cannot be determined.

Note : IOS also supports a new keyword "undetermined-transport" which matches any ipv6 packet where the layer4 cannot be determined

# Topics

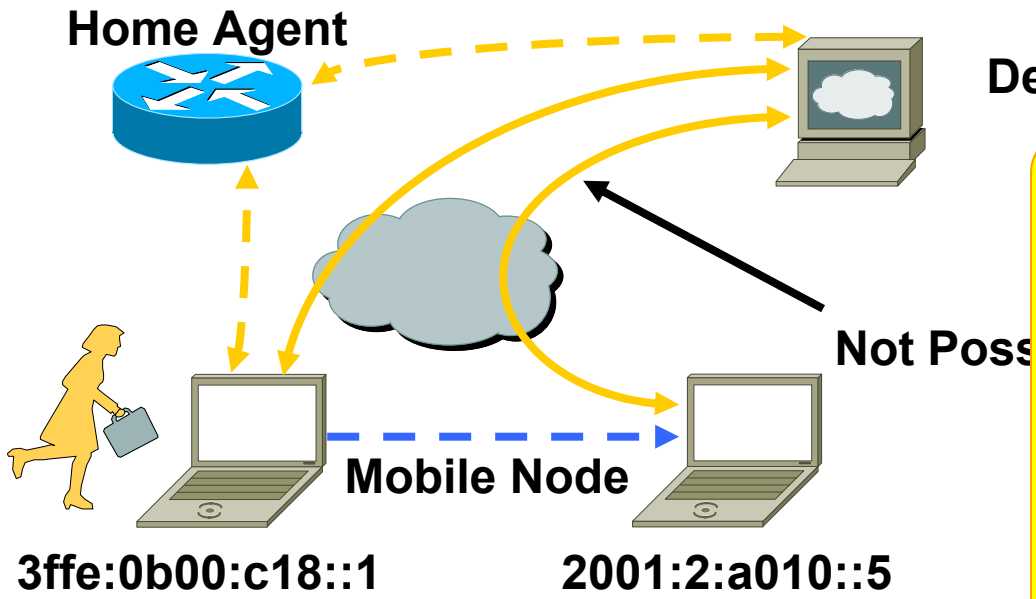




- Are some IPv4 security issues resolved with IPv6 ?
- Filtering IPv6
- Fragmentation
- Conclusion

# IP Mobility - security still work in progress






Mobility and security elements of mobile IPv6 still work in progress… (MIPv6 draft : Return Routability Test).

- **Mobility means:**
  - Mobile devices are fully supported while moving
  - Built-in on IPv6
    - Any node can use it
  - Efficient routing means performance for end-users

# Transition mechanisms  security



**http://www.6net.org/publications/**

**D6.2.2: Operational procedures for secured management with transition mechanisms**

**draft-savola-v6ops-6to4-security-02.txt**

**Processing of 6to4 packets :**

**o Relay Router**

**1. incoming from native, tunneled to 6to4**

**2. tunneled from 6to4, going to nativ**

**o Router**

**1. tunneled from relay, source is native**

**2. tunneled to relay, destination is native**

**3. tunneled directly, destination is 6to4**

**«…. in particular, checks that always match 2002:V4ADDR and V4ADDR must be implemented. »**

- **Anti-spoofing ACLs**
- **Use of IPsec for protecting manually configured tunnels**

# Conclusion



- IPsec is not the answer to every IPv6 security issues
- A new protocol brings new security issues with it
- Mobile IPv6 brings also many security challenges with it .
- Work in progress

# By the Way …
# IPv6 Hacking Tools



- Sniffers/packet capture
  - Snort
  - TCPdump
  - Sun Solaris snoop
  - COLD
  - Ethereal
  - Analyzer
  - Windump
  - WinPcap
  - NetPeek
  - Sniffer Pro
- Worms
  - Slapper


- Scanners
  - IPv6 Security Scanner
  - Halfscan6
  - Nmap
  - Strobe
  - Netcat
- DoS Tools
  - 6tunneldos
  - 4to6ddos
  - Imps6-tools
- Packet forgers
  - SendIP
  - Packit
  - Spak6

# By the Way (cont) …
# IPv6 Security Tools



- **IPTrap**
  - **Listens to ports and fakes services**
  - **Works with IPChains/Tables to Firewall clients**
- **AESOP**
  - **TCP Proxy**

# By the Way (cont) …



- « Recently one of the Honeynet Project's Solaris Honeynets was compromised. What made this attack unique was after breaking into the system, the attackers enabled IPv6 tunneling on the system, with communications being forwarded to another country. The attack and communications were captured using Snort, however the data could not be decoded due to the IPv6 tunneling. Also, once tunneled, this could potentialy disable/bypass the capabilities of some IDS systems. »

**Lance Spitzner**

http://www.securityfocus.com/archive/119/303782/2002-12-15/2002-12-21/0

# Questions?



# Thank you!



## “IPv6 Security ”

**Eric Marin**

**EMEA Senior Consulting Engineer**


CISCO SYSTEMS