
# IPv6 Security – problems and mitigations

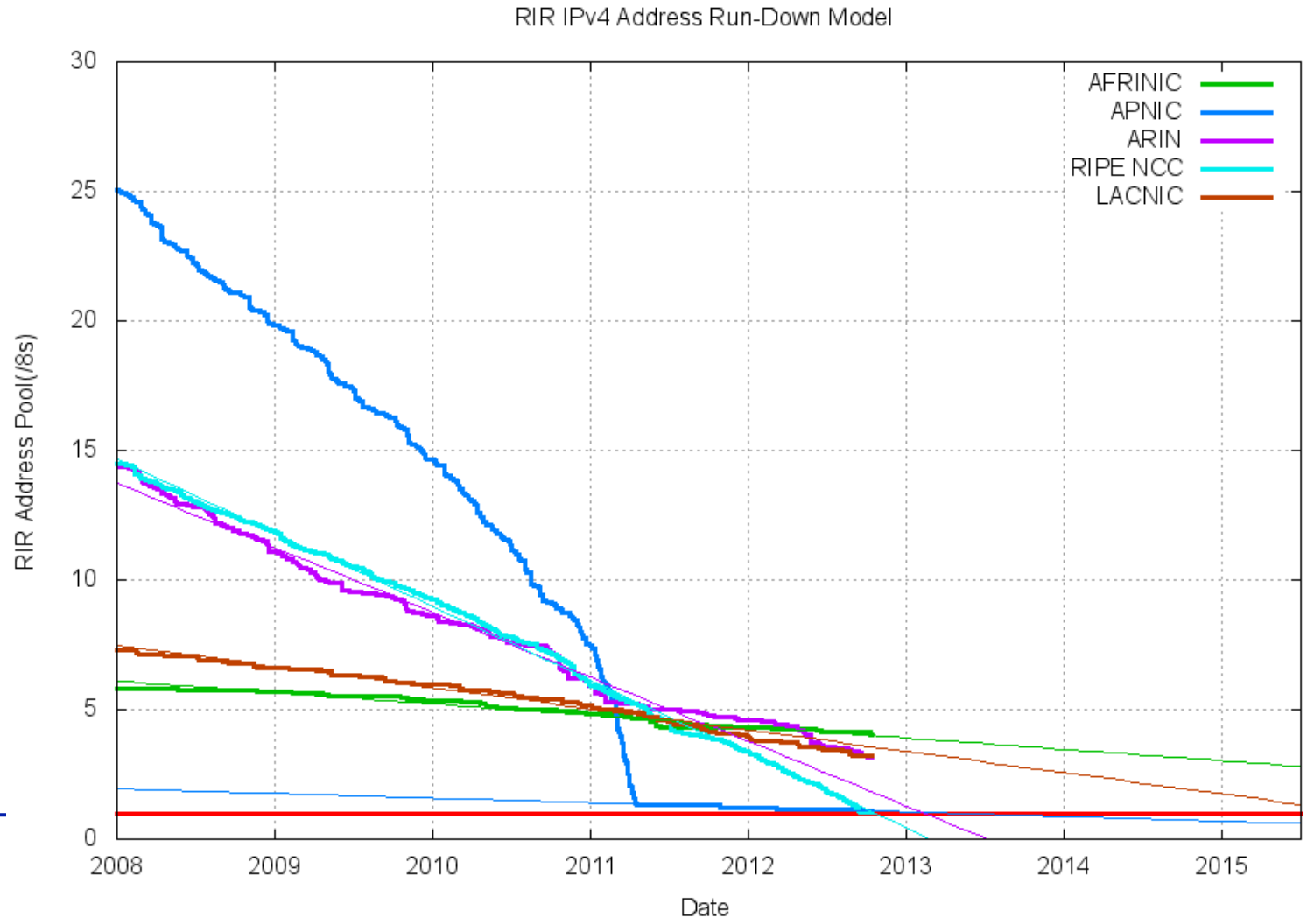
**János Mohácsi (mohacsi@niif.hu)**

**Hbone Workshop 2012**

# Agenda

- Motivation
- Comparison of IPv4 and IPv6
- Vulnerabilities and possible mitigation in IPv6
- Summary

# IP run out – Geoff Huston- October 2012

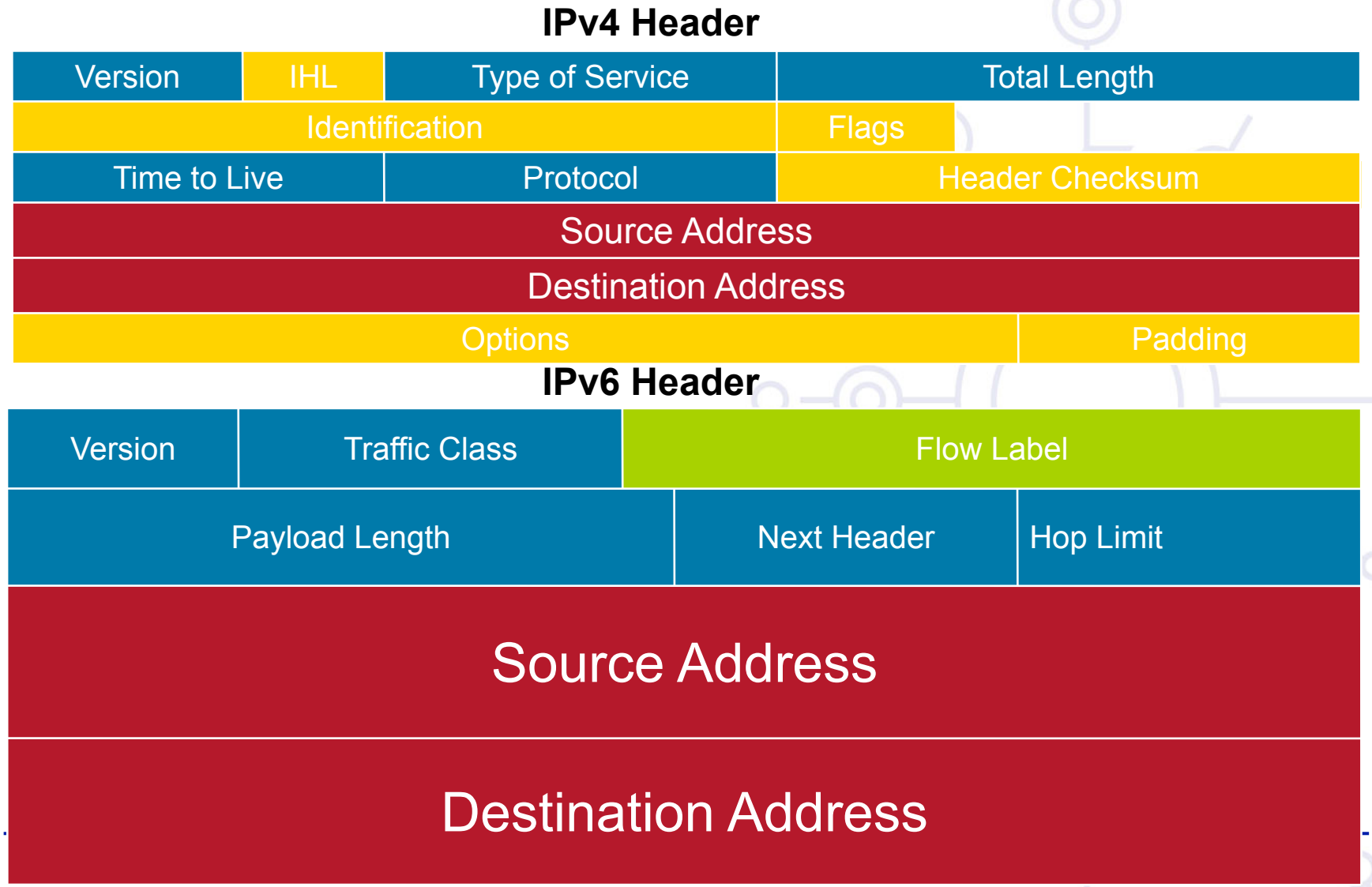


# Why security is difficult?

- If you believe that encryption (or firewalls or Intrusion Detection Systems) are the answer to all your security problems, then you probably asked the wrong question.
  - Security is about securing a *system*
  - Security is a *process NOT a product*
  - Over-concentration on technology is deeply naïve
  - However if you do major changes, like IPv4-IPv6, you must ensure you have not introduced new holes

# IPv4 / IPv6 Comparison

# Comparison of IPv4 & IPv6 header



# Comparing IPv4 / IPv6 /2

- IPv4 and IPv6 have very similar features. Major differences:

| | IPv4 | IPv6 |
|---|---|---|
| Addressing | 32 bits | 128 bits |
| HW address resolution | ARP | ICMPv6 ND/NA |
| Host auto-configuration | DHCP & (ICMP RS/RA) | ICMPv6 RS/RA & DHCPv6 (optional) |
| IPsec | Optional | Recommended (not mandatory) |
| Fragmentation | Both hosts and routers can fragment | Only hosts fragment packets |

# Addressing

- IPv6 uses 128 bit addresses
- In a similar way to IPv4
  - Addresses can be aggregated in prefix in order to simply routing
  - Different «types» of addresses are defined
    - unicast, anycast, multicast
  - Addresses can have different “scopes”
    - link-local, global
- A network host can use different addresses of different types and scopes at each given time
  - This is less common in IPv4, but it can also happen

# HW Address Resolution

- Hardware address resolution is needed when transmitting IP (v4/v6) datagrams over an Ethernet / 802.11 or similar layer 2 segment
- IPv4
  - ARP: address resolution protocol
    - A separate entity from the rest of the stack
- IPv6
  - ARP features are folded into ICMPv6's ND (neighbor discovery) sub-protocol

# Host Auto-Configuration

- Host-autoconfiguration allows “plug-and-play” network access
- IPv4
  - DHCP (+ some ICMP messages)
- IPv6
  - Two ways: stateless and stateful
  - SLAAC: Stateless Auto Configuration (ICMPv6)
  - DHCPv6: similar to v4 DHCP, stateful

# Fragmentation

- Packet fragmentation occurs when a packet being forwarded is too big for the outgoing link MTU
- IPv4
  - Any intermediate router can fragment and reassemble
- IPv6
  - Only hosts can fragment and reassemble
  - Path MTU discovery (ICMPv6)

# IPSec

- IPSec allows encryption of IP packet flows
- IPv4
  - IPSec was an afterthought and was implemented years after IPv4 was widely deployed
  - Thus IPSec support was never included in host requirements
- IPv6
  - IPv6 was born with IPSec support already considered
  - IPSec support is however a recommendation but it's not a mandatory requirement

# What is new with IPv6?

- Security was considered from the start in IPv6
- Some of the key improvements:
  - IPsec useable with the core protocols
  - Cryptographically Generated Addresses (CGA)
  - SEcure Neighbor discovery (SEND)
  - Making intrusion harder
- Tunneling and other transitions methods making security complex

# Inherent vulnerabilities

- Less experience working with IPv6
- New protocol stack implementations
- Security devices such as Firewalls and IDSs have less support for IPv6 than IPv4
- More complex networks
  - Overlay tunnels
  - Dual stack (two protocols on the same wire)

# Topics in this module

- Threats to be Countered in IPV6
  - Scanning
  - Multicast Addresses
  - Unauthorised Access Control
  - Protocol Weaknesses
  - Privacy
  - Transition Mechanisms
  - Worms/Viruses and other threats
    - There are already worms that use IPv6
      - e.g. Rbot.DUD
- Threats that are not IPv6 specifics not covered – application/browser/user insecurities

# Scanning in IPv4

- In IPv4, reconnaissance is relatively easy
  - 1. DNS/IANA crawling (whois) to determine ranges
  - 2. Ping sweeps and port scans:


  - 3. Application vulnerability scans:


```
mohacsi — bash — 69×19 — ⌘1
norfolk:~ mohacsi$ nmap -sP 192.168.0.0/24

Starting Nmap 6.01 ( http://nmap.org ) at 2012-09-09 20:31 CEST
Nmap scan report for 192.168.0.1
Host is up (0.0010s latency).
Nmap scan report for 192.168.0.104
Host is up (0.0012s latency).
Nmap scan report for 192.168.0.122
Host is up (0.0018s latency).
Nmap scan report for norfolk.lan (192.168.0.200)
Host is up (0.00032s latency).
Nmap scan report for 192.168.0.230
Host is up (0.0090s latency).
Nmap scan report for 192.168.0.237
Host is up (0.0037s latency).
Nmap scan report for airport-at-home.lan (192.168.0.241)
Host is up (0.0013s latency).
Nmap done: 256 IP addresses (7 hosts up) scanned in 2.64 seconds
norfolk:~ mohacsi$
```

# Scanning in IPv6

- Subnet Size is much larger
  - About ~50000 years to scan a /64 subnet@1M addresses/sec (exhaustive scan)
- But…
  - NMAP does NOT support IPv6 network scanning
  - IPv6 Scanning methods are changing
    - DNS based, parallelised scanning, common numbering
  - Compromising a router at key transit points
    - Can discover addresses in use
    - Scan from router?
- Avoid:
  - Using easy to guess addresses

# IPv6 addresses in the real world

- **Malone measured (*) the address generation policy of hosts and routers in real networks**

| Address type | Percentage |
|---|---|
| SLAAC | 50% |
| IPv4-based | 20% |
| Teredo | 10% |
| Low-byte | 8% |
| Privacy | 6% |
| Wordy | <1% |
| Others | <1% |

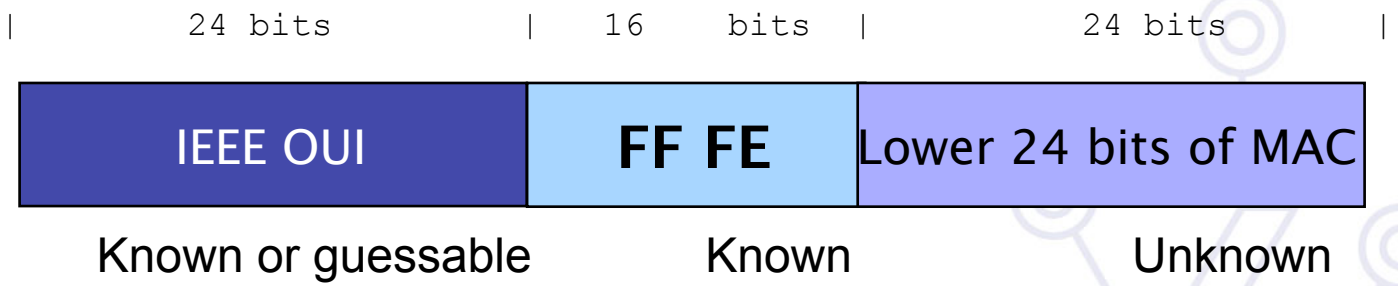
Hosts

| Address type | Percentage |
|---|---|
| Low-byte | 70% |
| IPv4-based | 5% |
| SLAAC | 1% |
| Wordy | <1% |
| Privacy | <1% |
| Teredo | <1% |
| Others | <1% |

Routers

Malone, D., "Observations of IPv6 Addresses", Passive and Active Measurement Conference (PAM 2008, LNCS 4979), April 2008, <http://www.maths.tcd.ie/~dwmalone/p/addr-pam08.pdf>.

# IPv6 addresses embedding IEEE IDs




- **Search space: ~2^24 bits – feasible!**
  - **Virtualisation environments: Well known OUIs**
  - **Same HW vendors**
- **The low-order 24-bits are not necessarily random:**
  - **An organization buys a large number of boxes**
  - **In that case, MAC addresses are usually close to each other**

# Easy to remember IPv6 addresses

- The IID is set to all-zeros, except for the last byte
  - e.g.: 2000:db8::1
  - There are other variants..
- Search space: usually 2^8 or 2^16
- Possible solution:
  - Microsoft uses randomized IIDs – non MAC-address-based - Essentially RFC 4941, without changing over time

# Scanning Multicast Addresses

- New Multicast Addresses - IPv6 supports new multicast addresses enabling attacker to identify key resources on a network and attack them
  - E.g. Site-local all DHCP servers (FF05::5), and All Routers (FF05::2)
  - Addresses must be filtered at the border in order to make them unreachable from the outside
    - To prevent smurf type of attacks: IPv6 specs forbid the generation of ICMPv6 packets in response to messages to global multicast addresses that contain requests

# IPv6 Scanning mitigation BCP

- Filter internal-use IPv6 addresses at organization border routers—prevent addresses like the all-nodes multicast address from becoming conduits for attack
- Use standard, but nonobvious static addresses for critical systems—try something a bit more complicated than ::1 for your default gateways (perhaps ::DEF1)
- Filter unneeded services at the firewall—just like in IPv4
- Selectively filter ICMP – more like IPv4!
- Maintain host and application security—just like in IPv4
- Filter Multicast at site boundary – more like IPv4

- Implement privacy extensions carefully—using them everywhere will complicate attack traceback and troubleshooting within your own organization

# Unauthorised Access Control

- Policy implementation in IPv6 with Layer 3 and Layer 4 is still done in firewalls
- Some design considerations!
  - Filter site-scoped multicast addresses at site boundaries
  - Filter IPv4 mapped IPv6 addresses on the wire

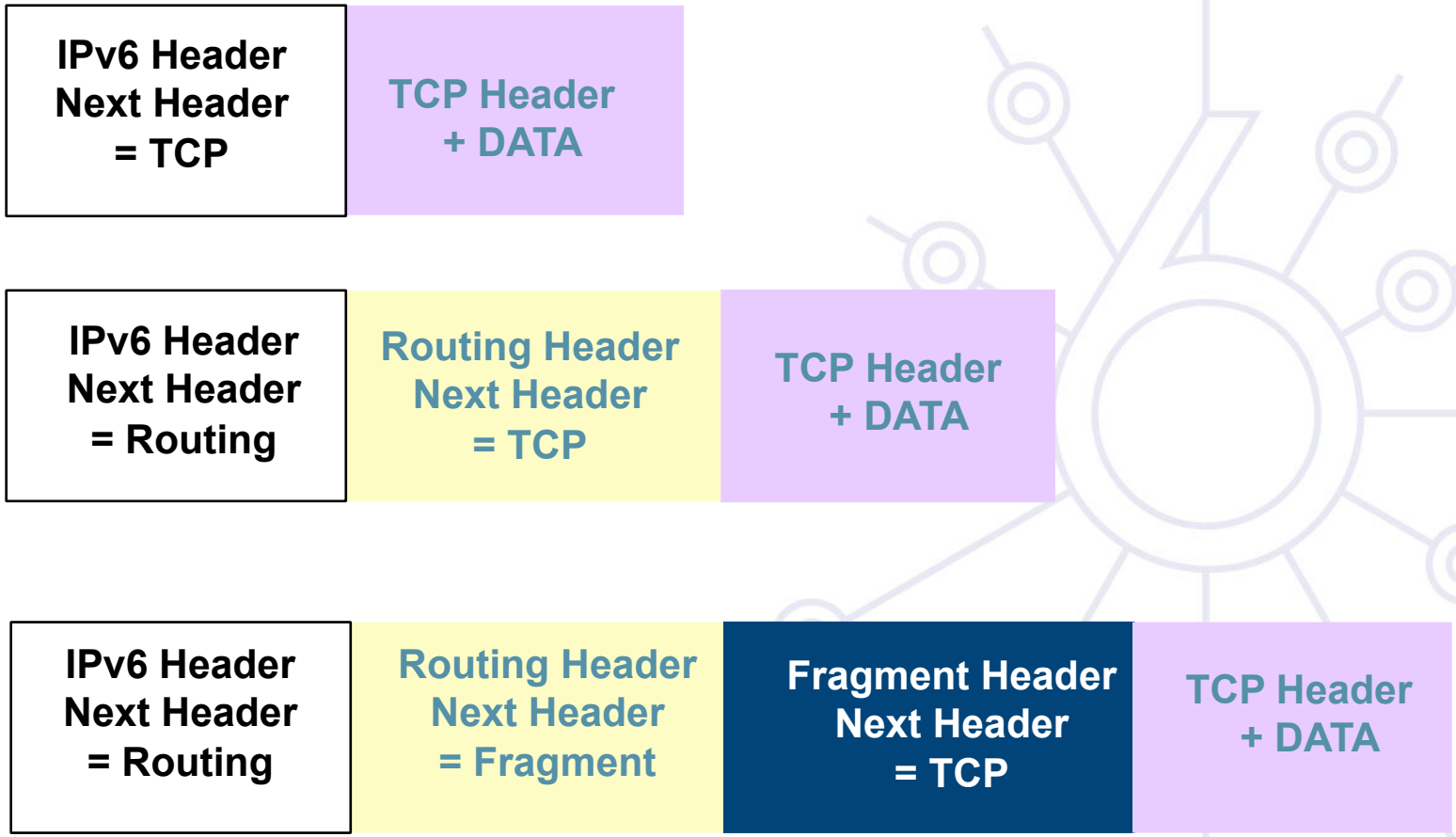
# Unauthorised Access control

- **Non-routable + bogon (unallocated) address filtering slightly different**
  - **in was IPv4 easier deny non-routable + bogons**
  - **in IPv6 simpler to permit legitimate (almost)**

| Action | Src | Dst | Src port | Dst port | |
|---|---|---|---|---|---|
| deny | 2001:db8::/32 | host/net | | | Doc prefix - NO |
| permit | 2001::/16 | host/net | any | service | |
| permit | 2002::/16 | host/net | any | service | 6to4 - YES |
| permit | 2003::/16 | host/net | any | service | |
| Deny | 3ffe::/16 | host/net | any | service | 6bone - NO |
| deny | any | any | | | |

Consult for non exisiting addresses at:
http://www.space.net/~gert/RIPE/ipv6-filters.html

# IPv6: Optional headers



# Problems with extension headers

- Routing header (RH0, deprecated by RFC 5095)
- Fragmentation - how can you determine in the fragment the upper layer protocols?
- Extension header tricking (reorder, long chains of headers, overlapping fragments)
- RFC 5722 updated the specs, forbidding overlapping fragments
- Impossible to filter without statefull firewall

- `deny ipv6 any any log undetermined transport`

# L3- L4 Spoofing

- While L4 spoofing remains the same, IPv6 address are globally aggregated making spoof mitigation at aggregation points easy to deploy
- Simpler to protect due to IPv6 address hierarchy
- However host part of the address is not protected
  - You need IPv6 <– >MAC address (user) mapping for accountability!
- Fragmented packets?

# Firewalls

- IPv6 architecture and firewall
  - NAT does not make secure – same level of security with IPv6 possible as with IPv4 (security and privacy)
    - Even better: e2e security with IPSec
  - Weaknesses of the packet filtering cannot be hidden by NAT
  - IPv6 does not require end-to-end connectivity, but provides end-to-end addressability
  - Support for IPv4/IPv6 transition and coexistence
  - Not breaking IPv4 security
- Most firewalls are now IPv6-capable
  - Cisco ACL/PIX, Juniper NetScreen, CheckPoint
  - Modern OSes now provide IPv6 capable firewalls

# Firewall setup

- No blind ICMPv6 filtering possible:

| | | | |
|---|---|---|---|
| IPv6 specific | Parameter problem | required | Error report (e.g. Extension header errors) |
| | NS/NA | | Required for normal operation – except static ND entry |
| | RS/RA | | For Stateless Address Autoconfigration |
| | Packet too big | | Path MTU discovery |
| | MLD | | Requirements in for multicast |

# Firewalls L4 issues

- Problem FTP
  - Complex: PORT, LPRT, EPRT, PSV, EPSV, LPSV (RFC 1639, RFC 2428)
  - No support in IPv6 firewalls for all the variants
- Solution: HTTP seems to be the next generation file transfer protocol with WEBDAV and DELTA
- Other non trivially proxy-able protocol:
  - No support (e.g.: H.323)

# IPv6 Unauthorized Access mitigation BCP

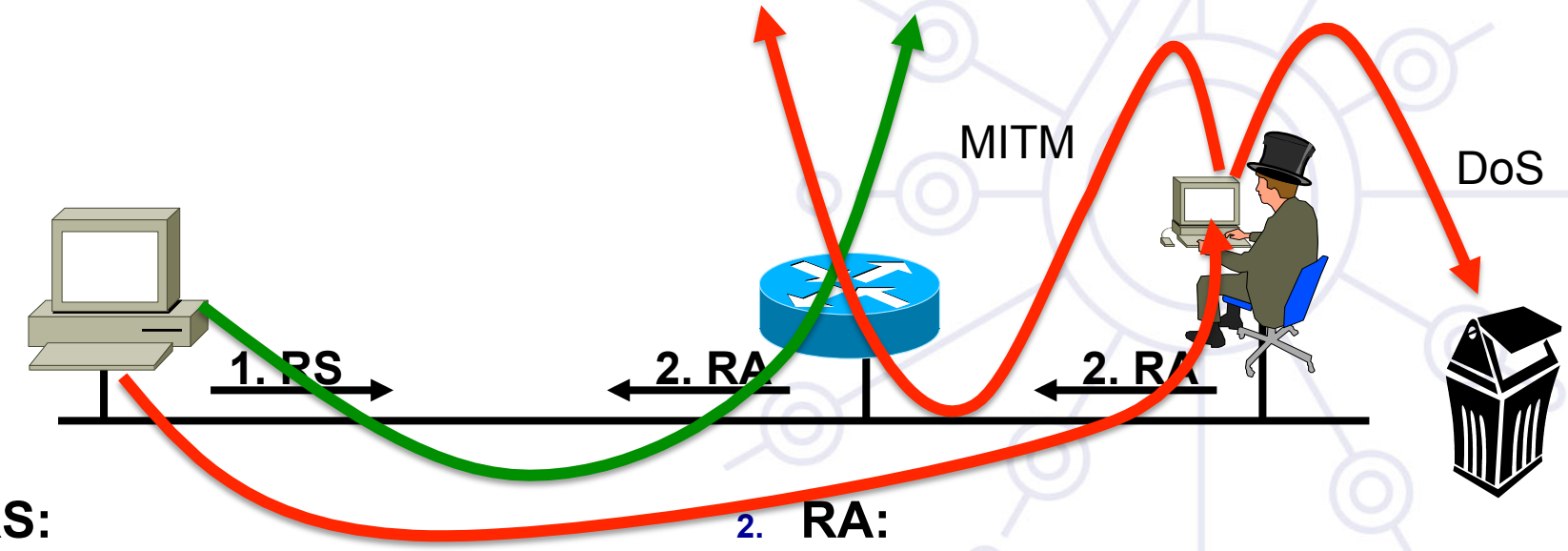
- Determine what extension headers will be allowed through the access control device—network designers should match their IPv6 extension header policy closely to their IPv4 IP options policy
- Ensure adequate IPv6 header filtering capabilities—for example, drop all packets with the routing header if you don't have MIPv6
- Deny IPv6 fragments destined to an internetworking device—used as a DoS vector to attack the infrastructure
- Determine which ICMPv6 messages are required through the access control device and apply filters appropriately—it is recommended that administrators map their ICMPv6 policy closely to the equivalent ICMPv4 policy with the following additions:
  - ICMPv6 Type 2—Packet too big
  - ICMPv6 Type 4—Parameter problem
  - ICMPv6 Type 130-132—Multicast listener
  - ICMPv6 Type 133/134—Router solicitation and router advertisement
  - ICMPv6 Type 135/126—Neighbor solicitation and neighbor advertisement
- Carefully select supported protocols – e.g. HTTP vs FTP

# PROTOCOL WEAKNESSES

# Rogue Router Advertisement

Router Advertisements contains:
-Prefix to be used by hosts
-Data-link layer address of the router
-Miscellaneous options: MTU, DHCPv6 use, …

**RA w/o Any Authentication Gives Exactly Same Level of Security as DHCPv4 (None)**




1. **RS:**
   - **Data = Query: please send RA**
2. **RA:**
   - **Data= options, prefix, lifetime, A+M+O flags**

# Effect of Rogue Router Advertisements

- Rogue RA [RFC 6104]
- Problem:
  - Denial of service: all traffic sent to a black hole
  - Man in the Middle attack: attacker can intercept, listen, modify unprotected data
- Also affects legacy IPv4-only network with IPv6-enabled hosts
- Most of the time from non-malicious users
- Requires layer-2 adjacency - some relief

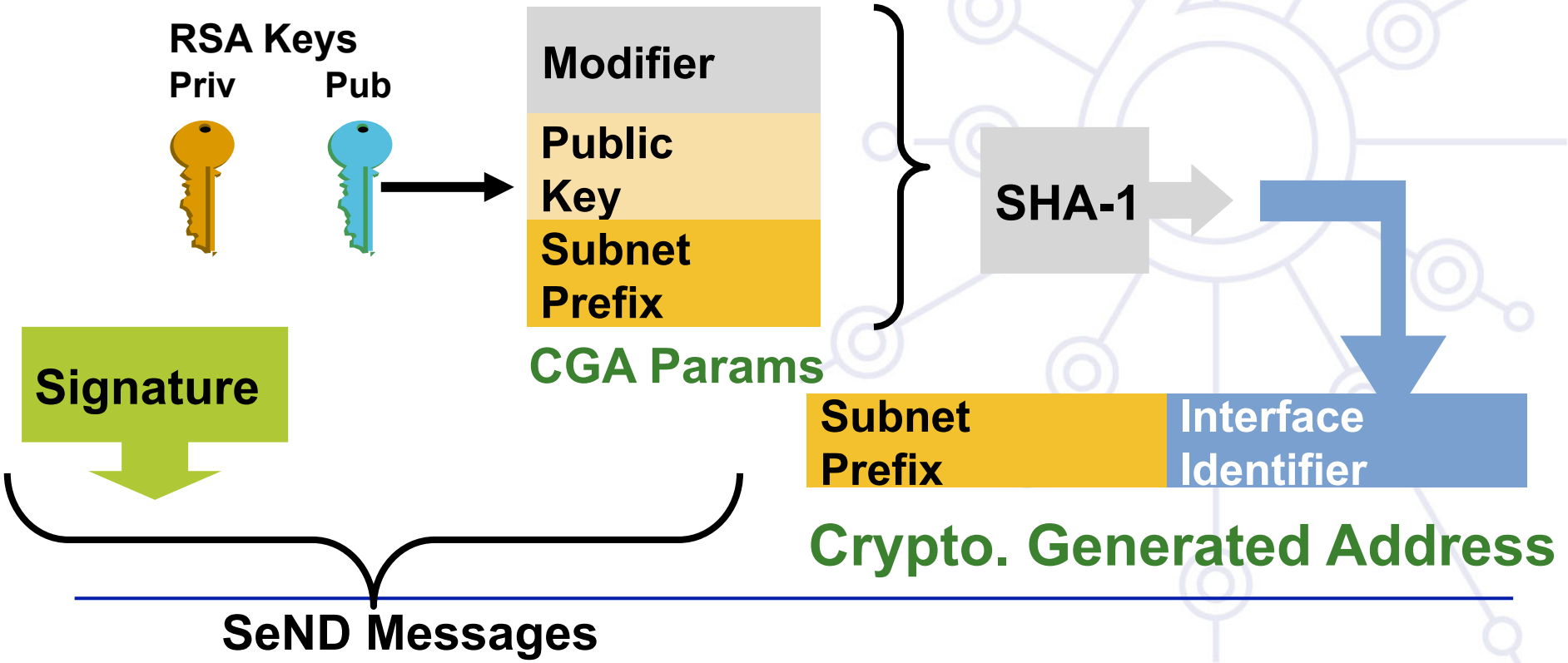
- A major blocking factor might be for enterprise IPv6 deployment

# Mitigation against Rogue RAs

1. RA snooping - RA Guard - as defined [RFC 6105]
2. ACL on switches/isolation of the Hosts
3. Usage of SEND
4. Using RA router preference – use high
5. Layer 2 admission control – like 802.1X
6. Host based filtering - unwanted RAs
7. Deprecation tools:
    1. **rafixd:** http://www.kame.net/dev/cvsweb2.cgi/kame/kame/kame/rafixd/
    2. **ramond:** http://ramond.sourceforge.net/
8. Using DHCPv6 with prefix and default gateway option

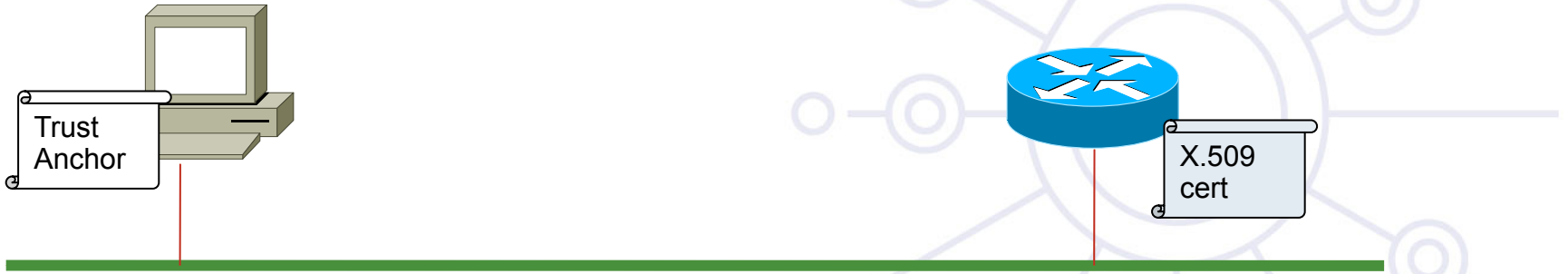
# Cryptographically Generated Addresses CGA RFC 3972 (Simplified)

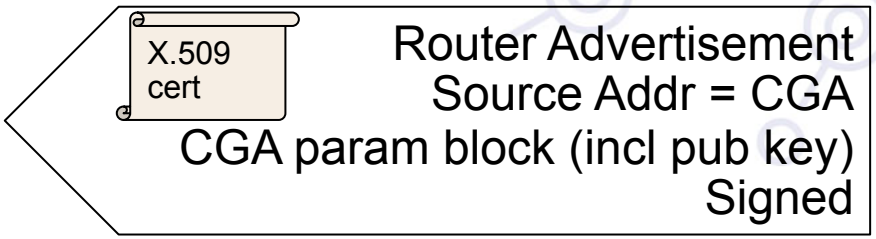
- Each devices has a RSA key pair (no need for cert)
- Ultra light check for validity
- Prevent spoofing a valid CGA address



# Securing Router Advertisements with SeND

- **Adding a X.509 certificate to RA**
- **Subject Name contains the list of authorized IPv6 prefixes**



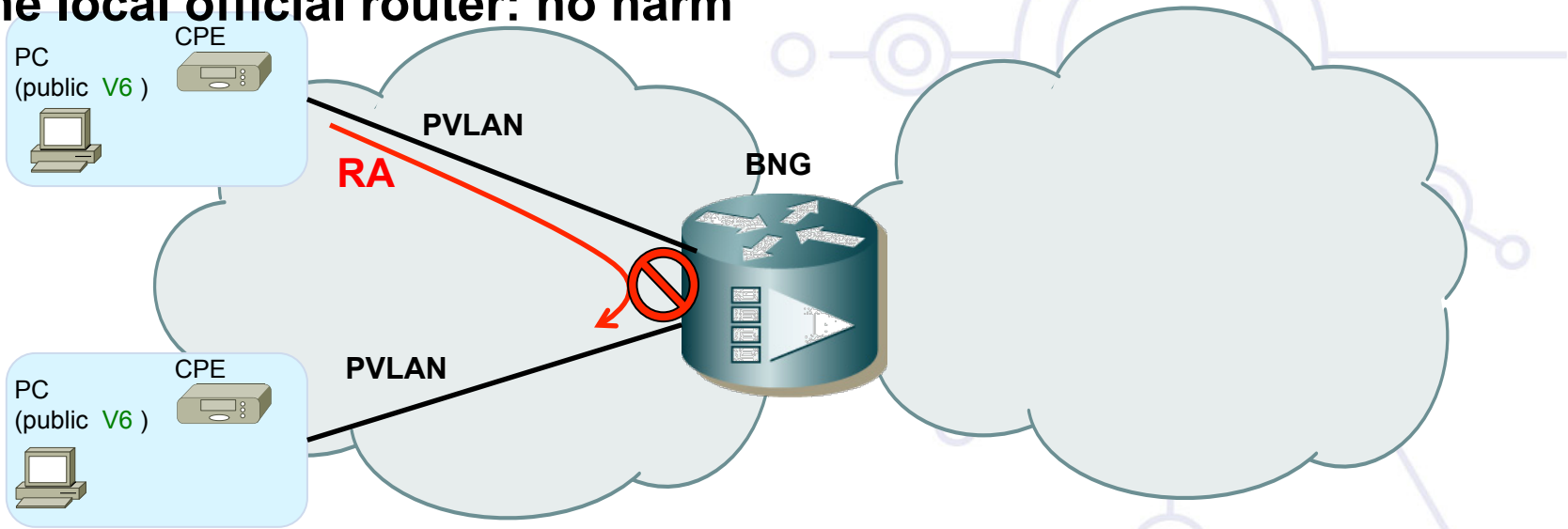



# Secure Neighbor Discovery (SeND) RFC 3971

- RFC 3972 Cryptographically Generated Addresses (CGA)
  - IPv6 addresses whose interface identifiers are cryptographically generated from node public key
- SeND adds a signature option to Neighbor Discovery Protocol
  - Using node private key
  - Node public key is sent in the clear (and linked to CGA)
- Very powerful
  - If MAC spoofing is prevented
  - But, not a lot of implementations: Cisco IOS, Linux,*BSD, third party for Windows (from Hasso-Plattner-Institut in Germany)

# Rogue RA: Host Isolation

- Prevent Node-Node Layer-2 communication by using:
  - 1 VLAN per host (SP access network with Broadband Network Gateway)
  - Private VLANs (PVLAN) where node can only contact the official router
  - Wireless in AP isolation mode
- Link-local scope multicast (RA, DHCP request, etc) sent only to the local official router: no harm



# Rogue RA: Port Access Control List

- **Port ACL blocks all ICMPv6 Router Advertisements from hosts**

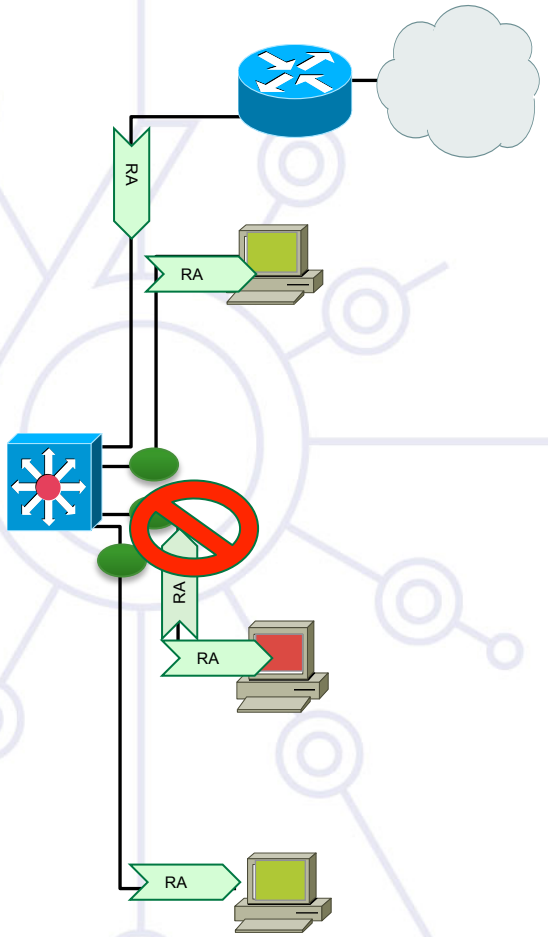
```
interface FastEthernet3/13
  switchport mode access
  ipv6 traffic-filter ACCESS_PORT in
  access-group mode prefer port
```

- ACL to filter RA and DHCPv6:

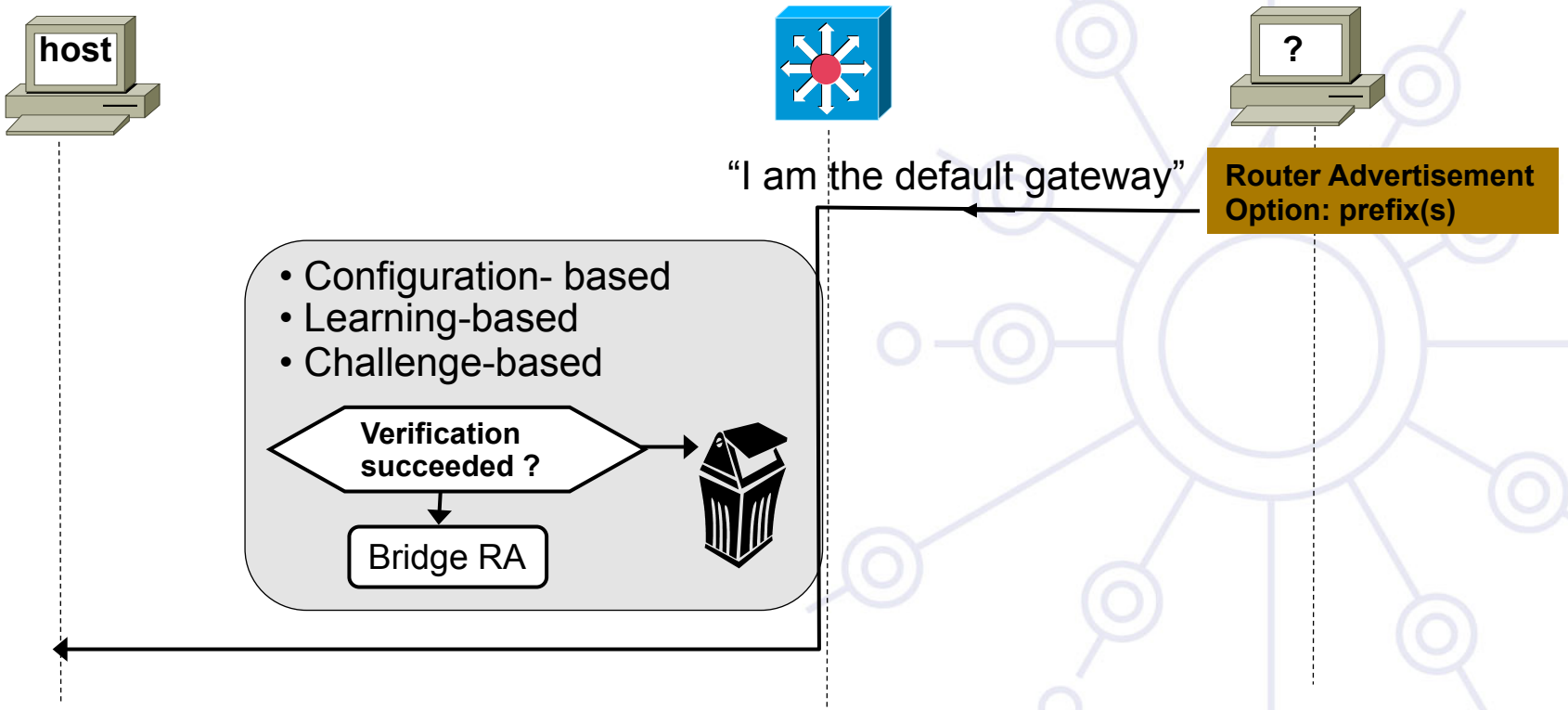
```
ipv6 access-list block-ra-dhcp
   10 deny icmp any any 134 0
   20 deny udp any eq 547 fe80::/64 eq
   546
   30 permit ipv6 any any
exit
```

- Apply for the interface:

```
interface 1-44
   ipv6 access-group block-ra-dhcp in
```



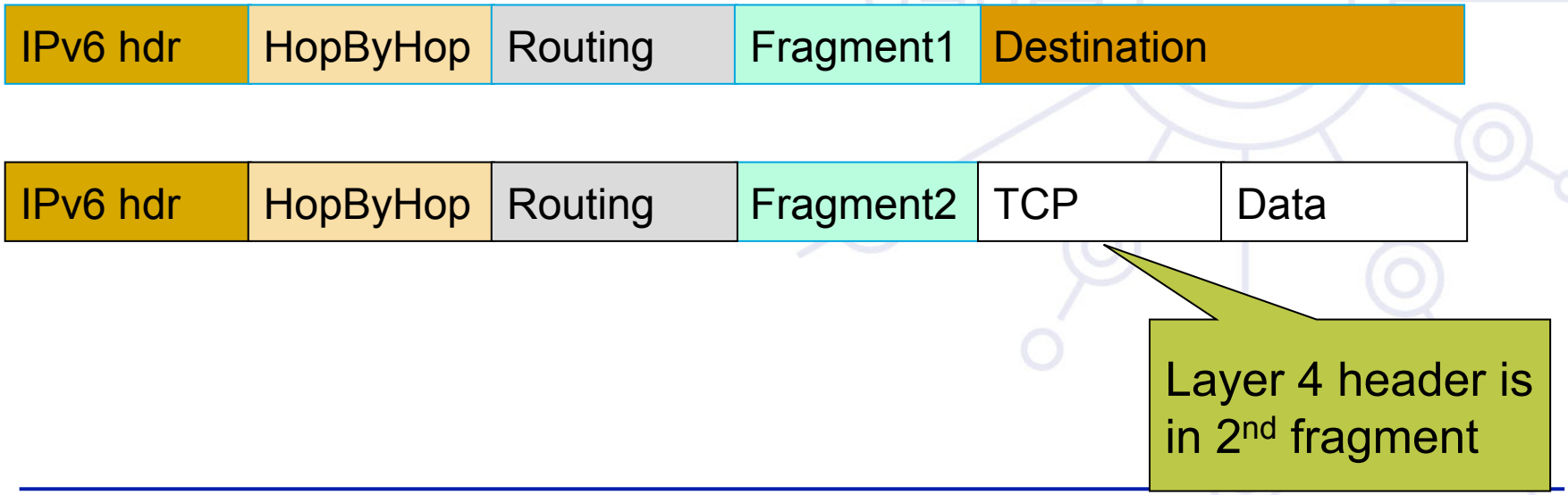
# RA-Guard (RFC 6105)




- Switch selectively accepts or rejects RAs based on various criteria's
- Can be ACL based, learning based or challenge (SeND) based.
- Hosts see only allowed RAs, and RAs with allowed content

# Here comes Fragmentation…

- **Extension headers chain can be so large than it is fragmented!**
- **RFC 3128 is not applicable to IPv6**
- **Layer 4 information could be in 2nd fragment**



# Parsing the Extension Header Chain Fragments and Stateless Filters (RA Guard)

- **RFC 3128 is not applicable to IPv6, extension header can be fragmented**
  - **IETF 6MAN could change it-** `atomic fragment draft`
- **ICMP header could be in 2nd fragment after a fragmented extension header**
- **RA Guard works like a stateless ACL filtering ICMP type 134**
- **THC fake_router6 –FD implements this attack which bypasses RA Guard**
- ***Partial work-around: block all fragments sent to ff02::1***
  - *'undetermined-transport' is even better*
  - *Does not work in a SeND environment (larger packets) but then no need for RA-guard* ☺



ICMP header is in 2nd fragment, RA Guard has no clue where to find it!

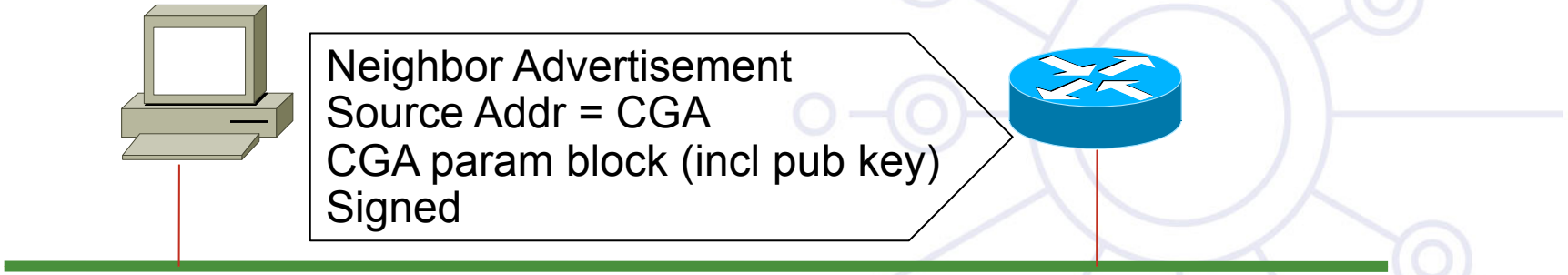
# Neighbor Advertisement can be Spoofed

- Pretty much like RA: no authentication
  - Any node can ‘steal’ the IP address of any other node
  - Impersonation leading to denial of service or MITM
  - Attack tool: THC parasite6
- Requires layer-2 adjacency
- IETF SAVI Source Address Validation Improvements (work in progress)

# NDP Spoofing Mitigations

| Where | What |
|---|---|
| Routers & Hosts | configure static neighbor cache entries |
| Routers & Hosts | Use CryptoGraphic Addresses (SeND CGA) |
| Switch (First Hop) | Host isolation |
| Switch (First Hop) | Address watch<br>• Glean addresses in NDP and DHCP<br>• Establish and enforce rules for address ownership |

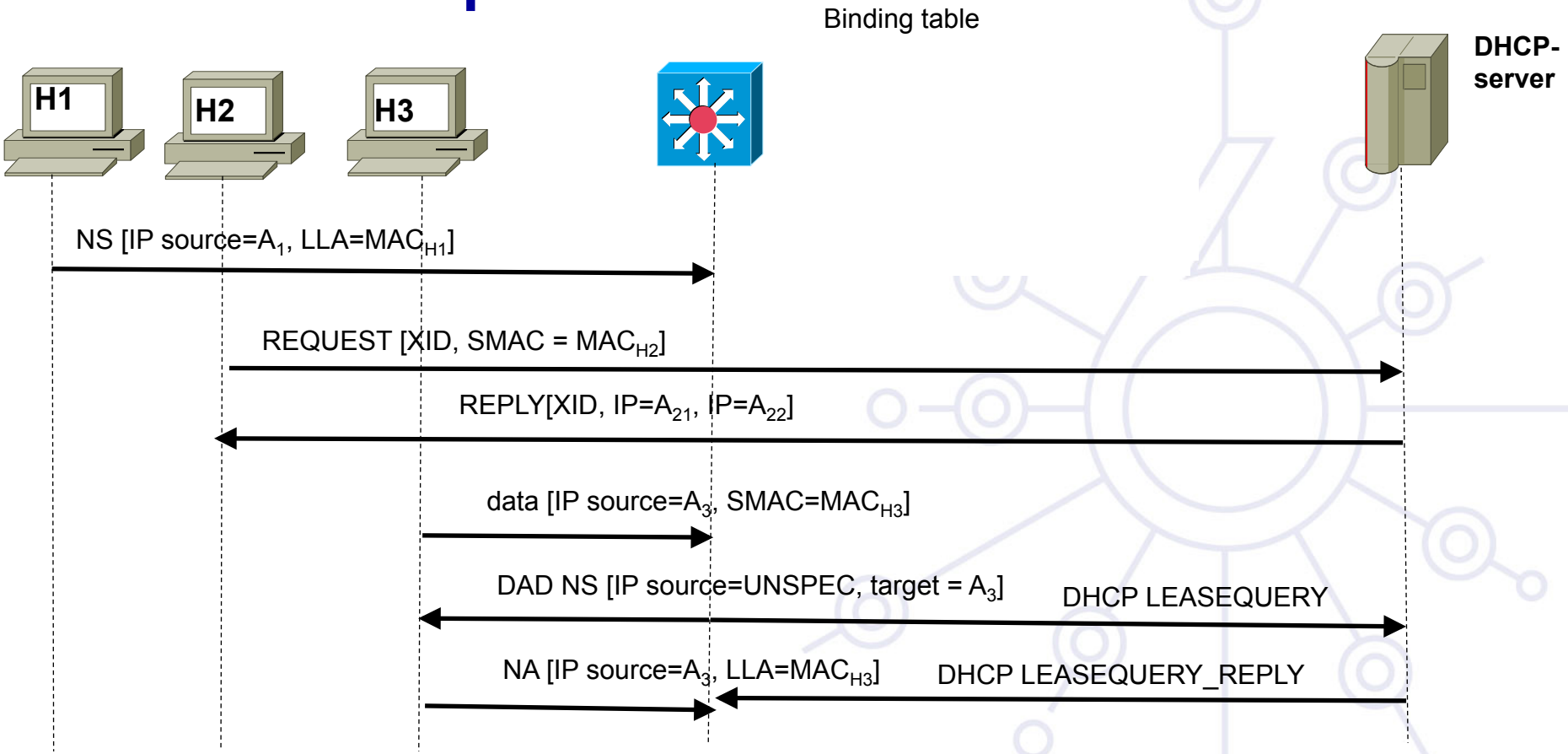
# Securing Neighbor Advertisements with SeND



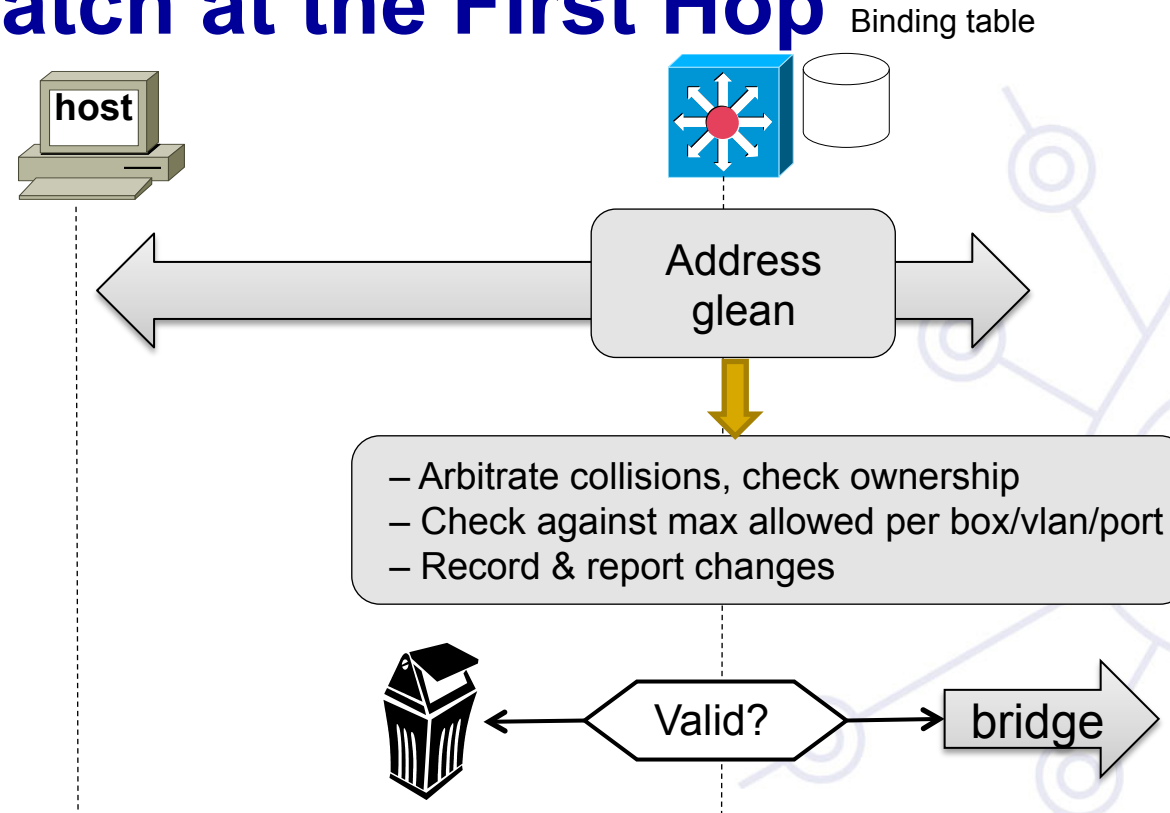
# SAVI: How to Learn?

- If a switch wants to enforce the mappings < *IP address, MAC address*> how to learn them?
- Multiple source of information
  - SeND: verify signature in NDP messages, then add the mapping
  - DHCP: snoop all messages from DHCP server to learn mapping (same as in IPv4)
  - NDP: more challenging, but '*first come, first served*'
  - The first node claiming to have an address will have it

# NDP Spoofing – Mitigation: Address Glean at the First Hop



# NDP Spoofing – Mitigation: Address Watch at the First Hop




- Preference is a function of: configuration, learning method, credential provided
- Upon collision, choose highest preference (for instance static, trusted, CGA, DHCP preferred over dynamic, not_trusted, not_CGA, SLACC)
- For collision with same preference, choose First Come, First Serve

# DHCPv6 problems

- Fake DHCPv6 server
  - Define who can act as DHCP server

# First Hop Security implementation

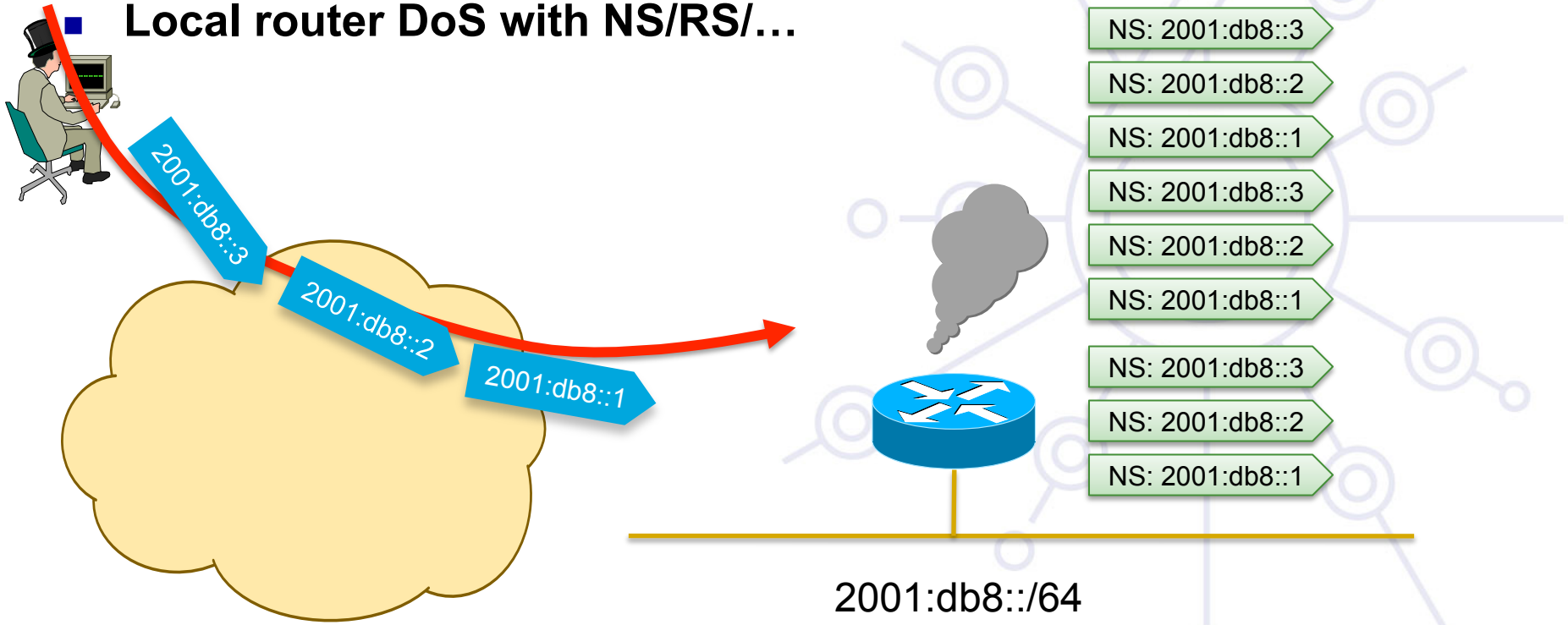
- Cisco
  - IPv6 VLAN ACL & RA Guard: 12.2(54)SG, 3.2.0SG, 15.0(2)SG, 12.2(33)SXI4
  - NDP inspection: 12.2(50)SY and 15.0(1)SY

For more Information:
http://www.cisco.com/en/US/docs/ios/ipv6/configuration/guide/ip6-roadmap.html
http://www.cisco.com/en/US/docs/ios-xml/ios/ipv6/configuration/15-2mt/ip6-first-hop-security.html

- Juniper
  - soon

# Scanning Made Bad for CPU
# Remote Neighbor Cache Exhaustion

- Remote router CPU/memory DoS attack if aggressive scanning
  - Router will do Neighbor Discovery... And waste CPU and memory
- Local router DoS with NS/RS/…

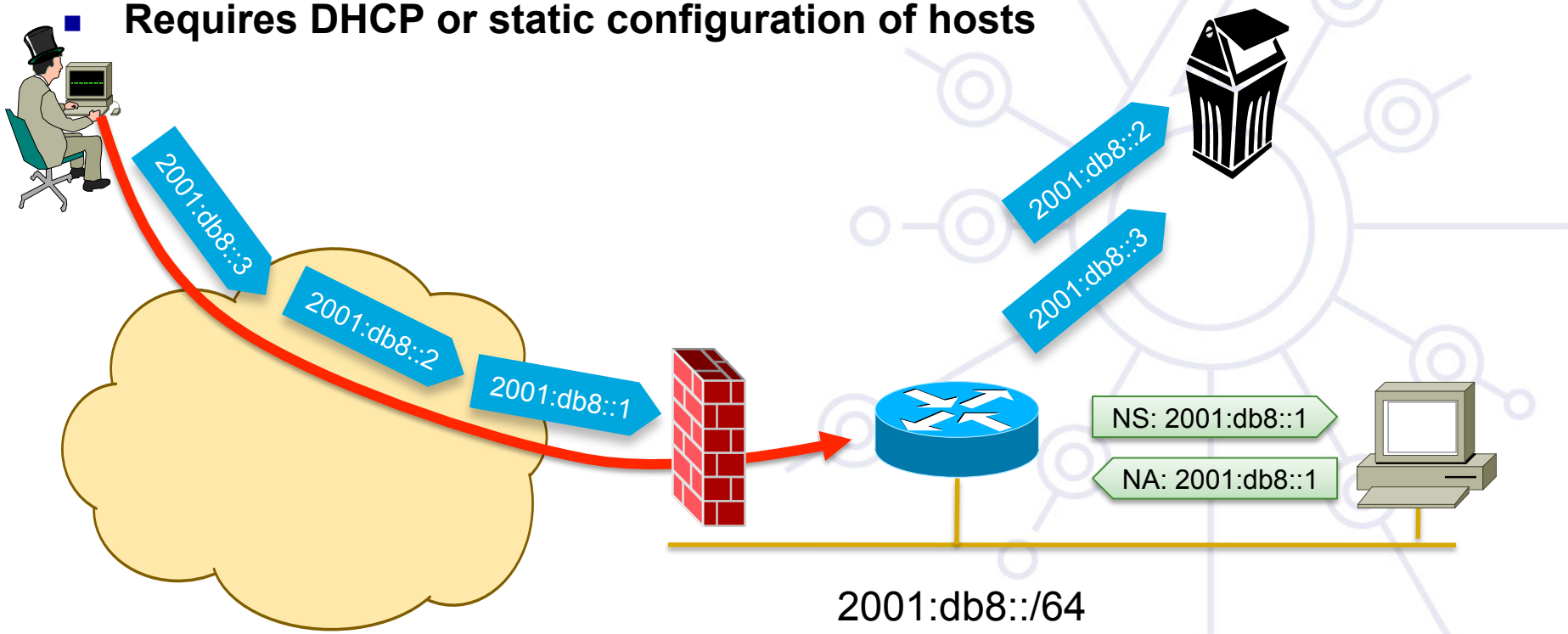


# Mitigating Remote Neighbor Cache Exhaustion

- Mainly an implementation issue
  - Rate limiter on a global and per interface
  - Prioritize renewal (PROBE) rather than new resolution
  - Maximum Neighbor cache entries per interface and per MAC address
- Internet edge/presence: a target of choice
  - Ingress ACL permitting traffic to specific statically configured (virtual) IPv6 addresses only
  - ⇒ Allocate and configure a /64 but uses addresses fitting in a /120 in order to have a simple ingress ACL
  - ⇒ Use of link local addresses

# Simple Fix for Remote Neighbor Cache Exhaustion

- Ingress ACL allowing only valid destination and dropping the rest
- NDP cache & process are safe
- Requires DHCP or static configuration of hosts



# Amplification (DDoS) Attacks

- There are no broadcast addresses in IPv6
  - This would stop any type of amplification attacks that send ICMP packets to the broadcast address
  - Global multicast addresses for special groups of devices, e.g. link-local addresses, etc.
- IPv6 specifications forbid the generation of ICMPv6 packets in response to messages to global multicast addresses
  - Many popular operating systems follow the specification
  - Still uncertain on the danger of ICMP packets with global multicast source addresses

# Mitigation of IPv6 amplification

- Be sure that your host implementations follow the ICMPv6 spec [RFC 4443]
- Implement Ingress Filtering
  - Defeats Denial of Service Attacks which employ IP Source Address Spoofing [RFC 2827]
- Implement ingress filtering of IPv6 packets with IPv6 multicast source address

# Privacy problem

- IIDs source: static, SLAAC, DHCPv6
- SLAAC: Modified EUI-64 IIDs are constant
- Host roaming: the prefix changes, IID constant – non HTTP cookie for tracking HOST

# Mitigation to host-tracking

- RFC 4941: privacy/temporary addresses
  - Random IIDs that change over time
  - Generated in addition to traditional SLAAC addresses
  - Traditional addresses used for server-like communications, temporary addresses for client-like communications
  - Privacy extension is not switched on by default – difficult to track
  - DAD for each temporary addresses
  - Privacy extension can be enabled per prefix based
- Operational problems:
  - Difficult to manage in LAN – changing over the time

# Stable privacy-enhanced addresses

- draft-gont-6man-stable-privacy-addresses proposes to generate Interface IDs as:
- F(Prefix, Interface_Index, Network_ID, Secret_Key)
- Where:
  - F() is a hash function
  - Prefix SLAAC or link-local prefix
  - Interface_Index is the (internal) small number that identifies the interface
  - Network_ID could be e.g. the SSID of a wireless network
  - Secret_Key is unknown to the attacker (and randomly generated by default)

# Stable privacy-enhanced addresses (II)

- As a host moves:
  - Prefix and Network_ID change from one network to another
  - But they remain constant within each network
  - F() varies across networks, but remains constant within each network
- This results in addresses that:
  - Are stable within the same subnet
  - Have different Interface-IDs when moving across networks
- Document already accepted as a 6man wg item

# IPv6 transition mechanisms

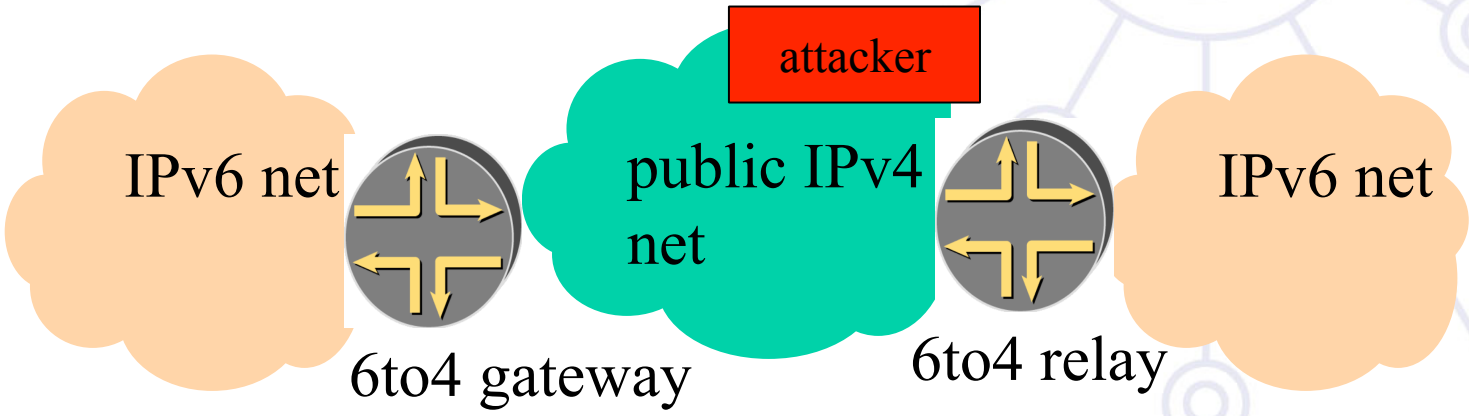
- ~15 methods possible in combination
- Dual stack:
  - enable the same security for both protocol
- Tunnels:
  - ip tunnel – punching the firewall (protocol 41)
  - gre tunnel – probably more acceptable since used several times before IPv6
  - l2tp tunnel – udp therefore better handled by NATs
  - Teredo tunnel – udp - better to avoid – host only solution

# Mixed IPv4/IPv6 environments

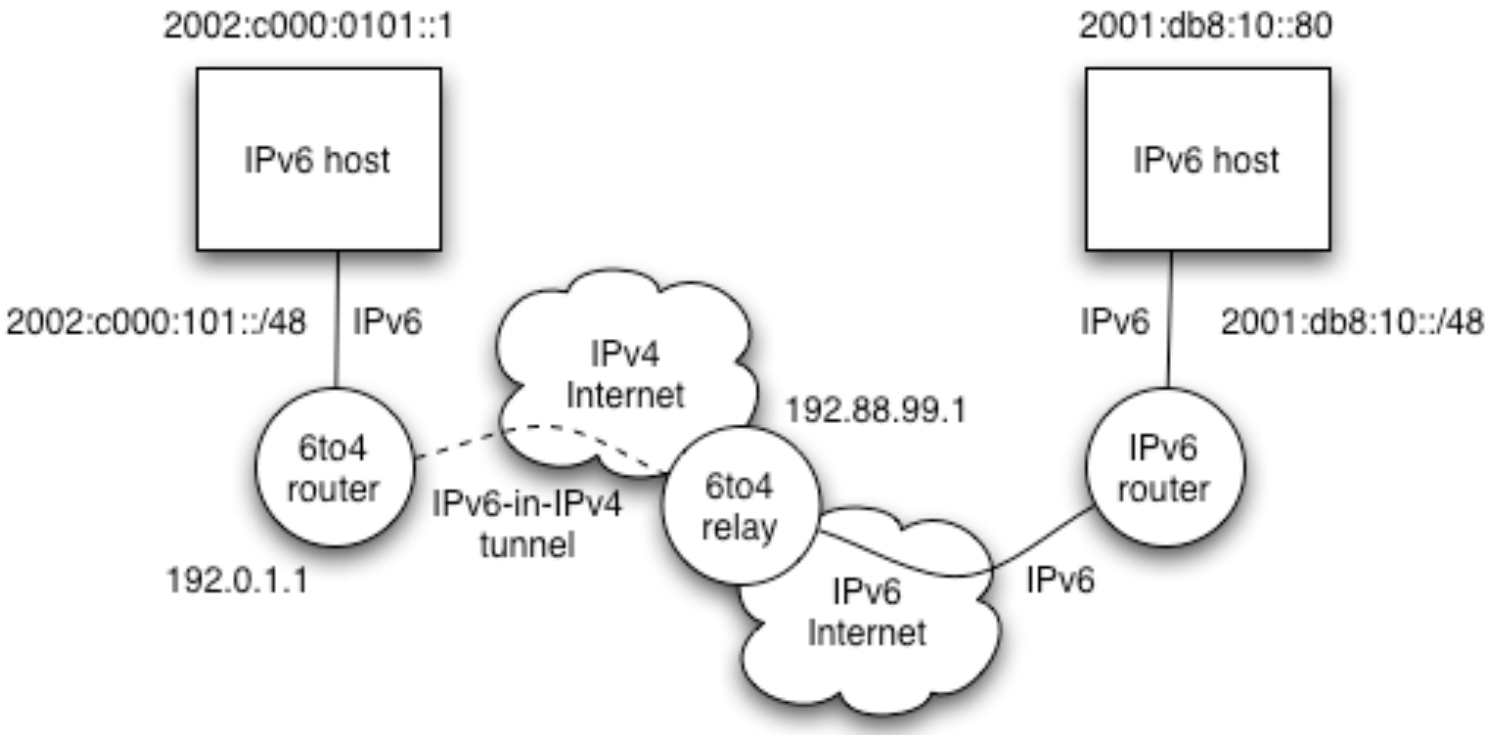
- Some security issues with transition mechanisms
  - Tunnels often interconnect networks over areas supporting the “wrong” version of protocol
  - Tunnel traffic often not anticipated by the security policies. It may pass through firewall systems due to their inability to check two protocols in the same time
- Do not operate completely automated tunnels
  - Avoid “translation” mechanisms between IPv4 and IPv6, use dual stack instead
  - Only authorised systems should be allowed as tunnel end-points

# L3 – L4 Spoofing in IPv4 with 6to4

- For example, via 6to4 tunnelling spoofed traffic can be injected from IPv4 into IPv6.
  - IPv4 Src: IPv4 Address
  - IPv4 Dst: 6to4 Relay Anycast (192.88.99.1)
  - IPv6 Src: 2002:: Spoofed Source
  - IPv6 Dst: Valid Destination



# 6to4 with relay



# Other threats

- IPv6 Routing Attack
  - Use traditional authentication mechanisms for BGP and IS-IS.
  - Use IPsec to secure protocols such as OSPFv3 and RIPng
- Viruses and Worms
- Sniffing
  - Without encryption, IPv6 is no more or less likely to fall victim to a sniffing attack than IPv4
- ICMP attacks – slight differences with ICMPv4
  - Recommendations for Filtering ICMPv6 Messages in Firewalls (RFC4890)
  - TCP ICMP attacks – slight differences with ICMPv6
    - http://tools.ietf.org/html/draft-ietf-tcpm-icmp-attacks-06
- Application Layer Attacks
  - Majority of vulnerabilities on the Internet today are at the application layer, something that nothing to do with IPv6
- Man-in-the-Middle Attacks (MITM)
  - Without proper encrytpion, any attacks utilizing MITM will have the same likelihood in IPv6 as in IPv4
- Flooding
  - Flooding attacks are identical between IPv4 and IPv6

# Vulnerability testing/assessment

- Testing tools
  - Nmap, Ettercap, Lsof, Snoop, DIG, Etherape, Wireshark, Fping, Ntop, SendIP, TCPDump, WinDump, IP6Sic, NetCat6, Ngrep, THC-IPv6, Amap
- Assessment tools
  - SAINT, nessus, ndpmon, ramond, rafixd

# Attacker tools

- Scanners: Nmap, halfscan6, Scan6, CHScanner
- Packet forgery: Scapy6, SendIP, Packit, Spak6
- DoS Tools: 6tunneldos, 4to6ddos, Imps6-tools
- THC IPv6 Attack Toolkit: parasite6, alive6, fake_router6, redir6, toobig6, detect-new-ip6, dos-new-ip6, fake_mld6, fake_mipv6, fake_advertiser6, smurf6, rsmurf6

  http://freeworld.thc.org/
- Si6Networks toolkit: Runs on Linux and *BSD

  http://www.si6networks.com/tools

# SUMMARY

# Beware of IPv6 In IPv4 Networks

- I do not have IPv6 in my network and I won't support it. I do not care then
- Well, you should
- Even though you do not run IPv6 in your network, you may be vulnerable:
  - Rogue RA (Windows Network Sharing)
  - 6to4, Teredo and other tunnel technologies
- All these may open holes in your network security

# Summary

- IPv6 has potential to be a foundation of a more secure Internet
- Elements of the IPv6 security infrastructure
  - Firewalls, IPSec, privacy enhanced address etc.

  are mature enough to be deployed in production environment.
- Other elements are in usable pilot state
  - CGA, SEND, VPNs, RA-Guard, DHCPv6 snooping etc.

  But even these are ready for deployment


6deploy
Questions
http://www.6deploy.eu
mohacsi@niif.hu