

IPv6 Security

APRICOT 14 – Manila, Philippines

February 2009

Merike Kaeo

merike@doubleshotsecurity.com



Agenda

- IPv6 Fundamentals Review
 - Addressing
 - Header Formats
 - ICMP Considerations
- Architectural Design Choices for IPv6 Networks
- IPv6 Device Security
- Securing The Data Plane for IPv6 Networks
- Securing The IPv6 Routing Infrastructure
- Where Does IPsec Fit In?



IPv6 Fundamentals Review (Addressing)



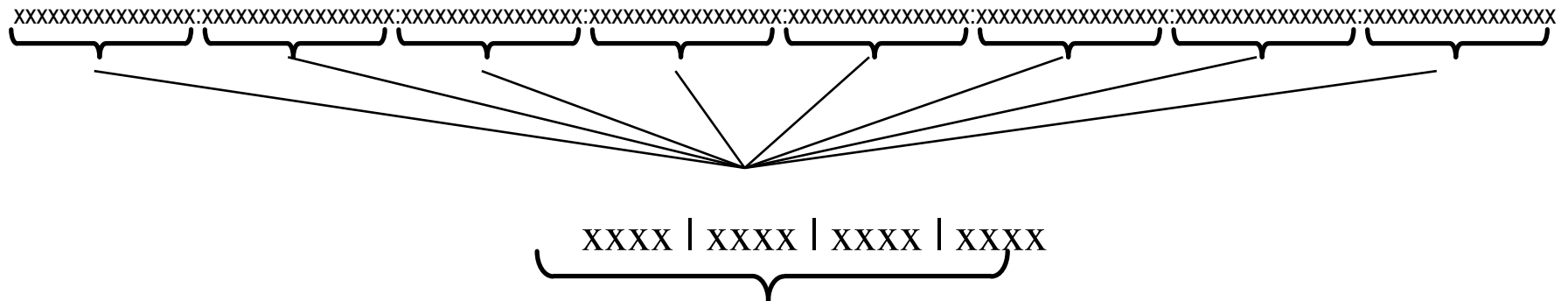
Security Workshop - APRICOT 14 Manila, Philippines - February 2009

IPv6 Address Representation

- Format
 - X:X:X:X:X:X:X:X
 - 2001:0DB8:6665:0036:0000:0000:0000:B100
 - Case insensitive
 - Leading Zero's are optional
 - 2001:0DB8:6665:36:0:0:0:B100
 - Successive 0's are represented as '::' but only once in an address
 - 2001:0DB8:6665:36::B100



IPv6 Addressing



0000: 0	0100: 4	1000: 8	1100: C
0001: 1	0101: 5	1001: 9	1101: D
0010: 2	0110: 6	1010: A	1110: E
0011: 3	0111: 7	1011: B	1111: F

2001:DB8::/32

2001:DB8:0:0::/64

2001:DB8:0:0:0:0:FFFE::/112

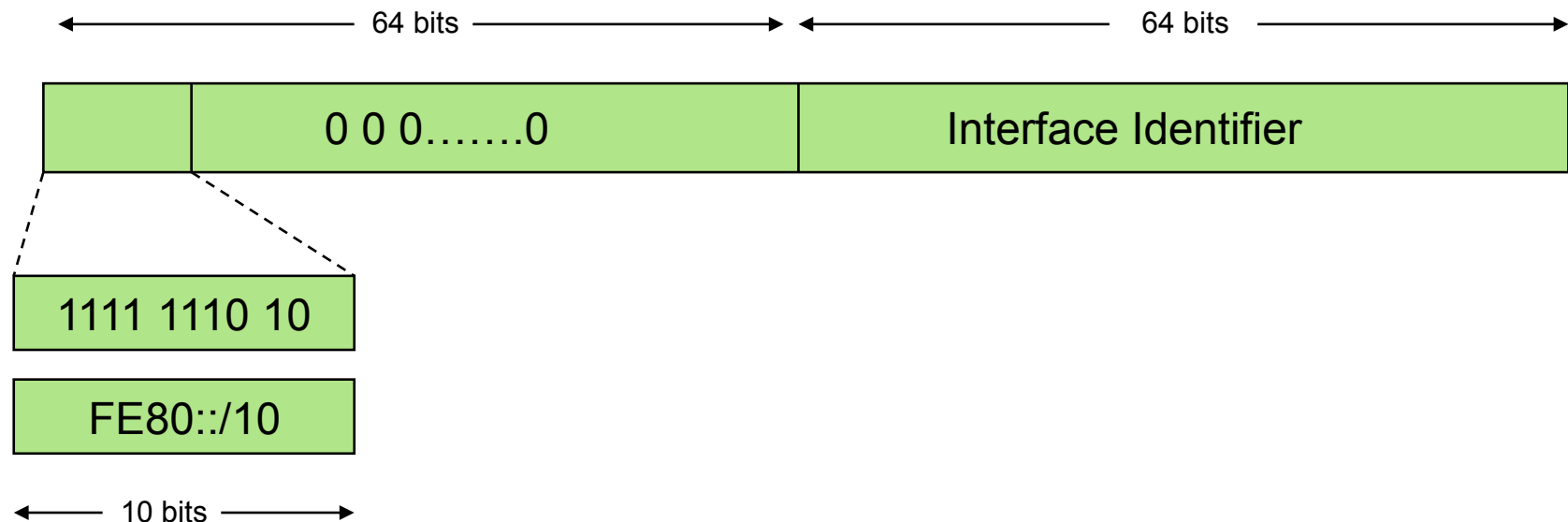


IPv6 Addressing Architecture

- Unicast Addresses
 - Global Unique
 - Link-Local
 - Site-Local (deprecated but still used)
 - Special-Use
 - Unspecified / Loopback / IPv4-Compatible / IPv4-Mapped
- Multicast Addresses
- Interface Identifiers



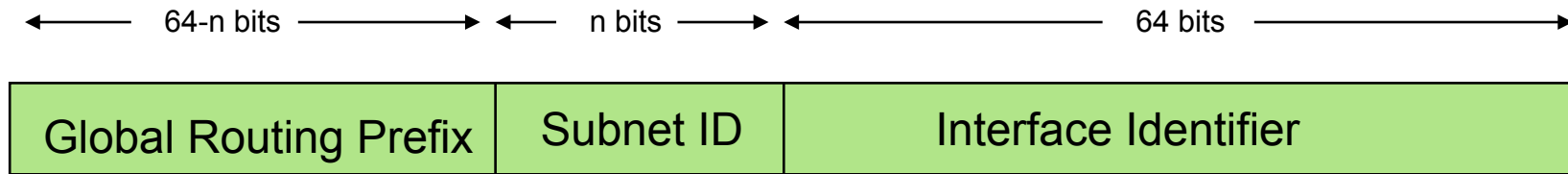
Link-Local Addresses



- Unique on the local link
- Automatically calculated and assigned by interface itself
- Nodes can communicate on local IPv6 link without presence of router or server
- Routers are not allowed to forward packets with link local src or dst off-link



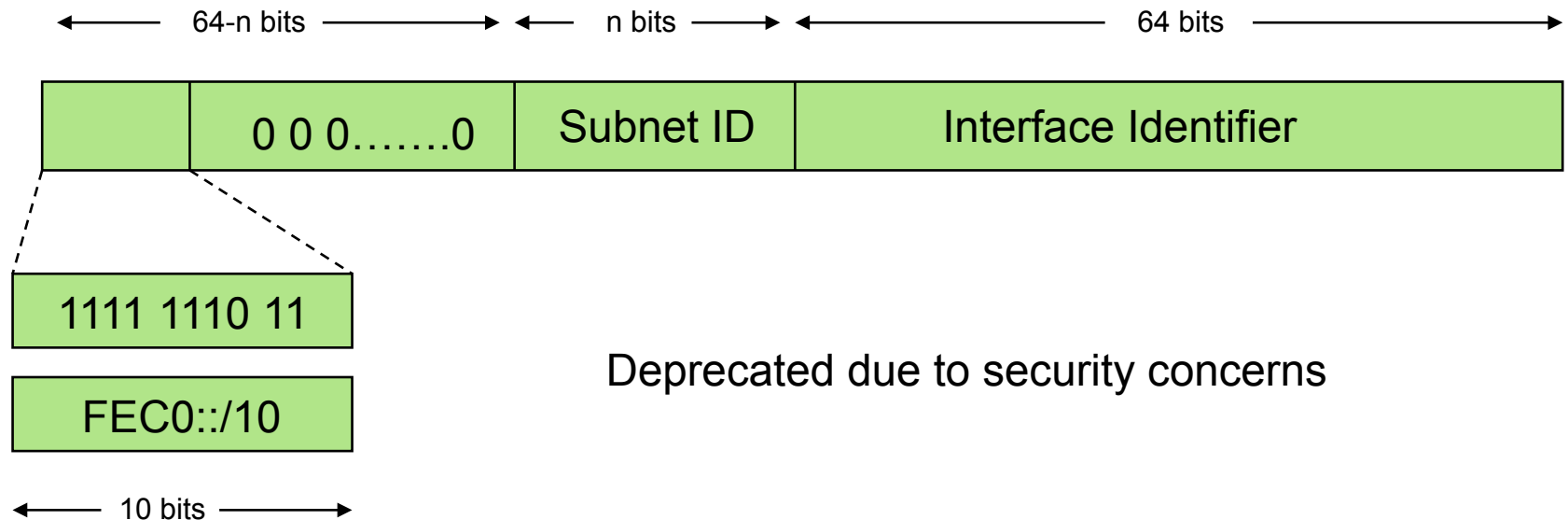
Global Unique Addresses



- First 64 bits are “network number”
 - Typical RIR allocations are /32 to ISPs
 - Typical ISP allocations are /48
- Network Number + Interface ID = Globally Unique Address



Site-Local Addresses



- Scope is limited to the local site
- Idea was to have something comparable to RFC1918
- Experience with NAT/RFC1918
- 'Replaced' by ULAs



Special Purpose Addresses

- Unspecified Address
 - 0:0:0:0:0:0:0:0
 - Used for bootstrapping, i.e. when no other address is available
 - Initial DHCP request
 - Duplicate address detection
- Loopback Address
 - 0::1
 - Identifies self (same as 127.0.0.1 in IPv4)

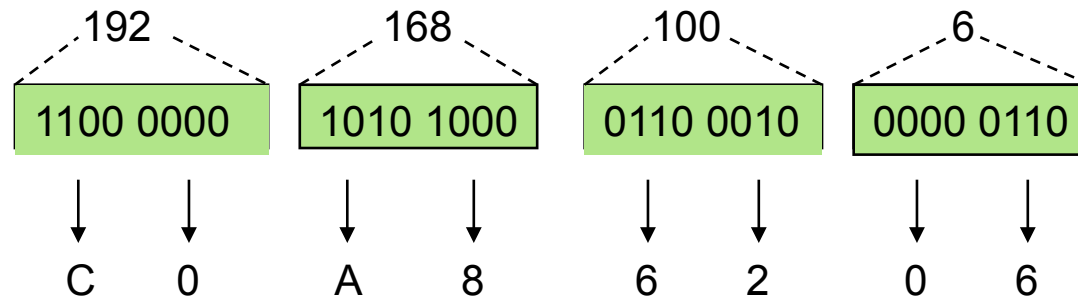


IPv4 Compatible Addresses

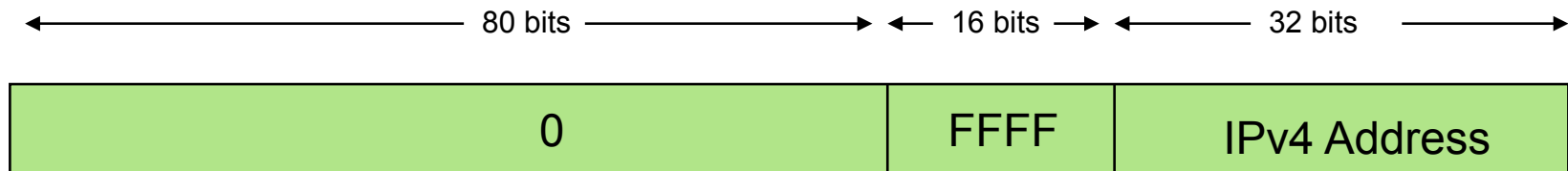


0:0:0:0:0:0:192.168.100.6 → ::C0A8:6206

XXXX XXXX : XXXX XXXX : XXXX XXXX : XXXX XXXX

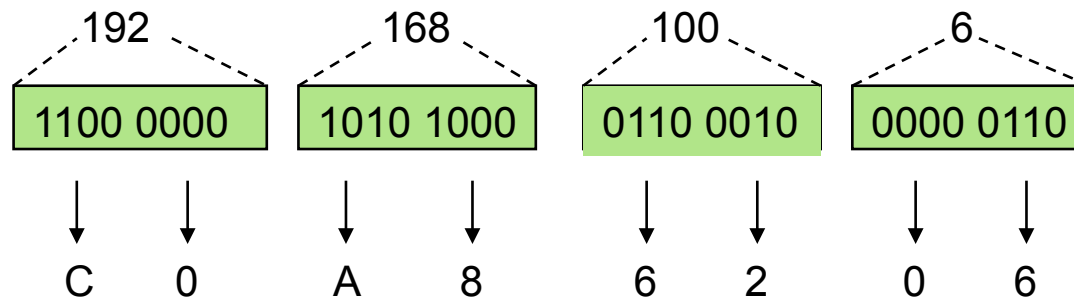


IPv4-Mapped Addresses

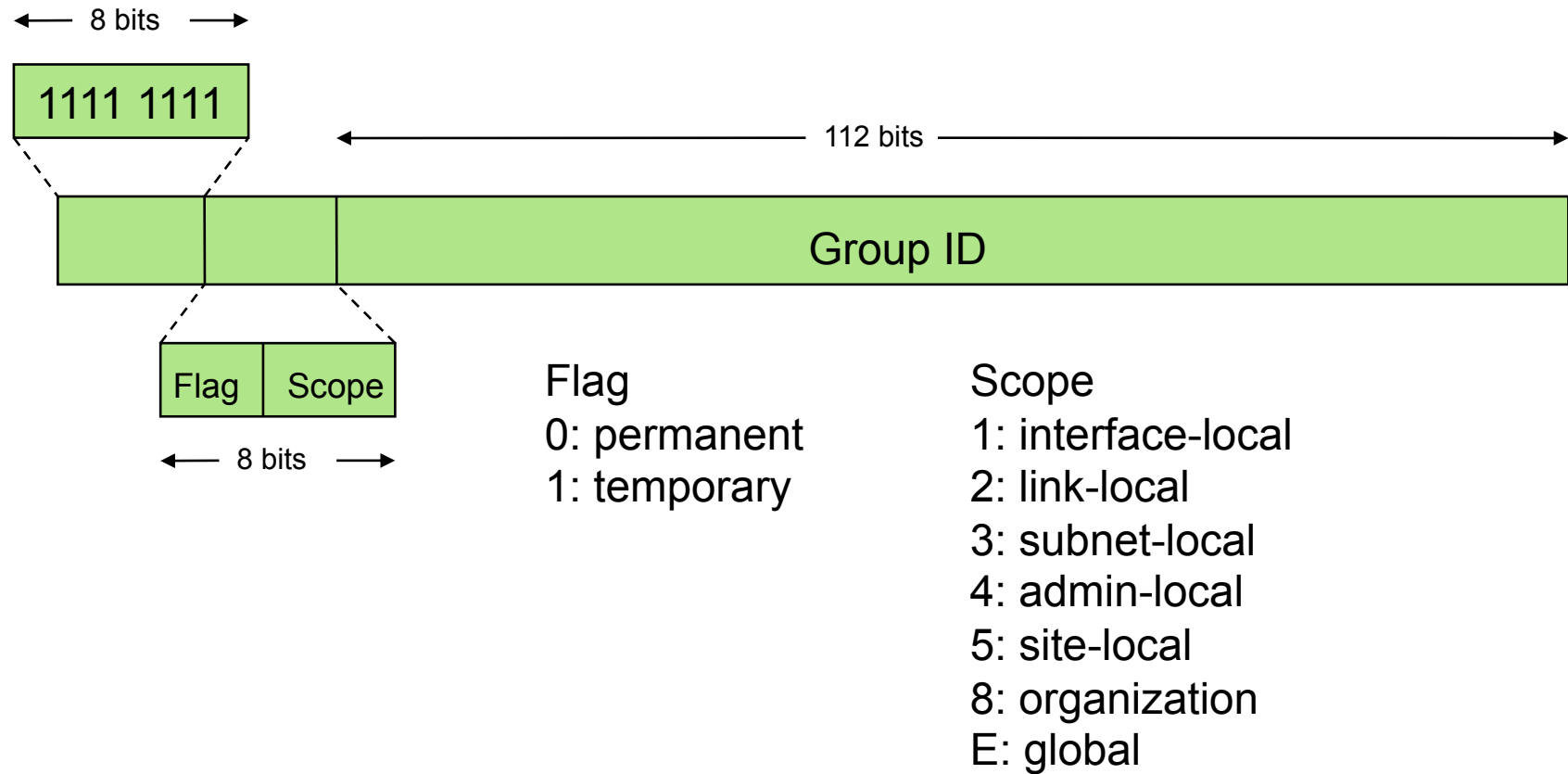


0:0:0:0:0:FFFF:192.168.100.6 → ::FFFF:C0A8:6206

XXXX XXXX : XXXX XXXX : XXXX XXXX : XXXX XXXX



Multicast Addresses



Stateless Address Autoconfiguration (SLAAC)

- RFC2462
- For autoconfiguration of IPv6 there are two options
 - Stateful (DHCPv6)
 - Stateless (via RA)
- For SLAAC this is done by combining address prefix advertised in the RA with the Interface ID
 - EUI-64 or RFC3041 (privacy addresses)
- Thought to help renumbering of a network
- Problem
 - How do I find a DNS server?
 - How do I send update to the DNS server?



IPv6 Fundamentals Review (Header Formats)



Security Workshop - APRICOT 14 Manila, Philippines - February 2009

IPv4 Header Format

20 octets + options: 13 fields, including 3 flag bits

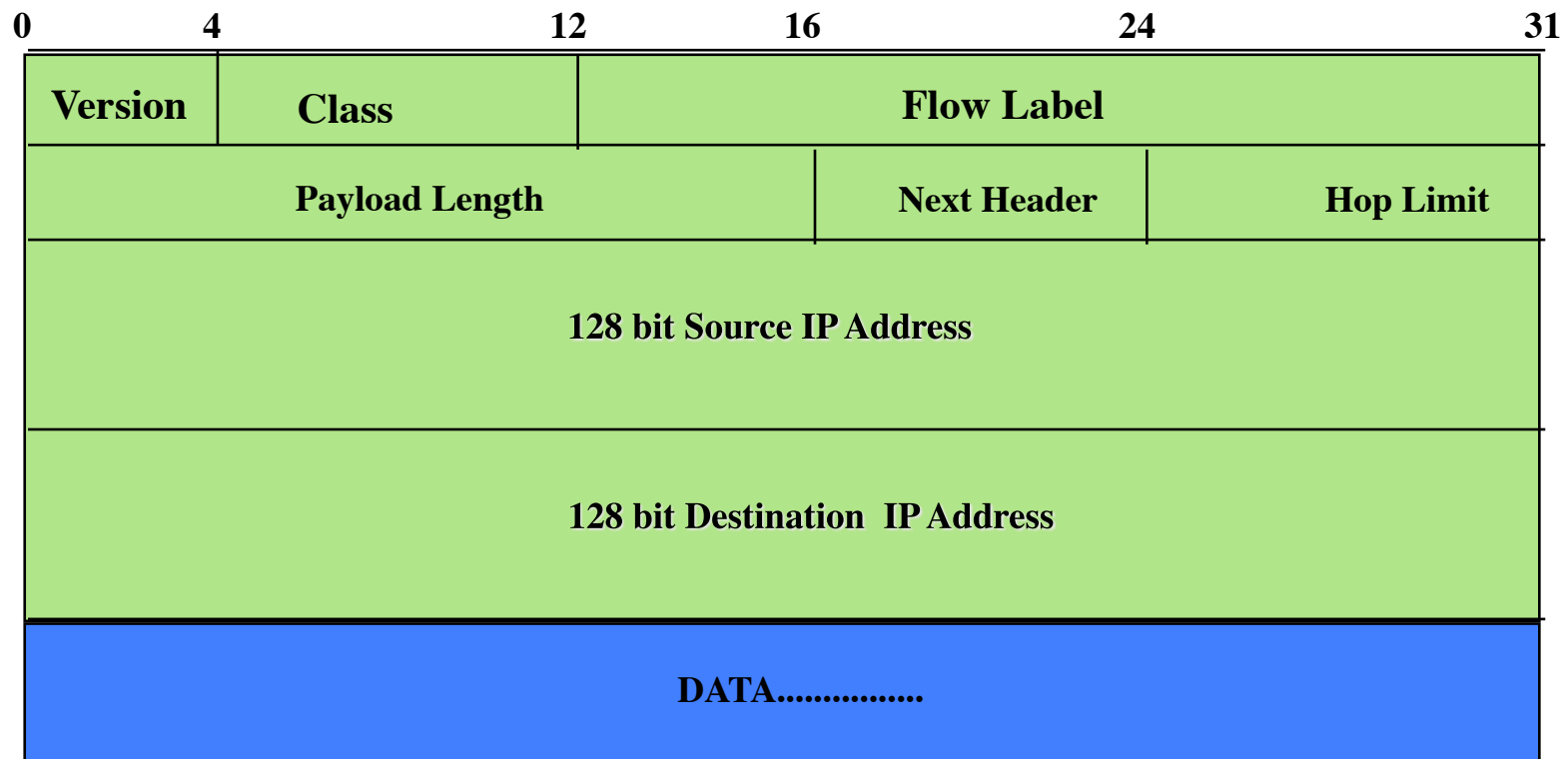
0	4	8	16	31
Version	IHL	Type of Service	Total Length (in bytes)	
Identification			Flags	Fragmentation Offset
Time to Live	Protocol		Header Checksum	
Source IP Address				
Destination IP Address				
Options (if any)				Padding
DATA.....				

Fields in red front are absent from IPv6 header



IPv6 Header Format

40 octets, 8 fields



Summary: IPv4/IPv6 Header Changes

- Streamlined
 - Fragmentation fields moved out of base header
 - IP options moved out of base header
 - Header Checksum eliminated
 - Header Length field eliminated
 - Length field excludes IPv6 header
- Revised
 - Time to Live = Hop Limit
 - Protocol = Next Header
 - Precedence & TOS = Traffic Class
 - Addresses increased from 32 bits to 128 bits
- Extended
 - Flow Label field added



IPv6 Extension Headers

- Carry the additional options and padding features that are part of the base IPv4 header
- Extension headers are optional and placed after the base header
- There can be zero, one, or more Extension Headers between the IPv6 header and the upper-layer protocol header
- Ordering is important

Currently Defined IPv6 Extension Headers:

- Hop-by-Hop Options (0)
- Routing Header (43)
- Fragment Header (44)
- ESP Header (50)
- Authentication Header (51)
- Destination Options (60)

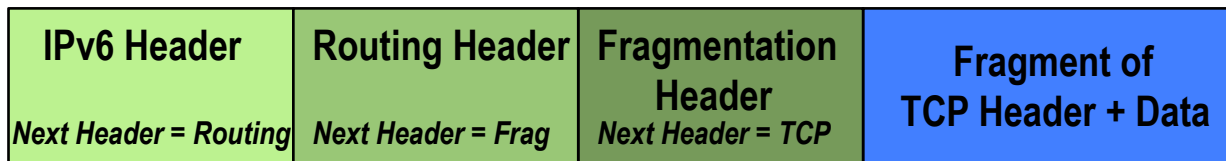
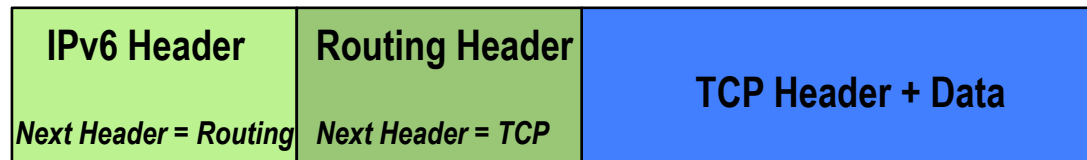
Other Extension Header Values:

- TCP upper-layer (6)
- UDP upper-layer (17)
- ICMPv6 (58)
- No Next Header Present (59)



Extension Header Chaining

There can be zero, one, or more Extension Headers between the IPv6 header and the upper-layer protocol header



Extension Header Ordering

- Hop-by-Hop
- Destination Options*
 - For options processed by the 1st destination address plus subsequent destinations listed in the routing header
- Routing
- Fragment
- Authentication
- Encapsulating Security Payload
- Destination Options
- Upper-Layer header



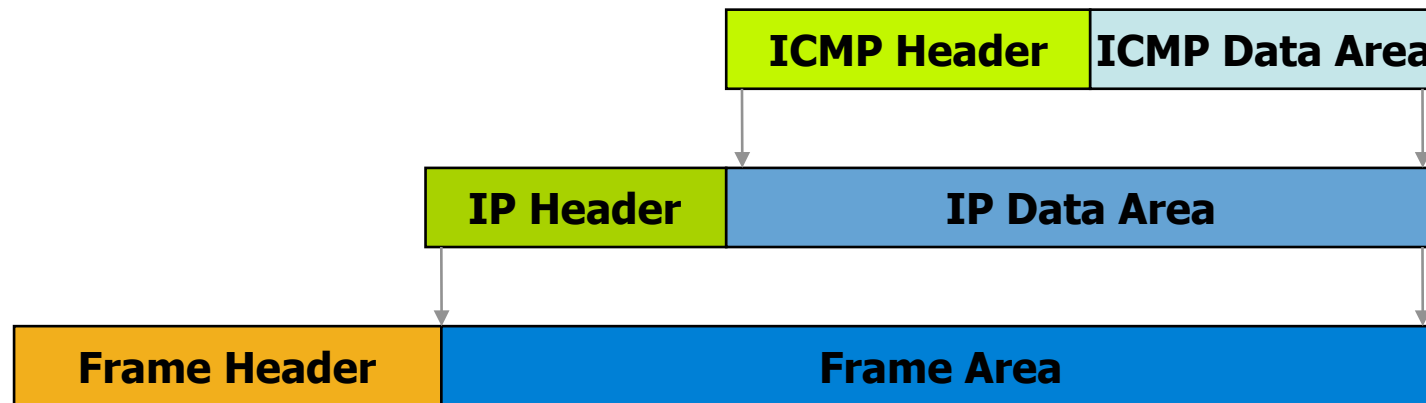
IPv6 Fundamentals Review (ICMP Considerations)



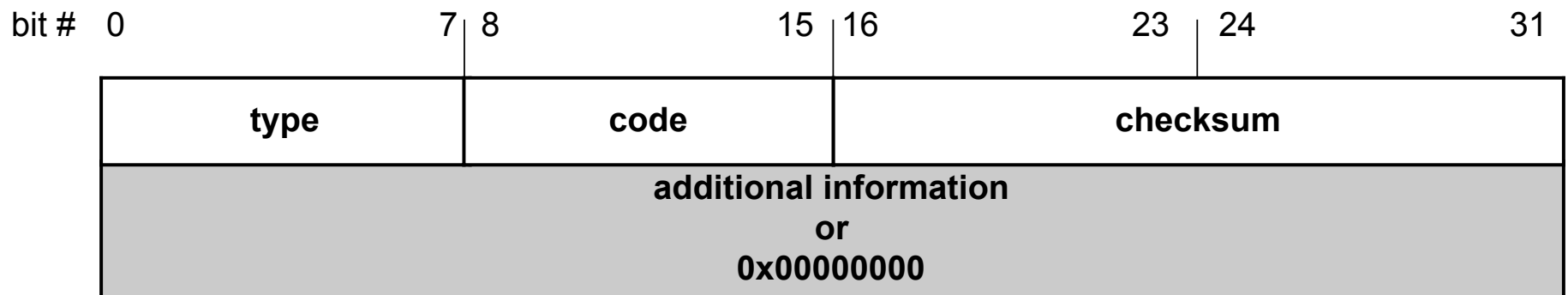
Security Workshop - APRICOT 14 Manila, Philippines - February 2009

Internet Control Message Protocol

- Original specification in RFC 792
- Used to report problems with delivery of IP packets
- Supports Path MTU (PMTU) Discovery between a sender and a receiver, which helps to optimize performance of data delivery between pairs or hosts by avoiding fragmentation en route



ICMP Message Format



4 byte header:

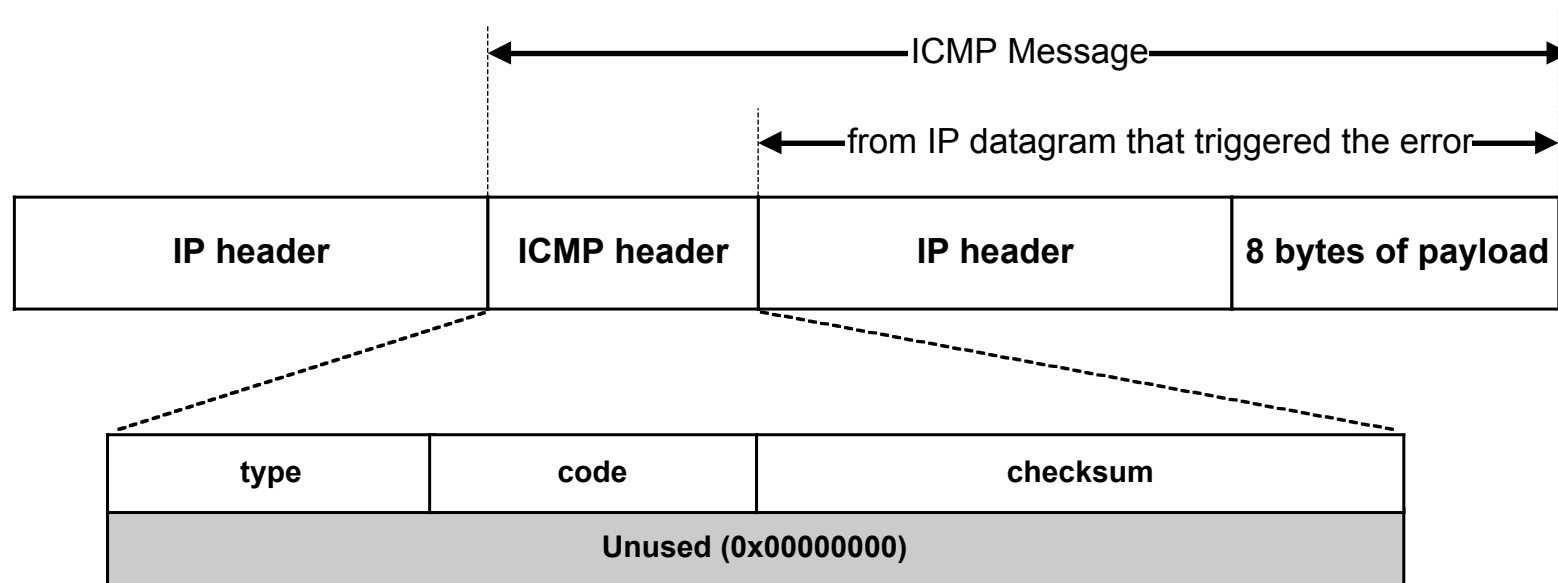
- **Type (1 byte):** type of ICMP message
- **Code (1 byte):** subtype of ICMP message
- **Checksum (2 bytes):** similar to IP header checksum. Checksum is calculated over entire ICMP message

If there is no additional data, there are 4 bytes set to zero.
Each ICMP messages is at least 8 bytes long



ICMP Error Messages

- ICMP error messages report error conditions
- Typically sent when a datagram is discarded
- ICMP error messages include the complete IP header and the first 8 bytes of the payload (typically: UDP, TCP)



ICMP Message Types

Type	Message Type	Description
3	Destination Unreachable	Packet could not be delivered
11	Time Exceeded	Time to live field hit 0
12	Parameter Problem	Invalid header field
4	Source Quench	Tell host to slow down transmission due to congestion
5	Redirect	Notification that packet seems to be routed wrong
8	Echo	Ask a machine if it is alive and reachable
0	Echo Reply	Yes, I am alive
13	Timestamp Request	Same as Echo request, but with timestamp
14	Timestamp Reply	Same as Echo reply, but with timestamp



Destination Unreachable Codes

Code	Definition
0	Network Unreachable [no routing table entry available for destination network]
1	Host Unreachable [destination host should be directly reachable but does not respond to ARP requests]
2	Protocol Unreachable [protocol in protocol field of IP header is not supported at destination]
3	Port Unreachable [transport protocol at destination host cannot pass datagram to an application]
4	Fragmentation needed & Don't Fragment was set
5	Source Route failed
6	Destination Network Unknown
7	Destination Host Unknown
8	Source Host Isolated
9	Communication Destination Network is Administratively Prohibited
10	Communication Destination Host is Administratively Prohibited
11	Destination Network Unreachable for Type of Service
12	Destination Host Unreachable for Type of Service
13	Communication Administratively Prohibited
14	Host Precedence Violation
15	Precedence Cutoff Violation



Additional Codes for ICMP Types

Redirect Codes

Code	Definition
0	Redirect Datagram for the Network (or subnet)
1	Redirect Datagram for the Host
2	Redirect Datagram for the Type of Service & Network
3	Redirect Datagram for the Type of Service & Host

Time Exceeded Codes

Code	Definition
0	Time to Live Exceeded in Transit
1	Fragment Reassembly Time Exceeded

Parameter Problem Codes

Code	Definition
0	Pointer Indicates the Error
1	Missing a Required Option
2	Bad Length



ICMPv6

- Is similar to IPv4 ICMP, with a few differences:
 - ICMP is carried in an IPv6 datagram
 - A checksum is computed since ICMP is a transport protocol, relative to IPv6
 - New messages are defined for the IPv6 specification
 - In an error message, the original datagram is included in the error packet for easier recovery by the source
- Identified by the Next Header value = 58
- ICMP header contains Type and Code fields to identify and qualify the message specifics
- Two defined ICMP classes in IPv6:
 - Error Messages
 - Informational



ICMPv6 Error Messages

- Identified by a Type field value between 0 and 127
- Message Types:
 - destination unreachable
 - no route
 - administratively prohibited (i.e. firewalls)
 - address unreachable
 - port unreachable
 - packet too big
 - time exceeded
 - parameter problem
 - erroneous header field
 - unrecognized next header type
 - unrecognized option



ICMPv6 Informational Messages

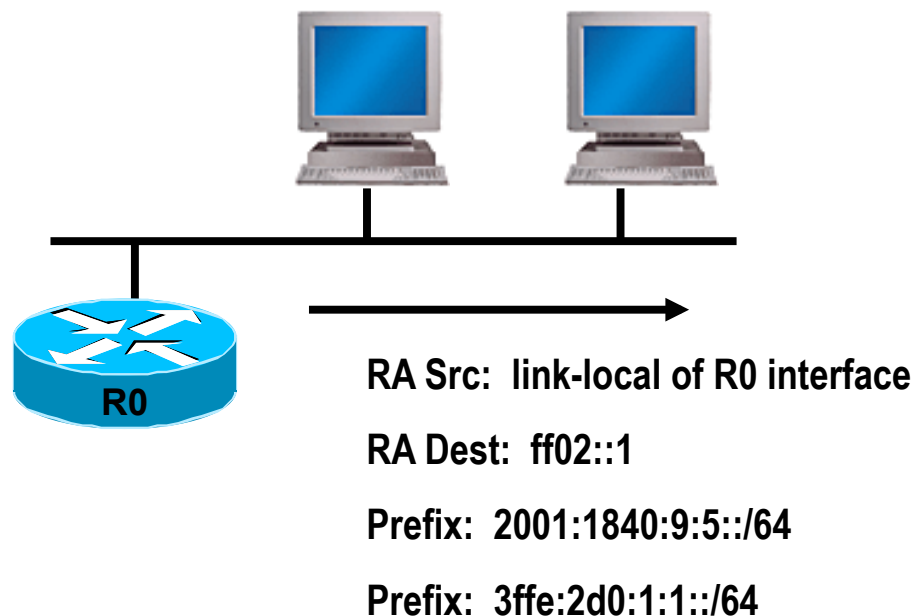
(Type: 128-255)

<i>ICMP Number</i>	<i>Message Type</i>
128	Echo request
129	Echo reply
130	Multicast group membership query
131	Multicast group membership report
132	Multicast group membership termination
133	Router Solicitation
134	Router Advertisement
135	Neighbor Solicitation
136	Neighbor Advertisement
137	Redirect
138	Router Renumbering
139	Node Information query
140	Node Information reply
141	Inverse Neighbor Solicitation
142	Inverse Neighbor Advertisement



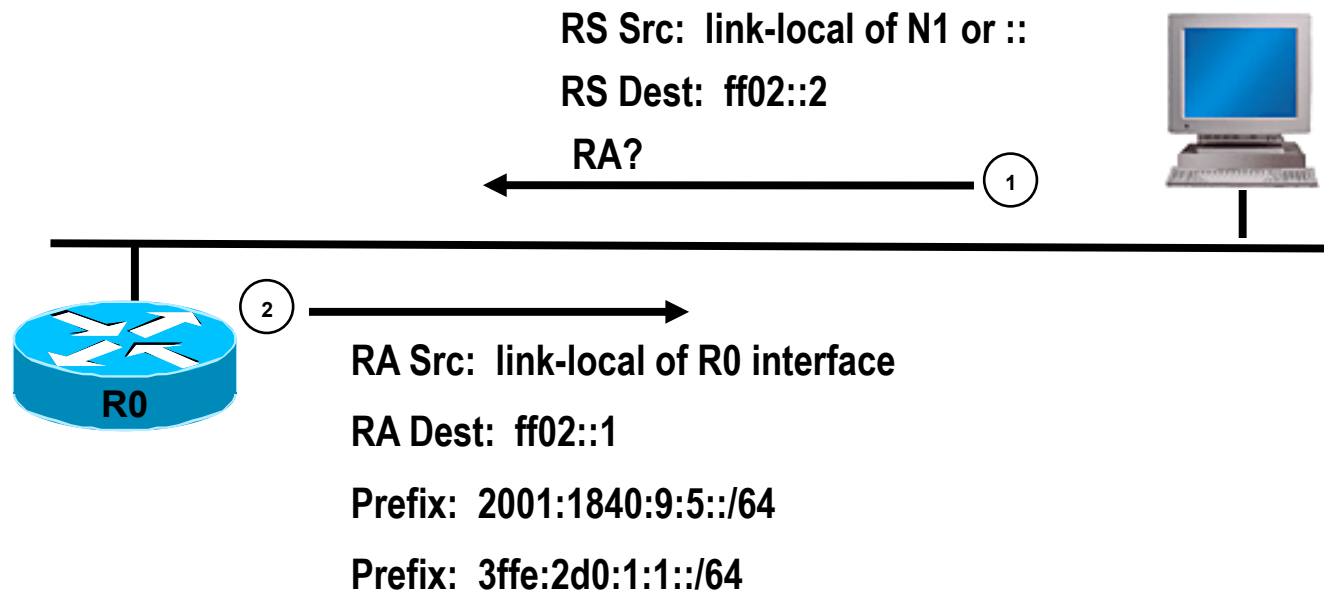
IPv6 Router Advertisement

- Sent periodically or in response to a Router Solicitation message
- Periodic RA's are sent to the all-nodes multicast address "ff02::1"
- RA messages contain information that inform the hosts about link information needed for auto-configuration



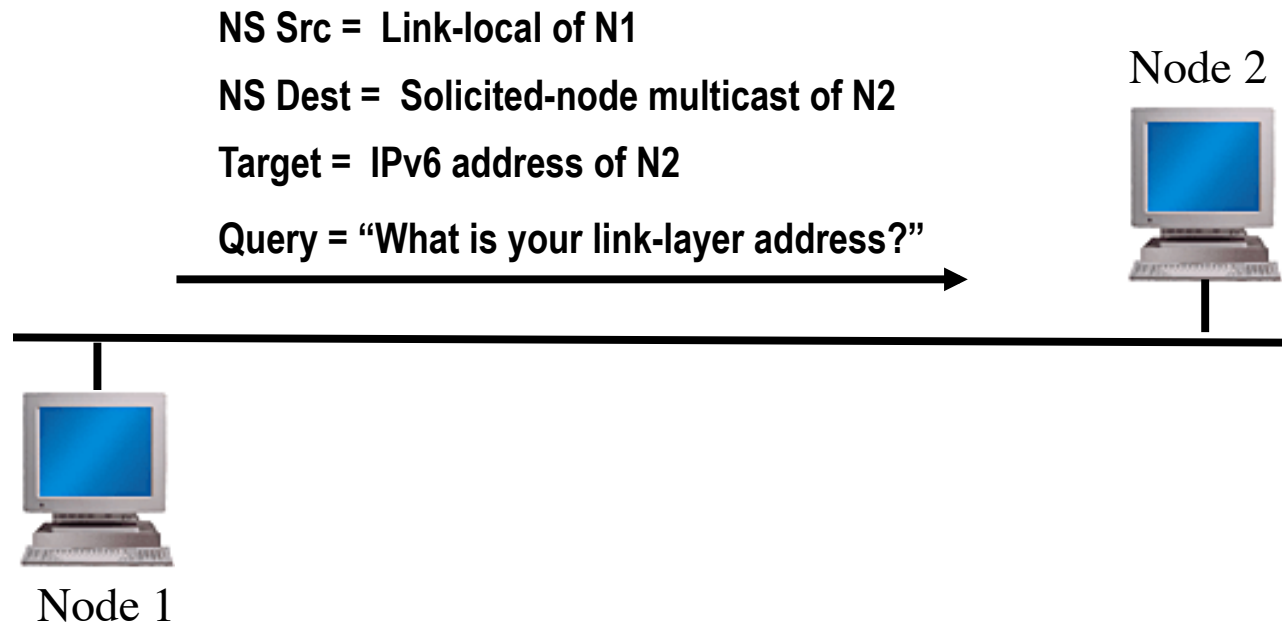
IPv6 Router Solicitation

- Sent at host start-up or to solicit a Router Advertisement immediately
- RS messages are usually sent to the all-routers multicast address “ff02::2”
- RS source address could be the link-local address of the sending node, or the unspecified “::” address



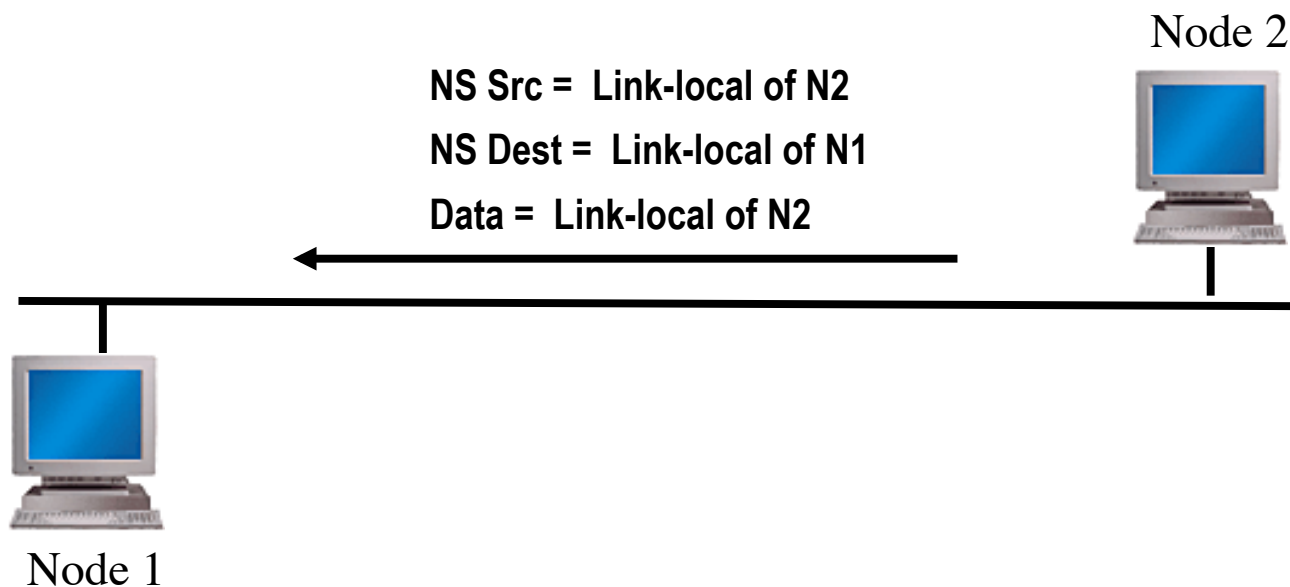
IPv6 Neighbor Solicitation

- Used by nodes for link-layer to IP-layer address resolution
- For link-layer address resolution, the solicited-node multicast address is used as the destination of the request (vs. broadcast in IPv4 ARP)
- Also used in the Duplicate Address Detection (DAD) and Neighbor Unreachability Detection (NUD) processes



IPv6 Neighbor Advertisement

- Sent in response to an NS or unsolicited to propagate new information
- Neighbor Advertisements contain:
 - Router flag: to indicate whether this neighbor is a router
 - Solicited flag: to indicate whether this NA is in response to a NS
 - Override flag: to indicate whether this information should override an existing neighbor cache entry
- NA's in response to an address resolution request are unicast to the solicitor



Architectural Design Choices for IPv6 Networks



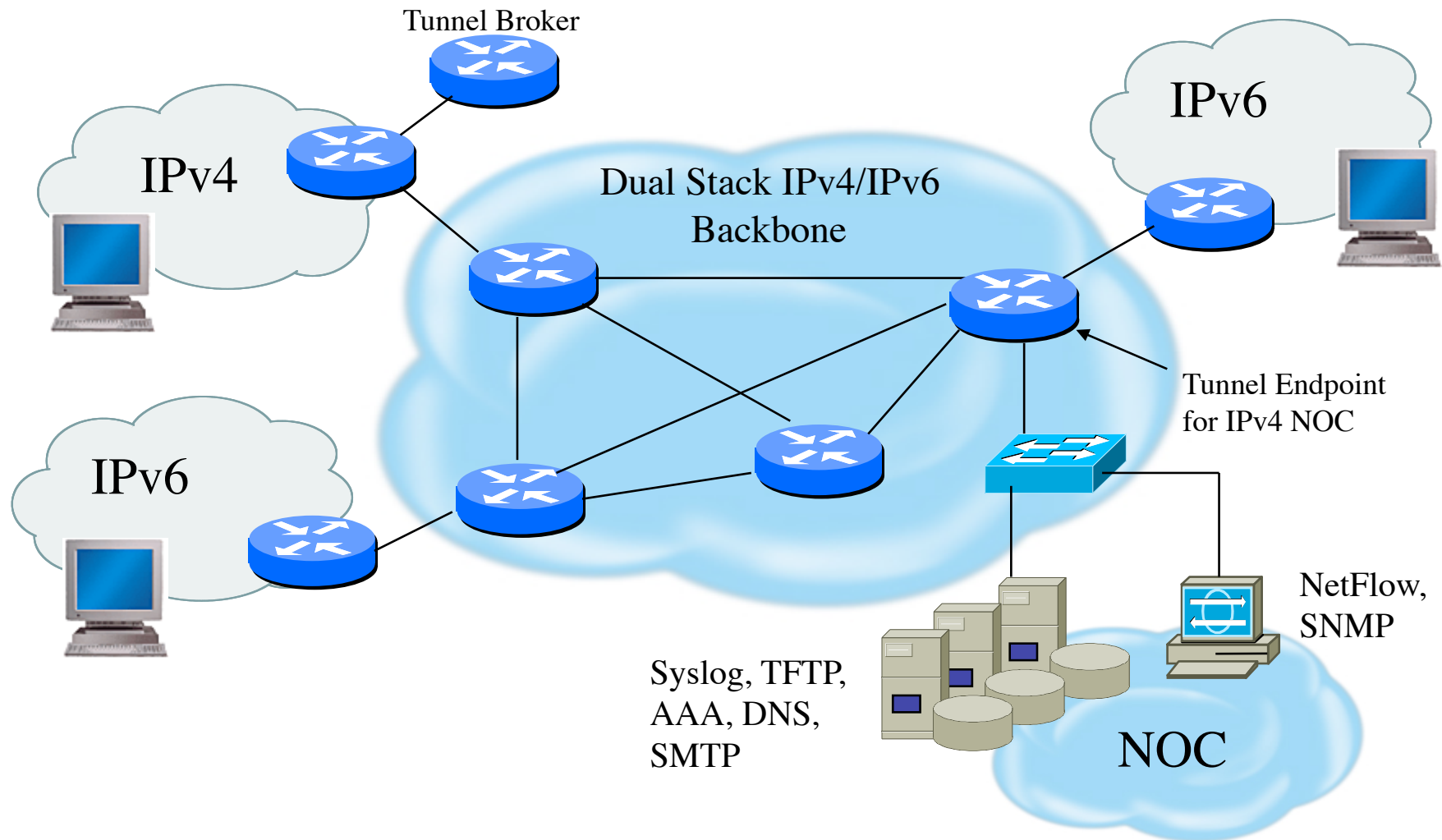
Security Workshop - APRICOT 14 Manila, Philippines - February 2009

What Is Same / What Is Different

- Same for IPv4 and IPv6
 - Security Properties
 - Security Services
- Different for IPv6 Architectures
 - Protocol Operation
 - More Automation
 - Scalable Mobile Hosts
 - Potential Application Integration



Sample IPv6 Infrastructure



What Needs To Be Considered

- Where is automation advantageous versus a security risk?
- How will IPv4 content be accessible?
 - Where is an address translation capability required?
- Where are network-based security mitigation techniques reliably advantageous versus a hindrance?
- What technologies need to be made easier to deploy to be operationally viable?
- What security services are being used to adhere to security policy requirements but are instantiations of IPv4 architecture limitations?



IPv6 Automation

- Protocol Capabilities
 - Neighbor Discovery allows nodes to easily find one another
 - Router Advertisements enable nodes to automatically create their own globally reachable IPv6 address
- Security Issues
 - Redirect attacks
 - Denial-of-Service attacks
 - Neighbor solicitation spoofing
 - Neighbor advertisement spoofing
 - Neighbor Unreachability Detection failure
 - Duplicate Address Detection DoS attack



Architecture Considerations

- Addressing / Naming
 - What subnet boundaries make sense
 - your own network infrastructure
 - filtering considerations
 - Endpoint Identifier management
 - address automation vs obscurity vs auditability
 - DNS and DHCPv6 Considerations
- Native Routing vs Tunnels
- Management
- Security



SeND Capabilities

- SeND protects against:
 - Spoofed Messages To Create False Entries In Neighbor Cache
 - Neighbor Unreachability Detection Failure
 - Duplicate Address Detection DoS Attack
 - Router Solicitation and Advertisement Attacks
 - Replay Attacks
 - Neighbor Discovery DoS Attacks
- SeND does NOT:
 - Protect statically configured addresses
 - Protect addresses configured using fixed identifiers (I.e.EUI-64)
 - Provide confidentiality
 - Compensate for unsecured link-layer
 - No guarantee that payload packets came from node that used SEND



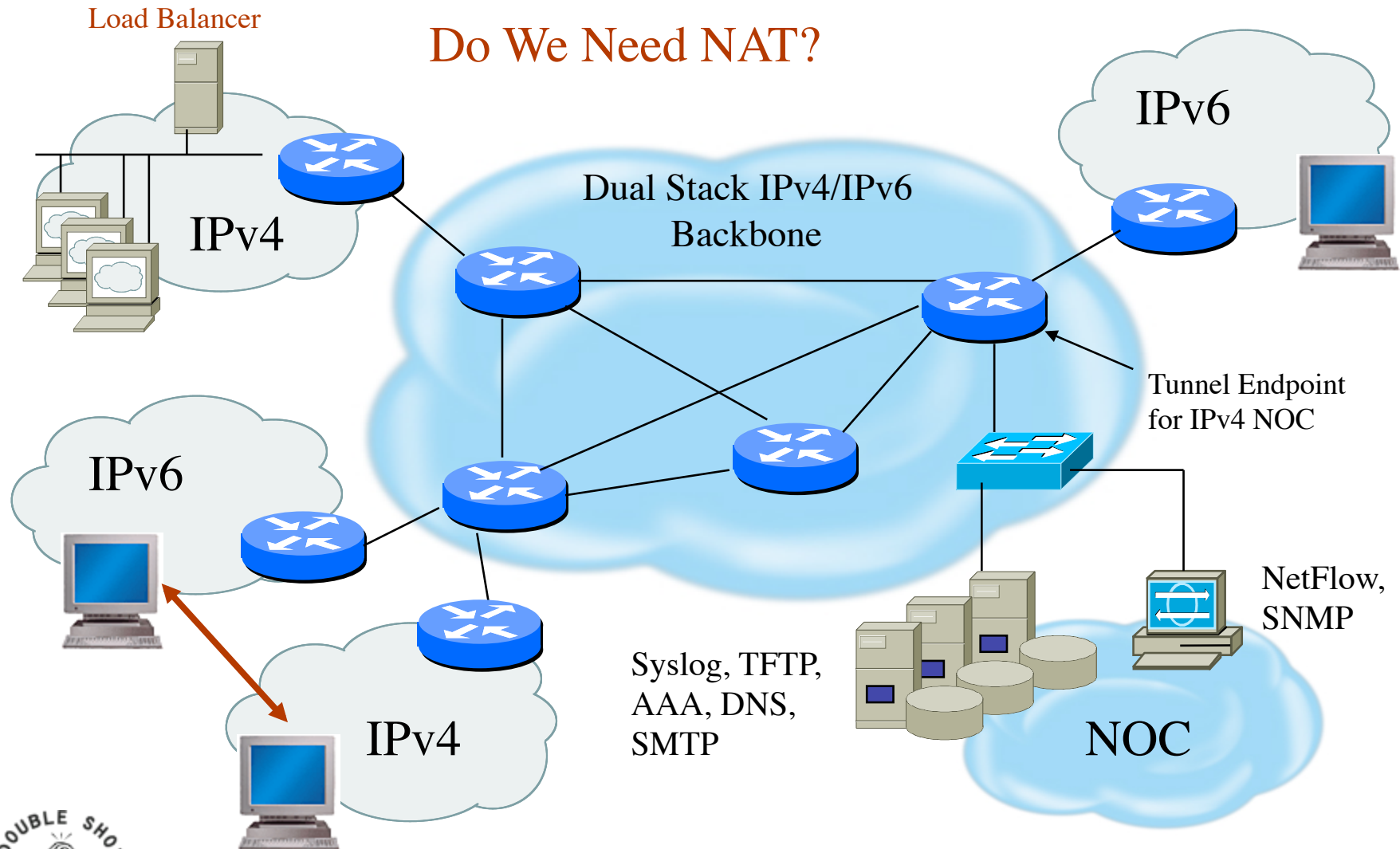
Tunneling Considerations

- Manually configured tunnels are not scalable
- Automated tunnels require more diligence to provide effective security services
- Deployments of 6to4, ISATAP and Teredo all require layered security models
 - Perform ingress firewall sanity checks
 - Log and audit tunneled traffic
 - Provide authentication where possible
 - Use IPsec where appropriate



Network Address Translation

Do We Need NAT?



IPv6 Security Enhancements

- Fragmentation
 - Prohibited by intermediary devices
 - Overlapping fragments are not allowed
 - Devices must drop reassembled packets that are less than 1280 bytes
- Broadcasts
 - Removes concept of dedicated broadcasts
 - Specific language to avoid ICMPv6 broadcast amplification attacks
- IPsec
 - Defined into the base protocol spec



IPv6 Device Security



Security Workshop - APRICOT 14 Manila, Philippines - February 2009

End Host IPv6 Security Guidelines

- Basic Principles
 - Address assignment is performed in a reliable manner and cannot be spoofed
 - Traffic sourced from or destined to an end-host can be protected from modification, deletion or spoofing
 - Malicious behavior can be detected and mitigated
- Addressing recommendations
 - Use stateless auto-configuration when low probability that spoofing can occur
 - Use DHCPv6 if need to have control over addresses
 - Use standard but non-obvious static addresses for critical systems
- Hardening the host
 - Restrict access to the client or server to authenticated and authorized individuals
 - Monitor and audit access to the client and server
 - Turn off any unused services on the end node
 - Use host firewall capabilities to control traffic that gets processed by upper layer protocols
 - Use virus scanners to detect malicious programs
- Protecting traffic between hosts
 - Use IPsec



Device Physical Access

- Equipment kept in highly restrictive environments
- Console access
 - password protected
 - access via OOB management
- Individual users authenticated
- Social engineering training and awareness



Device Access Considerations

- Console Port
 - Access via cable connected to the serial port
 - Only access to password recovery functions
- Auxiliary Port
 - Generally used for out of band (OOB) access
 - Also used for connecting to other console ports
- Virtual TTY (VTY)
 - Default access is via 'telnet'
- HTTP
- TFTP
- SNMP

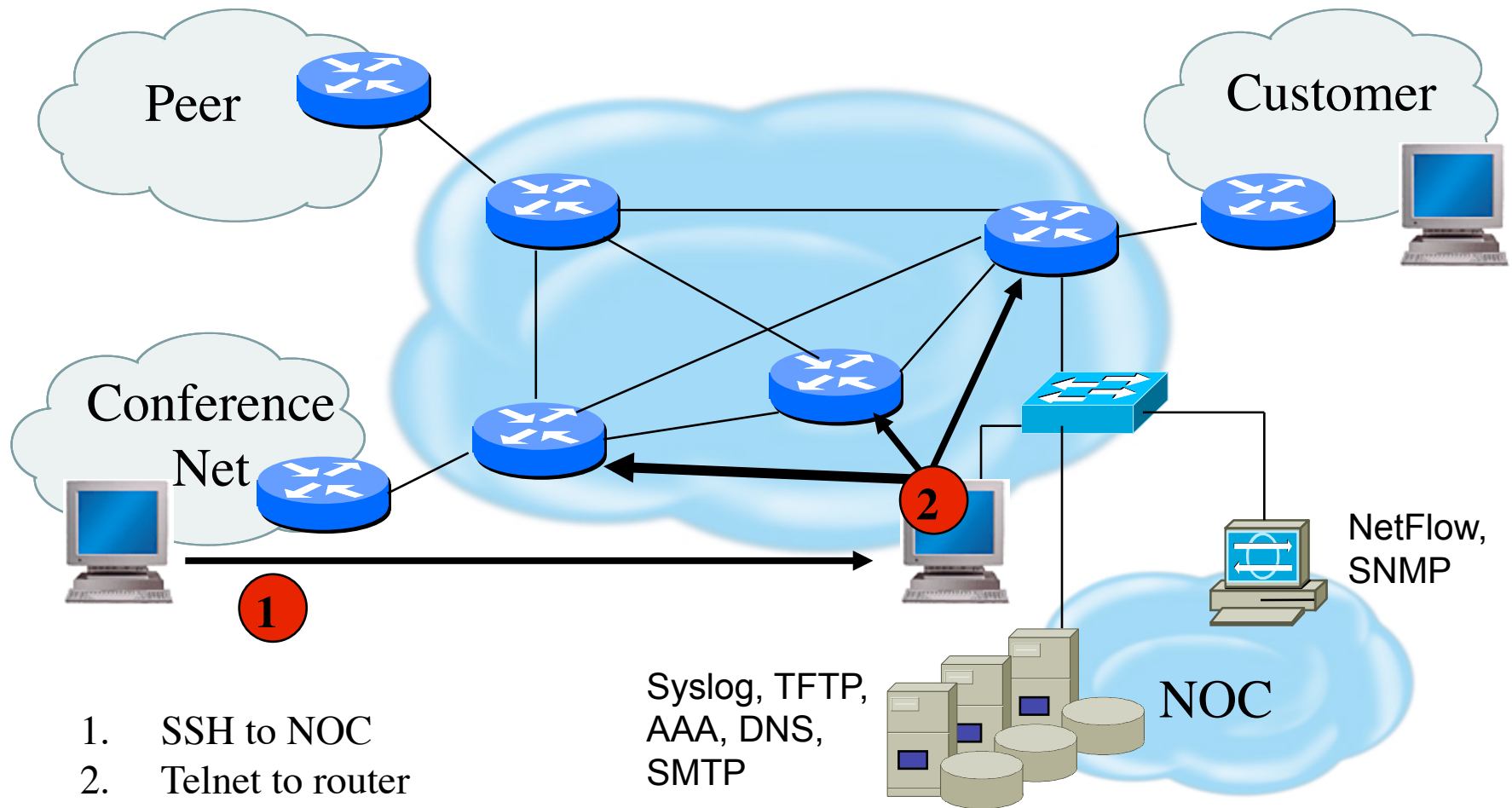


Management Plane Filters

- Authenticate Access
- Define Explicit Access To/From Management Stations
 - SNMP
 - Syslog
 - TFTP
 - NTP
 - AAA Protocols
 - DNS
 - SSH, Telnet, etc.



Telnet using SSH 'Jumphost'



Added Controls For SSH Access

Configure IPv6 vty-input access-list

```
ipv6 access-list vty-filter
```

```
permit host <ipv6 address> host <ipv6 address>
```

Apply vty-input access-list to vty 0 4

```
line vty 0 4
```

```
ipv6 access-class vty-filter in
```



Device Management Common Practice

- SSH primarily used; Telnet only from jumphosts
- HTTP access explicitly disabled
- All access authenticated
 - Varying password mechanisms
 - AAA usually used
 - Different servers for in-band vs OOB
 - Different servers for device authentication vs other
 - Static username pw or one-time pw
 - Single local database entry for backup
- Each individual has specific authorization
- Strict access control via filtering
- Access is audited with triggered pager/email notifications
- SNMP is read-only
 - Restricted to specific hosts
 - View restricted if capability exists
 - Community strings updated every 30-90 days

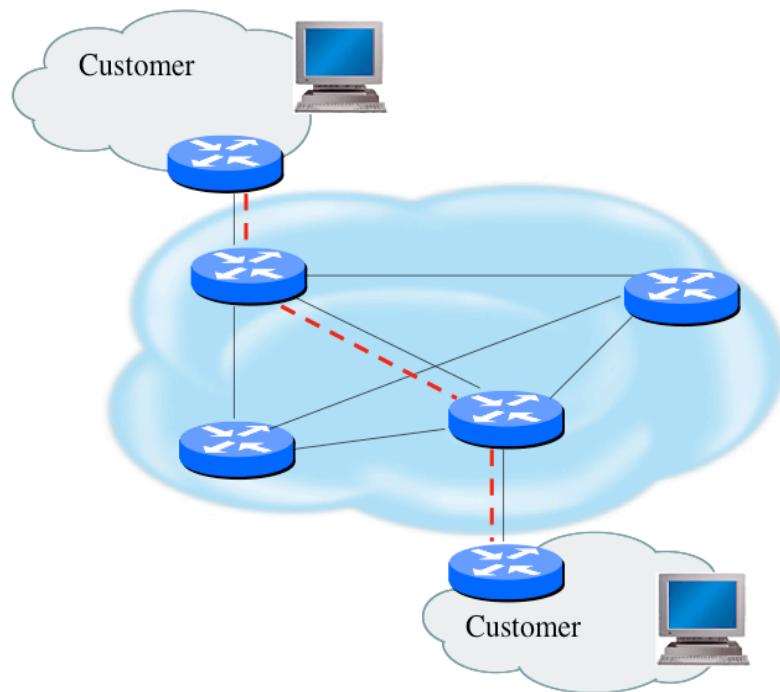


Securing the Data Plane for IPv6 Networks



Security Workshop - APRICOT 14 Manila, Philippines - February 2009

Securing The Data Path



- Filtering and rate limiting are primary mitigation techniques
- BCP-38 guidelines for ingress filtering
- Null-route and black-hole any detected malicious traffic
- Netflow is primary method used for tracking traffic flows
- Unicast Reverse Path Forwarding is not consistently implemented
- Logging of Exceptions



Data Plane (Packet) Filters

- Most common problems
 - Poorly-constructed filters
 - Ordering matters in some devices
- Scaling and maintainability issues with filters are commonplace
- Make your filters as modular and simple as possible
- Take into consideration alternate routes
 - Backdoor paths due to network failures



Filtering Deployment Considerations

- How does the filter load into the router?
- Does it interrupt packet flow?
- How many filters can be supported in hardware?
- How many filters can be supported in software?
- How does filter depth impact performance?
- How do multiple concurrent features affect performance?
- Do I need a standalone firewall?



Router Filter vs Standalone Firewall Tradeoffs

USING A ROUTER AS FIREWALL

- Increased CPU cycles and memory usage
- Single device to maintain

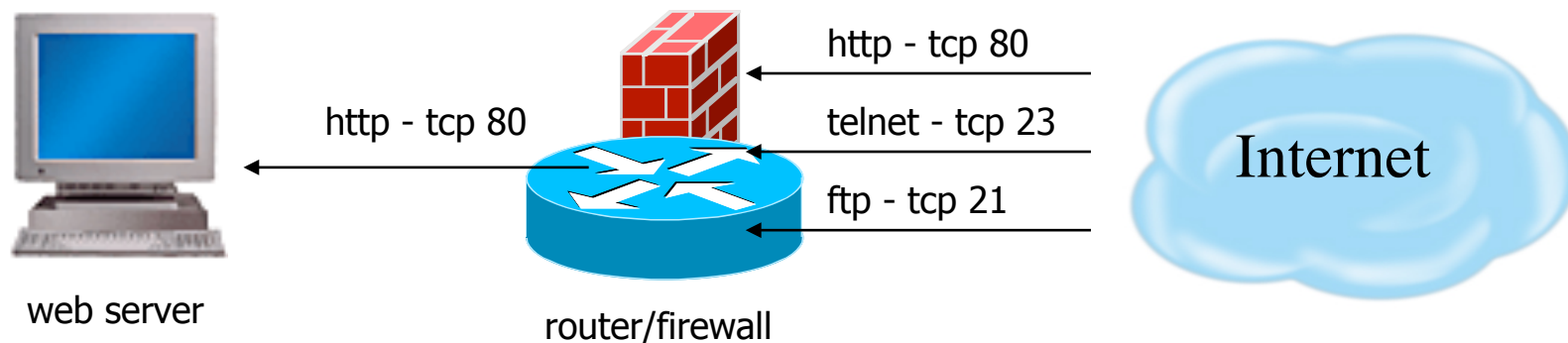
USING A STANDALONE FIREWALL

- Additional hardware cost and maintenance
- Additional software purchase and updates
- Administrative setup and training
- Offload resources used from router



Packet Filtering Firewall

- examines the source and destination address of the data packet and either allows or denies the packet from traveling the network
- blocks access through the firewall to any packets, which try to access ports which have been declared "off-limits"

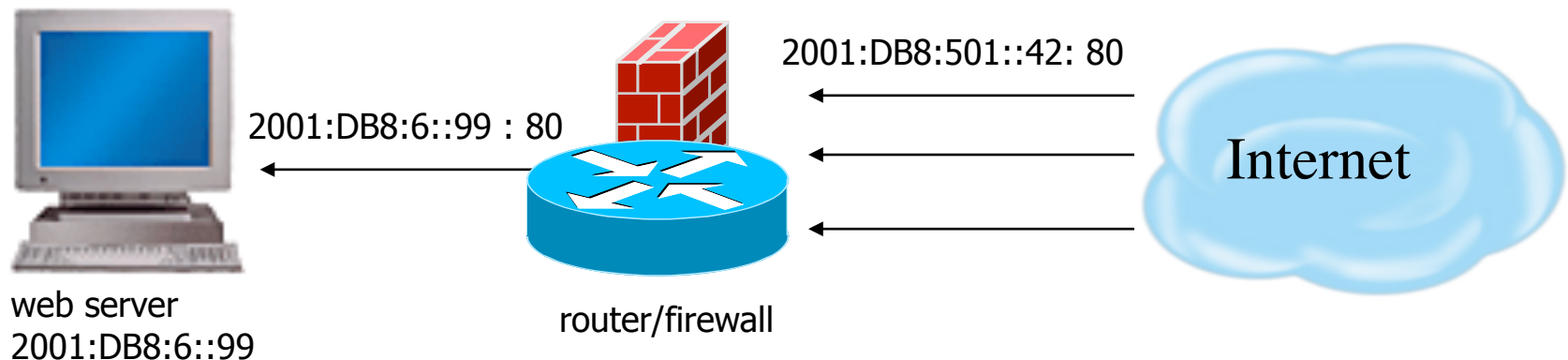


Allow only http - tcp 80
Drop anything else



Application Layer Firewall

- Also known proxy firewalls, application gateway
- attempts to hide the configuration of the network behind the firewall by acting on behalf of that network/servers
- All requests for access are translated at the firewall so that all packets are sent to and from the firewall, rather than from the hosts behind the firewall

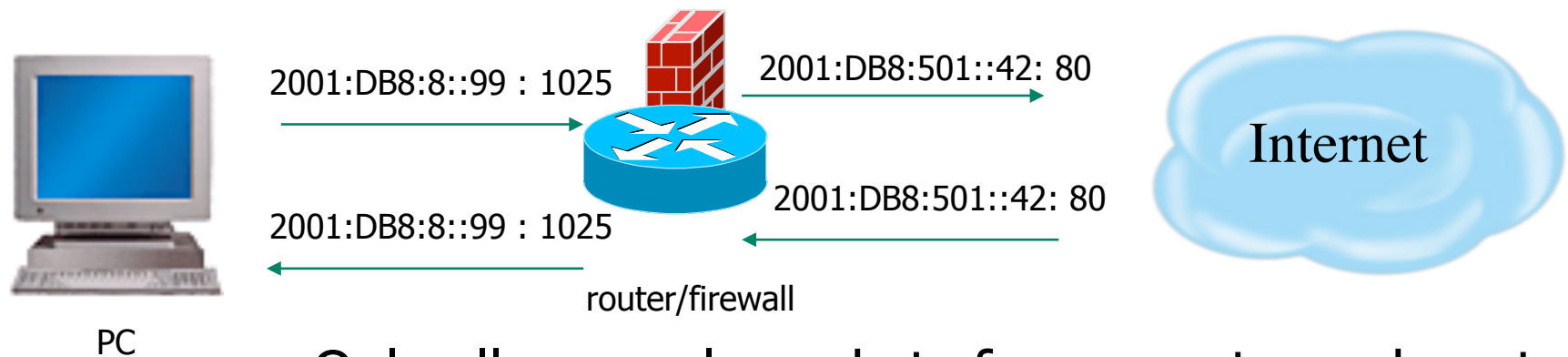


Translates 2001:DB8:501::42 : 80
To 2001:DB8:6::99 : 80



Stateful Inspection Firewall

- Examines the state and the context of the packets
- Remembers what outgoing requests have been sent and only allow responses to those requests back through the firewall
- Attempts to access the internal network that have not been requested by the internal network will be denied



Only allows reply packets for requests made out
Blocks other unregistered traffic



IPv6 Filtering Considerations

- IPv6 addressing architecture will simplify or complicate filters....carefully think about it.
- Routing filters are usually more optimal than packet filters but have less granularity
 - Routing filters affect the routes that are accepted and sent between routers and therefore forward or drop traffic based on reachability information
 - Packet filters are used to allow or deny data packets from being processed or forwarded by a device based on the IP header information.
- Best policy is to deploy filtering mechanisms that will drop any unwanted traffic as close to source as possible



General Firewall BCP

- Explicitly deny all traffic and only allow what you need
- The default policy should be that if the firewall doesn't know what to do with the packet, deny/drop it
- Don't rely only on your firewall for all protection of your network
- Implement multiple layers of network protection
- Make sure all of the network traffic passes through the firewall
- Log all firewall exceptions (if possible)



Ingress Packet Filters To Consider

- Accept all ICMPv6 packets for Neighbor Discovery and Path MTU Discovery that is a function necessary for the communication with IPv6
- Reject the packets which contain relevant special-use prefix in the **source** address field
 - ::1/128 : loop back address
 - ::/128 : unspecified address
 - ::/96 : IETF reserved address;IPv4-compatible IPv6 address
 - ::ffff:0:0/96 : IPv4-mapped IPv6 address
 - ::/8 : reserved
 - fc00::/7 : unique-local address
 - ff00::/8 : multicast address
 - 2001:db8::/3 : documentation addresses



Ingress Packet Filters To Consider(cont.)

- Reject the packets which contain relevant special-use prefix in the **destination** address field
 - `::1/128` : loop back address
 - `::/128` : unspecified address
 - `::/96` : IETF reserved address;IPv4-compatible IPv6 address
 - `::ffff:0:0/96` : IPv4-mapped IPv6 address
 - `::/8` : reserved
 - `fc00::/7` : unique-local [fc00::/16] and site-local [fc00::/10] address
 - `2001:db8::/32` : documentation address
- Reject the packets which have your own prefix in the source address field
- Reject packets that use the routing header Care must be taken not to reject ICMPv6 packets whose source address used with Duplicate Address Detection is the unspecified address (`::/128`). If all of ICMPv6 is accepted, then there is no problem although ordering of the filters needs to be carefully thought through.



Egress Packet Filters To Consider

- Permit sending all ICMPv6 packets for Neighbor Discovery and Path MTU Discovery that is a function necessary for the communication with IPv6
- Deny sending the packets which contain special-use prefix in the source address field
 - `::1/128` : loop back address
 - `::/128` : unspecified address
 - `::/96` : IETF reserved address; IPv4-compatible IPv6 address
 - `::ffff:0:0/96` : IPv4-mapped IPv6 address
 - `::/8` : reserved
 - `fc00::/7` : unique-local address
 - `ff00::/8` : multicast address
 - `2001:db8::/32` : documentation address
- Deny sending packets that use the routing header [unless using mobility features]
- Deny sending packets with destination address in the 6to4 reserved address range (`2002::/16`) if not supporting 6to4 services (i.e. relays) and not providing transit services
- Deny sending packets with destination address in the Teredo address range (`2001::/32`) if not running a Teredo relay or offering a Teredo transit service
- Multicast address should only be in source address field.



Allow Following ICMPv6 Through A Firewall

- ICMPv6 type 1 code 0: no route to destination
- ICMPv6 type 2: packet too big (required for PMTUD)
- ICMPv6 type 3: time exceeded
- ICMPv6 type 4: parameter problem (informational when IPv6 node has problem identifying a field in the IPv6 header or in an extension header)
- ICMPv6 type 128: echo request
- ICMPv6 type 129: echo reply



Allow Following ICMPv6 To/From A Firewall

- ICMPv6 type 2: packet too big – firewall device is not allowed to fragment IPv6 packets going through it and must be able to generate this message for correct PMTUD behavior
- ICMPv6 type 4: parameter problem
- ICMPv6 type 130-132: multicast listener messages – in IPv6 a routing device must accept these messages to participate in multicast routing
- ICMPv6 type 133-134: router solicitation and advertisement – needed for IPv6 autoconfiguration
- ICMPv6 type 135-136: neighbor solicitation and advertisement – used for duplicate address detection and layer2-to-IPv6 address resolution



Routing Header: RFC 2460 Text

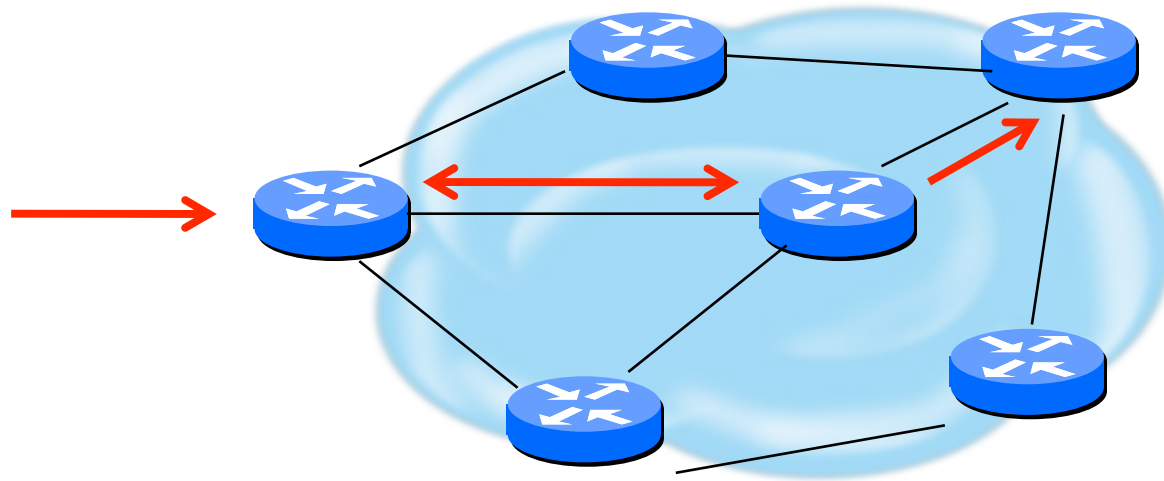
- The routing header is used by an IPv6 source to list one or more intermediate **nodes** to be “visited” on the way to packet’s destination.
- Each extension header should occur at most once, except for the destination options header which should occur at most twice.
- IPv6 nodes must accept and attempt to process extension headers ***in any order*** and ***occurring any number of times*** in the same packet.



Routing Header Issue

A single RH of Type 0 may contain multiple intermediate node addresses, and the same address may be included more than once in the same RH0.

If the routing header contains a repetition of a pair of addresses of the form A B A B A B ... If this A B pair were repeated 3 times then a single packet directed at A would traverse the path A B 3 times, and B A twice. If such packets were generated at a total rate of 1 Mbps then the path between A and B would experience a total of 5Mbps of traffic.



Routing Header Processing

- Disabling processing still allows all other hosts to be used for attack
- Dropping is required for ISP's
- RFC 5095 – Deprecation of RH0
- Until rfc5095 implemented:
 - Use ingress filtering for RH0 traffic
 - RH Type 2 is required for mobility so have to ensure that only RH0 traffic is blocked



RFC2827 (BCP38) – Ingress Filtering

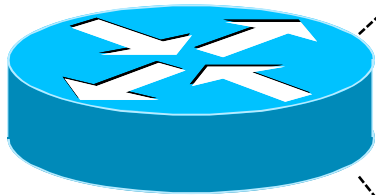
If an ISP is aggregating routing announcements for multiple downstream networks, strict traffic filtering should be used to prohibit traffic which claims to have originated from outside of these aggregated announcements.

The ONLY valid source IP address for packets originating from a customer network is the one assigned by the ISP (whether statically or dynamically assigned).

An edge router could check every packet on ingress to ensure the user is not spoofing the source address on the packets which he is originating.



Example Outgoing Packet Filter



```
ipv6 access-list extended DSL-ipv6-Outbound  
permit ipv6 2001:DB8:AA65::/48 any  
deny  ipv6 any any log
```

```
interface atm 0/0  
  ipv6 traffic-filter DSL-ipv6_Outbound out
```



Cisco Filters (Access-Lists)

- IPv6 access-lists (ACL) are used to filter traffic and restrict access to the router
- IPv6 extended access lists add support for option header and upper layer filtering
- Cisco specific filtering characteristics
 - A reference to an empty ACL will permit any any
 - Implicit permit rule for neighbor discovery
 - Implicit deny any any as final rule in each ACL



Configuring Cisco IPv6 ACLs

- Creating the IPv6 ACL

```
[no] ipv6 access-list <name>
```

- Defining the IPv6 ACL entry

```
[no] permit | deny ipv6 | <protocol> any | host <src> | src/len [sport] any | host <dest> |  
dest/len [dport] [reflect <name> [timeout <secs>]] [fragments] [routing] [dscp <val>]  
[flow-label <val>] [time-range <name>] [log | log-input] [sequence <num>]
```

- Applying an ACL to an interface

```
interface s0/0  
ipv6 traffic-filter ipv6_in in  
ipv6 traffic-filter ipv6_out out
```

- Restricting access to the router

```
line vty 0 4  
ipv6 access-class vty-filter in
```



Monitoring Cisco IPv6 ACLs

- Show the IPv6 ACL configuration

```
show ipv6 access-list [name]
```

- Clearing the IPv6 ACL match count

```
clear ipv6 access-list [name]
```

As with any filter configuration, ordering is important since all entries are checked sequentially. Ensure that most frequent 'hits' are on top of the list.



Cisco and RH0 Filtering

- To disable processing of all types routing headers on 12.2(15)T and up one can use:

```
no ipv6 source-route
```

Note that this will still forward these packets on to other hosts which can be vulnerable. This statement also affects perfectly valid Routing Headers of Type 2 which are used by Mobile IPv6.

- If possible upgrade to 12.4(2)T or higher and block only the Type 0 Routing Header (note interface specific config):

```
Router(config)#ipv6 access-list deny-sourcerouted
```

```
Router(config-ipv6-acl)#deny ipv6 any any routing-type 0
```

```
Router(config-ipv6-acl)#permit ipv6 any any
```

```
Router(config)#interface Ethernet0
```

```
Router(config-if)#ipv6 source-route
```

```
Router(config-if)#ipv6 traffic-filter deny-sourcerouted in
```



Cisco IPv6 NetFlow

- Netflow IPv6 support from 12.4 IOS releases
- Uses Netflow v9
- Activate per interface

ipv6 flow ingress

ipv6 flow egress

- Show status

show ipv6 flow cache



IPv6 Filtering References

- RFC 4890 'Recommendations for Filtering ICMPv6 Messages in Firewalls'
- RFC 5156 'Special-Use IPv6 Addresses'
- <http://www.space.net/~gert/RIPE/ipv6-filters.html>
- <http://www.cymru.com/Bogons/v6top.html>
- NSA Router Security Configuration Guide Supplement – Security for IPv6 Routers

Many filtering recommendations are not uniform and that while similarities exist, a definitive list of what to deny and what to permit does not exist. Any environment will need to determine what is most suitable for them by using these references as guidelines.



Securing The IPv6 Routing Infrastructure



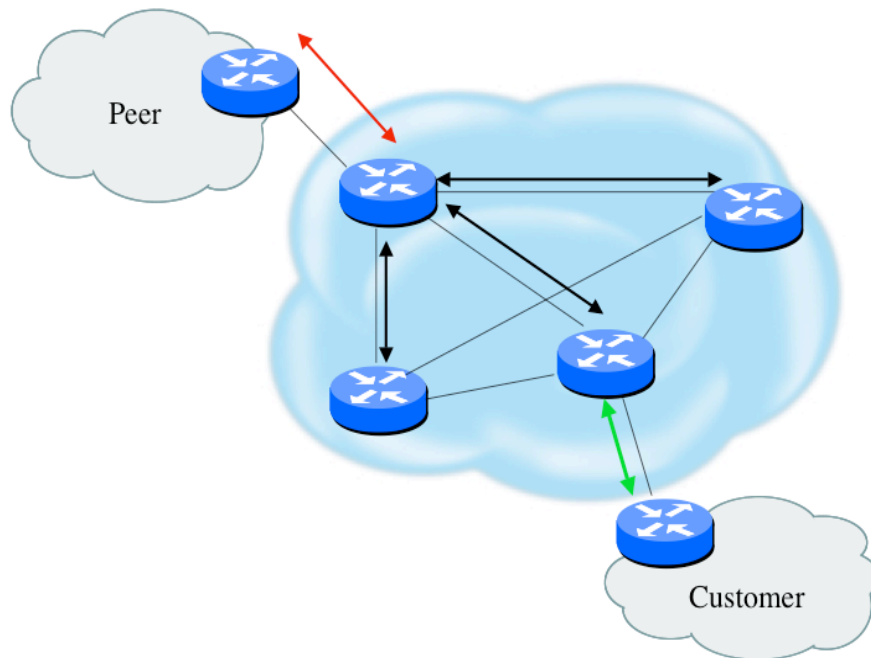
Security Workshop - APRICOT 14 Manila, Philippines - February 2009

Router Security Considerations

- Segment areas for route redistribution and ensure limited access to routers in critical backbone areas
- Design networks so outages don't affect entire network but only portions of it
- Control router access....watch against internal attacks on these systems. Use different passwords for router enable and monitoring system root access.
- Scanning craze for all kinds of ports – this will be never ending battle



Routing Control Plane



- MD-5 authentication
 - Some deploy at customer's request
- Route filters limit what routes are believed from a valid peer
- Packet filters limit which systems can appear as a valid peer
- Limiting propagation of invalid routing information
 - Prefix filters
 - AS-PATH filters (trend is leaning towards this)
 - Route dampening (latest consensus is that it causes more harm than good)
- Not yet possible to validate whether legitimate peer has authority to send routing update



Why Use Route Authentication

- Route Authentication equates to data origin authentication and data integrity
- In BGP, requires TCP resets to be authenticated so malicious person can't randomly send TCP resets
- In cases where routing information traverses shared networks, someone might be able to alter a packet or send a duplicate packet
- Routing protocols were not initially created with security in mind.....this needs to change....



Control Plane (Routing) Filters

- Filter traffic destined TO your core routers
- Develop list of required protocols that are sourced from outside your AS and access core routers
 - Example: eBGP peering, GRE, IPSec, etc.
 - Use classification filters as required
- Identify core address block(s)
 - This is the protected address space
 - Summarization is critical for simpler and shorter filter lists



BGP Prefix Filtering

- All BGP Prefixes coming into your network and leaving your network need to be filtered to enforce a policy.
- The problem is most ISPs are not:
 - Filtering Comprehensively
 - Filtering their customer's prefixes
 - Filtering prefixes going out of their network.

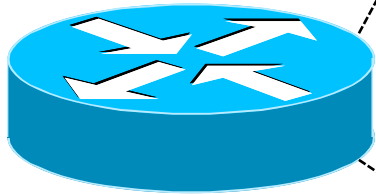


BGP Prefix Filters To Consider

- Special-use prefixes
 - `::/0` exact : default route
 - `::1/128` : loop back address
 - `::/128` : unspecified address
 - `::/96` : IPv4-compatible IPv6 address
 - `::ffff:0:0/96` : IPv4-mapped IPv6 address
 - `::/8` or longer : reserved
 - `fe80::/10` or longer : link-local address
 - `fc00::/7` or longer : unique-local address
 - `ff00::/8` or longer : multicast range (RFC3513)
 - `fe00::/9` or longer : multicast range (RFC3513)
 - `2001:db8::/32` or longer : documentation address
- Your own prefix
- The 6bone prefix (`3ffe::/16`)
- The 6to4 reserved address range (`2002::/16`) if not supporting 6to4 services (i.e. relays) and not providing transit services
- The Teredo address range (`2001::/32`) if not running a Teredo relay or offering a Teredo transit service



BGP Simple Bogon Prefix Filter Example



```
ipv6 prefix-list ipv6-special-use-pfx deny 0::/0 le 128
ipv6 prefix-list ipv6-special-use-pfx deny 0::1/128 le 128
ipv6 prefix-list ipv6-special-use-pfx deny 0::/128
ipv6 prefix-list ipv6-special-use-pfx deny 0::/96
ipv6 prefix-list ipv6-special-use-pfx deny 0::ffff:0:0/96
ipv6 prefix-list ipv6-special-use-pfx deny 0::/8 le 128
ipv6 prefix-list ipv6-special-use-pfx deny fe80::/10 le 128
ipv6 prefix-list ipv6-special-use-pfx deny fc00::/7 le 128
ipv6 prefix-list ipv6-special-use-pfx deny fe00::/9 le 128
ipv6 prefix-list ipv6-special-use-pfx deny ff00::/8 le 128
ipv6 prefix-list ipv6-special-use-pfx deny 2001:db8::/32 le 128
ipv6 prefix-list ipv6-special-use-pfx deny 3ffe::/16 le 128
```

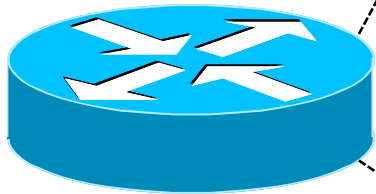


BGP Prefix Filters (RIR Allocations)

- APNIC
 - `ftp://ftp.apnic.net/stats/apnic/delegated-apnic-latest`
- RIPE NCC
 - `ftp://ftp.ripe.net/pub/stats/ripencc/delegated-ripencc-latest`
- ARIN
 - `ftp://ftp.arin.net/pub/stats/arin/delegated-arin-latest`
- LACNIC
 - `ftp://ftp.lacnic.net/pub/stats/lacnic/delegated-lacnic-latest`
- AfriNIC
 - `ftp://ftp.afrinic.net/pub/stats/afrinic/delegated-afrinic-latest`



BGP RIR Allocation Prefix Filter Example (Needs Constant Updating)



```
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2001:0500::/30 ge 48 le 48
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2001:0678::/29 ge 48 le 48
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2001::/16 ge 35 le 35
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2001::/16 ge 19 le 32
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2003::/18 ge 19 le 32
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2400::/12 ge 13 le 32
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2600::/12 ge 13 le 32
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2610::/23 ge 24 le 32
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2620::/23 ge 40 le 48
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2800::/12 ge 13 le 32
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2A00::/12 ge 13 le 32
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2C00::/12 ge 13 le 32
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2001:0DF0::/29 ge 40 le 48
ipv6 prefix-list ipv6-RIR-allocations-pfx permit 2001:43F8::/29 ge 40 le 48
```



Prefix Filter Bogons and RIR Blocks

- Templates available from the Bogon Project:
 - <http://www.cymru.com/Bogons/index.html>
- Cisco Template
 - <ftp://ftp-eng.cisco.com/cons/isp/security/Ingress-Prefix-Filter-Templates/>
- Juniper Template
 - <http://www.qorbit.net/documents.html>



Other BGP Security Techniques

- BGP Community Filtering
- MD5 Keys on the eBGP and iBGP Peers
- Max Prefix Limits
- Prefer Customer Routes over Peer Routes (RFC 1998)
- GTSM (i.e. TTL Hack)



Audit and Validate Your Routing Infrastructures

- Are appropriate paths used? [Tunneled 6to4 causes some 'interesting' routing]
 - check routing tables
 - verify configurations
- Is router compromised?
 - check access logs



Routing Security Conclusions

- Current routing protocols do not have adequate security controls
- Mitigate risks by using a combination of techniques to limit access and authenticate data
- Be vigilant in auditing and monitoring your network infrastructure
- Consider MD5 authentication
- Always filter routing updates....especially be careful of redistribution



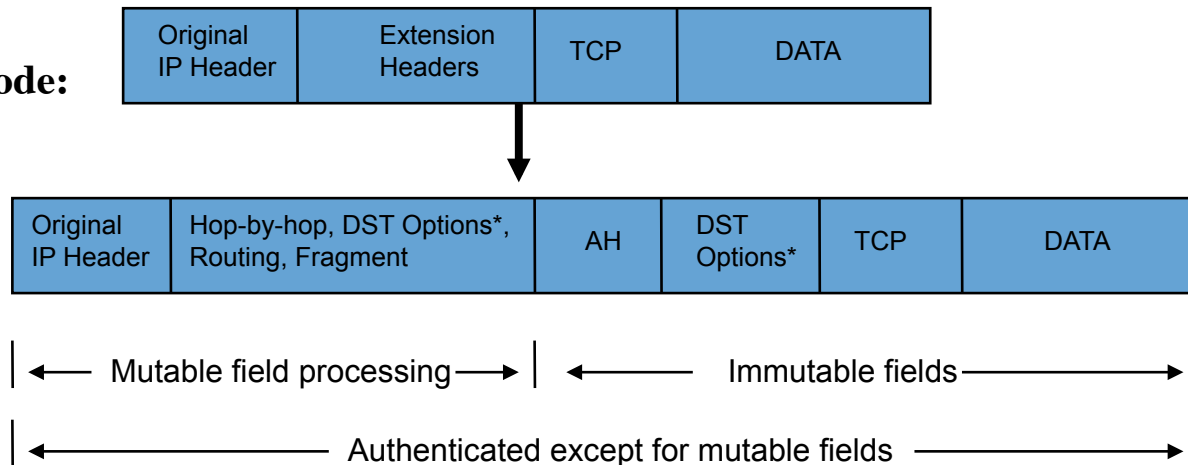
Where Does IPsec Fit In?



Security Workshop - APRICOT 14 Manila, Philippines - February 2009

IPv6 IPsec AH

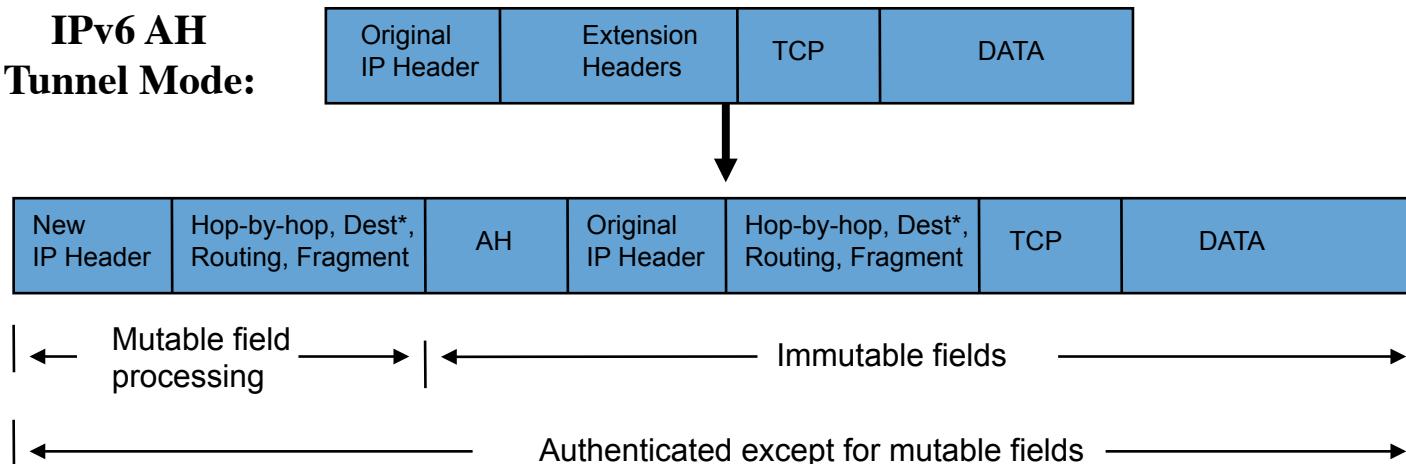
IPv6 AH Transport Mode:



Mutable Fields:

- DSCP
- ECN
- Flow Label
- Hop Limit

IPv6 AH Tunnel Mode:



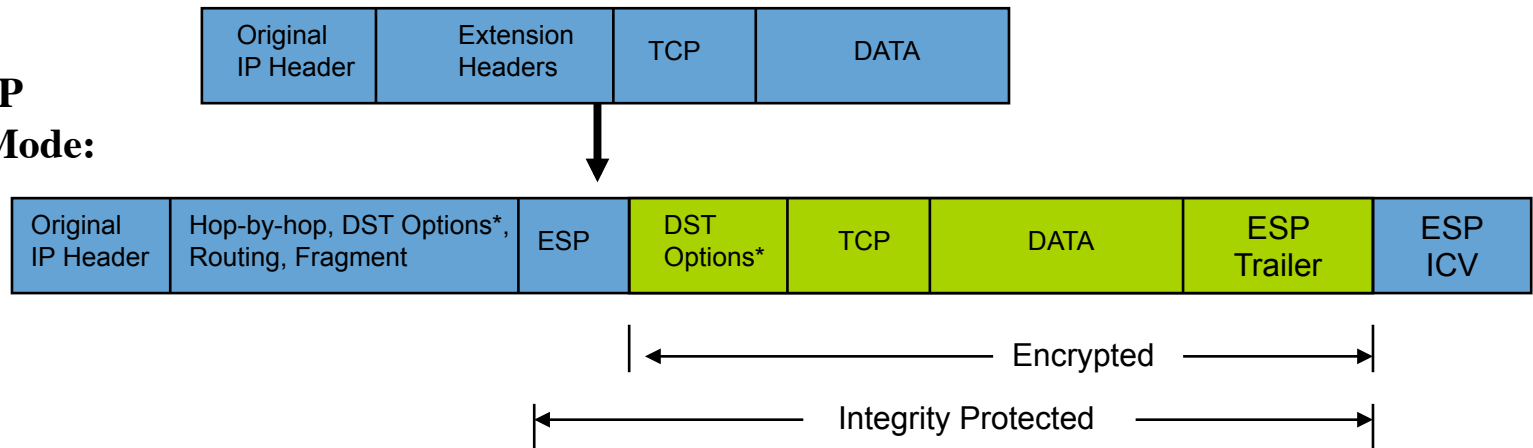
Mutable Fields:

- DSCP
- ECN
- Flow Label
- Hop Limit

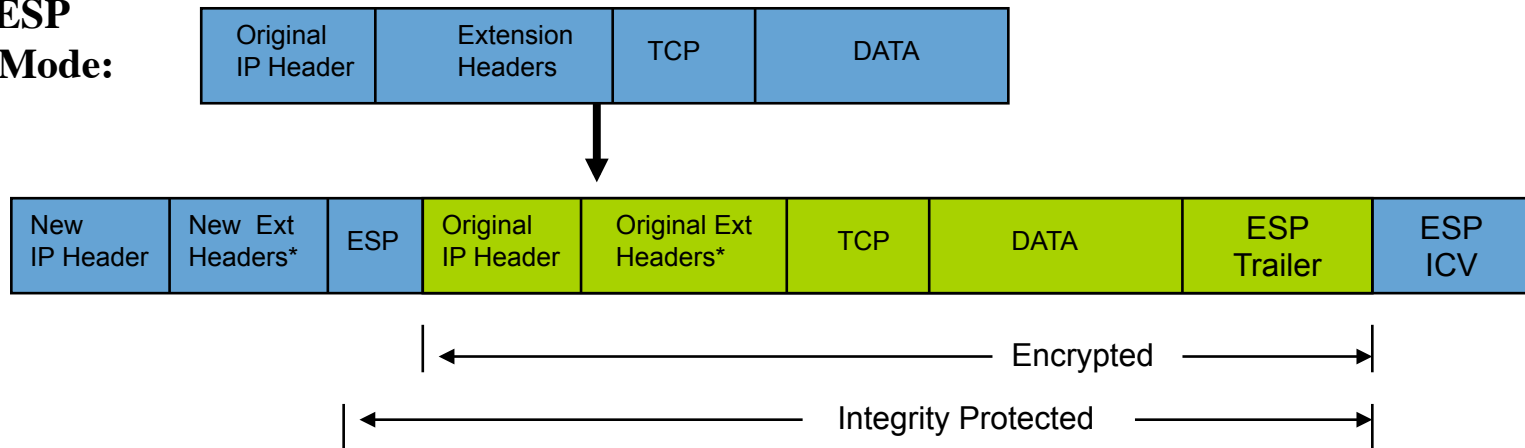


IPv6 IPsec ESP

IPv6 ESP Transport Mode:



IPv6 ESP Tunnel Mode:

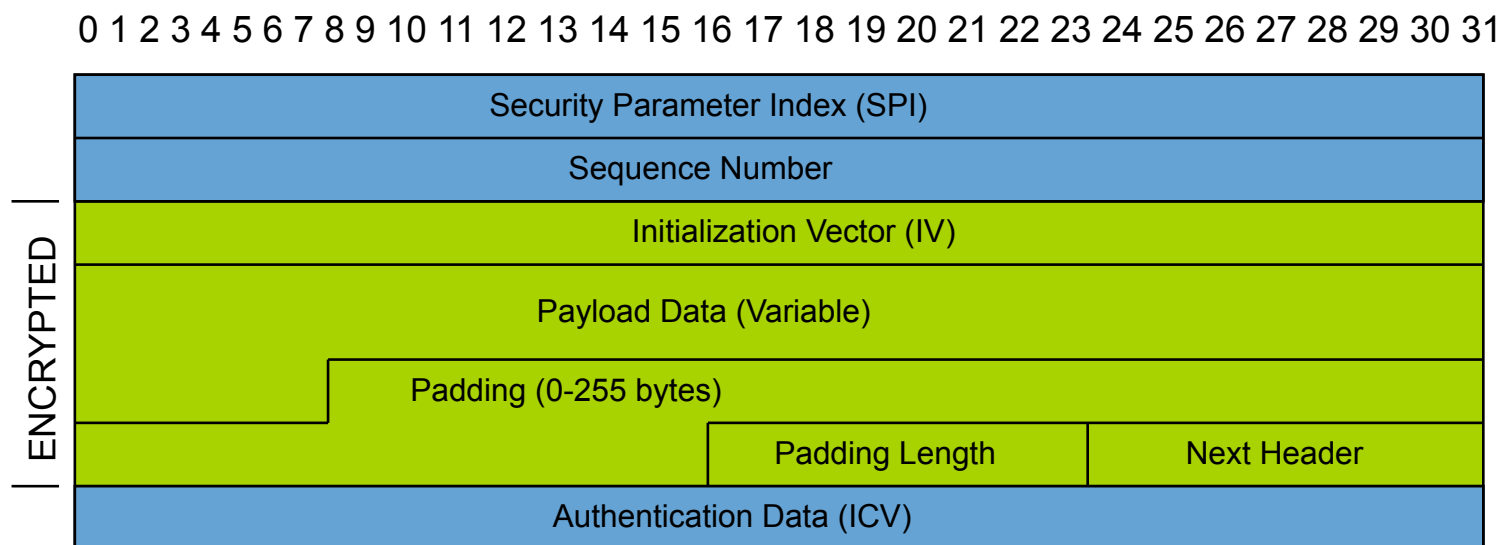


IPv6 Enhancements Needed

- Standards Modifications
 - Need to take into consideration Stateless Autoconfiguration where Router Advertisement sends network prefix
 - Need to be able to differentiate between encrypted versus integrity protected traffic
- Usability
 - Interoperable defaults
 - Consistent terminology



ESP Header Format



- SPI:** Arbitrary 32-bit number that specifies SA to the receiving device
- Seq #:** Start at 1 and must never repeat; receiver may choose to ignore
- IV:** Used to initialize CBC mode of an encryption algorithm
- Payload Data:** Encrypted IP header, TCP or UDP header and data
- Padding:** Used for encryption algorithms which operate in CBC mode
- Padding Length:** Number of bytes added to the data stream (may be 0)
- Next Header:** The type of protocol from the original header which appears in the encrypted part of the packet
- Auth Data:** ICV is a digital signature over the packet and it varies in length depending on the algorithm used (SHA-1, MD5)



Vendor Specific Deployment Issues

- Lack of interoperable defaults
 - A default does NOT mandate a specific security policy
 - Defaults can be modified by end users
- Configuration complexity
 - Too many knobs
 - Vendor-specific terminology
- Good News: IPv6 support in most current implementations



Default Issues

Vendor A

IKE Phase 1

- SHA1
- RSA-SIG
- Group 1
- Lifetime 86400 Sec
- Main Mode

IKE Phase 2

- PFS
- Group 1

Vendor B

IKE Phase 1

- MD5
- Pre-Share Key
- Group 5
- Lifetime 86400 Sec
- Main Mode

IKE Phase 2

- PFS
- Group 5

Vendor C

IKE Phase 1

- SHA1
- Pre-Share Key
- Group 2
- Lifetime 86400 Sec
- Aggressive Mode

IKE Phase 2

- PFS
- Group 2



Terminology Issues

IKE Phase 1

IKE Phase 1 SA

IKE SA

ISAKMP SA

Main Mode

DH Key Length

DH Group

Modp #

Group #

IKE Phase 2

IKE Phase 2 SA

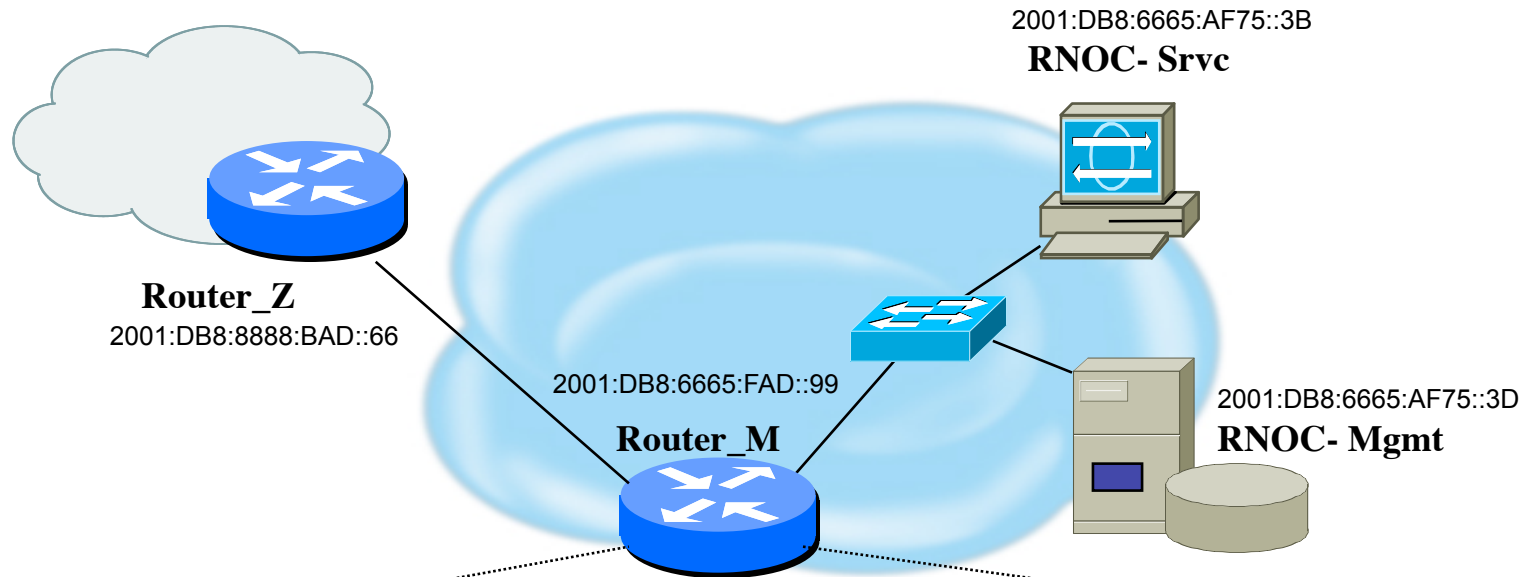
IPsec SA

Quick Mode

Configuration complexity increased with vendor
-specific configuration terms



Potentially Easy Configuration



Syslog server 2001:DB8:6665:AF75::3D authenticate esp-null sha1 pre-share 'secret4syslog'

TFTP server 2001:DB8:6665:AF75::3D authenticate esp-null aes128 pre-share 'secret4tftp'

BGP peer 2001:DB8:8888:BAD::66 authenticate esp-null aes128 pre-share 'secret4AS#XXX'



Interoperable Defaults For SAs

- Security Association groups elements of a conversation together
 - AH authentication algorithm and keys
 - ESP encryption algorithm and key(s)
 - Cryptographic synchronization
 - SA lifetime
 - SA source address
 - Mode (transport or tunnel)



How Do We Communicate Securely ?



Do we want integrity protection of data ?
Do we want to keep data confidential ?
Which algorithms do we use ?
What are the key lengths ?
When do we want to create new keys ?
Are we providing security end-to-end ?



Unix IPsec IKE Daemons

- Racoon2 (IKEv1 and IKEv2 and KINK)
 - <http://www.racoon2.wide.ad.jp/w/>
- Ipsec-tools (IKEv1)
 - port of KAME's IPsec utilities to the Linux-2.6 IPsec implementation; it supports NetBSD and FreeBSD as well
 - <http://ipsec-tools.sourceforge.net/>
- Strongswan (IKEv1 and IKEv2)
 - <http://www.strongswan.org/>
- Openikev2 (IKEv2)
 - <http://openikev2.sourceforge.net/>



LINUX and MACOSX machines

- Type command ' *man racoon* '
- Read how to set-up racoon, the name for this particular IKE software
- Type command ' *man setkey* '
- This command is used to set up the SA database
- The following files are located in */etc/racoon*:
 - ***psk.txt*** – file which contains the shared secrets
 - ***racoon.conf*** – file which configures IKE phase 1 and IKE phase 2 parameters



Set Up Security Policy Database

- Create a file named '***ipsec.conf***' which will be used with *setkey* to establish the correct security associations. The file should have the following information:
 - *flush;*
 - *spdflush;*
 - *spdadd 2001:DB8:6665:AF75::3D/128
2001:DB8:8888:BAD::66/128 any -P out ipsec esp/
transport//require ;*
 - *spdadd 2001:DB8:8888:BAD::66/128
2001:DB8:6665:AF75::3D/128 any -P in ipsec esp/
transport//require ;*



Creating SA Database

- Test to see what happens when you try and create an SA database:
- Type the following:
 - `setkey -f /etc/racoon/ipsec.conf`
- Use the ‘ `setkey -P -D` ’ command to see if appropriate entries have been created



Pre-Shared Key Configuration

- Edit the `psk.txt` file to add the peer IP address and the pre-shared secret key:

```
- # file for pre-shared keys used for IKE authentication
- # format is: 'identifier' 'key'
- # For example:
- # 10.1.1.1          flibbertigibbet
- # www.example.com  12345
- # foo@www.example.com micropachycephalosaurus
- <peer IPv6 address> <shared secret>
```

- Since the `psk.txt` file contains sensitive information make sure that the file is appropriately protected:
 - `chmod 600 /etc/racoon/psk.txt`

