Vulnerability Scanning in an IPv6 World

Richard Newman – m.richard.newman@gmail.com
Brett McKinney – mckinney.brett@gmail.com

Agenda

- IPv4… IPv5
- Intro to IPv6
- Threat Surface
- Network Scanning and Host Discovery
- Neighbor Discovery Demo using Metasploit
- Questions

What Happened to IPv4

- February 3, 2011?
- So what now…

IPv5… Huh?

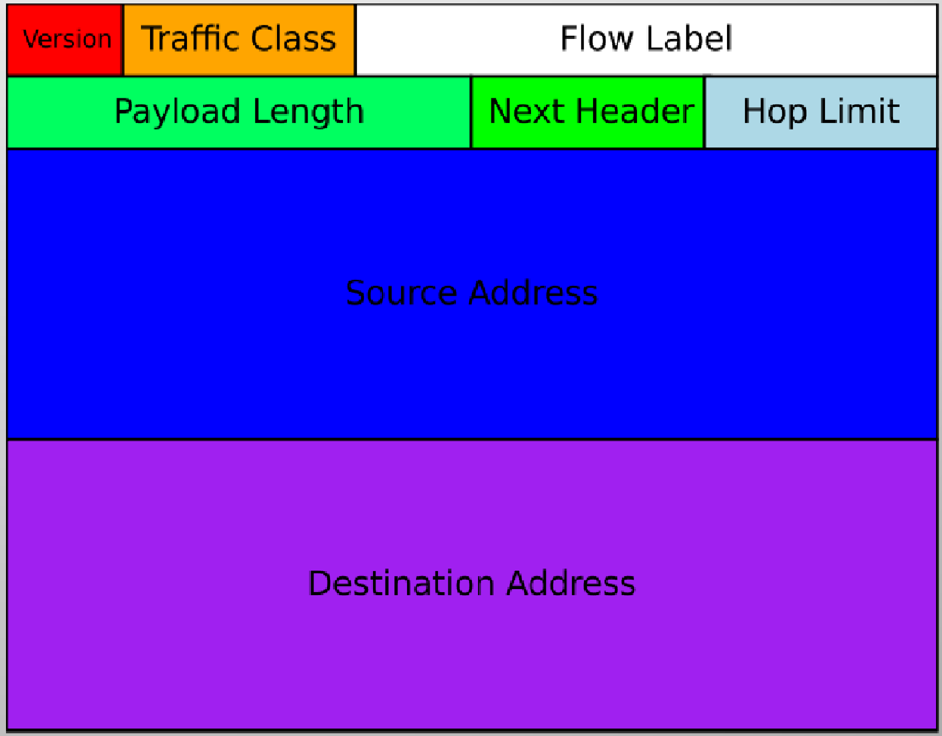
- AKA “Internet Stream Protocol” or ST
- Developed in 1978 and defined by RFCs 1190 and 1890
- ST-II added “5” to the protocol field in the IP header deeming it “IPv5”
  - Drafted in 1987 in RFC 1700 which was published in 1990
- Connection oriented compliment to IPv4
  - At layer 3
  - Much like TCP at layer 4
- First used for VoIP traffic
- ATM and MPLS also have features from IPv5
- Did not address diminishing address space

Changes in IPv6

- New Header
  - New Address format
  - New Address Scope
  - New Address Range
- Management of Addresses
- Auto Configuration
- IPv4 vs. IPv6
- Security
- Control Mechanisms

New Header

- 320 bits vs. 160 bits in IPv4
- Next header field
- Unlimited extensions
  - Routing
  - Fragmentation
  - Security



Address Classes

- Unicast
  - Identifies a single interface on the network
- Anycast
  - Groups similar interfaces
  - Packets are typically delivered to the nearest member of the anycast group
- Multicast
  - Groups of similar interfaces
  - Any multicast packet is delivered to all members of a multicast group
- Broadcast addresses are NOT supported

Address Scope

- Unicast addresses have two address scopes
  - Loopback
  - Link-local addresses
  - Not routable beyond local link
- Global
  - Everything else
  - Anycast addresses are globally routable

Address Scope, cont.

- Multicast addresses have 6 defined scopes
  - Interface local
  - Local link
  - Admin local
  - Site local
  - Organization local
  - global

Address Range

- 128 bit
- 340282366920938463463374607431768211456 hosts
  - 340 undecilion
  - Whole IPv4 per person on earth
  - IPv6 address per atom in the body
- IPv4
  - 32 bit – about 4 billion addresses

Management of Addresses

- Still handled by IANA and delegated to the local registrars
- Currently only 1/8 of the current possible addresses will be available for distribution
- Remaining are reserved for future use

Auto Configuration

- DHCP is no longer necessary but can be used to provide domain and DNS server information
- SLAAC (Stateless Address Auto Configuration)
  - Allows interface to obtain address in a configured routed network
  - Depends on ICMPv6 for neighbor and router discovery
  - Allows for seamless network renumbering

IPv4 vs. IPv6

IPV4

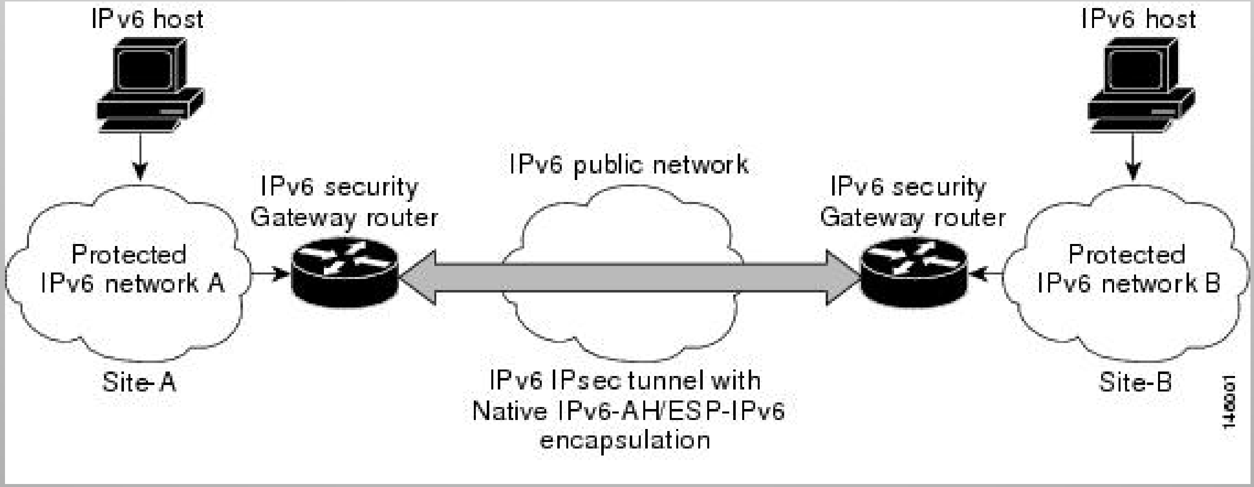
- 32 bit address
- Multicast optional
- ICMP optional
- 65535 max packet size

IPV6

- 128 bit address
- Multicast required
- ICMPv6 required
- 4GB max packet size

Built in Security

- Native support for IPSec
  - Support for native payload encryption
  - Support for native packet encryption (tunneling)



Control Mechanisms

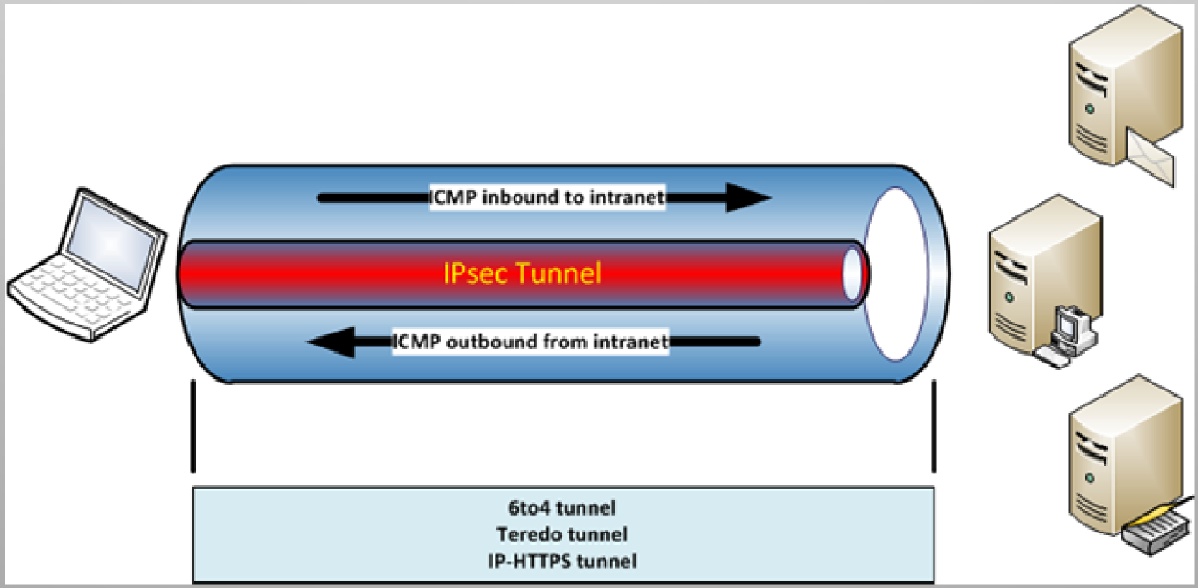
- ICMPv6
  - Max MTU path discovery
  - Router discovery
  - Neighbor discovery – Duplicate IP check

Threat Surface

- New multicast addresses could allow attackers to gain important information
  - FF02::1 - All nodes
  - FF05::2 - All routers
  - FF05::5 - All DHCP servers
- Rogue IPv6 devices
  - Fake IPv6 router can maliciously provide addressing and force traffic to route through it (MitM)
- Most dual stack firewalls only protect IPv4 by default, IPv6 traffic is passed transparently

Threat Surface, cont.

- IPv6 tunnels within IPv4 networks could allow IPv6 traffic to pass through firewalls undetected
  - Toredo
  - 6to4
  - Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)



Threat Surface, cont.

- Type 0 (zero) routing header
  - Akin to source routing
  - Allows the end device the ability to dictate the route through the network to the destination
- Built in ICMP and Multicast
  - Mandatory now for IPv6 to function
  - while some functions can be blocked at the edge, others have to be allowed in order to operate

Threat Surface, cont.

- An attacker can:
  - Claim to be another host's address using Neighbor Solicitation messages
  - Claim to be the default router using Router Advertisement messages
  - Claim all the addresses using Neighbor Solicitation messages
    - preventing hosts from obtaining an address
    - who needs DDoS…!!!
  - Advertise false prefixes using Router Advertisement messages

Changes in Scanning

- Port Scanning
  - Nothing
  - Just using a new address
- Host Discovery
  - Everything
  - Networks are too large

Changes in Scanning, cont.

- Workarounds
  - ICMPv6 Neighbor Discovery Protocol
    - Good if in a flat network
    - Doesn't work across routers or firewalls
  - Relying on sequential numbering or a common pattern for assigning addresses
  - Scanning around a known address
  - Reducing the search space by using Ethernet vendor prefix and "fffe" stuffing
  - DNS zone transfers
  - Log files may contain addresses
  - Sniffing traffic from a known host

Network Mapping

- Default subnet for IPv6 is 64 bits
  - giving 64 bits for hosts
  - 18 x 10^18 hosts
  - If you could scan 1,000,000 hosts per second it would still take over 500,000 years to complete the scan
- New methods for obtaining target host list will need to be developed
- Not new, but DNS will provide the initial target list

Network Mapping, cont.

- Tools
  - Nmap
    - Supports IPv6 networks
    - Current version will ONLY accept a single IPv6 address
    - No range support as of this writing
  - CHScanner
- Currently few tools support host discovery for IPv6 (including vulnerability and penetration testing tools)

Vulnerability Scanning Tools

- Nessus
  - Version 3.2
  - Linux only
- Saint
- Qualys
- Rapid 7 Nexpose?
- nCirlce?

Penetration Testing Tools

- Metasploit framework (added support in '08)
- Saint
- Core Impact

Web Application Testing Tools

- ALL
  - Web app testing is not affected

IPv6 Neighbor Discovery Demo

- Using Metasploit to discovery neighbor nodes
- nmap an IPv6 address

IPv6 Neighbor Discovery Demo

```
se@Ubuntu10: ~
File Edit View Terminal Help
msf auxiliary(ipv6_neighbor) > show options

Module options (auxiliary/scanner/discovery/ipv6_neighbor):

   Name       Current Setting    Required  Description
   ----       ---------------    --------  -----------
   INTERFACE  eth0               no        The name of the interface
   PCAPFILE                      no        The name of the PCAP capture file to
process
   RHOSTS     192.168.102.1-254  yes       The target address range or CIDR iden
tifier
   SHOST      192.168.102.168    no        Source IP Address
   SMAC       00:0c:29:c6:f4:c2  yes       Source MAC Address
   THREADS    1                  yes       The number of concurrent threads
   TIMEOUT    500                yes       The number of seconds to wait for new
 data

msf auxiliary(ipv6_neighbor) >
```

```
File Edit View Terminal Help
se@Ubuntu10:~$ ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0c:29:c6:f4:c2
          inet addr:192.168.102.168  Bcast:192.168.102.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fec6:f4c2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:9532 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8155 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6736683 (6.7 MB)  TX bytes:1025729 (1.0 MB)
          Interrupt:19 Base address:0x2000

se@Ubuntu10:~$
```

- Metasploit auxiliary module for IPv6 neighbor discovery
  - Required options are: INTERFACE, RHOSTS, SHOST, and SMAC
  - SMAC and SHOST are from the local interface
  - RHOSTS is the local broadcast domain the attack machine is on

IPv6 Neighbor Discovery Demo

```
File  Edit  View  Terminal  Help
   RHOSTS     192.168.102.1-254  yes       The target address range or CIDR iden
tifier
   SHOST      192.168.102.168    no        Source IP Address
   SMAC       00:0c:29:c6:f4:c2  yes       Source MAC Address
   THREADS    1                  yes       The number of concurrent threads
   TIMEOUT    500                yes       The number of seconds to wait for new
 data

msf auxiliary(ipv6_neighbor) > run

[*] Discovering IPv4 nodes via ARP...
[*]
[*]      192.168.102.1 ALIVE
[*]      192.168.102.2 ALIVE
[*]    192.168.102.158 ALIVE
[*]    192.168.102.173 ALIVE
[*]    192.168.102.254 ALIVE
[*] Discovering IPv6 addresses for IPv4 nodes...
[*]
[*]    192.168.102.158 maps to fe80::20c:29ff:fef8:4139
[*]    192.168.102.173 maps to fe80::20c:29ff:fe56:d3f
[*] Scanned 254 of 254 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(ipv6_neighbor) >
```

```
File  Edit  View  Terminal  Help
se@Ubuntu10:~$ ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0c:29:c6:f4:c2
          inet addr:192.168.102.168  Bcast:192.168.102.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fec6:f4c2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:9532 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8155 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6736683 (6.7 MB)  TX bytes:1025729 (1.0 MB)
          Interrupt:19 Base address:0x2000

se@Ubuntu10:~$ ping6 -I eth0 -c 3 fe80::20c:29ff:fe56:d3f
PING fe80::20c:29ff:fe56:d3f(fe80::20c:29ff:fe56:d3f) from fe80::20c:29ff:fec6:f
4c2 eth0: 56 data bytes
64 bytes from fe80::20c:29ff:fe56:d3f: icmp_seq=1 ttl=64 time=0.406 ms
64 bytes from fe80::20c:29ff:fe56:d3f: icmp_seq=2 ttl=64 time=0.368 ms
64 bytes from fe80::20c:29ff:fe56:d3f: icmp_seq=3 ttl=64 time=0.599 ms

--- fe80::20c:29ff:fe56:d3f ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.368/0.457/0.599/0.104 ms
se@Ubuntu10:~$
```

- Metasploit will return the link local IPv6 address for each IPv4 host discovered
- ping6 requires the interface to be specified for a link local address

IPv6 Neighbor Discovery Demo

IPv4 nmap

```
File  Edit  View  Terminal  Help
se@Ubuntu10:~$ nmap 192.168.102.173

Starting Nmap 5.00 ( http://nmap.org ) at 2011-03-21 12:39 EDT
Interesting ports on 192.168.102.173:
Not shown: 993 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
139/tcp   open  netbios-ssn
143/tcp   open  imap
445/tcp   open  microsoft-ds
5001/tcp open  commplex-link
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 1.40 seconds
se@Ubuntu10:~$
```

IPv6 nmap

```
File  Edit  View  Terminal  Help
se@Ubuntu10:~$ nmap -6 fe80::20c:29ff:fe56:d3f%eth0

Starting Nmap 5.00 ( http://nmap.org ) at 2011-03-21 12:40 EDT
Interesting ports on fe80::20c:29ff:fe56:d3f:
Not shown: 994 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
139/tcp   open  netbios-ssn
143/tcp   open  imap
445/tcp   open  microsoft-ds
5001/tcp open  commplex-link
8080/tcp open  http-proxy

Nmap done: 1 IP address (1 host up) scanned in 3.29 seconds
se@Ubuntu10:~$
```

- IPv4 nmap result – standard nmap syntax
  - Note: the results list the ports open from the IPv4 stack
- IPv6 nmap result – for a link local address the interface must be included in the address by appending %*interface* (%eth in our example)
  - Note: the results list the ports open from the IPv6 stack
- They may not match!

Questions

?

References

- http://playground.sun.com/pub/ipng/html/ipng-main.html
- http://www.6net.org/events/workshop-2005/mohacsi.pdf
- http://www.networkworld.com/news/2009/071309-ipv6-network-threat.html
- http://en.wikipedia.org/wiki/IPv6_address
- http://www.ietf.org/rfc/rfc5157.txt
- http://blogs.cisco.com/security/icmp-and-security-in-ipv6/#more-21899
- http://en.wikipedia.org/wiki/Internet_Stream_Protocol
- http://en.wikipedia.org/wiki/ICMPv6