Seguridad IPv6

Fernando Gont

WALC 2012
Ciudad de Panamá, Panamá. Octubre 18, 2012

Acerca de...

- He trabajado en análisis de seguridad de protocolos de comunicaciones para:
  - UK NISCC (National Infrastructure Security Co-ordination Centre)
  - UK CPNI (Centre for the Protection of National Infrastructure)
- Actualmente trabajando para SI6 Networks
- Miembro del grupo CEDI (I+D) de UTN/FRH, Argentina
- Participante activo de la Internet Engineering Task Force (IETF)
- Más información en: http://www.gont.com.ar

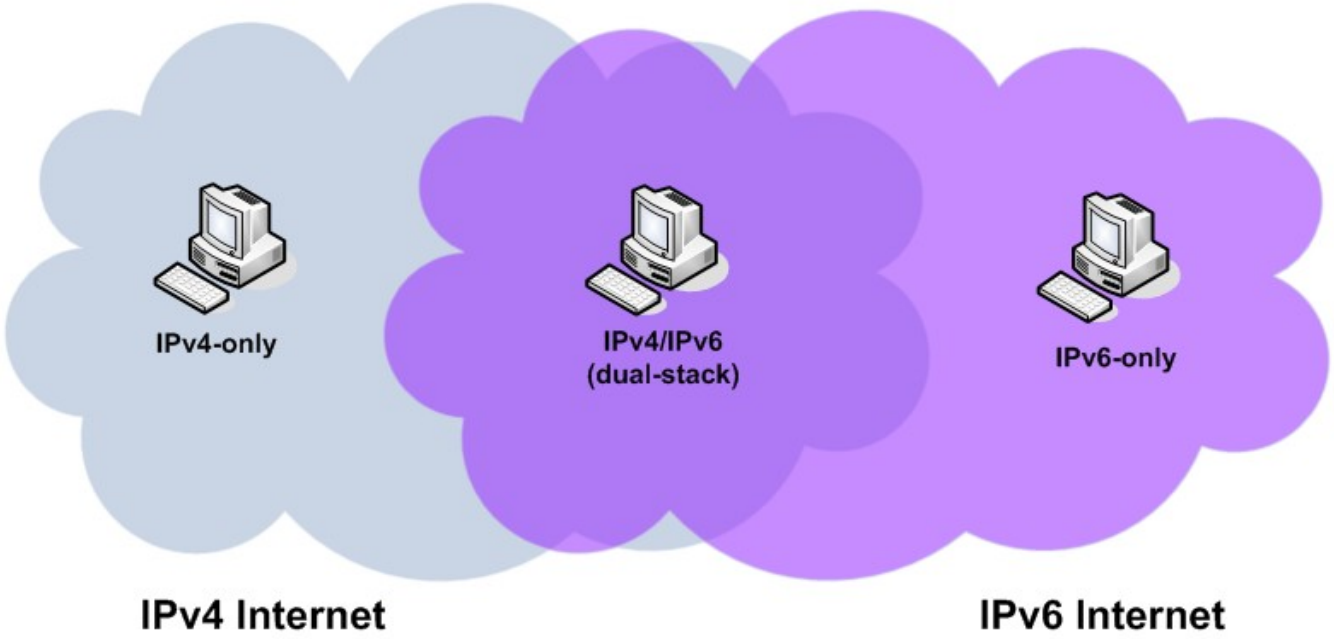
Agenda

- Motivación de esta presentación
- Breve comparación entre IPv6/IPv4
- Discusión de aspectos de seguridad de IPv6
- Implicancias de seguridad de los mecanismos de transición/co-existencia
- Implicancias de seguridad de IPv6 en redes IPv4
- Áreas en las que se necesita más trabajo
- Conclusiones
- Preguntas y respuestas

Breve reseña de IPv6

Pero... que es todo esto de IPv6?

- Soluciona el problema de escasez de direcciones
- Utiliza direcciones de 128 bits (vs. direciones IPv4 de 32 bit)
- Básicamente provee el mismo **servicio** que IPv4
- No es “compatible hacia atrás” con IPv4

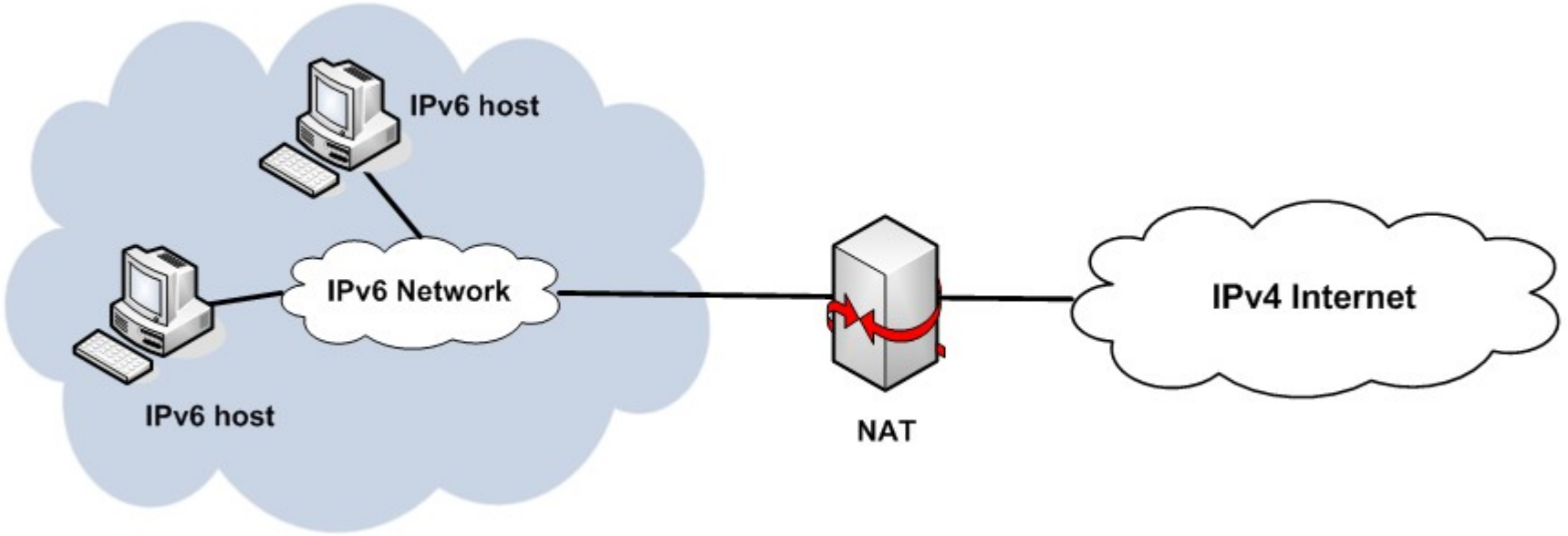


Pero... que es todo esto de IPv6? (II)

- Podemos interconectar “islas” de un protocolo a través de otro, mediante los llamados túneles

Pero... que es todo esto de IPv6? (III)

- Podemos interconectar hosts IPv4-only con hosts IPv6-only mediante “traductores”



Pero... que es todo esto de IPv6? (IV)

- Para un determinado nombre, el DNS puede contener
  - Registros A (direcciones IPv4)
  - Registros AAAA (direcciones IPv6)
- El sistema pedirá registros A y/o AAAA de según una variedad de criterios
- De acuerdo a los registros disponibles, y protocolos soportados, podrá utilizarse IPv4 y/o IPv6

Breve comparación entre IPv6/IPv4

- Muy similares en *funcionalidad*, pero no así en *mecanismos*

	IPv4	IPv6
Direccionamiento	**32 bits**	**128 bits**
Resolución de direcciones	**ARP**	**ICMPv6 NS/NA (+ MLD)**
Auto-configuración	**DHCP & ICMP RS/RA**	**ICMPv6 RS/RA & DHCPv6 (optional) (+ MLD)**
Aislamiento de fallos	**ICMPv4**	**ICMPv6**
Soporte de IPsec	**Opcional**	**Opcional**
Fragmentación	**Tanto en hosts como en routers**	**Sólo en hosts**

Motivación de esta presentación

Motivación de esta presentación

- Muchos mitos creados en torno a IPv6:
  - La seguridad fue considerada durante el diseño
  - El paradigma de seguridad cambiará a host-centric
  - Aumentará el uso de IPsec
  - etc.
- Estos mitos tienen y han tenido un impacto negativo
- Esta presentación intentará:
  - Separar “mito” de “realidad”
  - Influenciar como pensás sobre “seguridad IPv6”
  - Reproducir algunos ataques concretos, a modo de ejemplo

Consideraciones generales sobre seguridad IPv6

Algunos aspectos interesantes...

- Menor experiencia con IPv6 que con IPv4
- Implementaciones de IPv6 menos maduras que las de IPv4
- Menor soporte para IPv6 que para IPv4 en productos de seguridad
- La red Internet será mucho mas compleja:
  - Dos protocolos de Internet
  - Mayor uso de NATs
  - Mayor uso de túneles
  - Uso de otras tecnologías de transición co-existencia
- Pocos recursos humanos bien capacitados

… así y todo tal vez sea la única opción para permanecer en el negocio

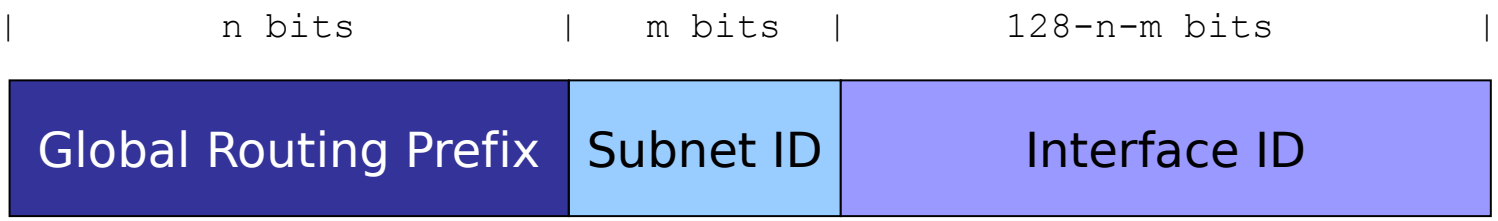
Implicancias de seguridad de IPv6

Direccionamiento IPv6
Breve reseña

Breve revisión de direccionamiento IPv6

- El mayor espacio de direcciones es el “motivador” de IPv6
- Se utilizan direcciones de 128 bits
- Semántica muy similar a IPv4:
  - Se agregan direcciones en “prefijos” para el ruteo
  - Existen distintos tipos de direcciones
  - Existen distintos alcances para las direcciones
- Cada interfaz utiliza multiples direcciones, de multiples tipos y alcances:
  - Una dirección link-local unicast
  - Una o mas direcciones global unicast
  - etc.

Formato de Direcciones Globales IPv6



- Diferentes políticas de selección de IID:
  - Embeber la MAC address (SLAAC tradicional)
  - Embeber la dirección IPv4 (por ej., 2001:db8::192.168.1.1)
  - Low-byte (por ej., 2001:db8::1, 2001:db8::2, etc.)
  - Wordy (por ej., 2001:db8::dead:beef)
  - Indicado por una tecnología de transición

Direccionamiento IPv6
Implicancias en address scanning remoto

Reseña de host-scanning en IPv4

- Utilizan direcciones de 32 bits
- Densidad de hosts en red elevada
- Técnica tradicional == fuerza bruta
  - Seleccionar el rango de direcciones deseado
  - Enviar pruebas (ICMP echo, TCP {SYN, ACK}, UDP
  - Esperar respuestas

La escala del problema es pequeña

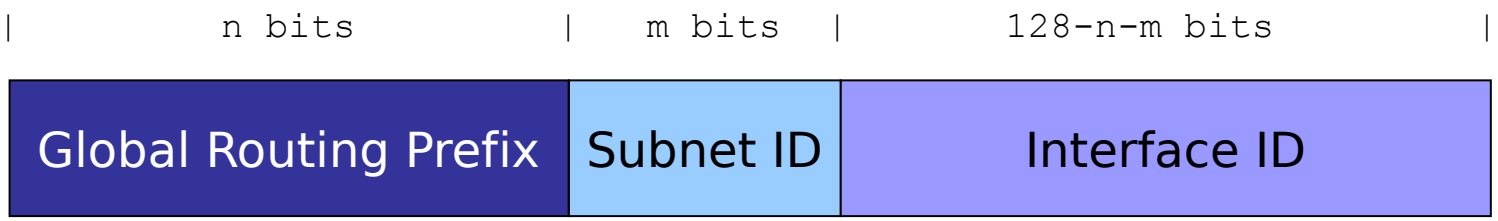
Fuerza bruta == “suficientemente bueno”

Leyendas urbanas sobre scanning IPv6

"Debido al gran espacio de direcciones IPv6, los ataques de escaneo son imposibles. Escanear un /64 tomaría 500.000.000 años"

Es realmente el espacio de búsqueda de un /64 2^{64} direcciones?

Formato de Direcciones Globales IPv6



- Diferentes políticas de selección de IID:
  - Embeber la MAC address (SLAAC tradicional)
  - Embeber la dirección IPv4 (por ej., 2001:db8::192.168.1.1)
  - Low-byte (por ej., 2001:db8::1, 2001:db8::2, etc.)
  - Wordy (por ej., 2001:db8::dead:beef)
  - Indicado por una tecnología de transición

Direcciones IPv6 en el mundo real

- Malone midió (*) las politicas de asignación de direcciones en escenarios reales

Tipo dirección	Porcentaje
SLAAC	50%
IPv4-based	20%
Teredo	10%
Low-byte	8%
Privacy	6%
Wordy	<1%
Otras	<1%

Hosts

Tipo dirección	Porcentaje
Low-byte	70%
IPv4-based	5%
SLAAC	1%
Wordy	<1%
Privacy	<1%
Teredo	<1%
Otras	<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>.

Direcciones con IEEE IDs embebidos

- En la práctica, el espacio de búsqueda es a lo sumo ~2^{24} bits – **posible!**
- Los 24 bits de bajo orden no son necesariamente aleatorios:
  - Una organización compra una gran cantidad de equipos
  - Usualmente, los equipos tienen direcciones MAC consecutivas
  - Las MACs se suelen distribuir por regiones geográficas

Direcciones con IEEE IDs embebidos (II)

- Las tecnologías de virtualización presentan un caso interesante
- Virtual Box utiliza el OUI 08:00:27 (espacio de busqueda: ~2^{24})
- VMWare ESX utiliza:
  - MACs automaticas: OUI 00:05:59, y siguientes 16 bits tomados de la dirección IPv4 del host real (espacio de búsqueda: ~2^{8})
  - MACs manualmente configuradas:OUI 00:50:56 y restantes bits en el rango 0x000000-0x3fffff (espacio de búsqueda: ~2^{22})

Direcciones IPv6 basadas en IPv4

- Simplemente incluyen una dirección IPv4 en el IID
- Ejemplo: 2000:db8::192.168.0.1
- El espacio de búsqueda es el mismo que el correspondiente a IPv4

Algunas conclusiones

- Los ataques de escaneo no son imposibles en IPv6
- Se han econtrado “in the wild”
- Es esperable que no sean de “fuerza bruta”, y aprovechen:
  - Patrones de las direcciones
  - “Leaks” de la capa de aplicación
  - etc.
- Recomendaciones:
  - Evitar patrones en las direcciones IPv6
  - Siempre considerar el uso de firewalls

Direccionamiento IPv6
Implicancias en address scanning local

IPv6 host scanning local

- Se trata de un problema completamente diferente
- Se reduce notablemente el espacio de búsqueda utilizando direcciones multicast (por ej. ff02::1)
- Pueden/deben utilizarse distintos paquetes de prueba:
  - ICMPv6 Echo Requests
  - Paquetes con opciones IPv6 de tipo 10xxxxxx no soportadas
- Técnica implementada en:
  - SI6 Networks IPv6 toolkit (http://www.si6networks.com/tools)
  - THC IPv6 Attack Toolkit (http://www.thc.org)
  - NMAP (http://www.insecure.org)

Ejemplo de scanning local

IPv6 host scanning local (II)

- Se pueden utilizar técnicas adicionales como:
  - mDNS
  - LLNR
- En el peor de los casos, puede utilizarse sniffing
- En síntesis, es un problema muy difícil de mitigar

Port scanning en IPv6

Port-scanning en IPv6

- Es igual que en el mundo IPv4
- Se puede realizar mediante nmap, del siguiente modo:

```
# nmap -6 -p1-10000 -n 2000:db8::1
80/tcp open http
135/tcp open msrpc
445/tcp open microsoft-ds
554/tcp open rtsp
1025/tcp open NFS-or-IIS
1026/tcp open LSA-or-nterm
1027/tcp open IIS
1030/tcp open iad1
1032/tcp open iad3
1034/tcp open unknown
1035/tcp open unknown
1036/tcp open unknown
1755/tcp open wms
9464/tcp open unknown
```

Conectividad Extremo a Extremo

Breve reseña

- La red Internet se basó en el principio de “extremo a extremo”
  - Red tonta, extremos (hosts) inteligentes
  - La comuncación es posible entre cualquier par de nodos
  - La red no examina el contenido de los paquetes IP
- Se suele argumentar que este principio permite la innovación
- Los NATs lo han eliminado de Internet
- Se espera que con IPv6 no existan NATs, y se retorne al principio “extremo a extremo”

IPv6 y el principio “extremo a extremo”

Mito: “*IPv6 devolverá a Internet el principio 'extremo a extremo*'”

- Se asume que el gran espacio de direcciones devolverá este principio
- Sin embargo,
  - Las direcciones globales no garantizan conectividad extremo a extremo
  - La mayoría de las redes no tiene interés en “innovar”
  - Los usuarios esperan en IPv6 los mismos servicios que en IPv4
  - Este principio aumenta la exposición de los sistemas
- En resumen,
  - La conectividad extremo a extremo no necesariamiente es deseable
  - La subred típica IPv6 solo permitirá “trafico saliente” (mediante firewalls)

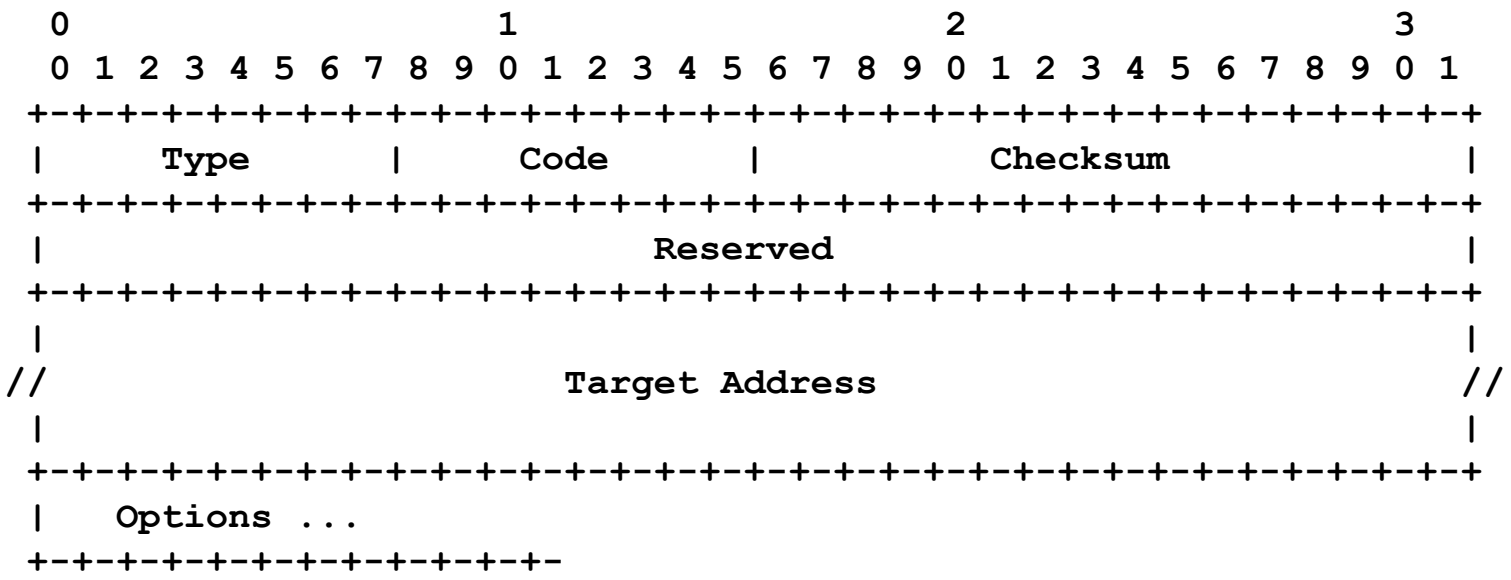
Resolución de direcciones

Resolución de Direcciones en IPv6

- Utiliza mensajes ICMPv6 Neighbor Solicitation y Neighbor Advertisement
- Funciona (aproximadamente) así:
  - El Host A envía un NS: Quién tiene la dirección IPv6 fc00:1::1?
  - El Host B responde con un NA: Yo tengo esa dirección IPv6, y la MAC address correspondiente es 06:09:12:cf:db:55.
  - El Host A “cachea” esa información en el “Neighbor Cache” por un período de tiempo
  - El Host A ahora puede enviar paquetes IPv6 al Host B

Mensajes Neighbor Solicitation

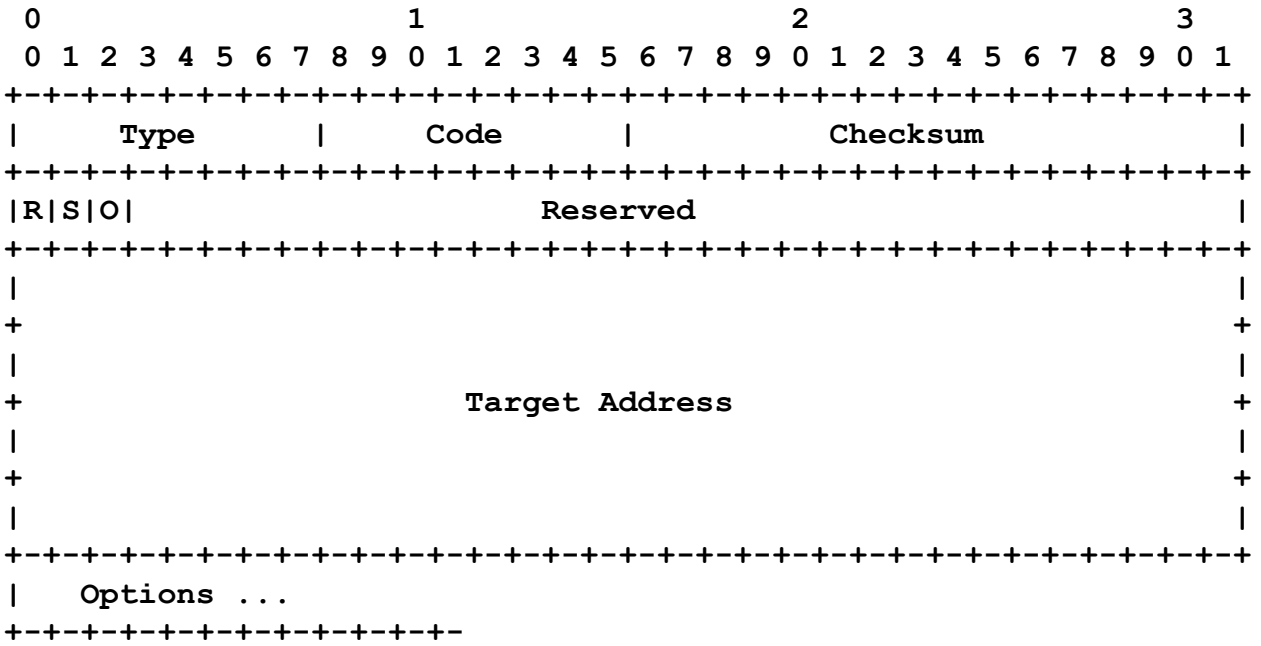
- Mensajes ICMPv6 de Type 135, Code 0
- Utilizados para solicitar el mapeo de una dirección IPv6 a una dirección de capa de enlace
- Única opción especificada: “Source Link-layer address”



SI6 NETWORKS

Mensajes Neighbor Advertisement

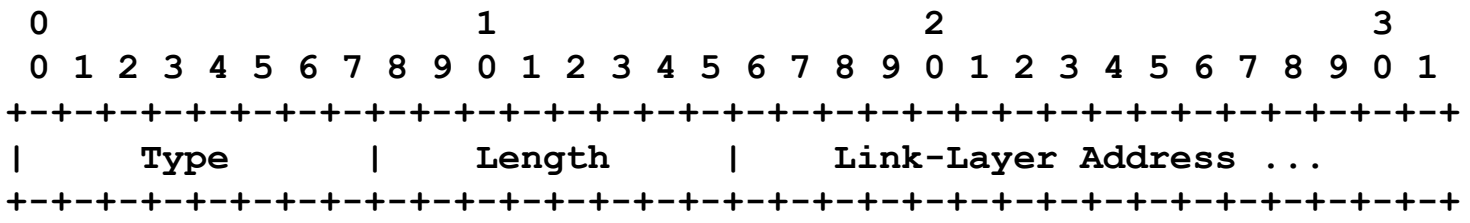
- Mensajes ICMPv6 de Type 136, Code 0
- Utilizados para informar el mapeo de una dirección IPv6 a una dirección de capa de enlace
- Única opción especificada: “Target Link-layer address”



SI6 NETWORKS

Source/Target Link-layer Address Option

- La Source Link-layer Address contiene la dirección link-layer correspondiente a la IPv6 Source Address del paquete
- La Target Link-layer address contiene la dirección link-layer correspondiente a la dirección peticionada



Type: 1 for Source Link-layer Address
2 for Target Link-layer Address

Ejemplo de tráfico de Address Resolution

```
% ping6 2004::1

12:12:42.086657 2004::20c:29ff:fe49:ebdd > ff02::1:ff00:1: icmp6: neighbor sol:
who has 2004::1(src lladdr: 00:0c:29:49:eb:dd) (len 32, hlim 255)
12:12:42.087654 2004::1 > 2004::20c:29ff:fe49:ebdd: icmp6: neighbor adv: tgt is
2004::1(RSO)(tgt lladdr: 00:0c:29:c0:97:ae) (len 32, hlim 255)
12:12:42.089147 2004::20c:29ff:fe49:ebdd > 2004::1: icmp6: echo request (len
16, hlim 64)
12:12:42.089415 2004::1 > 2004::20c:29ff:fe49:ebdd: icmp6: echo reply (len 16,
hlim 64)
```


SI6 NETWORKS

ndisc6: ND diagnostic tool

- Puede ser utilizada para enviar un NS para una dirección particular
- Ejemplo:

```
$ ndisc6 fc00:1::1 vboxnet0
Soliciting fc00:1::1 (fc00:1::1) on vboxnet0...
Target link-layer address: 08:00:27:9D:49:80
 from fe80::a00:27ff:fe9d:4980
```

Neighbor Cache

- Almacena información obtenida a través del proceso de Resolución de Direcciones
- Cada entrada (Dirección IPv6 address, Dirección link-layer) puede estar en uno de los siguientes estados:

NC entry state	Semantics
INCOMPLETE	Add. Res. Is in progress (not yet determined)
REACHABLE	Neighbor is reachable
STALE	Not known to be reachable
DELAY	Not known to be reachable (wait for indication)
PROBE	Not known to be reachble (probes being sent)

Neighbor Cache (contenido en *BSD)

- Ejemplo de salida de “ndp -a”:

```
% ndp -a
Neighbor                        Linklayer Address  Netif Expire    S Flags
2004:1::f8dd:347d:8fd8:1d2c     0:c:29:49:eb:e7      em1 permanent R
fe80::20c:29ff:fec0:97b8%em1    0:c:29:c0:97:b8      em1 23h48m16s S R
2004:1::20c:29ff:fe49:ebe7      0:c:29:49:eb:e7      em1 permanent R
fe80::20c:29ff:fe49:ebe7%em1    0:c:29:49:eb:e7      em1 permanent R
2004::1                         0:c:29:c0:97:ae      em0 23h49m27s S R
2004::20c:29ff:fe49:ebdd        0:c:29:49:eb:dd      em0 permanent R
fe80::20c:29ff:fe49:ebdd%em0    0:c:29:49:eb:dd      em0 permanent R
fe80::20c:29ff:fec0:97ae%em0    0:c:29:c0:97:ae      em0 23h48m16s S R
2004::d13e:2428:bae7:5605       0:c:29:49:eb:dd      em0 permanent R
```


SI6 NETWORKS

Neighbor Cache (contenido en Linux)

- Ejemplo de salida de “ip -6 neigh show”:

```
$ ip -6 neigh show
fe80::a00:27ff:fef9:7304 dev vboxnet0 lladdr 08:00:27:f9:73:04 router STALE
2000::4000 dev vboxnet0 lladdr 11:22:33:44:55:66 PERMANENT
2000:1::1 dev vboxnet0 lladdr 08:00:27:f9:73:04 router REACHABLE
fe80::fc8d:15ed:7f43:68ea dev wlan0 lladdr 00:21:5c:0b:5d:61 router STALE
```

Neighbor Discovery for IPv6
Ataques de Resolución de Direcciones

“Man in the Middle” ó Denial of Service

- Son la versión IPv6 de los ataques de “envenenamiento dde cache” de IPv4
- Sin mecanismos de autentificación, es trivial para un atacante local enviar información falsa
- Ataque:
  - “Escuchar” mensajes Neighbor Solicitation que tengan la dirección IPv6 de la víctima en el campo “Target Address”
  - Al recibir un NS responder con un Neighbor Advertisement con informaciń falsificada
- Si la “Target Link-layer address” es inexistente, el ataque será de DEnegación de Servicio (DoS)
- Si la “Target Link-layer address” es la del atacante, el ataque será de tipo “man in the middle”.

Realizando ataques con na6

- Correr la herramienta como:

  # ./na6 –i IFACE –W VICTIMADDR –L –E MACADDR –c –o

- Ahora enviar tráfico a la víctima – el mismo será enviado a MACADDR
- Podés verificarlo con tcpdump:

  # tcpdump –i em0 –e –vv ip6

Sniffing en switched networks

- En vez de desbordar la tabla del switch, se puede realizar un ataque mas elegante
- Mapear la dirección IPv6 de la víctima a:
  - La dirección Ethernet de broadcast (ff:ff:ff:ff:ff:ff)
  - Direcciones Ethernet multcast (por ej.,33:33:00:00:01)
- Esto hará que el tráfico se envíe a todos los nodos (incluyendo el atacante y la víctima)
- Ninguno de los BSD chequean estas direcciones especiales!

Realizando ataques con na6

- Correr la herramienta como:

  # ./na6 –i IFACE –W VICTIMADDR –L –E ff:ff:ff:ff:ff:ff –c –o

- Ahora enviar tráfico a la víctima – el mismo será enviado a MACADDR
- Podés verificarlo con tcpdump:

  # tcpdump –i em0 –e –vv ip6

- Es interesante:
  - Ejecutar tcpdump antes del ataque (no se recibirá el tráfico)
  - Ejecutar tcpdump luego del ataque (se verá que los paquetes van dirigidos a las direcciones Ethernet “especiales”)

Introduciendo un bucle en el router

- Responder los NS enviados por el router con un NA que contiene la dirección link-layer del propio router
- El router recibirá una copia del paquete (asumiendo que la tarjeta de red lo permite)
- Se decrementará el Hop Limit, y el paquete se reenviará
- El proceso se repetirá hasta que el Hop Limit se haga 0.

Realizando el ataque con na6

- Ejecutar la herramienta como:

  # ./na6 –i IFACE –W fc00:1::80 –L –E ROUTERMAC –c –o
- En algún host local, ejecutar:

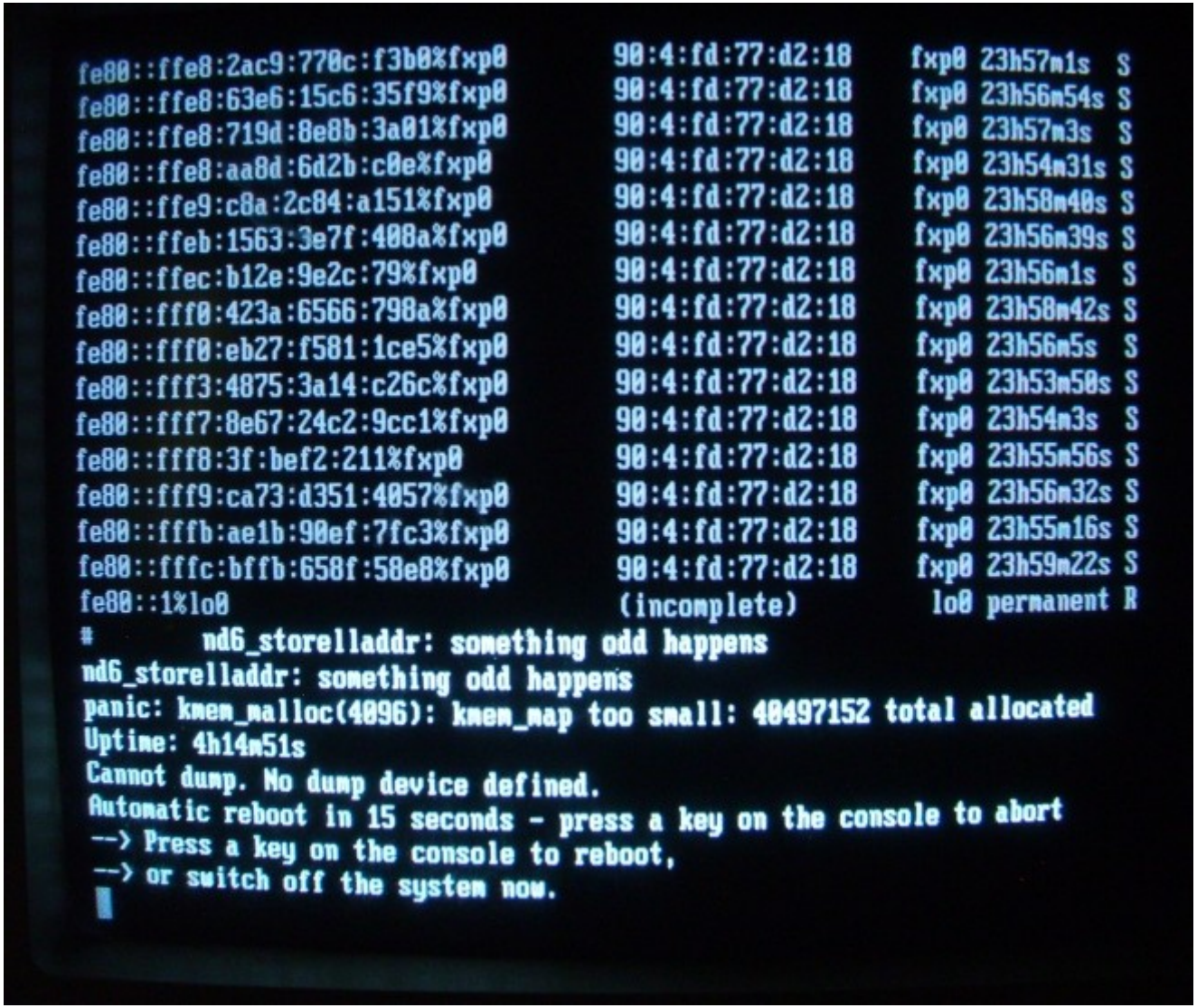
  % ping6 –i 20 –w 40 fc00:1::80
- Verificar el ataque corriendo tcpdump en el router:

  sudo tcpdump –i em1 –e –vv ip6
- Prestar atención al Hop Limit de los paquetes!

Desbordamiento del Neighbor Cache

- Algunas implementaciones (como FreeBSD y NetBSD) no impoen límites en el tamaño del Neighbor Cache
- Se puede lograr que se utilice toda la memoria del kernel para el Neighbor Cache, generando un kernel panic.
- Ataque:
  - Enviar un gran número de Neighbor Solicitation messages con una opción Source Link-layer address
  - Por cada paquete recibido, la víctima creara una entrada en el Neighbor Cache
  - Y si se insertan entradas mas rápido de lo que se eliminan las entradas “antiguas”....

Desbordamiento del Neighbor Cache (II)



SI6 NETWORKS

Algunas sysctl's para ND (OpenBSD)

- net.inet6.ip6.neighborgcthresh (defaults to 2048): Máxima cantidad de entradas en el Neighbor Cache
- net.inet6.icmp6.nd6_prune (defaults to 1): Intervalo entre Neighbor Cache babysitting (en seconds).
- net.inet6.icmp6.nd6_delay (defaults to 5): especifica la constanteDELAY_FIRST_PROBE_TIME de RFC 4861.
- net.inet6.icmp6.nd6_umaxtries (defaults to 3): especifica la constante MAX_UNICAST_SOLICIT de RFC 4861
- net.inet6.icmp6.nd6_mmaxtries (defaults to 3): especifica la constante MAX_MULTICAST_SOLICIT de RFC 4861.
- net.inet6.icmp6.nd6_useloopback (defaults to 1): Si es distinto de cero, usa la interfaz loopback para tráfico local.
- net.inet6.icmp6.nd6_maxnudhint (defaults to 0): Número maximo de avisos de reachability de la capa superior antes de realizar ND tradicional.

Neighbor Discovery for IPv6

Ataques contra Resolución de Direcciones – Contramedidas

Contramedidas para ataques ND

- Posibles contramedidas:
  - Desplegar SEND
  - Monitorear tráfico de Neighbor Discovery
  - Utilizar entradas estáticas en el Neighbor Cache
  - Restringir el acceso a al red local

Secure Neighbor Discovery (SEND)

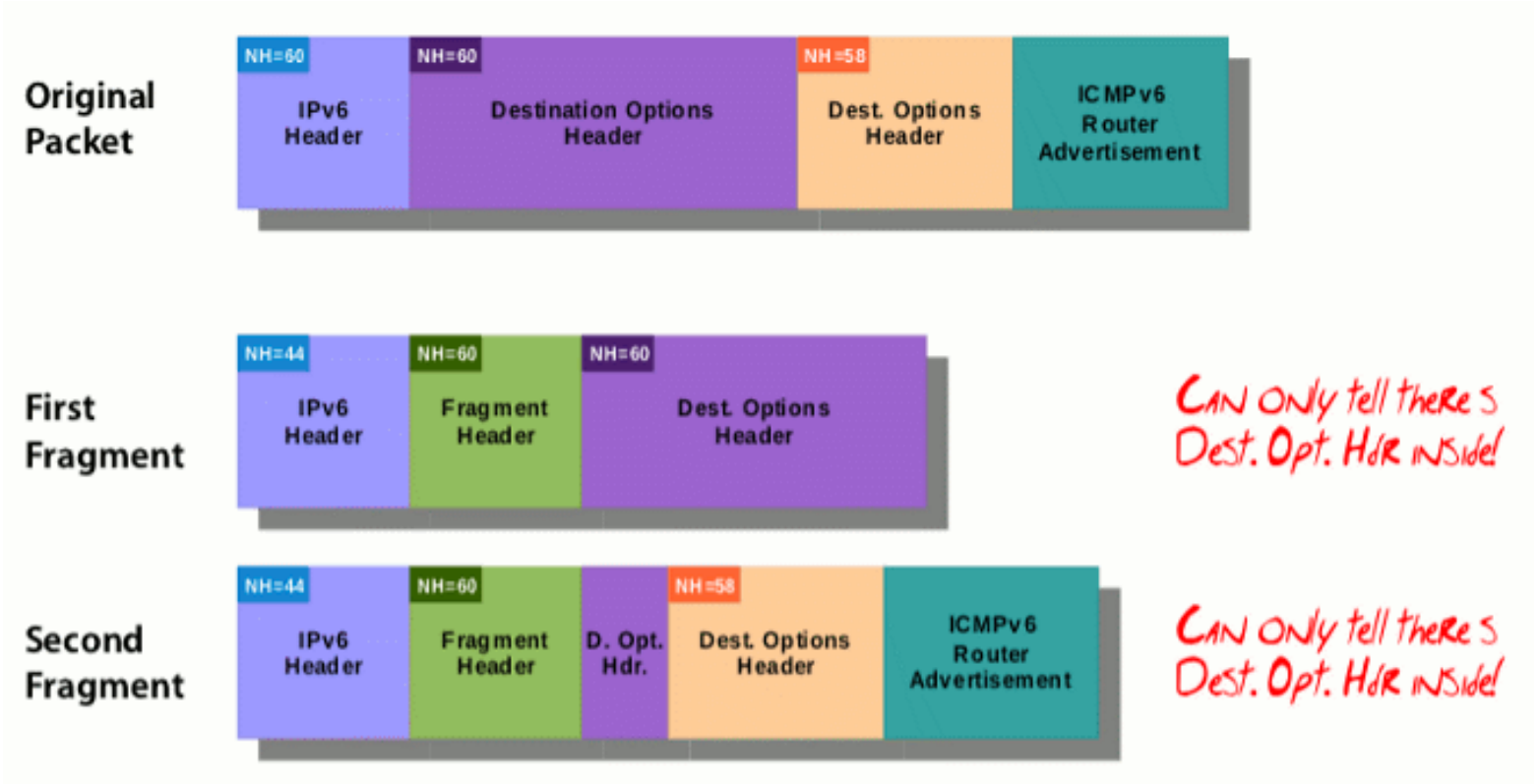
- Solución criptográfica el problema de los mensajes NS/NA faslsificados:
  - Se utilizan firmas criptograficas para proteger a todos los mensajes de Neighbor Discovery
  - Se utilizan certificados para verificar la autoridad de sistemas
- SEND es difícil de desplegar:
  - No está ampliamente soportado
  - El requerimiento de una PKI es uno de los obstaculos principales para su despliegue
  - El “retorno de inversión” es pequeño: otras piezas clave de la infraestructura de red siguen siendo inseguras (por ej., DNS)

Monitoreo de tráfico Neighbor Discovery

- Algunas herramientas registran los mapeos IPv6 -> Ethernet legítimos, y enván una alarma cuando los mismos cambian
- Similar al uso de arpwatch en IPv4
- Sin embargo, estas herramientas son facilmente evadibles:
  - ND corre sobre IPv6
  - Los paquetes pueden contener encabezados de extensión y/o ser fragmentados
  - Se vuelve muy dificultoso monitorear dicho tráfico

Evasión de monitoreo de tráfico ND

- Problema fundamental: Tráfico demasiado complejo
- Ejemplo:



SI6 NETWORKS

Entradas estáticas en el Neighbor Cache

- Consiste en configurar manualmente los mapeos IPv6->link-layer
- Análogo a incluir entradas estáticas en el ARP cache de IPv4
- Aplicable en escenarios muy particulares
- Es una solución que “no escala”

Entradas estáticas en *BSD

- El Neighbor Cache se manipula con el comando "ndp"
- Se pueden agregar entradsa estáticas así:

  # ndp –s IPV6ADDR MACADDR
- Si IPV6ADDR es una dirección link-local, se debe especificar el interface index de esta manera:

  # ndp –s IPV6ADDR%IFACE MACADDR

Restringir el acceso a la red local

- Principio fundamental: compartimentación
- Dividir la red de una organización en varias redes locales
- Se limita el alcance del daño de potenciales atacantes
- Puede implicar aumentos en los costos

Auto-configuración

Breve reseña

- Dos mecanismos de autoconfiguración en IPv6:
  - Stateless Address Auto-Configuration (SLAAC)
    - Basado en ICMPv6
  - DHCPv6
    - Basado en UDP
- SLAAC es mandatorio, mientras que DHCPv6 es opcional
- Funcionamiento básico de SLAAC:
  - Los hosts solicitan información mediante ICMPv6 Router Solicitations
  - Los routers responden con Router Advertisements:
    - Prefijos a utilizar
    - Rutas a utilizar
    - Parametros de red
    - etc.

Vulnerabilidades y contramedidas

- Falsificando Router Advertisements se puede realizar:
  - Man In the Middle
  - Denial of Service
- Posibles contramedidas:
  - Desplegar SEND
  - Monitorear mensajes RS/RA
  - Desplegar Router Advertisement Guard (RA-Guard)
  - Restringir el acceso a la red local

Vulnerabilidades y contramedidas (II)

- Lamentablemente,
  - SEND es difícil de desplegar
  - Las herramientas de monitoreo son facilmente evadibles
  - Implementaciones de RA-Guard actuales son fáciles de evadir
  - No siempre se puede limitar el acceso a la red local
- En síntesis,
  - La situación es similar al caso de IPv4

Neighbor Discovery for IPv6
Stateless Address Auto-configuration (SLAAC)

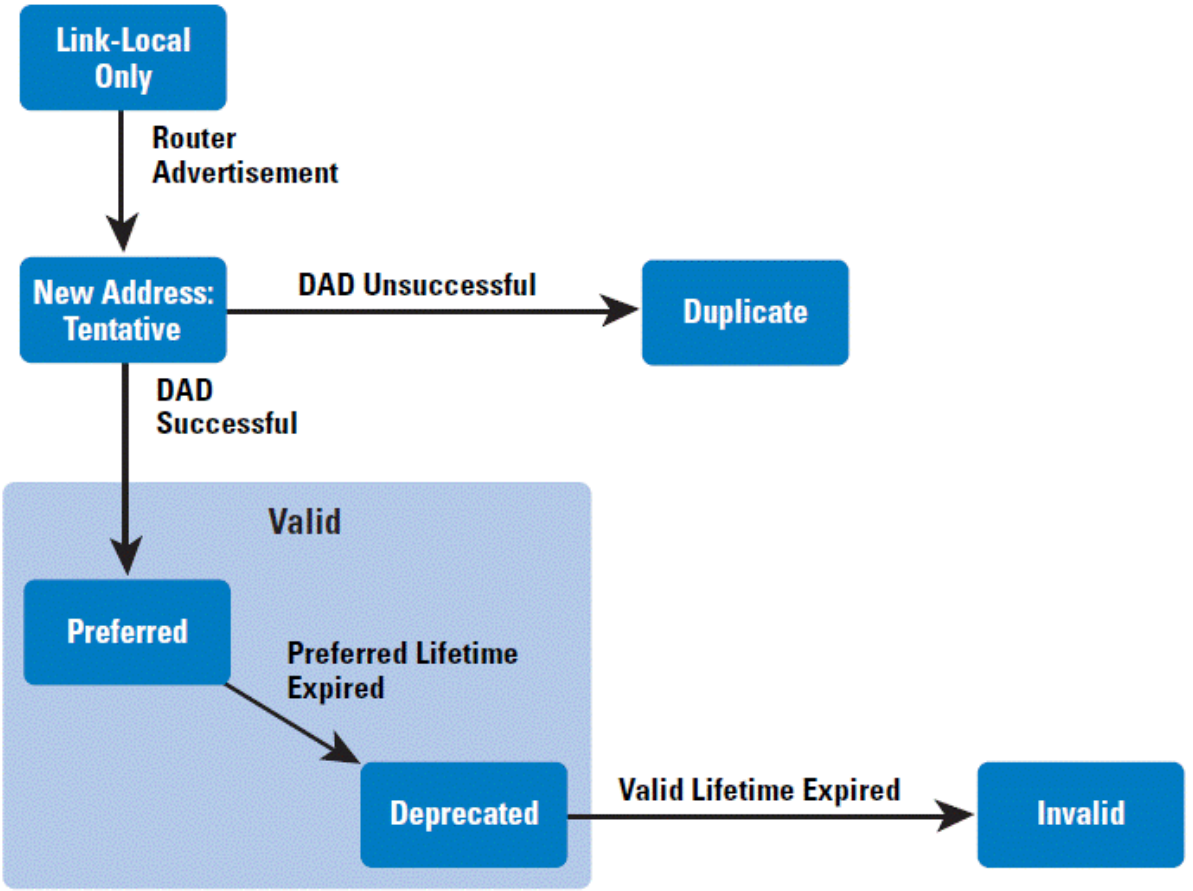
Breve reseña

- Existen dos mecanismos de autoconfiguración en IPv6:
  - Stateless Address Auto-Configuration (SLAAC)
    - Basado en mensajes ICMPv6
  - DHCPv6
    - Basado en paquestes UDP
- SLAAC es mandatorio, mientras que DHCPv6 es opcional
- Operación básica de SLAAC:
  - Los hosts solicitan información de configuración mediante mensajes Router Solicitation
  - Los routers proporcionan dicha información mediante Router Advertisements:
    - Prefijos de Auto-configuration
    - Rutas
    - Parámetros de red
    - etc.

SLAAC: Paso por paso

- Funciona (aproximadamente) así:
  1. El host genera una dirección link-local “tentativa”
  2. El host chequea que la dirección no está en uso – es decir, realiza Duplicate Address Detection (DAD) para esa dirección
     - Envía un NS, y espera respuestas
  3. El host envía un Router Solicitation
  4. Al recibir un Router Advertisement, genera una dirección IPv6 tentativa
  5. El host chequea que la dirección no está en uso – es decir, realiza Duplicate Address Detection (DAD) para esa dirección
     - Envía un NS, y espera respuestas
  6. Si la dirección es única, comienza a utilizarla

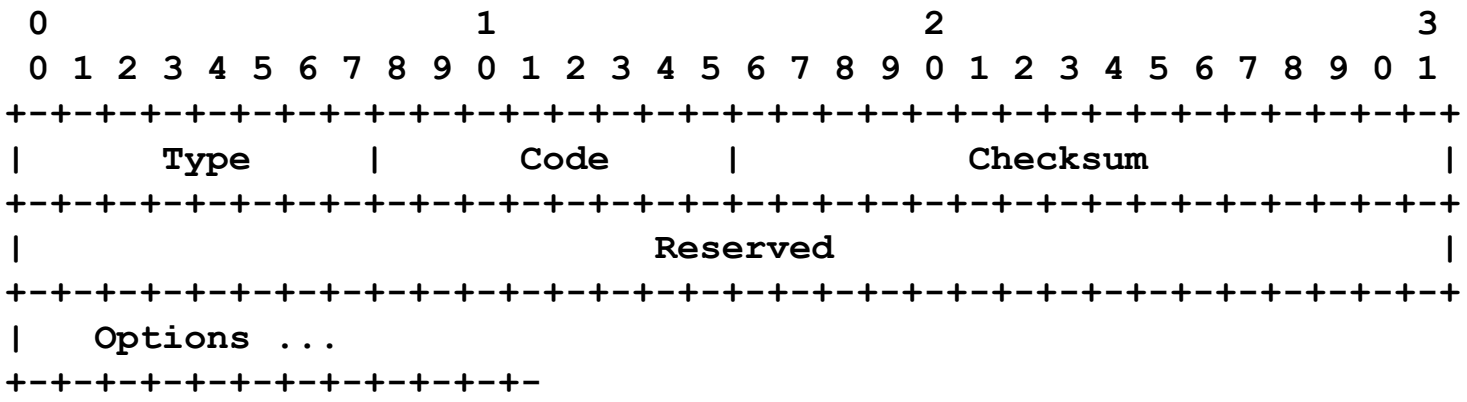
Diagrama de flujos de SLAAC



SI6 NETWORKS

Mensajes Router Solicitation

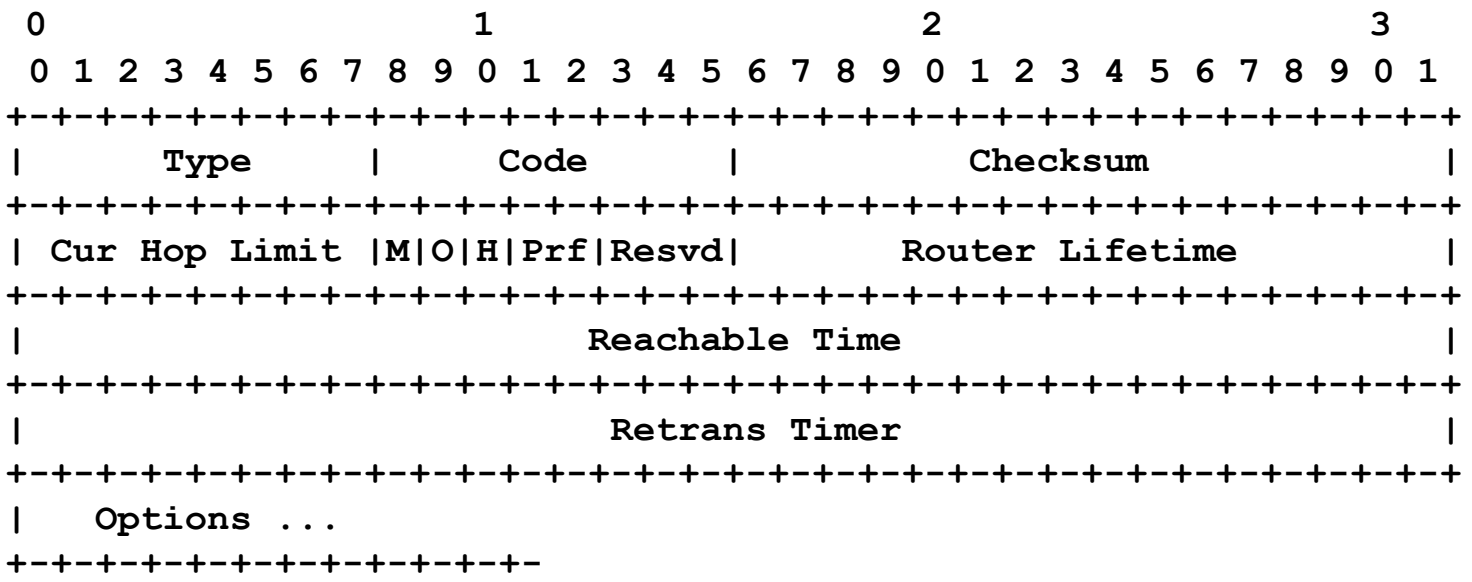
- Mensajes ICMPv6 de Type 133, Code 0
- Utilizados para solicitar información a un router local
- Unica opción permitida: Source Link-layer Address



SI6 NETWORKS

Mensajes Router Advertisement

- Mensajes ICMPv6 de Type 134, Code 0
- Utilizados para anunciar información de configuración



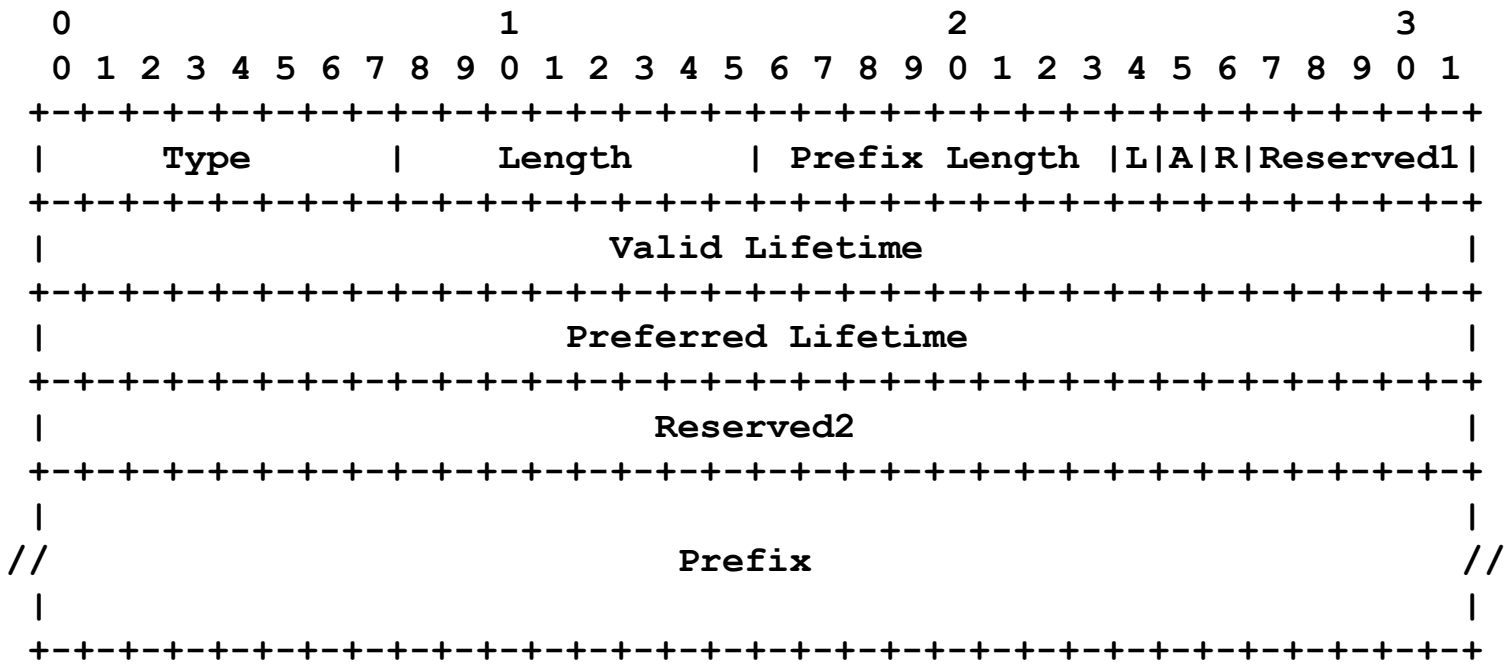
SI6 NETWORKS

Opciones disponibles en mensajes RA

- Los ICMPv6 Router Advertisements pueden contener las siguientes:
  - Source Link-layer address
  - Prefix Information
  - MTU
  - Route Information
  - Recursive DNS Server
- Usualmente incluyen varias de ellas

Prefix Information Option

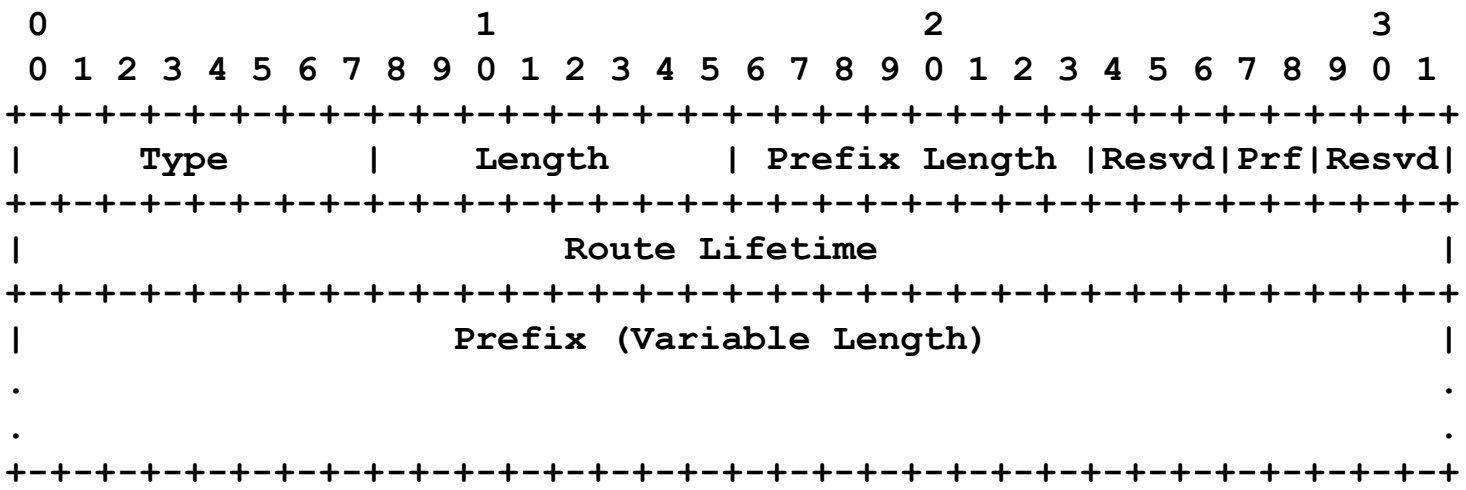
- dentificada por Type=3
- Especifica prefijos “on-link” y de “auto-configuración”



SI6 NETWORKS

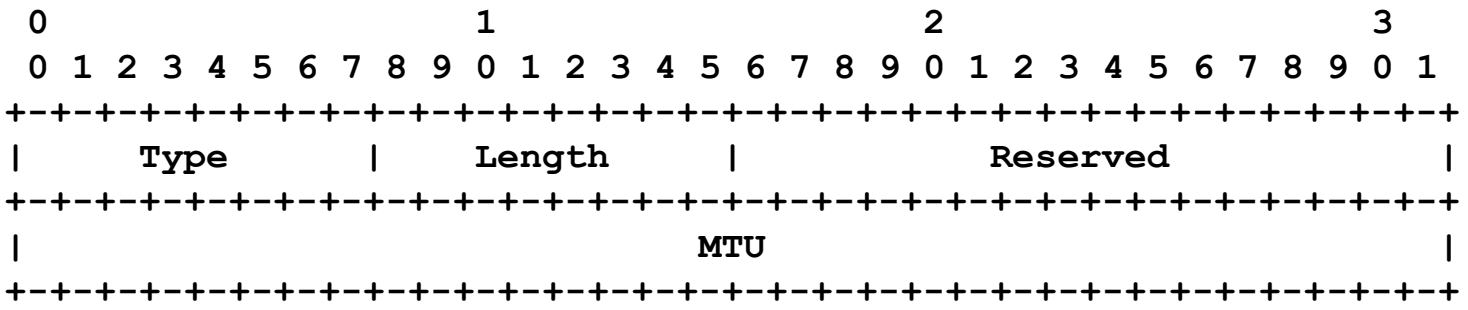
Router Information Option

- Type=24
- Anuncia rutas mas especificas, y con distintas prioridades



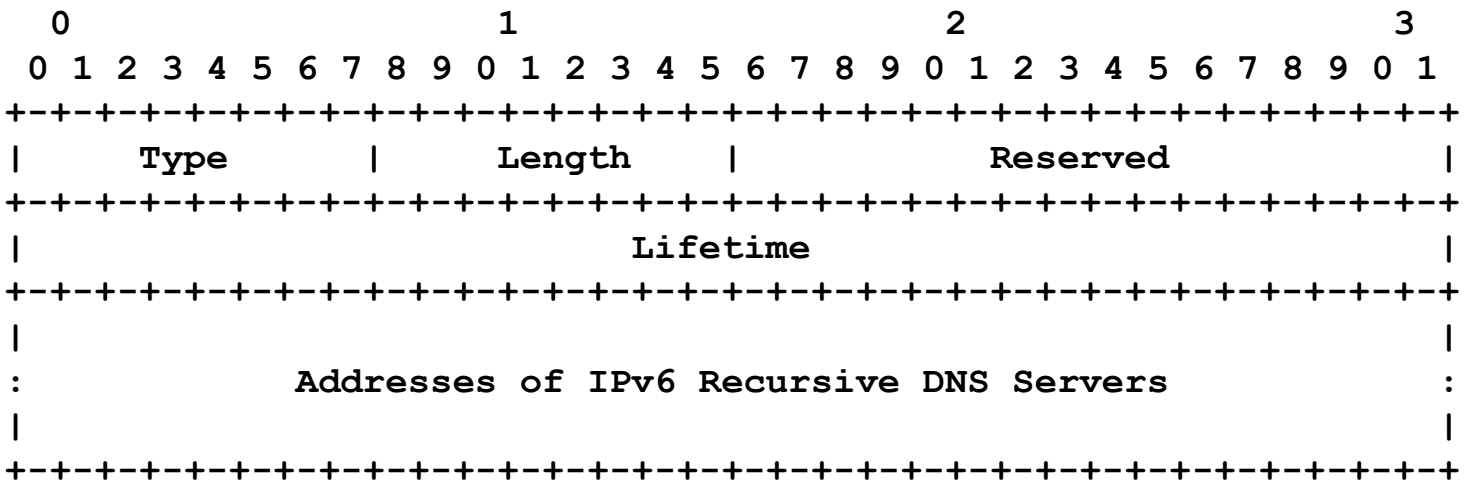
MTU Option

- Type=5
- Especifica el MTU a ser utilizado en este enlace



RDNSS Option

- Type=24
- Utilizada para anunciar servidores DNS recursivos



SI6 NETWORKS

SLAAC: Ejemplo de packet log

```
17:28:50 :: > ff02::1:ffaf:1958: icmp6: neighbor sol: who has
fe80::20c:29ff:feaf:1958 (len 24, hlim 255)

17:28:52 fe80::20c:29ff:feaf:1958 > ff02::2: icmp6: router solicitation (src
lladdr: 00:0c:29:af:19:58) (len 16, hlim 255)

17:28:52 fe80::20c:29ff:fec0:97b8 > ff02::1: icmp6: router advertisement(chlim=64,
router_ltime=1800, reachable_time=0, retrans_time=0)(src lladdr: 00:0c:29:c0:97:b8)
(prefix info: LA valid_ltime=2592000, preferred_ltime=604800, prefix=2004:1::/64)
(len 56, hlim 255)

17:28:52 :: > ff02::1:ffaf:1958: icmp6: neighbor sol: who has
2004:1::20c:29ff:feaf:1958 (len 24, hlim 255)
```

rdisc6: Troubleshooting tool

- Envia mensajes RS, y decodifica los mensajes RA responses
- Ejemplo de salida:

```
# rdisc6 -v eth0
Soliciting ff02::2 (ff02::2) on eth0...

Hop limit                 :           64 (      0x40)
Stateful address conf.    :           No
Stateful other conf.      :           No
Router preference         :       medium
Router lifetime           :           30 (0x0000001e) seconds
Reachable time            :  unspecified (0x00000000)
Retransmit time           :  unspecified (0x00000000)
 Prefix                   : fc00:1::/64
  Valid time              :      2592000 (0x00278d00) seconds
  Pref. time              :       604800 (0x00093a80) seconds
 Source link-layer address: 00:4F:4E:12:88:0F
 from fe80::24f:4eff:fe12:880f
```


SI6 NETWORKS

Información de prefijos (*BSD)

- % ndp -p

```
% ndp -p
2004::/64 if=em0
flags=LAO vltime=2592000, pltime=604800, expire=29d23h57m4s, ref=2
  advertised by
    fe80::20c:29ff:fec0:97ae%em0 (reachable)
2004:1::/64 if=em1
flags=LAO vltime=2592000, pltime=604800, expire=29d23h50m34s, ref=2
  advertised by
    fe80::20c:29ff:fec0:97b8%em1 (reachable)
fe80::%em1/64 if=em1
flags=LAO vltime=infinity, pltime=infinity, expire=Never, ref=0
  No advertising router
fe80::%em0/64 if=em0
flags=LAO vltime=infinity, pltime=infinity, expire=Never, ref=0
  No advertising router
fe80::%lo0/64 if=lo0
flags=LAO vltime=infinity, pltime=infinity, expire=Never, ref=0
  No advertising router
```

Default routers (*BSD)

- % ndp -r

```
% ndp -r
fe80::20c:29ff:fec0:97b8%em1 if=em1, flags=, pref=medium, expire=20m23s
fe80::20c:29ff:fec0:97ae%em0 if=em0, flags=, pref=medium, expire=26m53s
```

Routing table (*BSD)

- % netstat -r -p ip6

```
# netstat -r -p ip6
Internet6:
Destination        Gateway            Flags      Netif Expire
::                 localhost          UGRS        lo0 =>
default            fe80::20c:29ff:fec UG          em1
localhost          localhost          UH          lo0
::ffff:0.0.0.0     localhost          UGRS        lo0
2004:1::           link#2             U           em1
2004:1::20c:29ff:f link#2             UHS         lo0
2004:1::f8dd:347d: link#2             UHS         lo0
fe80::             localhost          UGRS        lo0
fe80::%em1         link#2             U           em1
fe80::20c:29ff:fe4 link#2             UHS         lo0
fe80::%lo0         link#5             U           lo0
fe80::1%lo0        link#5             UHS         lo0
ff01:1::           fe80::20c:29ff:fe4 U           em0
ff01:2::           fe80::20c:29ff:fe4 U           em1
ff01:5::           localhost          U           lo0
ff02::             localhost          UGRS        lo0
ff02::%em1         fe80::20c:29ff:fe4 U           em1
ff02::%lo0         localhost          U           lo0
```


SI6 NETWORKS

Algunas sysctl’s de SLAAC (OpenBSD)

- net.inet6.ip6.accept_rtadv (defaults to 1): Controla si se aceptan los Router Advertisements.
- net.inet6.ip6.dad_count (defaults to 1): Numero de pruebas DAD a enviar cuando se levanta una interfaz
- net.inet6.ip6.maxifprefixes (defaults to 16): Maximo número de prefijos por interfaz.
- net.inet6.ip6.maxifdefrouters (defaults to 16): Maximo numero de “default routers” por interfaz.

Direcciones SLAAC y privacidad

- Las direcciones SLAAC tradicionales incluyen la MAC address
- Esto permite el traceo de hosts
- Las “extensiones de privacidad” (RFC 4941) mitigan este problema
  - El Interface ID se setea a un número aleatorio
  - Las direcciones son temporales
  - Esto hace dificultosa la correlación de eventos por los administradores de red

Algunas sysctl's para Privacy Addresses

- Sysctl's que controlan la operación de Privacy Addresses (en FreeBSD):
  - net.inet6.ip6.use_tempaddr (defaults to 0)
    - Controls whether Privacy addresses are configured
  - net.inet6.ip6.tempppltime (defaults to 86400)
    - Specifies the "preferred lifetime" for privacy addresses
  - net.inet6.ip6.tempvltime (defaults to 604800)
    - Specifies the "valid lifetime" for privacy addresses
  - net.inet6.ip6.prefer_tempaddr (defaults to 0)
    - Controls whether privacy addresses are "preferred" (i.e., whether outgoing "conections" should use privacy addresses)

Neighbor Discovery for IPv6

Ataques contra SLAAC

Explotar DAD para ataques de DoS

- Escuchar mensajes NS con la Source Address igual a la IPv6 “unspecified” address (::).
- Responder dichos mensajes con un mensaje Neighbor Advertisement
- Como resultado, la dirección se considerará “en uso”, y DAD fallará
- El host no podrá utilizar dicha dirección “tentativa”
- Realizar el ataque con na6 de esta manera:

  # ./na6 -i IFACE -b ::/128 -c -o -L -vv

  O también:

  # ./na6 -i em0 -b ::/128 -B VICTIMMAC -c -o -L -vv

Anunciar un Hop Limit malicioso

- Anunciar un Hop Limit que haga que los paquetes sean descartados por alguno de los routers intermediarios
- Realizar el ataque de la siguiente manera:

  # ./ra6 –i IFACE –s ROUTERADDR –d TARGETADDR –c HOPS –v

Deshabilitar algún router existente

- Enviar un Router Advertisement con la Source Address del router local
- Setear el “Router Lifetime” a 0 (o a algún otro valor pequeño)
- Como resultado, la víctima quitará al router de la lista de “default routers”
- Realizar este ataque con la herramienta ra6:

  # ./ra6 -i IFACE -s ROUTERADDR -d TARGETADDR -t 0 -l 1 –v

Inundar a los hosts con prefijos SLAAC

- Algunos sistemas no imponen límites en la cantidad de direcciones que configuran
- Realizar este ataque con la herramienta ra6 de esta manera:

  # ./ra6 -i IFACE -d TARGETADDR --flood-prefixes 40 -P ::/64#LA -l -z 10 -e -vvv

Inundar a los hosts con rutas específicas

- Realizar el ataque con:

  # ./ra6 -i IFACE -d TARGETADDR --flood-routes 40 -R ::/64#1 -l -z 10 -e -vvv

Flood hosts with default routers

- Flood the local network with auto-configuration prefixes
- Perform this attack with the ra6 tool as follows:

  # ./ra6 -i IFACE -d TARGETADDR --flood-sources 40 -l -z 10 -e -vv

Neighbor Discovery for IPv6
SLAAC attacks – Countermeasures

Possible mitigations for SLAAC attacks

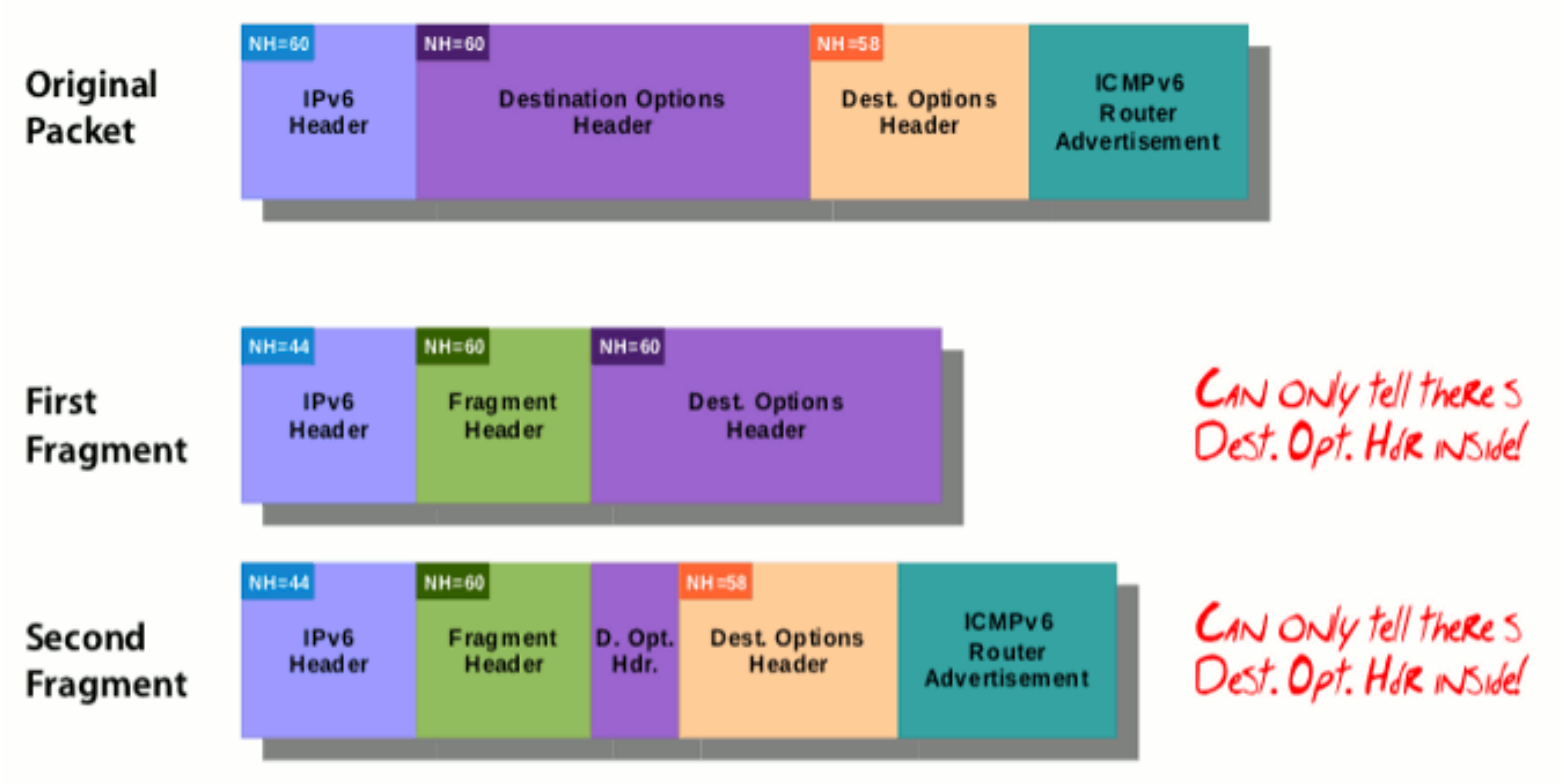
- Deploy SEND (SEcure Neighbor Discovery)
- Monitor Neighbor Discovery traffic (e.g., with NDPMon)
- Restrict access to the local network
- Deploy Router Advertisement Guard (RA-Guard)

RA-Guard (Router Advertisement Guard)

- Filtering policy enforced by layer-2 devices
- Works (roughly) as follows:
  - RA-Guard allows RAs only if they are received on pre-specified ports
  - Otherwise, they are dropped
- RA-Guard **asumes** that it is possible to identify RAs
- All known implementations can be evaded with IPv6 Extension Headers and/or fragmentation

RA-Guard evasion

- Fundamental problem: complexity of traffic to be “processed at layer-2”
- Example:



SI6 NETWORKS

Fixing RA-Guard

- In essence,
  - Follow the entire IPv6 header chain when trying to identify RAs
  - Drop the packet if it is an RA or you cannot positively determine that the packet is non-RA
- Ongoing work at the IETF to fix RA-Guard:
  - draft-ietf-v6ops-ra-guard-implementation
  - More human-readable explanation at: <http://blog.si6networks.com>

Soporte de IPsec

Breve reseña y consideraciones

Mito: "*IPv6 es mas seguro que IPv4 porque la seguridad fue considerada durante el diseño del protocolo*"

- Debe su origen a que IPsec era **opcional** para IPv4, y **mandatorio** para IPv6 (hoy es opcional para ambos)
- En la práctica, esto fue/es irrelevante:
  - Es mandatorio el soporte, pero no así su uso
  - Las implementaciones no respetan el estándar
  - Existen en IPv6 los mismos obstaculos para IPsec que en IPv4
- Incluso la IETF reconoció esta situación
- Conclusión:
  - El despliegue de IPv6 no implica un mayor uso de IPsec

Implicancias de seguridad de los mecanismos de transición

Breve reseña

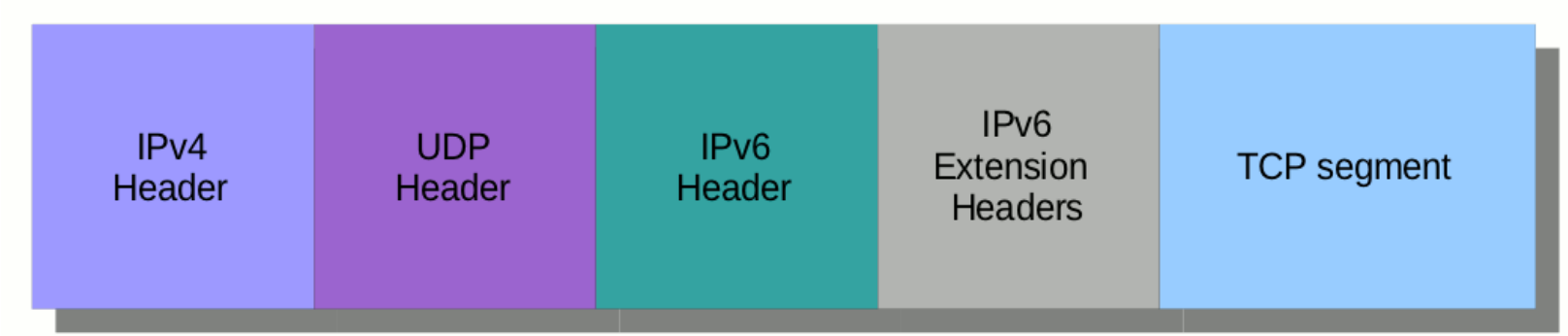
- Plan original de transición: doble pila (dual stack)
  - Desplegar IPv6 en paralelo con IPv4 **antes** de **necesitar** IPv6
  - Este plan **falló**
- La estrategia actual es transición/co-existencia basada en:
  - Doble pila
  - Túneles
    - Automáticos
    - Configurados
  - Traducción
    - CGN
    - NAT64
- La mayoria de los sistemas soportan algunos de estos mecanismos

Consideraciones de seguridad

- Se incrementa la complejidad de la red
- Se introducen “Puntos Únicos de Fallo” (Single Points of Failure)
- Se pueden utilizar las tecnologías de transición para evitar controles de seguridad
- Tecnologías como Teredo pueden aumentar la exposición de los hosts en la red
- Algunas tecnologías tienen implicancias de privacidad:
  - ¿Por dónde circula su tráfico Teredo o 6to4?
  - Esto puede (o no) ser problemático para su organización

Consideraciones de seguridad (II)

- La complejidad del tráfico aumenta notablemente
- Se dificulta la realización de “Deep Packet Inspection”
- Ejemplo: Estructura de un paquete “Teredo”:



- “Ejercicio”: construir filtro libpcap para capturar paquetes destinados al host 2001:db8::1, puerto TCP 25

Implicancias de seguridad de IPv6 en redes IPv4

Breve reseña

- La mayoría de los sistemas tiene algún tipo de soporte IPv6 habilitado “por defecto”
  - Doble pila
  - Teredo
  - ISATAP
  - etc
- Por ende,
  - La mayoría de las “redes IPv4” tienen al menos un **despliegue parcial de IPv6**
  - **IPv6 también afecta a redes que supuestamente “solo soportan IPv4”**

Fuga de tráfico en VPNs

- Escenario típico:
  - Te conectás a una red insegura
  - Establecés una VPN con tu casa/oficina
  - **Tu cliente VPN no soporta IPv6**
- Escape de tráfico en escenario legítimo:
  - Tu red local soporta IPv6
  - Si tu sistema prefiere IPv6 sobre IPv4, se generará tráfico IPv6 por fuera de la VPN
- Escape de tráfico en escenario malicioso:
  - Falsificá RA's o paquetes DHCPv6 que configuren el servidor DNS recursivo, y habiliten la conectividad IPv6
  - El tráfico IPv6 circulará por fuera de la VPN

Consideraciones de seguridad

- Se puede habilitar la conectividad IPv6 “durmiente”
  - Enviando Router Advertisements
  - Habilitando tecnologías de transición/co-existencia
- Las tecnologías de transición pueden aumentar la exposición de sistemas
  - Teredo permite el “traspaso” de NATs por sistemas externos
- En conclusión,
  - No existen redes IPv4 “puras”
  - Siempre se deben considerar las implicancias de seguridad de IPv6
  - Si no desea utilizar IPv6, asegurese que ese sea el caso

Áreas en las que se necesita más trabajo

Áreas en las que se necesita mas trabajo

- Seguridad de implementaciones
  - Todavía no han sido foco de ataque
  - Pocas herramientas de auditoria
  - Se descubrirán muchos bugs y vulnerabilidaes
- Soporte de IPv6 en dispositivos de seguridad
  - Se necesita paridad de funcionalidad IPv6/IPv4
  - Caso contrario, no se pueden aplicar las misms políticas de seguridad
- Educación/Entrenamiento
  - Es una locura desplegar IPv6 con “recetas de cocina”
  - Se necesita entrenamiento para todo el personal involucrado
  - Primero entrenarse, luego desplegar IPv6

Algunas conclusiones

Algunas conclusiones....

- Estar atentos al marketing y mitología sobre IPv6
  - Confiar en ellos tiene sus implicancias
- IPv6 provee una *funcionalidad* similar a IPv4
  - Los *mecanismos* utilizados son distintos
  - En dichas diferencias pueden aparecer las “sorpresas”
- La mayoría de los sistemas tiene soporte IPv6
  - Usualmente no existen redes IPv4 “puras”
  - Toda red debe considerar las implicancias de seguridad de IPv6
- Tarde o temprano desplegarás IPv6
  - Es hora de capacitarse y expetimentar con IPv6
  - Sólo después debe desplegarse el mismo

Preguntas?

Gracias!

Fernando Gont

fgont@si6networks.com

IPv6 Hackers mailing-list

http://www.si6networks.com/community/

www.si6networks.com