

Desplegando la Red IPv6

El reto IPv6

Seguridad

Lic. Oscar G. Acosta





Introducción

En mas de un 95% de los casos las vulnerabilidades que pueden ser aprovechadas de forma maligna para ganar el control de una máquina o con otros fines no las proporciona el protocolo de red como tal, sino las propias Aplicaciones o el Sistema Operativo que tiene instalado la máquina.

Por ello IPv6 no se puede considerar más inseguro que su predecesor IPv4 y la gran mayoría de las medidas de protección que en la actualidad se usan para IPv4 son igualmente válidas para IPv6.

Desde el punto de vista de seguridad con IPv6 las amenazas se pueden clasificar en tres grandes grupos:

- Las que ya existían con IPv4 y tienen un comportamiento similar con IPv6.
- Las que ya existían con IPv4 e introducen nuevas consideraciones.
- Las que aparecen debido a las características propias de IPv6.



IPv4

IPv4 ... (1970)

- Ataques de denegación (distribuída) de servicios, DoS y DDoS. Las inundaciones de información en *broadcasting* y los ataques Smurf, Fraggels, etc.
- La distribución de código malicioso automatizado, ya que el espacio IPv4 es corto y está saturado con lo que se da en una IP tirando el dardo sin mirar.
- Los ataques Man in the Middle, IPv4 no tiene características propias para proporcionar autenticación fuerte por sí mismo.
- Ataques de fragmentación, en los que se aprovecha lo mal que gestionan a veces las pilas de protocolo de algunos sistemas operativos la información fragmentada. Ej. Nukes OOB (Out Of Band).
- Ataques de reconocimiento y de escaneo de servicios. Escanear una clase C entera puede llevar bastante poco tiempo.
- Envenenamiento ARP y redirección de eco ICMP, o cómo desviar el tráfico a una dirección maliciosa.

SoftPerfect Network Protocol Analyzer

File Edit View Tools Filters Capture Analysis Help

Network Interface: Marvell Yukon 88E8055 PCI-E Gigabit Ethernet Controller [10.30.10.60] Start Capture

IPv4 Cabecera

Capture Data Flows Packet Builder

Ethernet header

- Destination: FF:FF:FF:FF:FF:FF
- Source: 00:11:11:75:19:53
- Type: 0x0004 (Internet Protocol)

IPv4 header

- Version: 4 (IP, Internet Protocol)
- Header length: 5 (20 bytes)
- Type of service: 0x00
 - 000... (Routine)
 - ...0... (Normal delay)
 - ...0... (Normal throughput)
 - ...0... (Normal reliability)
 - ...0... (Normal monetary cost)
 - ...0 (Reserved flag is not set)
- Total length: 78 bytes
- Identification: 51672
- Flags: 0x00
 - DF 0... (May fragment)
 - MF 0... (Last fragment)
- Fragment offset: 0
- Time to live: 128 hops (seconds)
- Protocol: UDP (0x11)
- Header checksum: 0xEE10 (Correct)
- Source IP: 192.168.0.102
- Destination IP: 192.168.0.255
- No options

UDP header

- Source port: 137 Netbios-ns
- Destination port: 137 Netbios-ns
- Length: 58 bytes
- Checksum: 0x440D (Correct)

Data: 50 bytes

Time	MAC Source	MAC Destination	Frame	Protocol	IP Source	IP Destination	Port So...	Port ...
13:52:25.515	00:1A:80:D1:6C:4B	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-ns	192.168.0.5	192.168.0.255	137	137
13:42:27.203	00:1A:80:D1:6C:4B	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-ns	192.168.0.5	192.168.0.255	137	137
13:52:32.359	00:11:11:75:19:53	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-ns	192.168.0.102	192.168.0.255	137	137
13:52:33.109	00:11:11:75:19:53	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-ns	192.168.0.102	192.168.0.255	137	137
13:42:27.265	00:13:20:5F:EE:95	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-ns	192.168.0.168	192.168.0.255	137	137
13:52:33.859	00:11:11:75:19:53	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-ns	192.168.0.102	192.168.0.255	137	137
13:40:46.531	00:40:F4:69:32:30	FF:FF:FF:FF:FF:FF	IP	UDP->Locus-map	192.168.0.253	192.168.0.255	127	125
14:04:43.062	00:11:11:75:18:03	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-ns	192.168.0.202	192.168.0.255	137	137
13:52:37.062	00:13:20:74:70:82	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-dgm	192.168.0.241	192.168.0.255	138	138
13:42:27.843	00:11:11:75:18:03	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-ns	192.168.0.202	192.168.0.255	137	137
13:52:35.281	00:13:20:99:8A:23	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-dgm	192.168.0.139	192.168.0.255	138	138
13:42:27.125	00:11:11:75:19:53	FF:FF:FF:FF:FF:FF	IP	UDP->Netbios-ns	192.168.0.102	192.168.0.255	137	137
14:03:26.390	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-1-3	57906	5355
14:03:26.484	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-1-3	57906	5355
14:03:23.640	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-1-3	49232	5355
14:05:03.343	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-1-3	60717	5355
14:03:23.734	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-1-3	49232	5355
14:05:03.250	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-1-3	60717	5355
14:02:56.484	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-1-3	52275	5355
14:02:56.375	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-1-3	52275	5355
13:52:58.671	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:58.062	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:57.921	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:58.781	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:59.031	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-1-3	61773	5355
13:52:59.031	00:1A:80:D1:6C:4B	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-16	---	---
13:52:59.031	00:1A:80:D1:6C:4B	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-16	---	---
13:52:57.171	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900

0x00 FFFF FFFF FFFF 0011 1175 1953 0800 4500 004E C9D8 0000 8011 EE10 C0A8 0066

0x1E C0A8 00FF 0089 0089 003A 440D 862A 0110 0001 0000 0000 0000 2045 4F46 4443

0x3C 4F46 4A45 4245 4945 5045 5043 4F45 4445 5045 4E43 4143 4143 4141 4100 0020

0x5A 0001

Captured 0 Filtered 5954 Link Speed 0 Memory 2% CPU usage 0%



IPv6

IPv6 ... (1992)

- Proporción un espacio mucho mayor. Pasamos de 2^{32} a 2^{128} .
- Direccionamiento jerárquico. Desde el *unicast* asignadas a un sólo nodo IPv6, al *anycast* de ciertos integrantes de un grupo determinado , al *multicast* (un grupo de nodos). ¿Resultado? Tablas de enrutamiento mucho menores. Comunicaciones más optimizadas.
- Calidad de servicio, QoS. Las cabeceras IPv6 contienen información específica que facilita la gestión del Quality of Service tanto para servicios diferenciados como integrados.
- Más rendimiento. Mejoran la gestión de paquetes y los tiempos de proceso.
- Pensado para la seguridad IPsec en IPv6.
- Extensibilidad. Las cabeceras IPv6 doblan en tamaño a las IPv4, pero sin embargo, las direcciones IPv6 son cuatro veces más largas. Las cabeceras IPv6 no contienen campos opcionales, lo que queramos enviar como opcional se hace vía cabeceras auxiliares. Esto reduce cómputo y tiempos, y simplifica la gestión.
- Movilidad. Trasladar nodos sin perder tiempo de operación es algo asumible en IPv6, mucho más fácil de lograr que en IPv4.

El tráfico enviado a una dirección anycast llega únicamente a un destinatario. Siendo el propio protocolo IPv6 el que selecciona, de entre múltiples posibles destinatarios, el más conveniente.

SoftPerfect Network Protocol Analyzer

File Edit View Tools Filters Capture Analysis Help

Network Interface: Marvell Yukon 88E8055

IPv6 ya existe en nuestras LAN !!!!

Capture Data Flows Packet Builder

Ethernet header

- Destination: 33:33:00:01:00:02
- Source: 00:1C:C0:22:57:94
- Type: 0x86DD Internet Protocol v6

IPv6 header

- Version: 6 (SIP, SIPP or IPv6)
- Traffic Class: Uncharacterized traffic (0x00)
- Flow Label: 0x00000
- Payload Length: 93 bytes
- Next header: UDP (0x11)
- Hop limit: 1
- Source Address: FE80:0-0-0-3C66-B28E-9BF1-202A
- Destination Address: FF02:0-0-0-0-0-1-2

UDP header

- Source port: 546 Dhcipv6-client
- Destination port: 547 Dhcipv6-server
- Length: 93 bytes
- Checksum: 0xA190 (Correct)
- Data: 85 bytes

MAC Source	MAC Destination	Frame	Protocol	IP Source	IP Destination	Port So...	Port ...	SEQ	A..	Size
00:1C:C0:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-3	52988	5355	---	---	84
00:1C:C0:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcipv6-server	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-2	546	547	---	---	147
00:1C:C0:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcipv6-server	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-2	546	547	---	---	147
00:1C:C0:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-3	51259	5355	---	---	84
00:1C:C0:22:57:94	33:33:00:00:00:16	IPv6	IPv6HOP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-0-16	---	---	---	---	90
00:1C:C0:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcipv6-server	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-2	546	547	---	---	147
00:1C:C0:22:57:94	33:33:00:00:00:16	IPv6	IPv6HOP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-0-16	---	---	---	---	90
00:1C:C0:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcipv6-server	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-2	546	547	---	---	147
00:1C:C0:22:57:94	33:33:00:00:00:16	IPv6	IPv6HOP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-0-16	---	---	---	---	90
00:1C:C0:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcipv6-server	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-2	546	547	---	---	147
00:1C:C0:22:57:94	33:33:00:00:00:0C	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-0-C	51167	1900	---	---	181
00:1C:C0:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-3	55971	5355	---	---	85
00:1C:C0:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-3	55971	5355	---	---	85
00:1C:C0:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcipv6-server	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-2	546	547	---	---	147
00:1C:C0:22:57:94	33:33:00:00:00:16	IPv6	IPv6HOP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-0-16	---	---	---	---	90
00:1C:C0:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-3	49432	5355	---	---	86
00:1C:C0:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcipv6-server	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-2	546	547	---	---	147
00:1C:C0:22:57:94	33:33:00:00:00:0C	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-0-C	51167	1900	---	---	181
00:1C:C0:22:57:94	33:33:00:00:00:0C	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-0-C	51167	1900	---	---	179
00:1C:C0:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcipv6-server	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-2	546	547	---	---	147
00:1C:C0:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-3	54750	5355	---	---	86
00:1C:C0:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-3	49432	5355	---	---	86
00:1C:C0:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcipv6-server	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-2	546	547	---	---	147
00:1C:C0:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcipv6-server	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-2	546	547	---	---	147
00:1C:C0:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80-0-0-0-3C66-B28E-9BF1-202A	FF02-0-0-0-0-0-1-3	54750	5355	---	---	86

0x00

0x1E

0x3C

0x5A

0x78

```

3333 0001 0002 001c c022 5794 86DD 6000 0000 005D 1101 FE80 0000 0000 0000
3C66 B28E 9BF1 202A FF02 0000 0000 0000 0000 0001 0002 0222 0223 005D
A190 017B EF3C 0008 0002 012c 0001 000E 0001 0001 1570 2EDC 001c c022 5794
0003 000c 0E00 1cc0 0000 0000 0000 0000 0027 0007 0005 7A61 6861 7900 1000
0E00 0001 3700 084D 5346 5420 352E 3000 0600 0800 1800 1700 1100 27

```

```

33.....À"w"tY`...].p€......
<f*žžñ *ÿ.....".#.]
{<.....p+ü..À"w"
.....À.....'.....zahay...
....7..MSFT 5.0.....'

```

Captured 12276 | Filtered 12276 | Link Speed 0 | Memory 6% | CPU usage 0%

SoftPerfect Network Protocol Analyzer

File Edit View Tools Filters Capture Analysis Help

Network Interface: Marvell Yukon 88E8055 PCI

IPv6 plug and play !!!!

Capture Data Flows Packet Builder

Ethernet header

- Destination: 33:33:00:00:00:0C
- Source: 00:1C:00:22:57:94
- Type: 0x86DD Internet Protocol v6

IPv6 header

- Version: 6 (SIP, SIPP or IPv6)
- Traffic Class: Uncharacterized traffic (0x00)
- Flow Label: 0x00000
- Payload Length: 127 bytes
- Next header: UDP (0x11)
- Hop limit: 1
- Source Address: FE80:0:0:3C66:B28E:9BF1:202A
- Destination Address: FF02:0:0:0:0:0:0:0C

UDP header

- Source port: 51167
- Destination port: 1900
- Length: 127 bytes
- Checksum: 0xB538 (Correct)

Data: 119 bytes

MAC Source	MAC Destination	Frame	Protocol	IP Source	IP Destination	Port So...	Port ...	SEQ	A...	Size
00:1C:00:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:0:1-3	52988	5355	---	84	
00:1C:00:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-2	546	547	---	147	
00:1C:00:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-2	546	547	---	147	
00:1C:00:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-3	51259	5355	---	84	
00:1C:00:22:57:94	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:0-16	---	---	---	90	
00:1C:00:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-2	546	547	---	147	
00:1C:00:22:57:94	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:0-16	---	---	---	90	
00:1C:00:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-2	546	547	---	147	
00:1C:00:22:57:94	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:0-16	---	---	---	90	
00:1C:00:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-2	546	547	---	147	
00:1C:00:22:57:94	33:33:00:00:00:0C	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:0-C	51167	1900	---	181	
00:1C:00:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-3	55971	5355	---	85	
00:1C:00:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-3	55971	5355	---	85	
00:1C:00:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-2	546	547	---	147	
00:1C:00:22:57:94	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:0-16	---	---	---	90	
00:1C:00:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-3	49432	5355	---	86	
00:1C:00:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-2	546	547	---	147	
00:1C:00:22:57:94	33:33:00:00:00:0C	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:0-C	51167	1900	---	181	
00:1C:00:22:57:94	33:33:00:00:00:0C	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:0-C	51167	1900	---	179	
00:1C:00:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-2	546	547	---	147	
00:1C:00:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-3	54750	5355	---	86	
00:1C:00:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-3	49432	5355	---	86	
00:1C:00:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-2	546	547	---	147	
00:1C:00:22:57:94	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-2	546	547	---	147	
00:1C:00:22:57:94	33:33:00:01:00:03	IPv6	UDP	FE80:0:0:3C66:B28E:9BF1:202A	FF02:0:0:0:0:0:1-3	54750	5355	---	86	

0x00 3333 0000 000C 001C C022 5794 86DD 6000 0000 007F 1101 FE80 0000 0000 0000 33.....À"m"†Ý`....□...þ€.....

0x1E 3C66 B28E 9BF1 202A FF02 0000 0000 0000 0000 0000 0000 000C C7DF 076C 007F <f²Ž>ñ *ÿ.....Çß.1.□

0x3C B538 4D2D 5345 4152 4348 202A 2048 5454 502F 312E 310D 0A48 6F73 743A 5B46 µ8M-SEARCH * HTTP/1.1..Host:[F

0x5A 4630 323A 3A43 5D3A 3139 3030 0D0A 5354 3A75 726E 3A73 6368 656D 6173 2D75 F02::C]:1900..ST:urn:schemas-u

0x78 706E 702D 6F72 673A 6465 7669 6365 3A4D 6564 6961 5265 6E64 6572 6572 3A31 pnp-org:device:MediaRenderer:1

0x96 0D0A 4D61 6E3A 2273 7364 703A 6469 7363 6F76 6572 220D 0A4D 583A 330D 0A0D ..Man:"ssdp:discover"..MX:3...

Captured 12276 | Filtered 12276 | Link Speed 0 | Memory 6% | CPU usage 14%

SoftPerfect Network Protocol Analyzer

File Edit View Tools Filters Capture Analysis Help

Network Interface

IPv6Home agent address discovery request !!!!

Capture Data Flows Packet Builder

Ethernet header

- Destination: 33:33:00:00:00:16
- Source: 00:1A:80:D1:6C:4B
- Type: 0x86DD Internet Protocol v6

IPv6 header

- Version: 6 (SIP, SIPP or IPv6)
- Traffic Class: Uncharacterized traffic (0x00)
- Flow Label: 0x00000
- Payload Length: 36 bytes
- Next header: IPv6HOP (0x00)
- Hop limit: 1
- Source Address: FE80:0-0-0-E987-B9DB-CE09-3E06
- Destination Address: FF02:0-0-0-0-0-0-16

IPv6 Options

- Next header: ICMPv6 (0x3A)
- Length: 0x00 (8 bytes)

Router Alert

- Length: 4 bytes
- Router Alert: MLD message (0x00)

PADN

- Length: 2 bytes
- Option data: No data

ICMPv6 header

- Type: 143 (Home Agent Address Discovery Request)
- Code: 0 (None)
- Checksum: 0xC093 (Correct)
- Data: 24 bytes

Time	MAC Source	MAC Destination	Frame	Protocol	IP Source	IP Destination	Port So...	Port ...
13:52:58.062	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:57.921	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:58.781	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:59.031	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	61773	5355
13:52:59.031	00:1A:80:D1:6C:4B	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-16	---	---
13:52:59.031	00:1A:80:D1:6C:4B	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-16	---	---
13:52:57.171	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:55.671	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:55.625	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	52681	3702
13:52:55.500	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	52681	3702
13:52:55.765	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:56.375	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	61174	5355
13:52:56.265	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	61174	5355
13:52:56.093	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:59.109	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:03.921	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:03.171	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:02.109	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:04.062	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:31.437	00:1A:80:D1:6C:4B	33:33:00:01:00:02	IPv6	UDP>Dhcpv6-ser...	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-2	546	547
13:42:23.406	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	58341	5355
13:42:23.500	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	58341	5355
13:53:01.781	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:59.484	00:1A:80:D1:6C:4B	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-16	---	---
13:52:59.437	00:1A:80:D1:6C:4B	33:33:00:01:00:02	IPv6	UDP>Dhcpv6-ser...	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-2	546	547
13:52:59.140	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	61773	5355
13:53:00.171	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:01.671	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900

0x00 3333 0000 0016 001A 80D1 6C4B 86DD 6000 0000 0024 0001 FE80 0000 0000 0000 33.....€Ñlk+Y`....\$.p€.....

0x1E E987 B9DB CE09 3E06 FF02 0000 0000 0000 0000 0000 0000 0016 3A00 0502 0000 é†¹ûî.>.ÿ.....:.....

0x3C 0100 8F00 C093 0000 0001 0400 0000 FF02 0000 0000 0000 0000 0000 0001 0003 ..□.À"......ÿ.....

Captured 0 Filtered 5954 Link Speed 0 Memory 2% CPU usage 0%

SoftPerfect Network Protocol Analyzer

File Edit View Tools Filters Capture Analysis Help

Network Interface

IPv6 DHCP server request !!!!

Capture Data Flows Packet Builder

Ethernet header

- Destination: 33:33:00:01:00:02
- Source: 00:1A:80:D1:6C:4B
- Type: 0x86DD Internet Protocol v6

IPv6 header

- Version: 6 (SIP, SIPP or IPv6)
- Traffic Class: Uncharacterized traffic (0x00)
- Flow Label: 0x00000
- Payload Length: 96 bytes
- Next header: UDP (0x11)
- Hop limit: 1
- Source Address: FE80:0-0-0-E987-B9DB-CE09-3E06
- Destination Address: FF02:0-0-0-0-0-1-2

UDP header

- Source port: 546 Dhcpv6-client
- Destination port: 547 Dhcpv6-server
- Length: 96 bytes
- Checksum: 0x62C7 (Correct)
- Data: 88 bytes

Time	MAC Source	MAC Destination	Frame	Protocol	IP Source	IP Destination	Port So...	Port ...
13:53:03.921	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:03.171	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:02.109	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:04.062	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:31.437	00:1A:80:D1:6C:4B	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-2	546	547
13:42:23.406	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	58341	5355
13:42:23.500	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	58341	5355
13:53:01.781	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:59.484	00:1A:80:D1:6C:4B	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-16
13:52:59.437	00:1A:80:D1:6C:4B	33:33:00:01:00:02	IPv6	UDP->Dhcpv6-server	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-2	546	547
13:52:59.140	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	61773	5355
13:53:00.171	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:01.671	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:01.062	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:53:00.921	00:1A:80:D1:6C:4B	33:33:00:00:00:0C	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-C	1900	1900
13:52:55.484	00:1A:80:D1:6C:4B	33:33:00:00:00:16	IPv6	IPv6HOP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-0-16
13:50:03.171	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	60765	5355
13:50:03.078	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	60765	5355
13:49:26.328	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	50918	5355
13:52:23.515	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	58575	5355
13:52:26.359	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	59622	5355
13:52:26.265	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	59622	5355
13:52:23.609	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	58575	5355
13:49:26.234	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	50918	5355
13:48:53.578	00:1A:80:D1:6C:4B	33:33:00:01:00:03	IPv6	UDP	FE80:0-0-0-E987-B9DB-CE09-3E06	FF02:0-0-0-0-0-1-3	60829	5355

0x00 3333 0001 0002 001A 80D1 6C4B 86DD 6000 0000 0060 1101 FE80 0000 0000 0000 33.....€ÑlK†Ÿ`....`...p€.....

0x1E E987 B9DB CE09 3E06 FF02 0000 0000 0000 0000 0001 0002 0222 0223 0060 é†¹ûî.>..ÿ.....".#.`

0x3C 62C7 0141 A0FB 0008 0002 189C 0001 000E 0001 0001 0FA9 B464 001A 80D1 6C4B bÇ.A û.....œ.....@'d.€ÑlK

0x5A 0003 000C 0C00 1A80 0000 0000 0000 0000 0027 000A 0008 5065 6472 6F2D 5043€.....'.....Pedro-PC

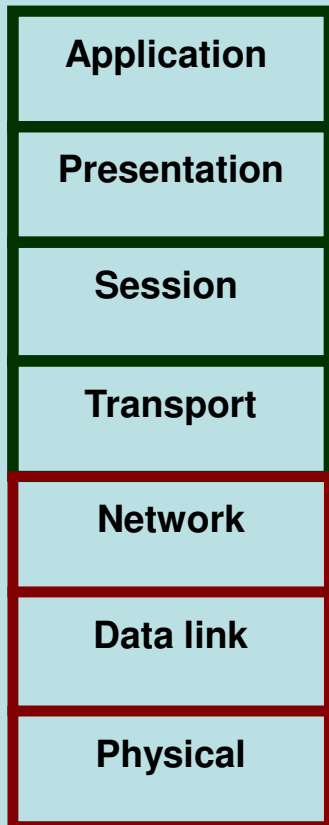
0x78 0010 000E 0000 0137 0008 4D53 4654 2035 2E30 0006 0008 0018 0017 0011 00277..MSFT 5.0.....'

Captured 0 Filtered 5954 Link Speed 0 Memory 2% CPU usage 0%



IPSec

IPsec (abreviatura de **Internet Protocol security**) es un conjunto de protocolos cuya función es asegurar las comunicaciones sobre el Protocolo de Internet (IP) autenticando y/o cifrando cada paquete IP en un flujo de datos. IPsec también incluye protocolos para el establecimiento de claves de cifrado.



Los protocolos de **IPsec** actúan en la capa de red, la capa 3 del modelo OSI. Otros protocolos de seguridad para Internet de uso extendido, como SSL, TLS y SSH operan de la capa de transporte (capas OSI 4 a 7) hacia arriba.

Esto hace que IPsec sea más flexible, ya que puede ser utilizado para proteger protocolos de la capa 4, incluyendo TCP y UDP, los protocolos de capa de transporte más usados. Una ventaja importante de IPsec frente a SSL y otros métodos que operan en capas superiores, es que para que una aplicación pueda usar IPsec no hay que hacer ningún cambio, mientras que para usar SSL y otros protocolos de niveles superiores, las aplicaciones tienen que modificar su código.



IPSec

Propósito de diseño.

Como el Protocolo de Internet no provee intrínsecamente de ninguna capacidad de seguridad, IPsec se introduce para proporcionar servicios de seguridad tales como:

- Cifrar el tráfico (de forma que no pueda ser leído por nadie más que las partes a las que está dirigido)
- Validación de integridad (asegurar que el tráfico no ha sido modificado a lo largo de su trayecto)
- Autenticar a los extremos (asegurar que el tráfico proviene de un extremo de confianza)
- Anti-repetición (proteger contra la repetición de la sesión segura).

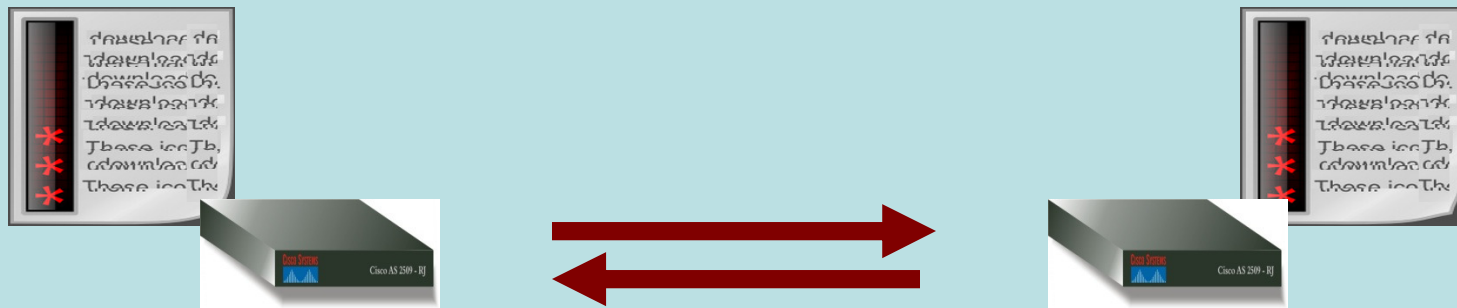


IPSec

Que IPSec sea nativo de IPv6 no significa que esté habilitado por defecto. Por ello, no supone ningún cambio con respecto a IPv4: la mayoría de las implementaciones de IPv4 soportan IPSec en la actualidad, pero otra cosa muy diferente es que pueda usarse IPSec para cualquier tipo de comunicación.

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 PKIs distan mucho de ser usables a nivel global. Además, IPSec no está soportado para comunicaciones como multicast, o broadcast, lo que limita en parte su usabilidad en redes globales.





IPSec

Para decidir qué protección se va a proporcionar a un paquete saliente, IPSec utiliza el índice de parámetro de seguridad (SPI), un índice a la base de datos de asociaciones de seguridad (SADB), junto con la dirección de destino de la cabecera del paquete, que juntos identifican de forma única una asociación de seguridad para dicho paquete.

Para un paquete entrante se realiza un procedimiento similar; en este caso IPSec toma las claves de verificación y descifrado de la base de datos de asociaciones de seguridad.

En el caso de multicast, se proporciona una asociación de seguridad al grupo, y se duplica para todos los receptores autorizados del grupo. Puede haber más de una asociación de seguridad para un grupo, utilizando diferentes SPIs, y por ello permitiendo múltiples niveles y conjuntos de seguridad dentro de un grupo. De hecho, cada remitente puede tener múltiples asociaciones de seguridad, permitiendo autenticación, ya que un receptor sólo puede saber que alguien que conoce las claves ha enviado los datos. Hay que observar que el estándar pertinente no describe cómo se elige y duplica la asociación a través del grupo; se asume que un interesado responsable habrá hecho la elección..



IPSec

IPsec puede utilizarse para crear VPNs en los dos modos, y este es su uso principal. Hay que tener en cuenta, sin embargo, que las implicaciones de seguridad son bastante diferentes entre los dos modos de operación (Transporte y Túnel).

Parte de la razón a esto es que no ha surgido infraestructura de clave pública universal o universalmente de confianza y el servicio **DNSSEC** fue originalmente previsto para esto.

La seguridad de comunicaciones extremo a extremo a escala Internet se ha desarrollado más lentamente de lo esperado.



IPSec

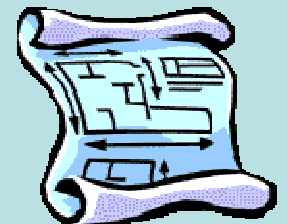
Detalles técnicos

IPsec consta de dos protocolos que han sido desarrollados para proporcionar seguridad a nivel de paquete.

Authentication Header (AH) proporciona integridad, autenticación y no repudio si se eligen los algoritmos criptográficos apropiados.

Encapsulating Security Payload (ESP) proporciona confidencialidad y la opción - altamente recomendable- de autenticación y protección de integridad.

Los algoritmos criptográficos definidos para usar con IPsec incluyen HMAC- SHA-1 para protección de integridad, y Triple DES-CBC y AES-CBC para confidencialidad. (RFC 4305).





IPSec

AH está dirigido a garantizar integridad sin conexión y autenticación de los datos de origen de los datagramas IP. Para ello, calcula un Hash Message Authentication Code (HMAC) a través de algún algoritmo hash operando sobre una clave secreta, el contenido del paquete IP y las partes inmutables del datagrama. Este proceso restringe la posibilidad de emplear NAT, que puede ser implementada con NAT transversal. Por otro lado, AH puede proteger opcionalmente contra ataques de repetición utilizando la técnica de ventana deslizante y descartando paquetes viejos. AH protege la carga útil IP y todos los campos de la cabecera de un datagrama IP excepto los campos mutantes, es decir, aquellos que pueden ser alterados en el tránsito. En IPv4, los campos de la cabecera IP mutantes (y por lo tanto no autenticados) incluyen TOS, Flags, Offset de fragmentos, TTL y suma de verificación de la cabecera. AH opera directamente por encima de IP, utilizando el protocolo IP número 51.

0 - 7 bit	8 - 15 bit	16 - 23 bit	24 - 31 bit
Next header	Payload length	RESERVED	
Security parameters index (SPI)			
Sequence number			
Hash Message Authentication Code (variable)			

Next header

Identifica el protocolo de los datos transferidos.

Payload length

Tamaño del paquete AH.

RESERVED

Reservado para uso futuro (todo ceros).

Security parameters index (SPI)

Indica los parámetros de seguridad que, en combinación con la dirección IP, identifican la asociación de seguridad implementada con este paquete.

Sequence number

Un número siempre creciente, utilizado para evitar ataques de repetición.

HMAC

Contiene el valor de verificación de integridad (ICV) necesario para autenticar el paquete; puede contener relleno.



IPSec

ESP proporciona autenticidad de origen, integridad y protección de confidencialidad de un paquete. ESP también soporta configuraciones de sólo cifrado y sólo autenticación, pero utilizar cifrado sin autenticación está altamente desaconsejado porque es inseguro. Al contrario que con AH, la cabecera del paquete IP no está protegida por ESP (aunque en ESP en modo túnel, la protección es proporcionada a todo el paquete IP interno, incluyendo la cabecera interna; la cabecera externa permanece sin proteger). ESP opera directamente sobre IP, utilizando el protocolo IP número 50.

0 - 7 bit	8 - 15 bit	16 - 23 bit	24 - 31 bit
Security parameters index (SPI)			
Sequence number			
Payload data (variable)			
Padding (0-255 bytes)			
		Pad Length	Next Header
Authentication Data (variable)			

Security parameters index (SPI)

Identifica los parámetros de seguridad en combinación con la dirección IP.

Sequence number

Un número siempre creciente, utilizado para evitar ataques de repetición.

Payload data

Los datos a transferir.

Padding

Usado por algunos algoritmos criptográficos para rellenar por completo los bloques.

Pad length

Tamaño del relleno en bytes.

Next header

Identifica el protocolo de los datos transferidos.

Authentication data

Contiene los datos utilizados para autenticar el paquete.



IPSec

Modos

Así pues y dependiendo del nivel sobre el que se actúe, podemos establecer dos modos básicos de operación de **IPsec**: **modo transporte** y **modo túnel**.

Modo transporte

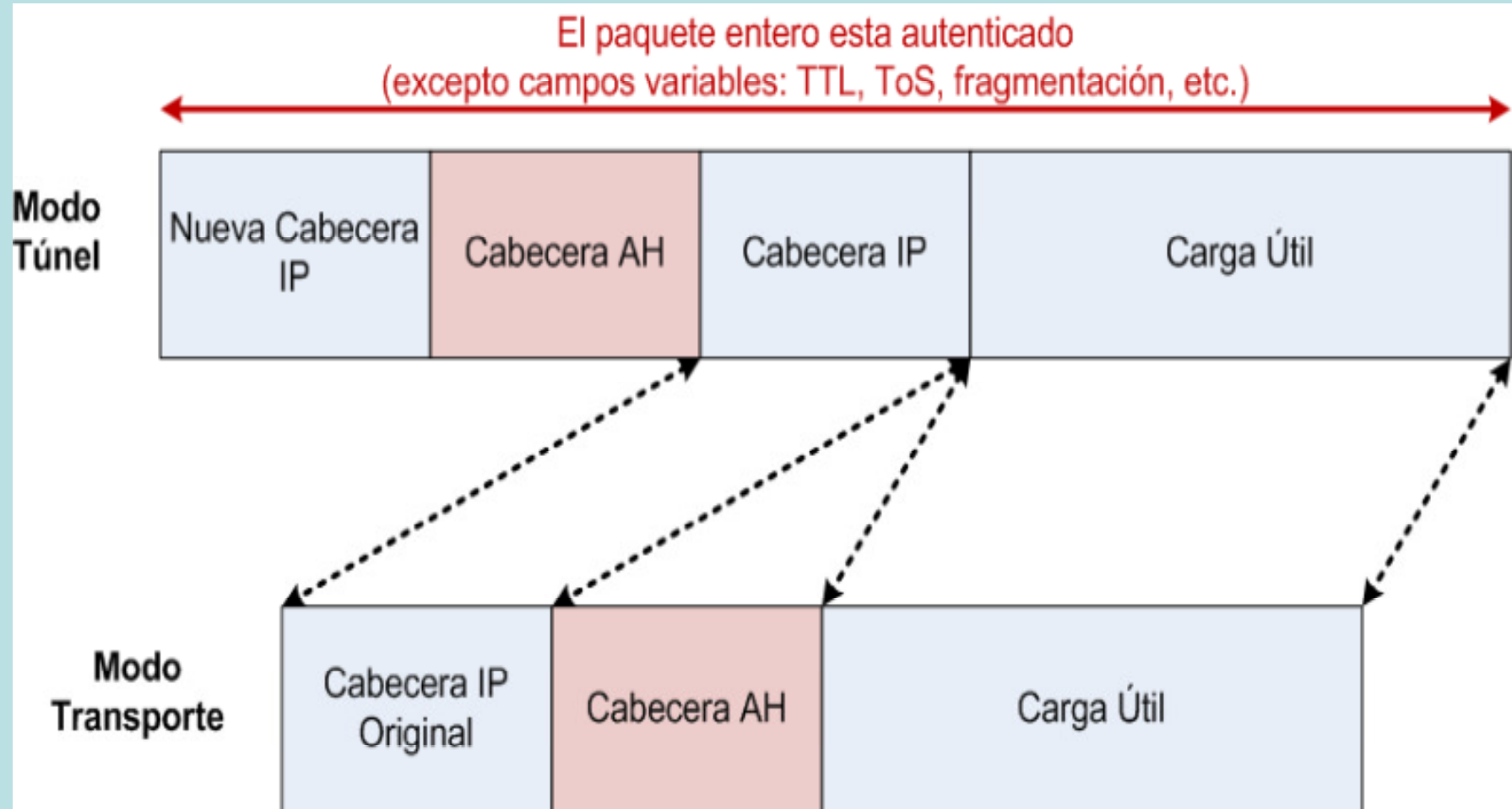
En **modo transporte**, **sólo la carga útil (los datos que se transfieren) del paquete IP es cifrada o autenticada**. El enrutamiento permanece intacto, ya que no se modifica ni se cifra la cabecera IP; sin embargo, cuando se utiliza la cabecera de autenticación (AH), las direcciones IP no pueden ser traducidas, ya que eso invalidaría el hash. Las capas de transporte y aplicación están siempre aseguradas por un hash, de forma que no pueden ser modificadas de ninguna manera (por ejemplo traduciendo los números de puerto TCP y UDP). El **modo transporte** se utiliza para comunicaciones ordenador a ordenador. Una forma de encapsular mensajes IPsec para atravesar NAT ha sido definido por RFCs que describen el mecanismo de NAT transversal.

Modo túnel

En el **modo túnel**, **todo el paquete IP (datos más cabeceras del mensaje) es cifrado o autenticado**. Debe ser entonces encapsulado en un nuevo paquete IP para que funcione el enrutamiento. El **modo túnel** se utiliza para comunicaciones red a red (túneles seguros entre routers, p.e. para VPNs) o comunicaciones ordenador a red u ordenador a ordenador sobre Internet.

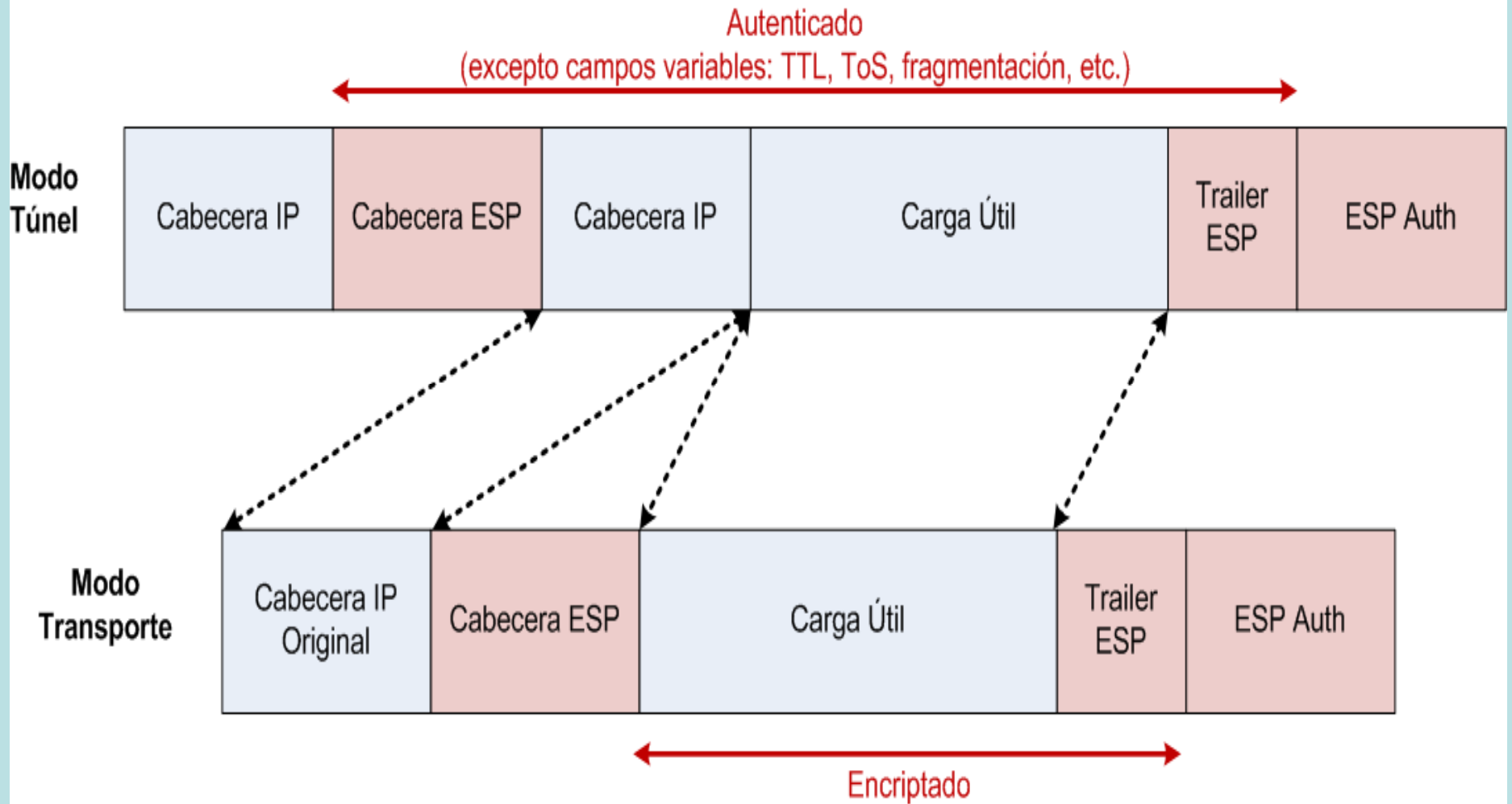


IPSec Protocolo AH





IPSec Protocolo ESP



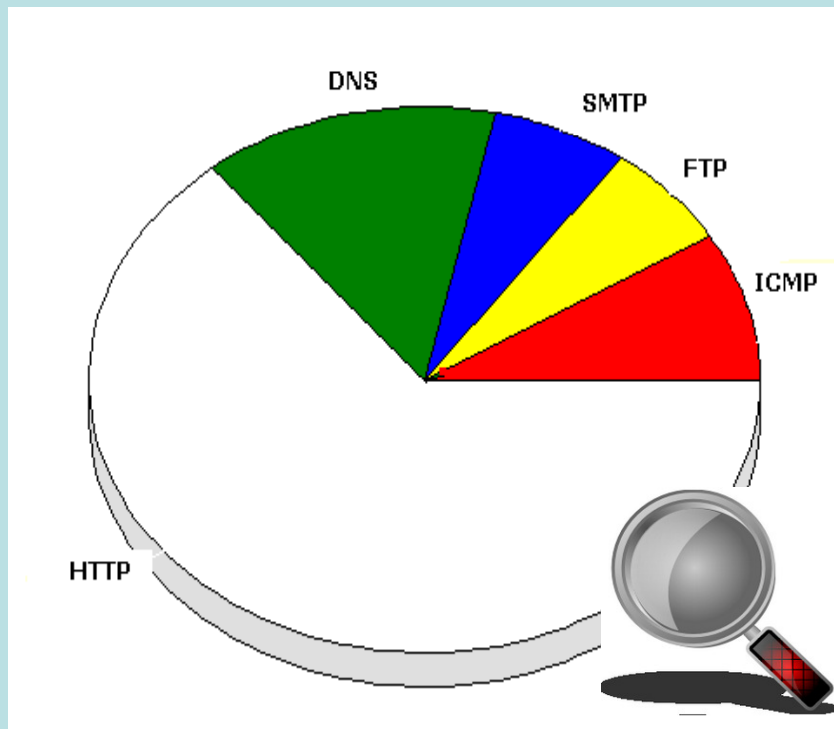


IPSec

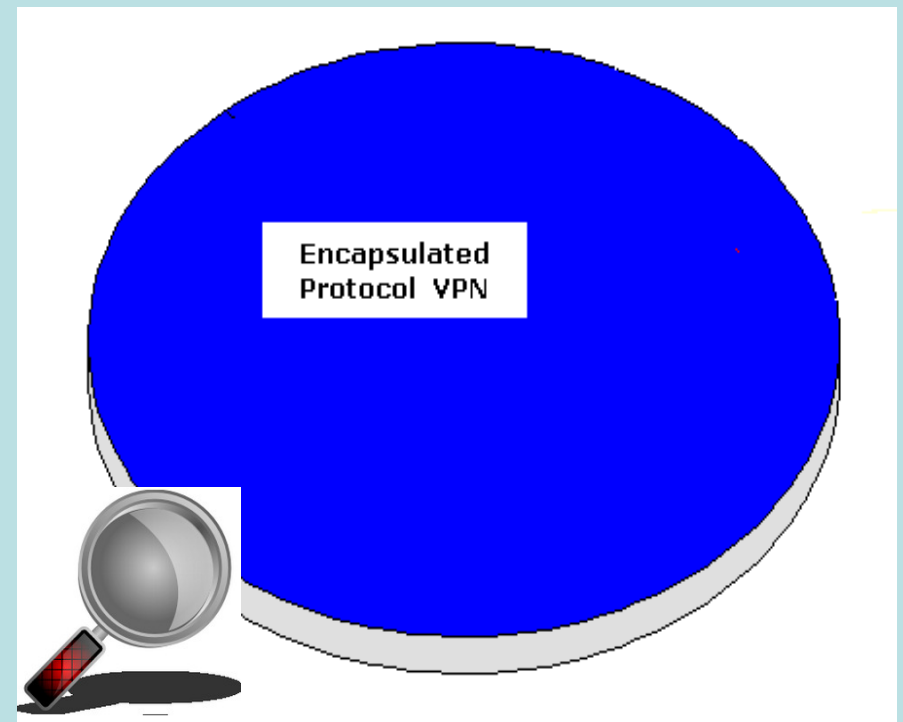
Problema ... ????

Si todo el paquete IP (datos más cabeceras del mensaje) es cifrado...!!!

Antes



Despues





NUEVAS AMENAZAS CON IPv6

IPv6 al igual que su hermano mayor IPv4 puede ser aprovechado de forma maliciosa para llevar a cabo actividades fraudulentas, no solo cuando se disfruta de una red IPv6 nativa, sino también cuando se utilizan mecanismos de transición!!!.

- La coexistencia de IPv4 e IPv6. Mientras se traslada lo que hay en IPv4 a IPv6, es posible hallar problemas relacionados con la dualidad de pilas (dos infraestructuras, cada una con sus problemas propios) , S.O. , además del hardware!!!.
- Manipulación de cabeceras. Pese a su diseño contra este tipo de actividad, no existe seguridad al 100%. No son descartables acciones futuras que burlen parte o la totalidad de los mecanismos de autenticación, especialmente en la fase de dualidad durante la migración.
- Ataques de inundación. El flood sólo se puede capear procesando la inundación y el tráfico, con lo que este tipo de ataques siempre estará ahí, si bien será más complicado para los atacantes.
- Movilidad. Al no existir este concepto en IPv4, nadie sabe a ciencia cierta cómo responderá realmente en IPv6. Todo un misterio pendiente de resolver.



NUEVAS AMENAZAS CON IPv6

IPv6 trae asociado el regreso del paradigma de comunicación extremo a extremo (*peer-to-peer* o *p2p*) con el que se concibió Internet en un principio.

No se utiliza NAT sino que las direcciones IPv6 globales son alcanzables desde cualquier punto de Internet.

Esto permite la proliferación de sistemas *p2p* donde los dispositivos finales envían y reciben contenidos de forma independiente a un servidor.

El administrador de red/seguridad debe tener esto en cuenta y tener claro que IPv6 permite, pero no obliga, la conectividad extremo-a-extremo.

- Existencia de al menos un *firewall* perimetral con soporte IPv6 en la red en la que se encuentra el usuario.
- Existencia en cada nodo IPv6 de un *firewall* local instalado en el propio nodo que filtre las comunicaciones indeseadas.



Autoconfiguración IPv6

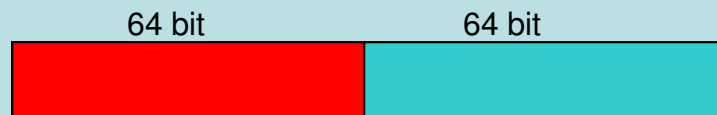
Los nodos IPv6 se autoconfiguran usando el protocolo NDP (*Neighbor Discovery Protocol*)

Además NDP permite descubrir a otros nodos IPv6 en el mismo enlace, determinar su dirección de nivel de red, encontrar otros routers IPv6 en su mismo enlace y mantener información de la ruta IPv6 hacia otros nodos activos.

NDP es similar en muchos aspectos al protocolo ARP de IPv4 y funciona básicamente enviando paquetes ICMPv6 a la LAN (u otro enlace) para encontrar otros nodos IPv6 vecinos con los que poder contactar.

Sin embargo NDP también posee una nueva función en relación a su homólogo ARP, que es la autoconfiguración de la dirección IPv6 por medio del envío de paquetes *Router Advertisement* (RA), tipo especial de paquete ICMPv6 enviado por un router IPv6.

El paquete tiene información del prefijo IPv6 de 64 bits de longitud (/64) que tiene asignado un determinado segmento de red por el que router envía el paquete. Los nodos IPv6 utilizan los paquetes RA para extraer el prefijo y formar su dirección IPv6 global añadiendo los 64 bits correspondientes al identificador del interfaz de red. Además de la formación de la dirección IPv6, el nodo IPv6 utiliza el paquete IPv6 para saber cual es el router que se debe usar para conectarse con la Internet IPv6 (*gateway* por defecto).





Autoconfiguración IPv6

Como se puede deducir, el proceso de autoconfiguración es básico para que los nodos IPv6 puedan obtener conectividad IPv6 global.

Sin embargo NDP tal y como está definido es vulnerable a varios ataques si no se utilizan las herramientas de seguridad adecuadas. En particular, es relativamente fácil poder realizar un bloqueo a todos los nodos IPv6 conectados en un segmento de red (denegación de servicio) sin más que un nodo IPv6 inyecte paquetes RA con un prefijo erróneo de forma intencionada o por descuido.

En este caso, todos los nodos del segmento de red configurarán una dirección IPv6 basada en el prefijo erróneo y por tanto nunca serán alcanzables por un nodo IPv6 externo !!!!!.





Autoconfiguración IPv6

Según las especificaciones de IPv6, los mensajes de ND (*Neighbor Discovery*) se pueden proteger con la cabecera de autenticación (AH -*Authentication Header*) de IPsec con el fin de establecer una relación de confianza con el supuesto router que envía los paquetes RA. Sin embargo existen limitaciones prácticas en el uso de la gestión automática y dinámica de claves.

Con el fin de minimizar este tipo de ataque en entornos no controlados, se ha definido un mecanismo llamado SEND (*Secure Neighbor Discovery*) que proporciona seguridad a los mensajes de NDP.

SEND define una serie de extensiones y mejoras de NDP que posibilita a los nodos IPv6 estar seguros de que el nodo que envía paquetes RA es adecuado y que el prefijo que anuncia es correcto.

De esta manera los administradores de un dominio IPv6 tienen una herramienta que garantiza que su red no va a ser contaminada (intencionadamente o por descuido) con el envío de paquetes RA erróneos, garantizando por tanto la seguridad de la autoconfiguración de los equipos IPv6 en ambientes donde la seguridad en el nivel físico está comprometida, como por ejemplo en las redes públicas inalámbricas.

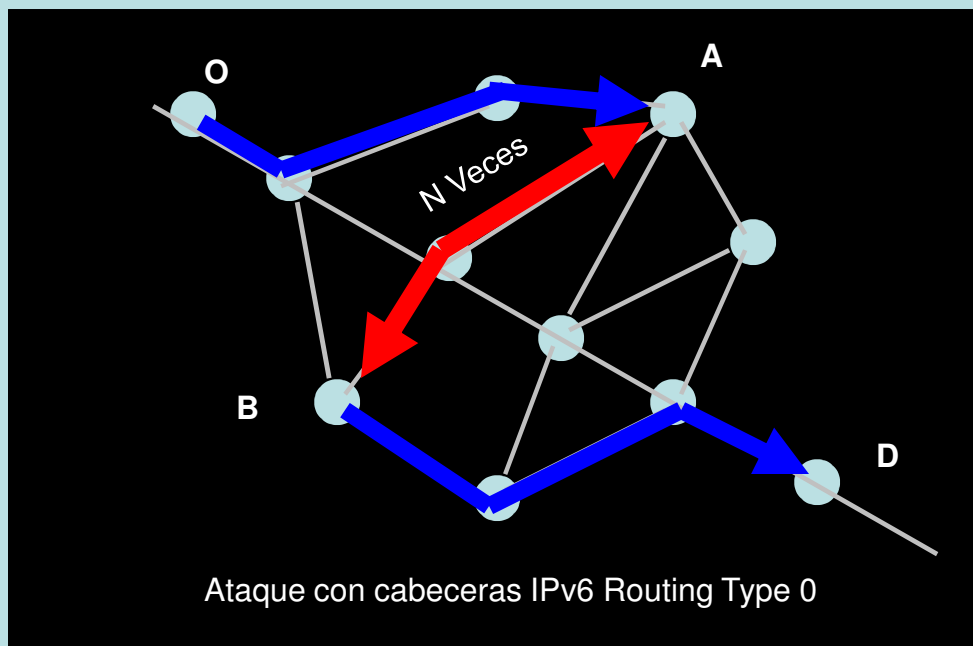


Routing Type 0 ... IPv6

La cabecera de extensión IPv6 *Routing Type 0* podría ser utilizada para llevar a cabo un ataque de denegación de servicio IPv6 de un determinado servidor, dominio o incluso de un determinado camino entre dos dominios IPv6 diferentes.

Este posible ataque se fundamenta en el hecho de que con esta cabecera no se sigue la ruta propuesta por los protocolos de encaminamiento para enviar tráfico desde un origen *O* a un destino *D*. Por el contrario, con esta cabecera se obliga a los paquetes IPv6 a pasar por uno o más nodos IPv6 intermedios. Además no hay ninguna restricción en la especificación IPv6 para que uno o varios de los nodos intermedios aparezcan más de una vez.

Como consecuencia se puede llevar a cabo un ataque de tipo DoS para obligar a pasar una gran cantidad de tráfico masivo entre dos nodos y de manera repetida.



Como se puede deducir, este ataque es lo suficientemente serio como para inhabilitar este tipo de cabecera, de manera que la cabecera *Routing Type 0* ha sido desaprobada. Así pues una pila IPv6 que cumpla con lo especificado en RFC5095 Dic 2007 y que reciba un paquete con esta cabecera descartará el paquete y procederá como si fuera una cabecera de tipo no reconocido. Puede pasar cierto tiempo hasta que los equipos con soporte IPv6 (sobre todo *firewalls* y routers) estén actualizados por lo que es recomendable que un administrador de dominio ponga los medios a su alcance para evitar que se cursen paquetes IPv6 con la cabecera *Routing Type 0*.



Transición

Teredo fue desarrollado por Christian Huitema en Microsoft, y estandarizado por IETF como RFC 4380.

Es una tecnología de transición que permite el establecimiento automático de túneles IPv6 entre hosts que se encuentran situados en diversos dispositivos NAT IPv4, definiendo una manera de encapsular paquetes IPv6 en datagramas UDP IPv4 que pueden ser dirigidos a través de dispositivos NAT y en Internet IPv4.



Transición



Teredo ... Suplantación de identidades

El cliente Teredo debe contactar con un servidor y con un *relay* con el fin de cursar tráfico IPv6 de forma efectiva.

Esto hace que ambos agentes, que por lo general están ubicados fuera del dominio del cliente Teredo, sean objeto de ataques por dos motivos:

- 1ero. Provocar un ataque de denegación de servicio (DoS)
- 2do. Interceptar tráfico de un cliente Teredo con diversos fines.

El segundo es el que tiene peores consecuencias puesto que un ataque de denegación de servicio, si bien no deja de ser un inconveniente al impedir que el cliente tenga conectividad IPv6, no compromete la seguridad de las comunicaciones del cliente.

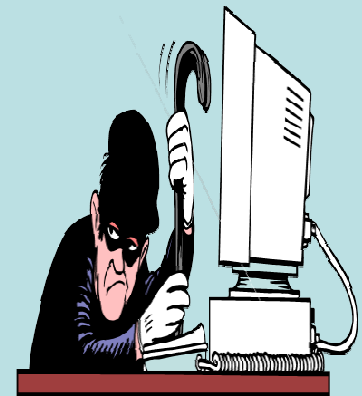


Transición

Teredo ... Suplantación de identidades

La suplantación de dirección más comprometedora es la del agente *relay* puesto que este componente se encarga de retransmitir tráfico IPv6 desde/hacia la Internet IPv6 hacia/desde clientes Teredo. Por tanto está manejando tráfico de datos, mientras que con la suplantación del servidor Teredo solo se intercepta tráfico de señalización del cliente Teredo.

Suplantar un agente Teredo *relay* no es trivial y consiste en hacer creer al cliente Teredo que el *relay* que debe usar para alcanzar un nodo IPv6 nativo es el atacante. Teóricamente el atacante podría realizar momentáneamente un ataque de tipo DoS al *relay* a ser suplantado y arreglárselas para interceptar el tráfico dirigido a él desde el cliente Teredo. Dadas las características de Teredo, a partir de ese momento el atacante puede capturar todo el tráfico hacia el nodo IPv6 nativo.





Transición

Teredo ... Suplantación de identidades

Otra forma de conseguir convertirse en un Teredo *relay* maligno para capturar tráfico IPv6 Teredo, es por medio del anuncio del prefijo 2001::/32 a través de BGP a toda la Internet IPv6. Este prefijo es el estandarizado para el protocolo Teredo y cuando se realiza este anuncio, los nodos IPv6 nativos cercanos (en términos de BGP) al *relay* maligno le envían al anunciante del prefijo Teredo todo el tráfico IPv6 que tiene como destino un cliente Teredo.



Lo mismo ocurre con el tráfico de regreso desde esos clientes Teredo hacia los nodos IPv6 nativos cercanos al *relay*. Esto es debido al funcionamiento intrínseco de Teredo. Como consecuencia, dicho tráfico pasa a través de una entidad maligna y por tanto es susceptible de ser manipulado o simplemente espiado. De esta forma, en caso de tener acceso a las rutas IPv6 BGP, un ataque de tipo *man-in-the-middle* es realmente fácil de llevar a cabo.



Transición

Teredo ... Suplantación de identidades

Con el fin de minimizar el impacto de un ataque de suplantación de identidad del Teredo *relay*, el protocolo Teredo define un procedimiento básico llamado “Test de conectividad IPv6 directa”, implementado por los clientes Teredo para asegurarse de que el *relay* que se está usando no ha cambiado en el transcurso de una comunicación con un nodo IPv6 nativo.

El procedimiento consiste en generar un número aleatorio y enviar un paquete ICMPv6 de tipo *echo request* incluyendo dicho número al nodo IPv6 nativo con el que se trata de contactar. La respuesta siempre viene a través de un Teredo *relay* cuya dirección IPv4 debe coincidir con la dirección del Teredo *relay* que se haya estado usando. Sin embargo este procedimiento no es útil en el caso de que un *relay* esté publicando el prefijo Teredo a través de BGP.

Por tanto la única manera de estar seguro de que las comunicaciones IPv6 a través de Teredo no son vulnerables a un ataque de suplantación de la identidad es protegerlas por medio de IPsec ya que proporciona seguridad extremo-a-extremo impidiendo que en el caso de que existan servidores/*relays* que intercepten tráfico.





Transición

Teredo ... Otros ataques ...

- Desbordamiento de la memoria *cache* del cliente Teredo. Las implementaciones de Teredo almacenan en una tabla *cache* diversos datos/estados correspondientes a las comunicaciones que se llevan a cabo con otros nodos IPv6. Un envío masivo de paquetes a un cliente Teredo desde diversos nodos IPv6 diferentes puede ocasionar un desbordamiento del cliente que consiga que deje de funcionar la implementación Teredo.



- Ataques contra los servidores y *relays* Teredo. Se puede realizar un ataque por fuerza bruta mediante el envío masivo de tráfico. En este caso dichos agentes se ven desbordados y no pueden cursar el tráfico Teredo por lo que se impide las comunicaciones tanto de los nodos IPv6 nativos (tratando de contactar con clientes Teredo) como de los propios clientes Teredo. Se puede luchar contra este tipo de ataques filtrando tráfico que cumpla un determinado patrón, como protocolo UDP, puerto Teredo, etc.



Transición

Teredo ... Otros ataques ...

- Ataques contra clientes Teredo a través de los *relays* Teredo. Un ataque similar al anterior consiste en que diversos nodos IPv6 nativos envíen tráfico masivo a un determinado cliente Teredo con una dirección IPv6 falsa. El tráfico será retransmitido a través de los diversos *relays* Teredo desbordando los recursos del cliente Teredo y sin posibilidad de encontrar a los causantes del ataque. Se puede luchar contra este tipo de ataques filtrando tráfico que cumpla un determinado patrón, como protocolo UDP, puerto Teredo, etc.





Transición

6to4

Gracias a 6to4 los nodos que se encuentran en una red que sólo es IPv4 pueden obtener conectividad IPv6 de una forma sencilla sin ningún despliegue especial en la red en la que se encuentra.

Al igual que ocurre con Teredo, este mecanismo no es inseguro en sí mismo. Tan solo ofrece nuevas posibilidades de realizar ataques a gente sin escrúpulos.

Los tipos de ataque que se pueden llevar a cabo con 6to4 se fundamentan en las características del mecanismo: los nodos 6to4 deben aceptar tráfico IPv4/IPv6 enviado desde cualquier lugar, sin comprobar identidades y sin una relación de confianza previa.

Con esta premisa es fácil entender que 6to4 se pueda usar para llevar a cabo ataques de tipo DoS y de suplantación de la identidad.

1ero. Ataques de denegación de servicio (DoS)

2do. Interceptar tráfico de un cliente Teredo con diversos fines.



Transición

6to4 Puerta abierta a Internet

Al igual que ocurre con Teredo, al usar 6to4 el nodo cliente se conecta a la Internet IPv6 global de forma directa, puenteando las posibles medidas de seguridad perimetral instaladas habitualmente en el la red para IPv4.

Por tanto se pueden aplicar en este caso las mismas consideraciones que con Teredo en cuanto a las medidas de seguridad en las comunicaciones extremo-a-extremo.



Transición

6to4 Suplantación de identidades

Puesto que un nodo 6to4 necesita de un *relay* 6to4 para conectar con un nodo IPv6 nativo, un atacante podría hacerse pasar por un *relay* 6to4 con el fin de espiar el tráfico intercambiado entre ambos.

La única forma de hacer esto es conseguir que el *relay* 6to4 haga pensar al nodo 6to4 que él es el *relay* adecuado (bien porque tenga la dirección *anycast* asignada a este servicio o bien porque se use una dirección *unicast* y se haga pasar por un *relay* confiable) y además sea capaz de anunciar por BGP el prefijo IPv6 correspondiente a 6to4: 2002::/16.

Sin embargo este ataque es más difícil de llevar a cabo que en el caso de Teredo debido a la asimetría en las rutas que sigue el tráfico intercambio entre ambos nodos: el *relay* 6to4 que usa el nodo 6to4 y el nodo IPv6 nativo puede ser diferente en la comunicación llevada a cabo entre ambos.





Transición

6to4 Ataques de denegación de servicio

Este ataque es muy similar al descrito para Teredo. La denegación del servicio se puede producir por una o varias de las siguientes razones:

- Falsos *relays* 6to4. Un nodo atacante se puede hacer pasar por un *relay* 6to4 y no retransmitir el tráfico 6to4 recibido. En este caso estará denegando la posibilidad de comunicarse con/a los dominios 6to4 afectados.
- Ataques contra *relays* 6to4. Se puede realizar un ataque por fuerza bruta mediante el envío masivo de tráfico hacia los *relays* 6to4. En este caso un ataque exitoso podría hacer caer el servicio 6to4 a nivel global debido a que todos los relays 6to4 públicos poseen la dirección IPv4 *anycast*.



- Ataques contra nodos finales 6to4.

Debido a las características propias de 6to4, los nodos finales 6to4 se ven obligados a recibir todo el tráfico 6to4 encapsulado en IPv4. Esta circunstancia se puede aprovechar para lanzar una ataque DoS distribuido contra un nodo IPv4. Es suficiente con que uno o diversos nodos IPv4 envíen a un nodo 6to4 tráfico masivo de tipo 6to4 con una dirección origen falsa.



Transición

6to4 Ataques de denegación de servicio

- Ataques con paquetes *Router Advertisement* (RA). Aunque la pseudointerfaz 6to4 no debería tener direcciones de ámbito local, esto no se cumple siempre, en parte por culpa de las implementaciones 6to4 y en parte porque un mismo nodo puede tener más de un pseudo-interfaz de tipo túnel en un interfaz físico. Además 6to4 asume que el resto de nodos 6to4 se encuentran en su mismo enlace por lo que la recepción de un paquete de tipo RA en un interfaz 6to4 resulta de procedencia ambigua y por tanto puede ser considerado válido cuando en realidad es fruto de un ataque.

El uso de paquetes RA malignos provoca como consecuencia la denegación de conectividad IPv6.





Transición

6to4 Recomendaciones

Con el fin de proteger adecuadamente el servicio 6to4 es recomendable seguir las siguientes recomendaciones. Algunas de ellas son consideraciones que deben ser tomadas en cuenta por los desarrolladores de las implementaciones 6to4, mientras que otras son recomendaciones para los administradores de un dominio.

- Eliminar tráfico 6to4 dirigido o proveniente de direcciones IPv4 de tipo privado, broadcast o multicast.
- Eliminar tráfico 6to4 recibido de un nodo 6to4 cuya dirección IPv4 no coincide con su prefijo 6to4.
- Eliminar tráfico 6to4 cuyo destino no es una dirección IPv6 global.
- Descartar tráfico dirigido a otro dominio 6to4 a través de otro *relay* 6to4.
- Descartar tráfico proveniente de otro dominio 6to4 recibido a través de otro *relay* 6to4.
- Descartar en un nodo 6to4 cualquier tipo de tráfico IPv6 que no esté dirigido al prefijo 6to4 que posee.
- Utilizar SEND o IPSec en el caso de necesitar autoconfiguración en la interfaz física correspondiente a la pseudo-interfaz 6to4.



Transición

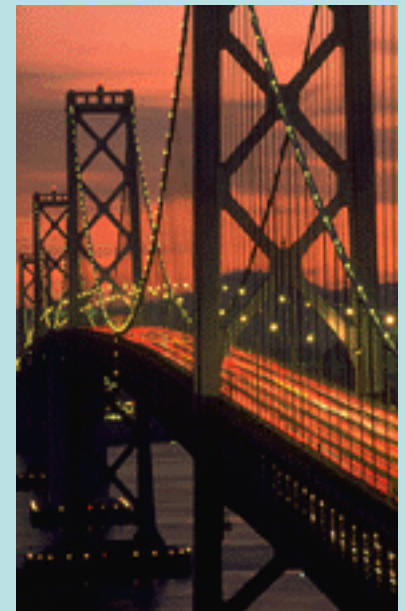
Tunnel Brokers Otro uso de IPv6 con fines maliciosos (túneles 6in4, 6to4, Teredo).

Los mecanismos de transición IPv6 basados en túneles podrían ser usados para puentear las medidas de seguridad perimetrales (*firewalls, proxies, etc.*) instaladas en cualquier dominio puesto que tienen la facilidad de ocultar el tráfico TCP/UDP transportado a través de IPv6 dentro del túnel IPv4.

Esta característica podría ser aprovechada para llevar a cabo actividades ilícitas como:

- Visitas web a sitios no autorizados.
- Comunicación o coordinación entre grupos delictivos.
- Ataques coordinados de DoS.
- Compartición ilegal de Doc., tráfico no controlado, ...etc.

Todo ello con la novedad de que ser indetectables en la Internet actual en el caso de que las medidas de control de tráfico solo controlen paquetes TCP/UDP transportados sobre paquetes IPv4.





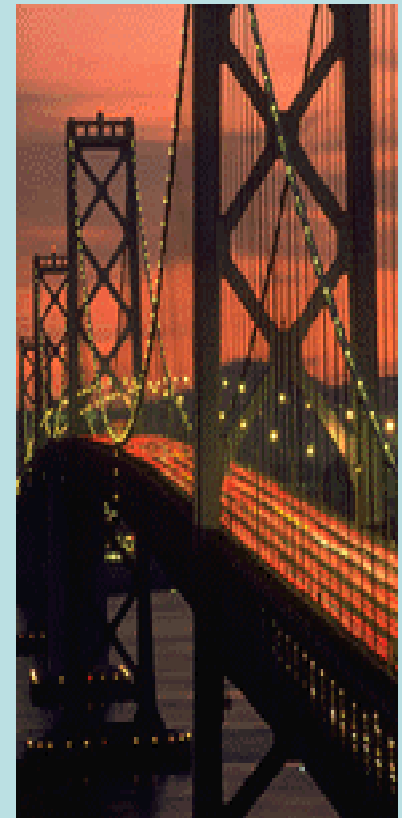
Transición

Tunnel Brokers Otro uso de IPv6 con fines maliciosos (túneles

6in4, 6to4, Teredo).

Un ejemplo de esto podría ser el uso de servidores de túneles IPv6, también llamados *Tunnel Brokers* (TB), los cuales los usuarios lícitos usan para obtener lícitamente conectividad IPv6 y que los usuarios fraudulentos pueden utilizar para construir una red "virtual" distribuida mundialmente sobre la actual Internet IPv4 que les posibilita la comunicación entre ellos sin que sea detectable por sistemas que no soportan comunicaciones IPv6.

Al usar un TB se puede crear una red virtual IPv6 sobre la Internet IPv4.





Hoy Estado del Arte

19 ... (Municiones) payloads de ataques sobre la Pila Ipv6 (Metaexploit).

linux/x86/shell/bind_ipv6_tcp normal Linux Command Shell, Bind TCP Stager (IPv6)
linux/x86/shell/reverse_ipv6_tcp normal Linux Command Shell, Reverse TCP Stager (IPv6)
linux/x86/shell_bind_ipv6_tcp normal Linux Command Shell, Bind TCP Inline (IPv6)

windows/dllinject/bind_ipv6_tcp normal Reflective Dll Injection, Bind TCP Stager (IPv6)
windows/dllinject/reverse_ipv6_tcp normal Reflective Dll Injection, Reverse TCP Stager (IPv6)

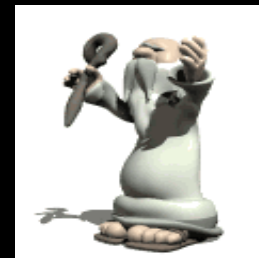
windows/meterpreter/bind_ipv6_tcp normal Windows Meterpreter (Reflective Injection), Bind TCP Stager (IPv6)
windows/meterpreter/reverse_ipv6_tcp normal Windows Meterpreter (Reflective Injection), Reverse TCP Stager (IPv6)

windows/patchupdllinject/bind_ipv6_tcp normal Windows Inject DLL, Bind TCP Stager (IPv6)
windows/patchupdllinject/reverse_ipv6_tcp normal Windows Inject DLL, Reverse TCP Stager (IPv6)
windows/patchupmeterpreter/bind_ipv6_tcp normal Windows Meterpreter (skape/jt injection), Bind TCP Stager (IPv6)
windows/patchupmeterpreter/reverse_ipv6_tcp normal Windows Meterpreter (skape/jt injection)
windows/patchupvncinject/bind_ipv6_tcp normal Windows VNC Inject (skape/jt injection), Bind TCP Stager (IPv6)
windows/patchupvncinject/reverse_ipv6_tcp normal Windows VNC Inject (skape/jt injection)

windows/shell/bind_ipv6_tcp normal Windows Command Shell, Bind TCP Stager (IPv6)
windows/shell/reverse_ipv6_tcp normal Windows Command Shell, Reverse TCP Stager (IPv6)

windows/upexec/bind_ipv6_tcp normal Windows Upload/Execute, Bind TCP Stager (IPv6)
windows/upexec/reverse_ipv6_tcp normal Windows Upload/Execute, Reverse TCP Stager (IPv6)

windows/vncinject/bind_ipv6_tcp normal VNC Server (Reflective Injection), Bind TCP Stager (IPv6)
windows/vncinject/reverse_ipv6_tcp normal VNC Server (Reflective Injection), Reverse TCP Stager (IPv6)

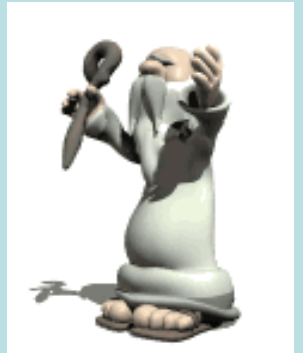




CONCLUSIONES

ASPECTOS DE SEGURIDAD MEJORADOS CON IPv6

- Ataques *broadcast (smurf)*. En IPv4 se generan enviando un paquete ICMP *echo request* a la dirección de *broadcast* de la red con la dirección origen del nodo atacado. En IPv6 no existe el concepto de *broadcast*. IPv6 especifica que no se debe responder con un mensaje ICMP a ningún paquete que tenga dirección destino de nivel tres *multicast* o direcciones de nivel dos *multicast* o *broadcast*. Por lo tanto si las pilas IPv6 siguen la especificación este problema desaparece.
- Fragmentación de paquetes. En IPv6 la fragmentación de paquetes sólo se puede realizar en los extremos de la comunicación, por lo que se reducen los riesgos por ataques con fragmentos superpuestos o de pequeño tamaño. Las consideraciones sobre fragmentos fuera de secuencia serán las mismas que para IPv4, pero en el nodo final. Los *firewalls* no deberán filtrar los fragmentos de paquetes.





CONCLUSIONES

ASPECTOS DE SEGURIDAD MEJORADOS CON IPv6

- Ataques *broadcast (smurf)*. En IPv4 se generan enviando un paquete ICMP *echo request* a la dirección de *broadcast* de la red con la dirección origen del nodo atacado. En IPv6 no existe el concepto de *broadcast*. IPv6 especifica que no se debe responder con un mensaje ICMP a ningún paquete que tenga dirección destino de nivel tres *multicast* o direcciones de nivel dos *multicast* o *broadcast*. Por lo tanto si las pilas IPv6 siguen la especificación este problema desaparece.
- Fragmentación de paquetes. En IPv6 la fragmentación de paquetes sólo se puede realizar en los extremos de la comunicación, por lo que se reducen los riesgos por ataques con fragmentos superpuestos o de pequeño tamaño. Las consideraciones sobre fragmentos fuera de secuencia serán las mismas que para IPv4, pero en el nodo final. Los *firewalls* no deberán filtrar los fragmentos de paquetes.



CONCLUSIONES



Además de solventar la escasez de direcciones IP, el protocolo IPv6 ha sido creado, desde un inicio, con la seguridad y eficiencia como objetivos, medidas como la implantación de IPsec, el nuevo diseño del paquete o la manera de asignar las direcciones IP son la prueba de ello.

No obstante, sustituir un protocolo tan extendido e importante como IPv4 supondrá un desafío de gestión y técnico con implicaciones en la seguridad de los sistemas de información.

CONCLUSIONES



La mayor parte de sistemas operativos tiene la posibilidad de utilizar IPv6, es necesario comenzar a realizar una política de seguridad que lo contemple y tomar las medidas de seguridad apropiadas para cumplirla.

Debido al agotamiento de direcciones IPv4, a que cada vez existirán más servicios sobre IPv6 y a la aparición de otros nuevos que aprovechan la explosión del número de direcciones IP disponibles, es necesario comenzar a obtener conocimiento y experiencia sobre la implantación y la administración de este protocolo y de los mecanismos de interoperabilidad con IPv4. La mejor forma es hacerlo gradualmente, habilitándolo en unos servicios de manera muy controlada.

El momento mas crítico será aquel donde la transición obligue la convivencia de ambos protocolos, Hoy
Urge prepararse para enfrentar el Reto



REFERENCIAS

- Informe sobre las implicaciones de seguridad en la implantación de IPV6, INTECO-CERT
- Comisión Europea: http://ec.europa.eu/information_society/policy/ipv6/index_en.htm
- Nuevas amenazas de seguridad con IPV6, Miguel Á Díaz Fernández, Álvaro Vives, César Olvera Morales, CONSULINTEL
- ENISA: <http://www.enisa.europa.eu/act/res/files/resilience-features-of-ipv6-dnssec-and-mpls/?searchterm=ipv6>
- SecurityFocus: <http://www.securityfocus.com/news/11463>
- IETF: <http://www.ietf.org/rfc/rfc3971.txt>
- Microsoft: <http://technet.microsoft.com/en-us/network/bb530961.aspx>
- Consulintel: http://www.mundointernet.es/IMG/pdf/ponencia162_1.pdf
- NetworkWorld: <http://www.networkworld.com/news/2009/071309-ipv6-network-threat.html>
- Portal IPv6: <http://www.ipv6tf.org>
- IPv6-To-Standard: <http://www.ipv6-to-standard.org>
- RFC3756 Nikander, P., Kempf, J., and Nordmark, E., "IPv6 Neighbor Discovery (ND) Trust Models and Threats", May 2004,
- RFC3971 Arkko, J., Kempf, J., Zill, B., and Nikander, P., "Secure Neighbor Discovery (SEND)", March 2005, IETF
- RFC4193 Hinden, and R., Haberman, B., "Unique Local IPv6 Unicast Addresses", October 2005, IETF Request For Comment
- RFC4861 Narten, T., Nordmark, E., Simpson, W., and Soliman, H., "Neighbor Discovery for IP version 6 (IPv6)", 2007
- RFC4941 Narten, T., and Draves, R., "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", January 2001,
- RFC5095 Abley, J., Savola, P., and Neville-Neil, G., "Deprecation of Type 0 Routing Headers in IPv6", December 2007,
- RFC5157 T., Chown, "IPv6 Implications for Network Scanning", March 2008, IETF Request For Comment
- Scott Hogg, Eric Vyncke, "IPv6 Security", Cisco Press, 2008
- Daniel Minoli, Jake Kouns "Security in an IPv6 Environment", Auerbach Publications, 2008