Coexistencia y transición

Módulo 7

Coexistencia y transición

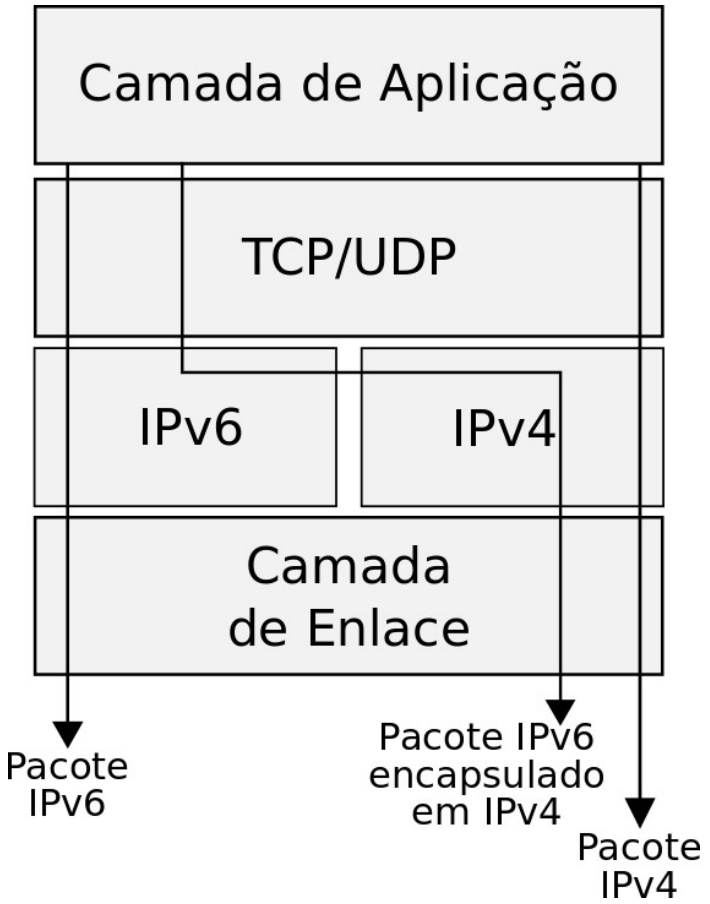
- Toda la estructura de Internet está basada en IPv4.
- Un cambio inmediato de protocolo es inviable debido al tamaño y a la proporción que tiene esta red.
- La adopción de IPv6 se debe realizar de manera gradual.
- Inicialmente habrá un período de transición y de coexistencia entre los dos protocolos.
- Las redes IPv4 necesitarán comunicarse con las redes IPv6 y viceversa.
- Para facilitar este proceso se han desarrollado algunas técnicas que buscan mantener la compatibilidad de toda la base instalada de redes IPv4 con el nuevo protocolo IPv6.

Coexistencia y transición

- Estas técnicas de transición se dividen en 3 categorías:
  - **Doble pila**
    - Provee soporte a ambos protocolos en el mismo dispositivo.
  - **Tunelización**
    - Permite el tráfico de paquetes IPv6 sobre la estructura de la red IPv4 existente.
  - **Traducción**
    - Permite la comunicación entre nodos que solo soportan IPv6 y nodos que solo soportan IPv4.

Doble pila

- Los nodos se vuelven capaces de enviar y recibir paquetes tanto para IPv4 como para IPv6.
- Al comunicarse con un nodo IPv6, un nodo IPv6/IPv4 se comporta como un nodo IPv6; al comunicarse con un nodo IPv4, como un nodo IPv4.
- Necesita al menos una dirección para cada pila.
- Utiliza mecanismos IPv4, como por ejemplo DHCP, para adquirir direcciones IPv4, y mecanismos de IPv6 para direcciones IPv6.



Doble pila

- Una red doble pila es una infraestructura capaz de encaminar ambos tipos de paquetes.

- Exige analizar algunos aspectos:
  - Configuración de los servidores de DNS;
  - Configuración de los protocolos de enrutamiento;
  - Configuración de los *firewalls*;
  - Cambios en la administración de las redes.

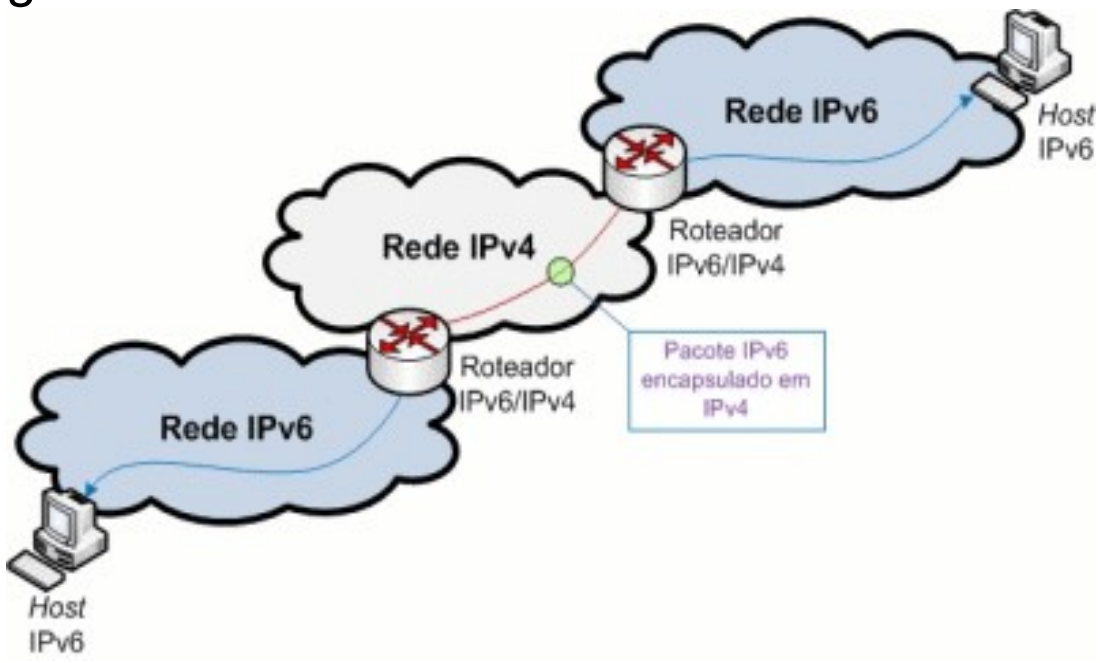
Cuando no es posible Dual Stack

Seguir utilizando IPv4

- Mientras se consiguen direcciones IPv4
  - Técnicas de transición para IPv6
  - Problemas de acceso
  - Problemas de performance
  - Costos incrementales para las direcciones IPv4
- Usar CGN cuando no reciban mas IPv4
  - Direcciones privadas + doble NAT
  - Costos crecientes
  - Equipos costosos con obsolescencia
-

Técnicas de tunelización

- También llamado encapsulamiento.
- El contenido del paquete IPv6 se encapsula en un paquete IPv4.
- Se pueden clasificar de la siguiente manera:
  - *Router-a-Router*
  - *Host-a-Router*
  - *Router-a-Host*
  - *Host-a-Host*



Técnicas de tunelización

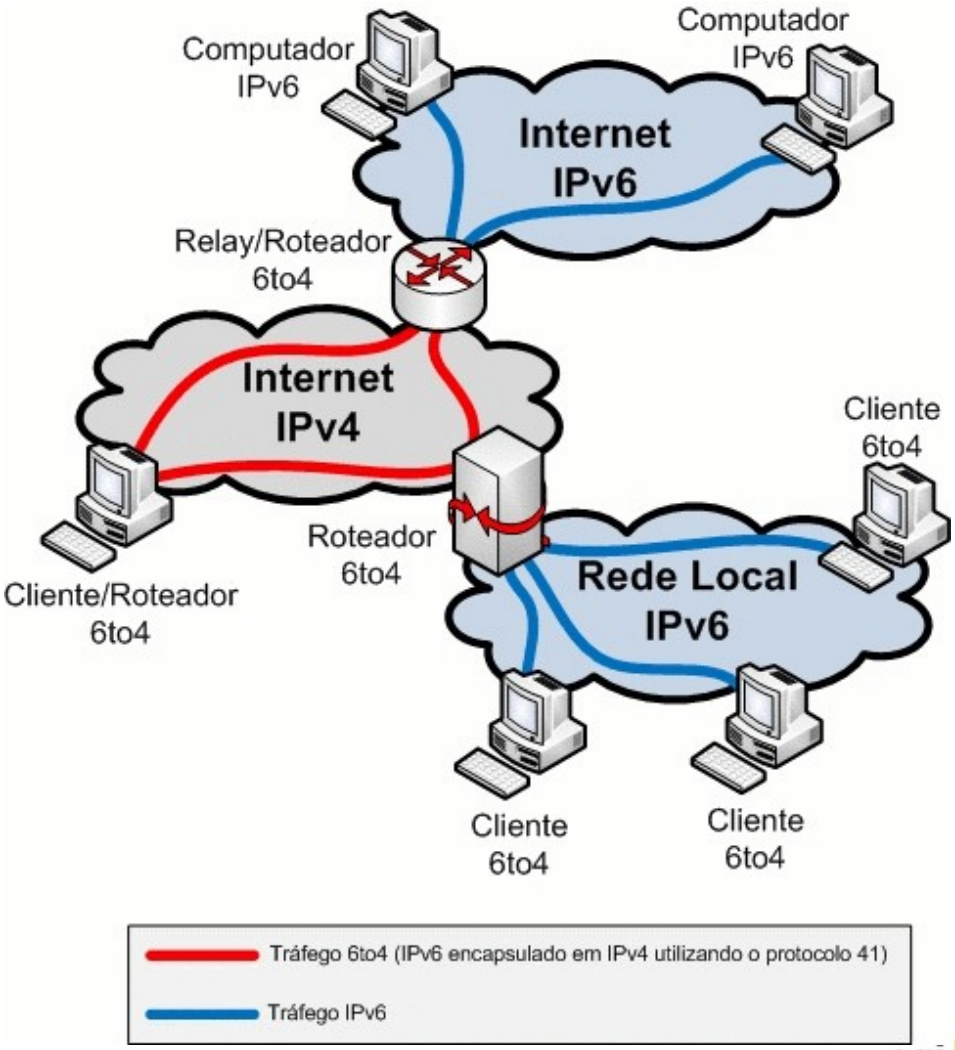
- Existen diferentes formas de encapsulamiento:
  - Paquetes IPv6 encapsulados en paquetes IPv4;
    - Protocolo 41.
    - 6to4, ISATAP y *Tunnel Brokers.*
  - Paquetes IPv6 encapsulados en paquetes GRE;
    - Protocolo GRE.
  - Paquetes IPv6 encapsulados en paquetes UDP;
    - TEREDO.

Tunnel Broker

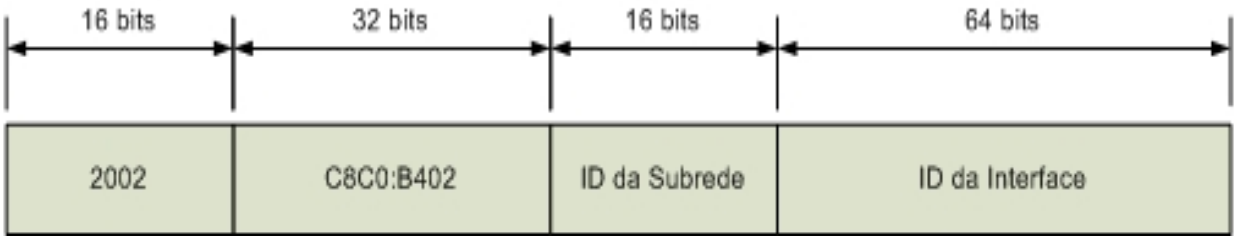
- Consiste en un túnel IPv6 dentro de la red IPv4, que se crea entre su computadora o red y el proveedor que suministrará la conectividad IPv6.
- Basta con registrarse en un proveedor de acceso *Tunnel Broker* y descargar un *software* o *script* de configuración.
- La conexión del túnel se realiza solicitando el servicio al Servidor Web del proveedor.
- Indicado para redes pequeñas o para un único *host* aislado.

6to4

- Forma de tunelización router-a-router.
- Provee una dirección IPv6 única al *host.*
- La dirección está formada por el prefijo de dirección global **2002:wwxx:yyzz::/48**, donde **wwxx:yyzz** es la dirección IPv4 pública del *host* convertida a formato hexadecimal.
- El *relay* 6to4 se identifica mediante la dirección *anycast* **192.88.99.1**.
- Encaminamiento asimétrico.
- Se puede utilizar con relays públicos cuando no hay conectividad v6 nativa.
- Cuando hay conectividad y servicios nativos se debe implementar para facilitar la comunicación con clientes 6to4.



6to4

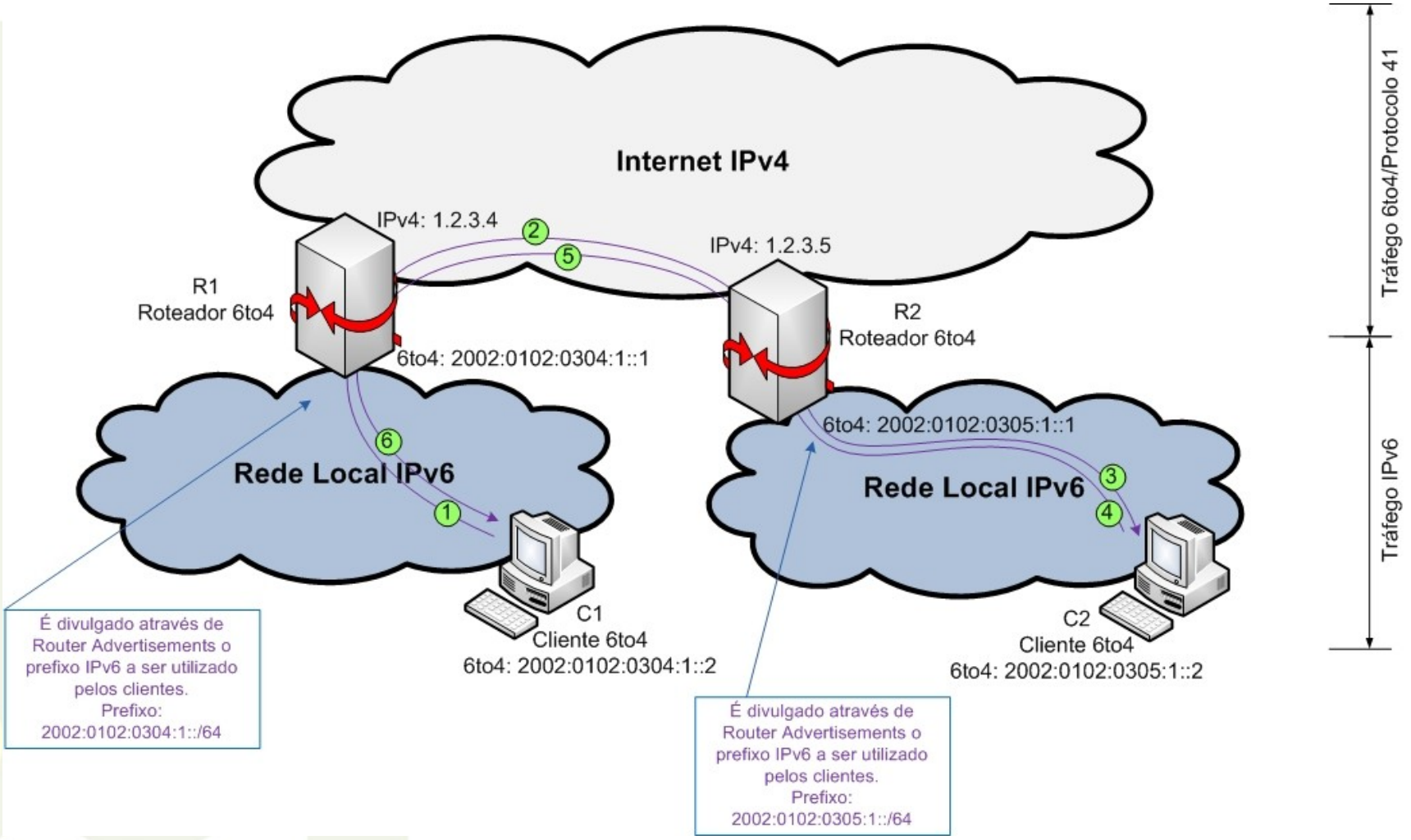


- El prefijo 6to4 siempre es **2002**.
- El siguiente campo, IPv4 pública del cliente, se crea convirtiendo la dirección a formato hexadecimal.
- El ID de la subred se puede usar para segmentar la red IPv6 6to4 en hasta 2^{16} subredes con 2^{64} direcciones cada una; se puede utilizar, por ejemplo, 0, 1, 2, 3, 4...
- El ID de la interfaz puede ser igual al segundo campo (Windows lo hace de este modo) o a cualquier otro número en el caso de configuración manual (Linux utiliza numeración secuencial: 1, 2, 3, 4...).

6to4

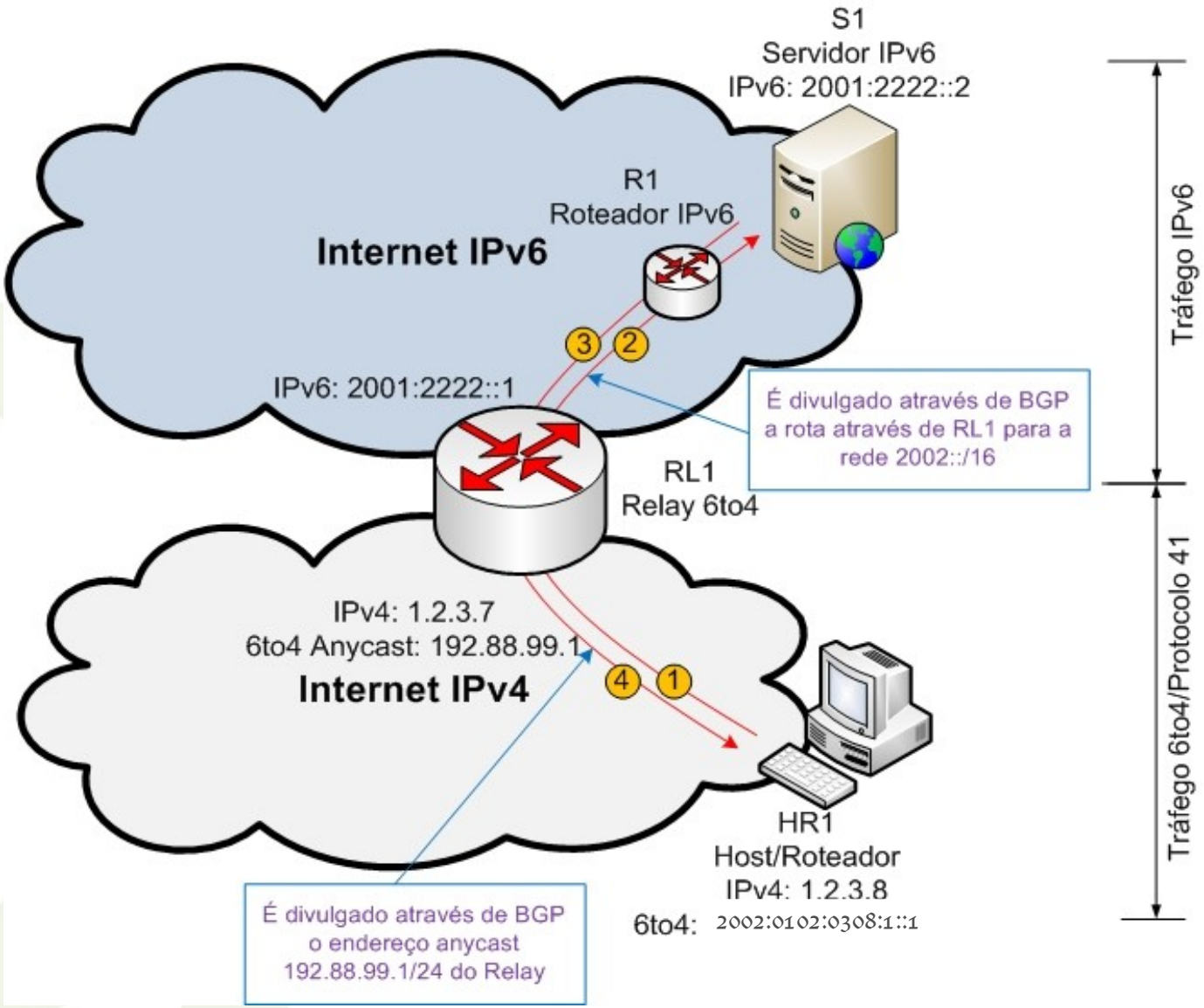
Comunicación entre Clientes 6to4 que están en redes diferentes

- Observe que el tráfico en la red local es nativo IPv6, solo está encapsulado entre los routers 6to4



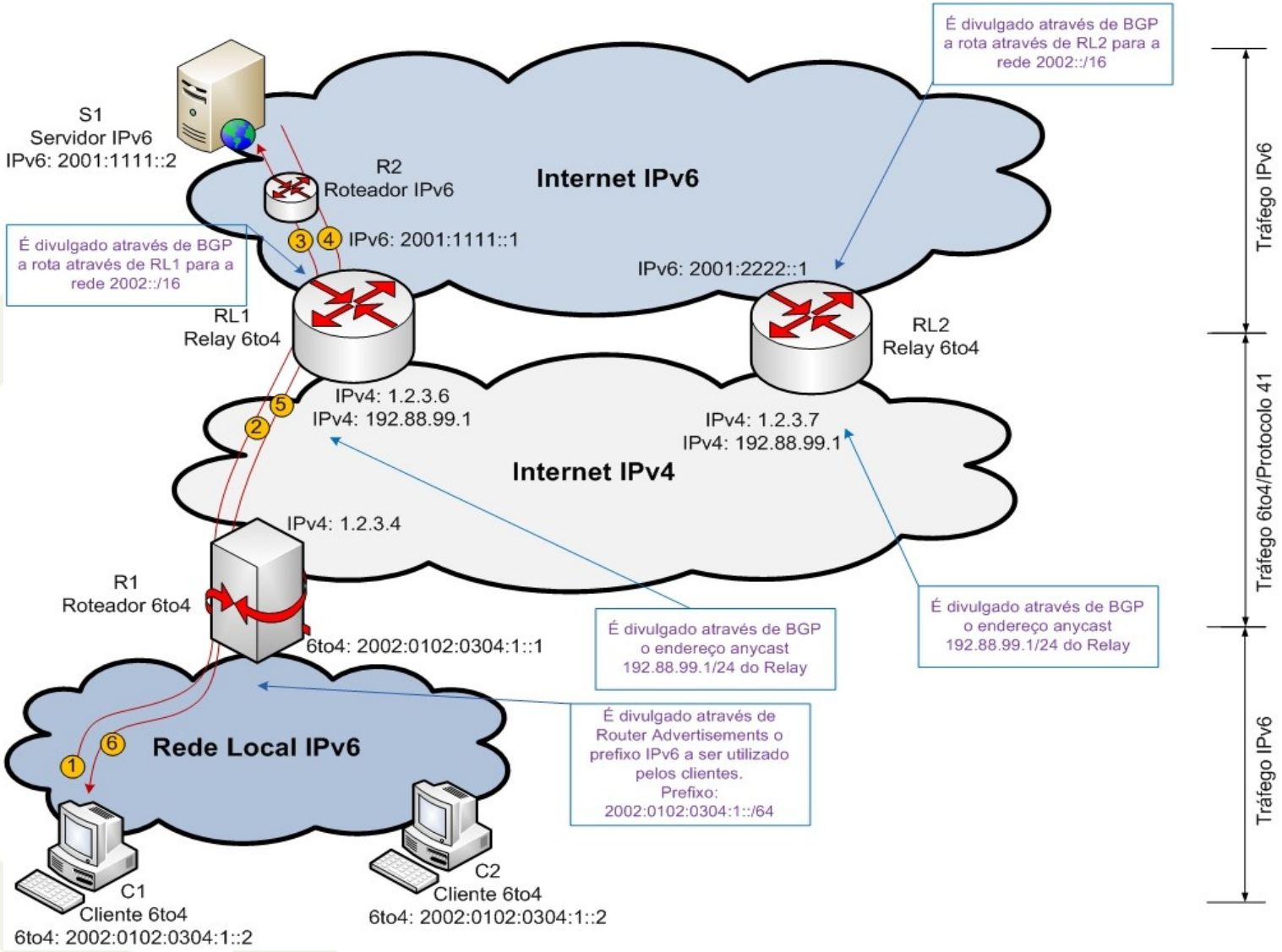
6to4

Comunicación Cliente/Router 6to4 con un servidor IPv6



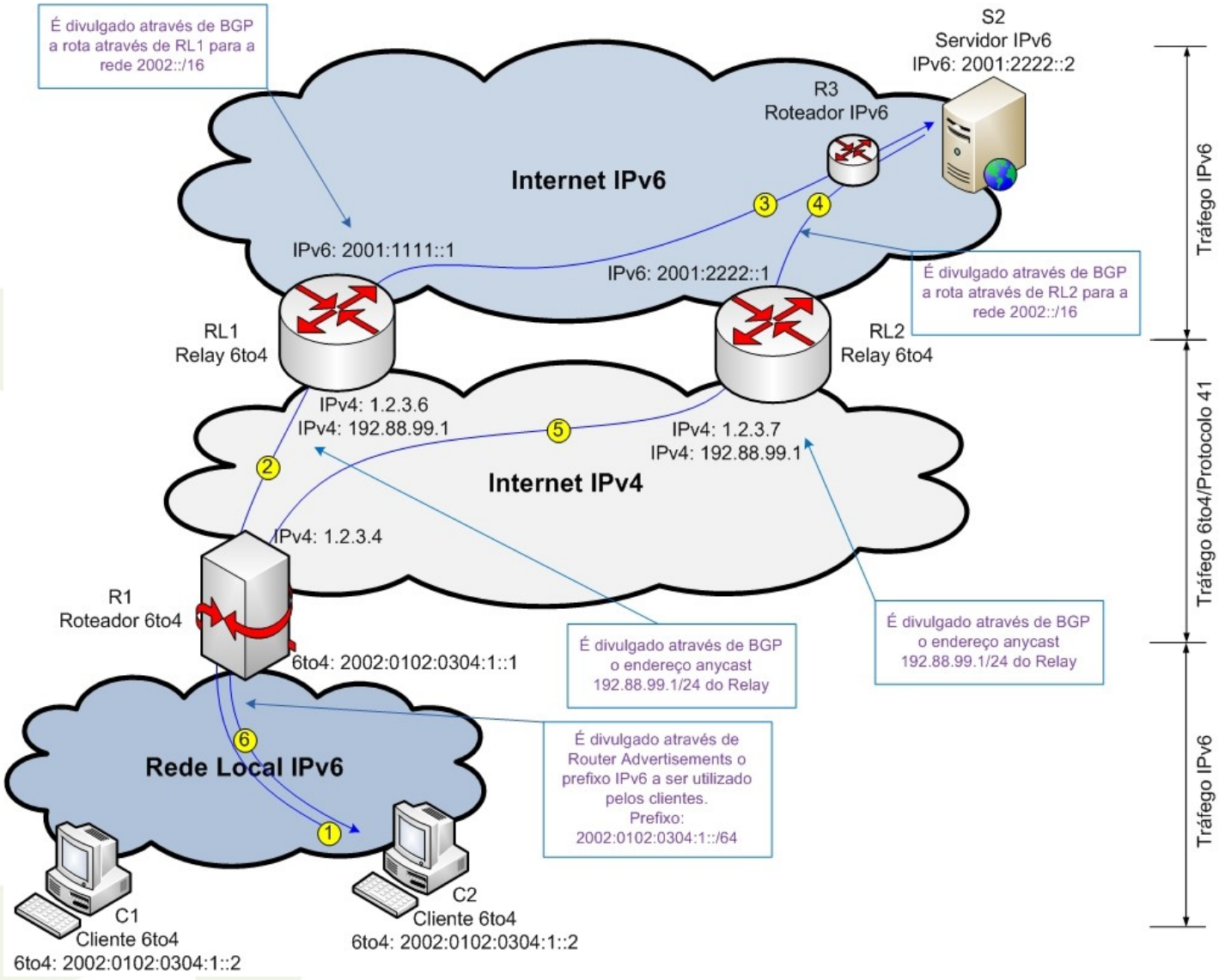
6to4

Comunicación Cliente 6to4 con un servidor IPv6 utilizando solo un Relay 6to4 (rutas de ida y vuelta iguales):



6to4

Comunicación Cliente 6to4 con un servidor IPv6 utilizando dos relays 6to4 diferentes (rutas de ida y vuelta diferentes)



6to4

- **Seguridad**

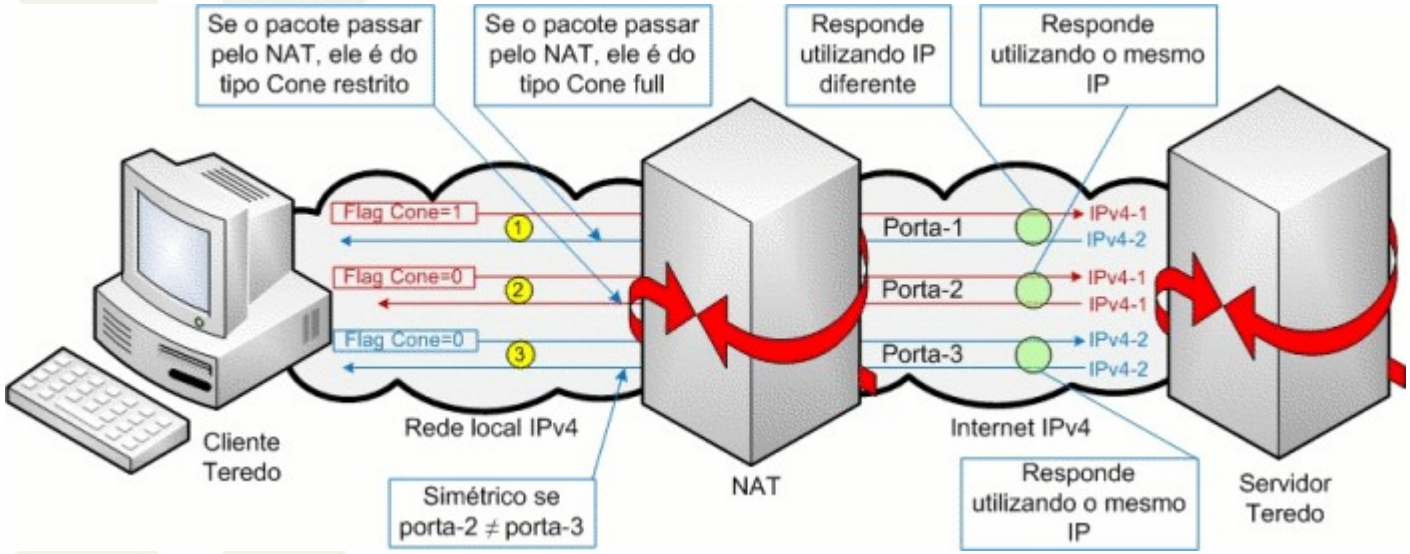
  - Los routers relay no verifican los paquetes IPv6 que están encapsulados en IPv4, a pesar de que sí los encapsula ny desencapsulan;

  - El spoofing de direcciones es un grave problema de los túneles 6to4 que puede ser fácilmente explorado;

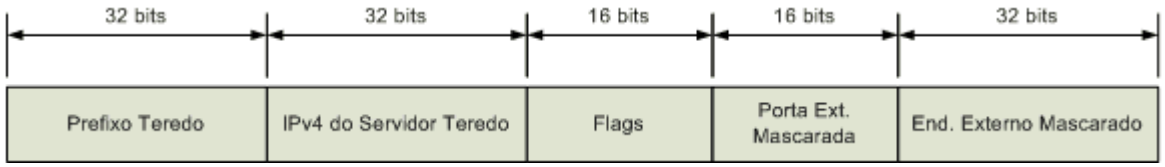
  - No hay un sistema de autenticación entre el router y el router relay, lo que facilita la exploración de la seguridad mediante el uso de routers relay falsos.

Teredo

- Encapsula el paquete IPv6 en paquetes UDP.
- Funciona a través de NAT tipo cono y cono restringido.
- Envía paquetes *bubles* periódicamente al servidor para mantener las configuraciones iniciales de la conexión UDP.
- Su funcionamiento es complejo y tiene *overhead.*

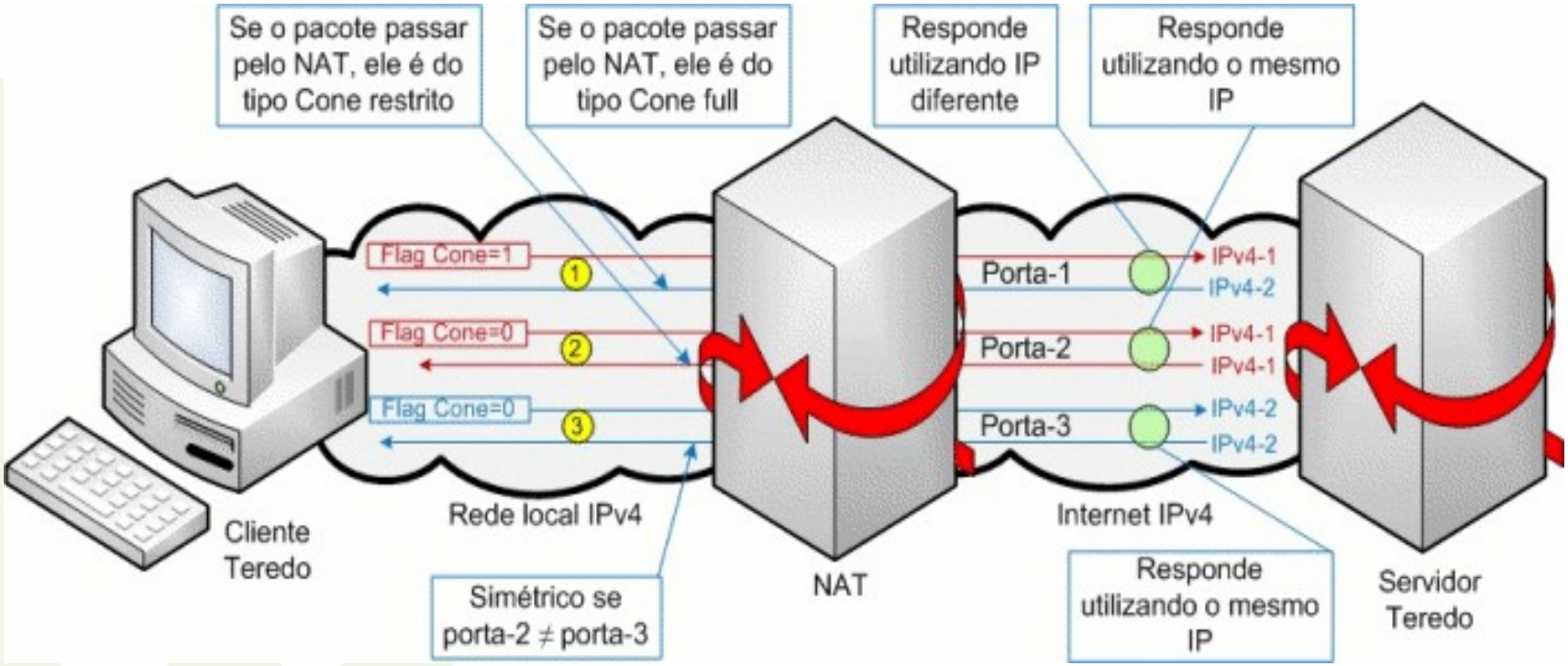


Teredo



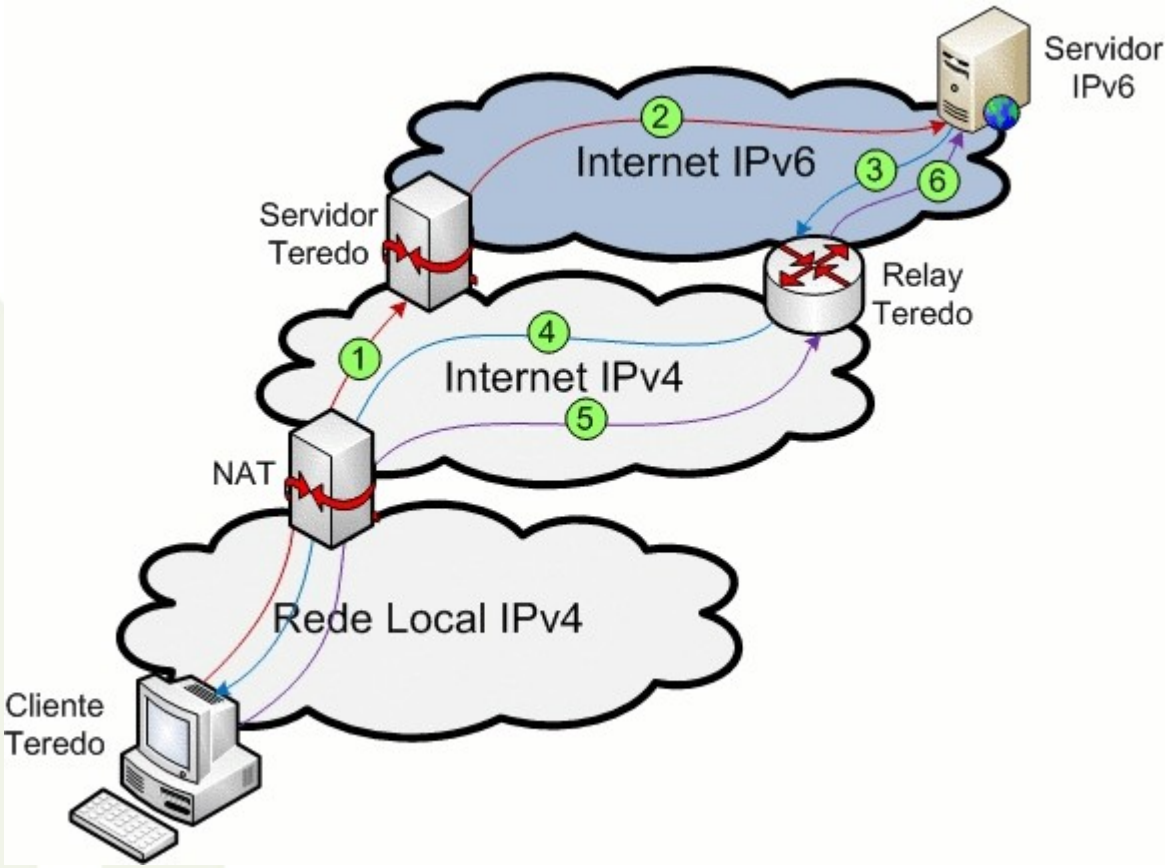
- Utiliza el prefijo **2001:0000::/32**.
- Los 32 bits siguientes contienen la dirección IPv4 del Servidor Teredo.
- Los 16 bits siguientes se utilizan para definir *flags* que indican el tipo de NAT utilizado e introducen una protección adicional contra los ataques de barrido.
- Los siguientes 16 bits indican el puerto UDP de salida del NAT.
- Los últimos 32 bits representan la dirección IPv4 pública del servidor NAT.

Teredo


Se o pacote passar pelo NAT, ele é do tipo Cone restrito
Se o pacote passar pelo NAT, ele é do tipo Cone full
Responde utilizando IP diferente
Responde utilizando o mesmo IP
Flag Cone=1
Flag Cone=0
Flag Cone=0
1
2
3
Porta-1
Porta-2
Porta-3
IPv4-1
IPv4-2
IPv4-1
IPv4-1
IPv4-2
IPv4-2
Cliente Teredo
Rede local IPv4
Simétrico se porta-2 ≠ porta-3
NAT
Internet IPv4
Responde utilizando o mesmo IP
Servidor Teredo

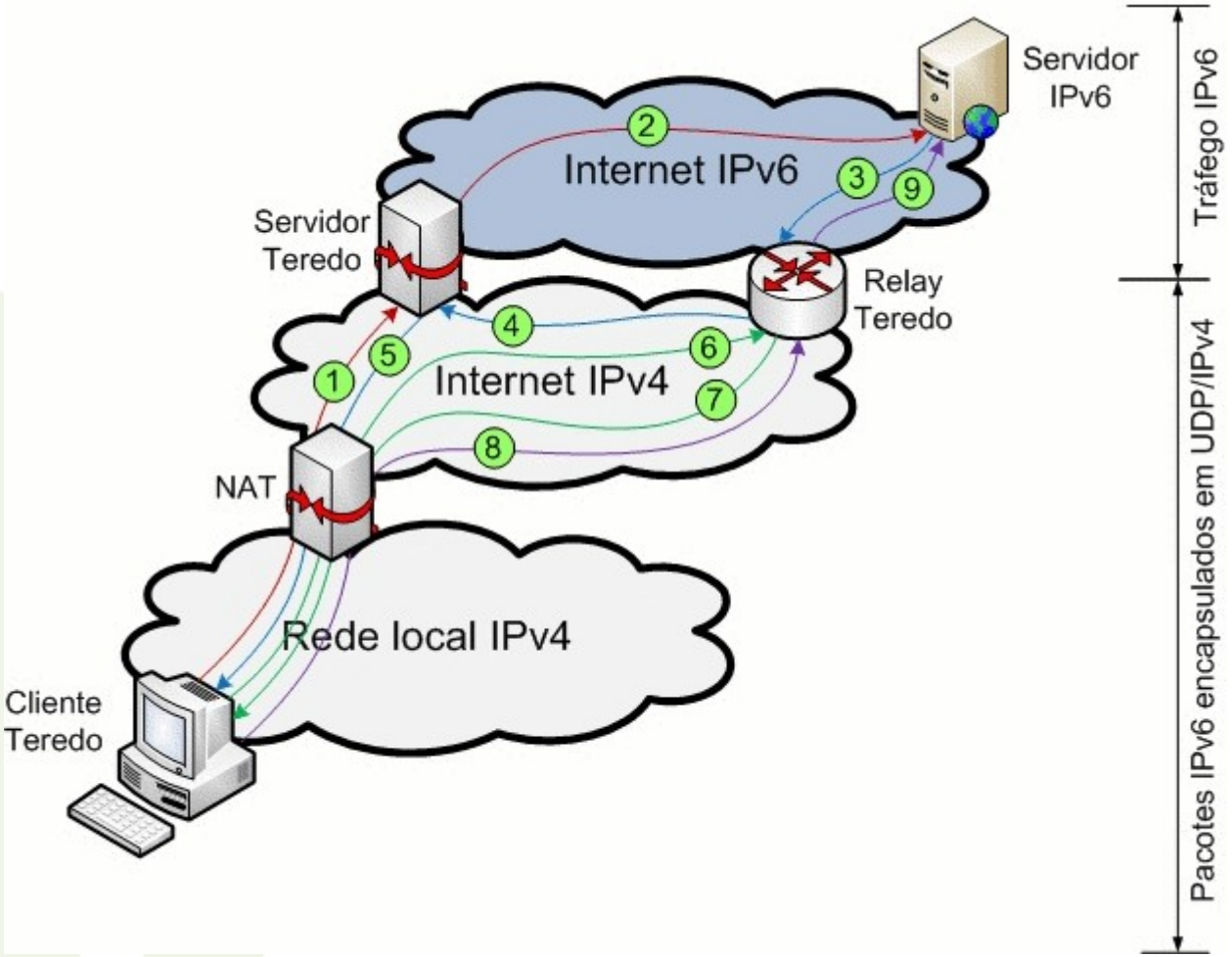
Teredo

Comunicación a través de NAT tipo CONO



Teredo

Comunicación a través de NAT restringido



Teredo

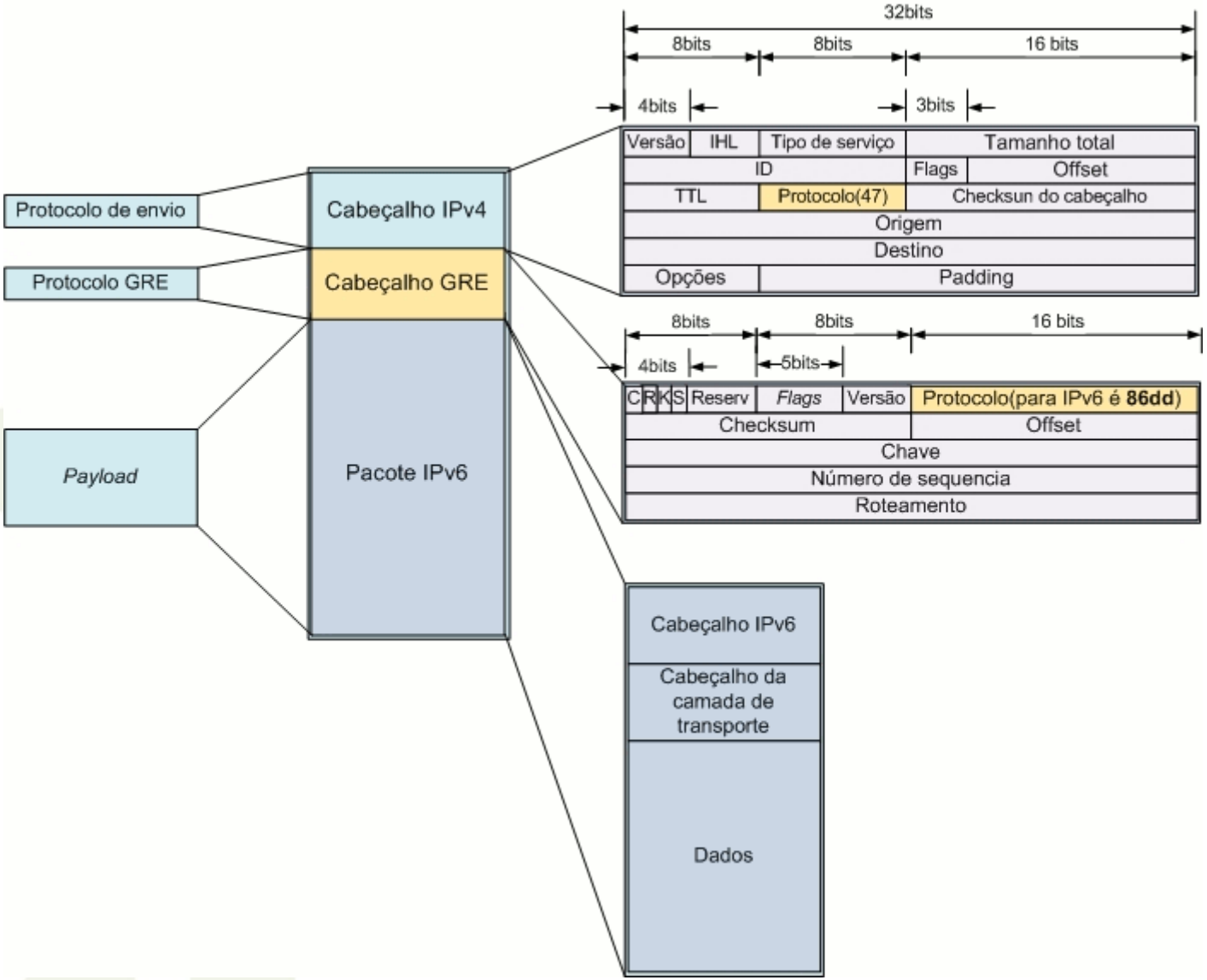
El principal problema de seguridad cuando se utiliza Teredo es que su tráfico puede pasar desapercibido por los filtros y *firewalls* si los mismos no están preparados para interpretarlo, por lo tanto las computadoras y la red interna quedan totalmente expuestos a ataques provenientes de la Internet IPv6. Para resolver este problema, antes de implementar Teredo se deben revisar los filtros y *firewalls* de la red o al menos de las computadoras que utilizarán esta técnica. Además de este problema, también existen los siguientes:

- El cliente Teredo publica en la red el puerto que abre en el NAT y el tipo de NAT que está utilizando, posibilitando así un ataque a través del mismo;
- El número de direcciones Teredo es mucho menor que las direcciones IPv6 nativas, lo que facilita la ubicación de computadoras vulnerables;
- Es fácil aplicar un ataque de denegación de servicio tanto en el cliente como en el *relay*;
- Debido al método de selección del *relay* por parte del *host* de destino, se puede crear un *relay* falso y utilizarlo para recolectar la comunicación de este *host* con sus clientes.

GRE

- GRE (*Generic Routing Encapsulation*) - Túnel estático *host*-a-*host* desarrollado para encapsular diferentes tipos de protocolos.
- Soportado por la mayoría de los sistemas operativos y routers.
- Funciona tomando los paquetes originales, agregando el encabezado GRE y enviándolos a la IP de destino.
- Cuando el paquete encapsulado llega a la otra punta del túnel, el encabezado GRE se elimina y queda solo el paquete original.

GRE


Protocolo de envio
Protocolo GRE
Payload
Cabeçalho IPv4
Cabeçalho GRE
Pacote IPv6
32bits
8bits
8bits
16 bits
4bits
3bits
Versão
IHL
Tipo de serviço
Tamanho total
ID
Flags
Offset
TTL
Protocolo(47)
Checksun do cabeçalho
Origem
Destino
Opções
Padding
8bits
8bits
16 bits
4bits
5bits
C
R
K
S
Reserv
Flags
Versão
Protocolo(para IPv6 é 86dd)
Checksum
Offset
Chave
Número de sequencia
Roteamento
Cabeçalho IPv6
Cabeçalho da camada de transporte
Dados

MPLS 6PE

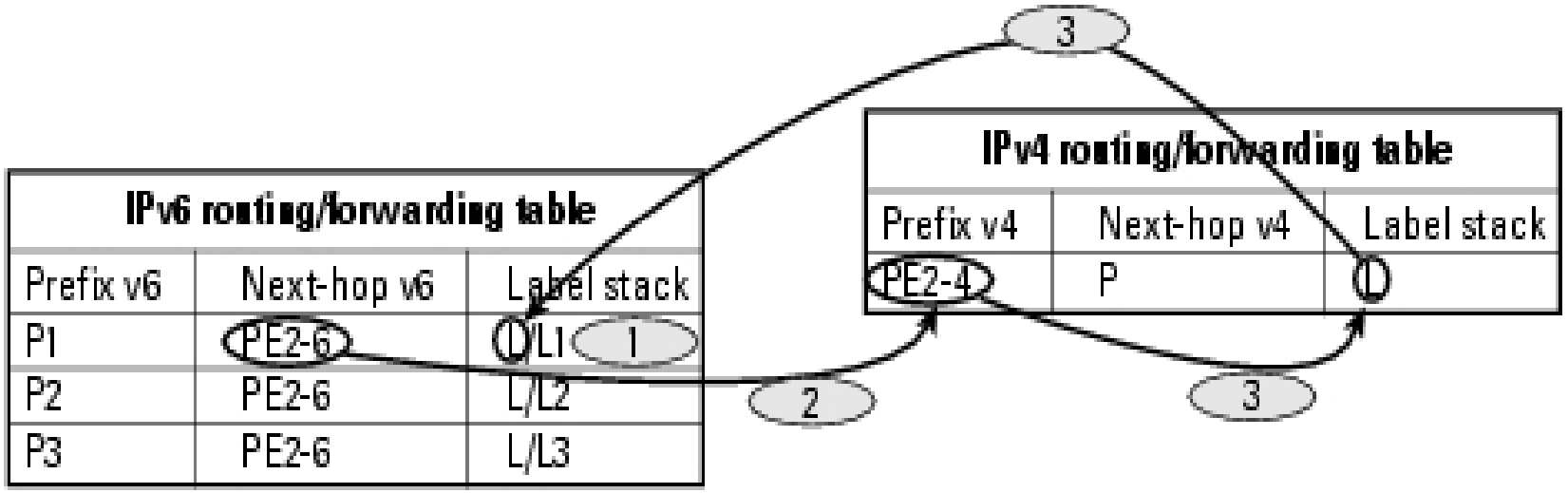
- Permite implementar IPv6 sin tocar el CORE
- Solo se configuran los PE
- Se pueden elegir los PE en los que se habilitará
- Los PE serán dual stack y no perderán funcionalidad IPv6
- No tiene problemas de performance o escalabilidad
- Requiere tener MPLS en el backbone

6PE en Cisco (Plano de Control)

Por cada prefijo 6PE recibido se realizan los siguientes pasos para determinar el next hop:

1.Cada prefijo se envia por BGP usando "IPv6+etiqueta". La etiqueta será la nueva entrada creada en CEF al recibir los bloques

2.El next-hop será la dirección IPv4 mapeada en IPv6 del neighbor



6PE Packet forwarding

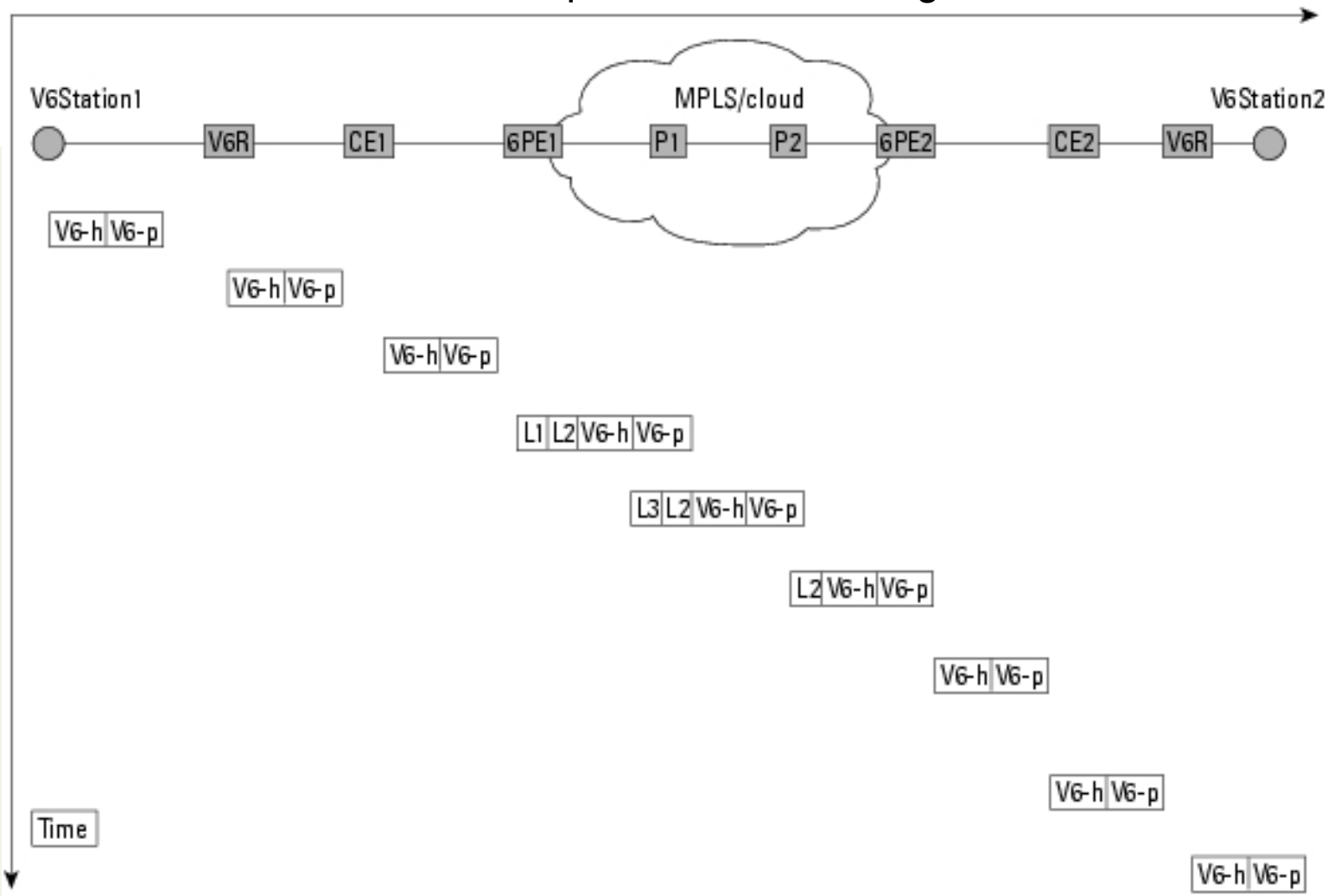
1.Forwarding IPv6 Normal (cuando tiene IPv6 Nativo para el destino)

2.Cuando recibe un paquete Ipv6, busca el IP destino en el FIB para saber que etiquetas colocar. Coloca primero la correspondiente al bloque IPv6 aprendido y luego la del LSP Ipv4 para llegar al router 6PE de salida.

3.Se envia el paquete hasta el destino usando MPLS común (top label swaping)

4.En el 6PE de salida, la etiqueta del bloque IPv6 informa al router 6PE que debe buscar el destino en la table FIB IPv6.

V6-h: Header Ipv6
V6-p: payload
L1: etiqueta forwarding (LSP)
L2: etiqueta del bloque IPv6 en el edge
L3, L4, L5...: etiquetas de forwarding en la nube MPLS



Técnicas de traducción

- Posibilitan un enrutamiento transparente en la comunicación entre los nodos de una red IPv6 y los nodos de una red IPv4 y viceversa.
- Pueden actuar de diversas maneras y en capas distintas:
  - Traduciendo encabezados IPv4 en encabezados IPv6 y viceversa;
  - Convirtiendo direcciones;
  - Convirtiendo APIs de programación;
  - Actuando en el intercambio de tráfico TCP o UDP.

SIIT

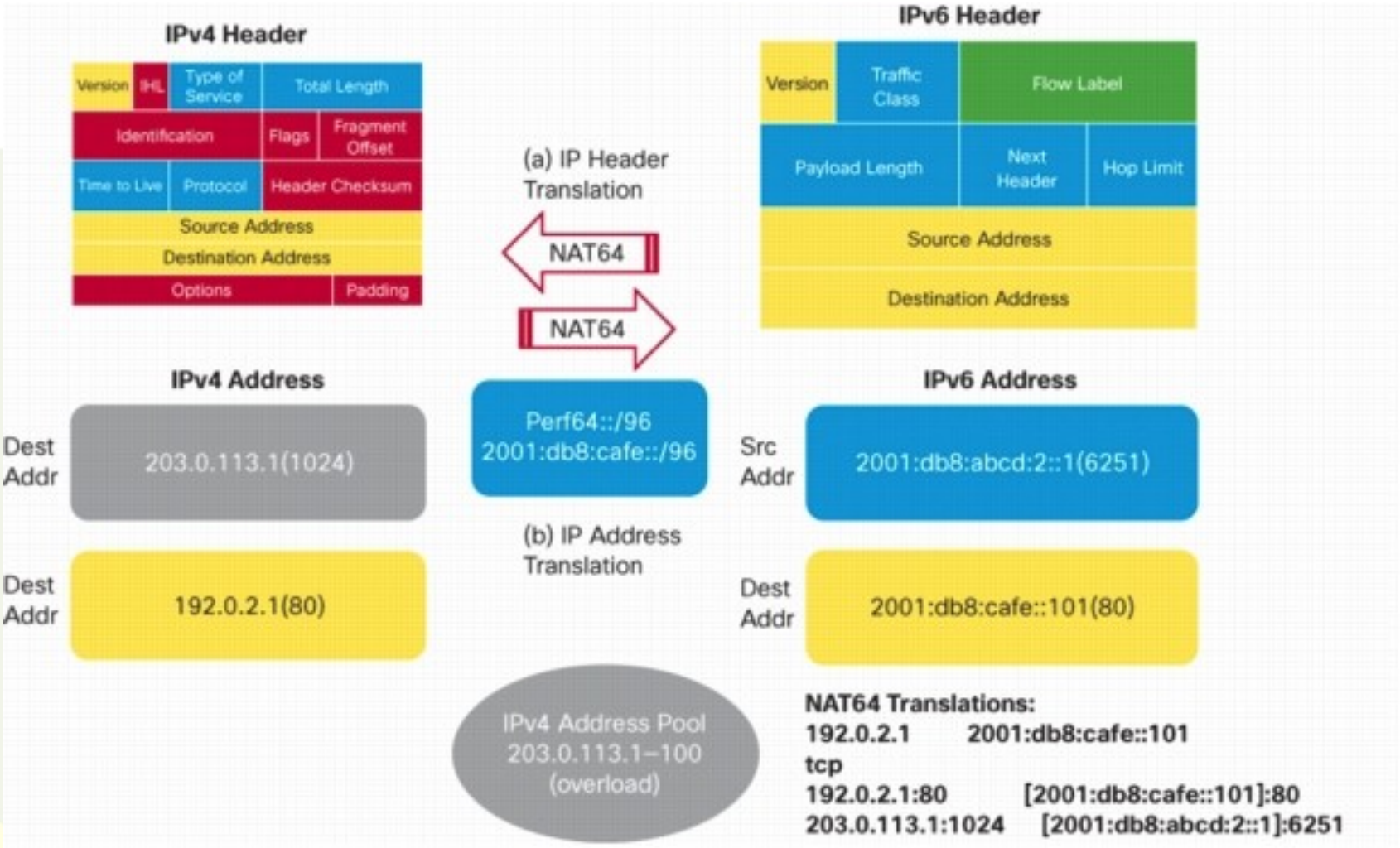
- SIIT (*Stateless IP/ICMP Translation*) - Permite la comunicación entre nodos que solo soportan IPv6 y nodos que solo soportan IPv4.
- Utiliza un traductor ubicado en la capa de red de la pila, el cual convierte campos específicos de los encabezados de paquetes IPv6 en encabezados de paquetes IPv4 y viceversa.
- Los encabezados TCP y UDP generalmente no se traducen.
- Utiliza una dirección IPv4-mapeada en IPv6, en el formato **0::FFFF:a.b.c.d**, que identifica el destino IPv4, y una dirección IPv4-traducida, en el formato **0::FFFF:0:a.b.c.d**, para identificar el nodo IPv6.
- Utiliza rangos de direcciones IPv4 para identificar nodos IPv6.
- Traduce mensajes ICMPv4 en ICMPv6 y viceversa.

ALG y DNS-ALG

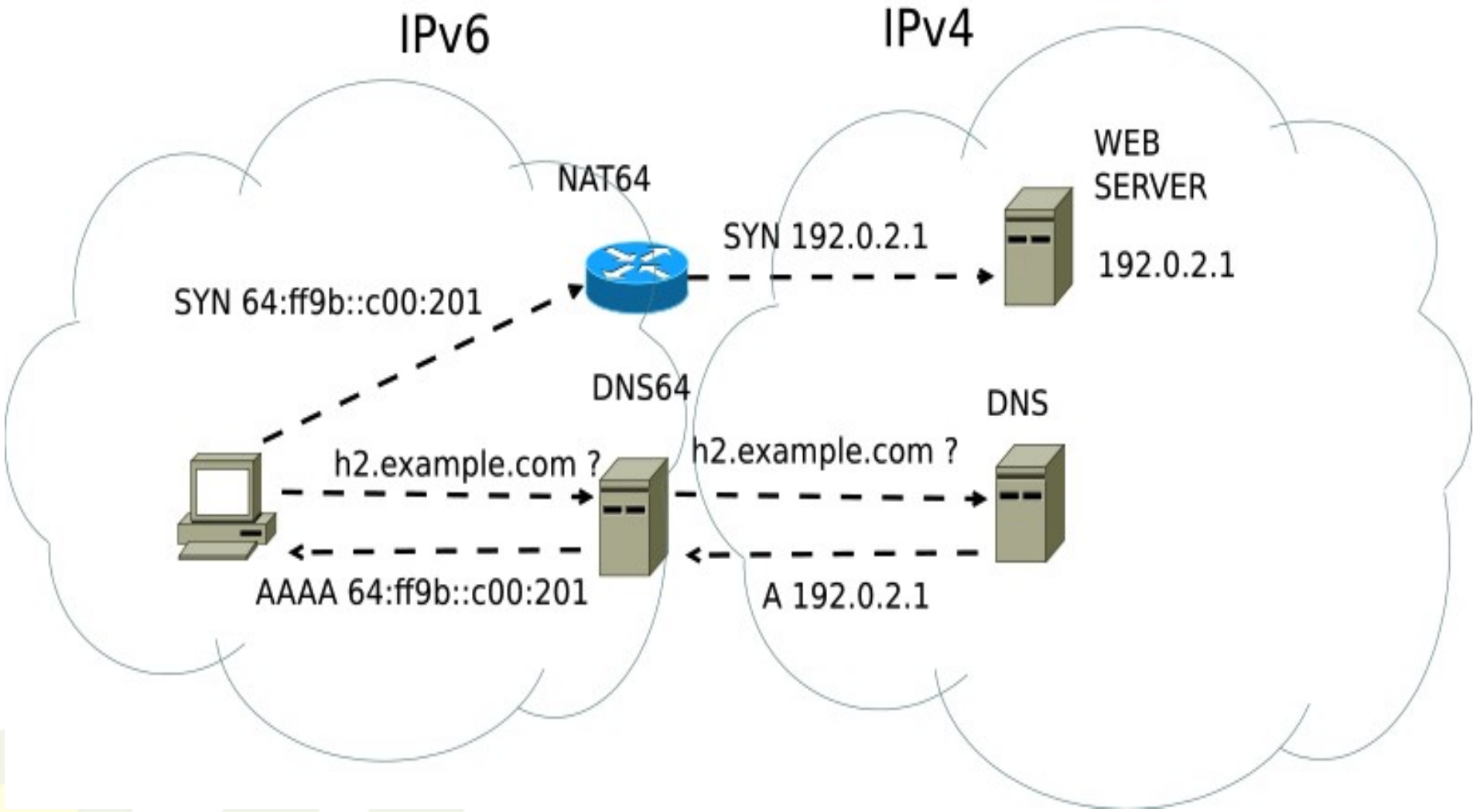
- ALG (*Application Layer Gateway*) - Trabaja como un *proxy* HTTP.
- El cliente inicia la conexión con el ALG, que establece una conexión con el servidor, retransmitiendo las solicitudes de salida y los datos de entrada.
- En redes solo IPv6, el ALG habilita la comunicación de los *hosts* con servicios en redes solo IPv4, configurando el ALG en nodos con doble pila.
- Normalmente se utiliza cuando el *host* que desea acceder a la aplicación en el servidor IPv4 está detrás de NAT o de un *firewall*.

- DNS-ALG - Traduce consultas DNS de tipo AAAA provenientes de un *host* IPv6 a consultas tipo A, en caso que el servidor de nombres a ser consultado se encuentre en el entorno IPv4, y viceversa.

NAT64


IPv4 Header
Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options
Padding
IPv6 Header
Version
Traffic Class
Flow Label
Payload Length
Next Header
Hop Limit
Source Address
Destination Address
(a) IP Header Translation
NAT64
NAT64
IPv4 Address
Dest Addr
203.0.113.1(1024)
Dest Addr
192.0.2.1(80)
Perf64::/96
2001:db8:cafe::/96
(b) IP Address Translation
IPv6 Address
Src Addr
2001:db8:abcd:2::1(6251)
Dest Addr
2001:db8:cafe::101(80)
IPv4 Address Pool
203.0.113.1–100
(overload)
NAT64 Translations:
192.0.2.1 2001:db8:cafe::101
tcp
192.0.2.1:80 [2001:db8:cafe::101]:80
203.0.113.1:1024 [2001:db8:abcd:2::1]:6251

NAT64 + DNS64


IPv6
IPv4
NAT64
WEB
SERVER
SYN 192.0.2.1
192.0.2.1
SYN 64:ff9b::c00:201
DNS64
DNS
h2.example.com ?
h2.example.com ?
AAAA 64:ff9b::c00:201
A 192.0.2.1