Router Teldat

Interfaz Túnel IP (TNIP)

Doc. *DM719* Rev. *10.10*
Abril, 2003

ÍNDICE

Capítulo 1 Interfaz túnel IP (TNIP) 1

1. Descripción 2
   - 1.1. Introducción 2
   - 1.2. Ventajas del tunneling 2
   - 1.3. Consideraciones especiales 2
2. Estructura de la trama encapsulada 4
   - 2.1. IP sobre IP con GRE 4
   - 2.2. IP sobre SRT con GRE 5
3. Paquetes de mantenimiento “keepalive” 7
   - 3.1. Paquete de petición “keepalive” 7
   - 3.2. Paquete de respuesta “keepalive” 7
4. Referencias 8

Capítulo 2 Configuración del Interfaz túnel IP (TNIP) 9

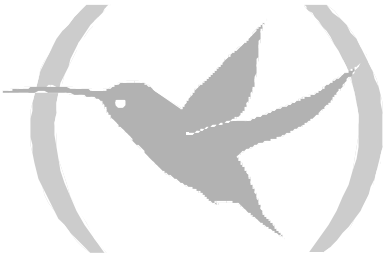
1. Creación del Interfaz túnel IP (TNIP) 10
2. Configuración del interfaz Túnel IP (TNIP) 12
   - 2.1. DESTINATION 12
   - 2.2. DISABLE 12
   - 2.3. ENABLE 13
   - 2.4. ENCAPSULATION 13
   - 2.5. KEEPALIVE 13
   - 2.6. LIST 14
   - 2.7. MODE 14
   - 2.8. QOS-PRE-CLASSIFY 14
   - 2.9. SOURCE 15
3. Configuración del protocolo de encapsulado GRE (Generic Routing Encapsulation) 16
   - 3.1. CHECKSUM 16
   - 3.2. CIPHER 16
   - 3.3. CIPHER-KEY 17
   - 3.4. KEY 17
   - 3.5. LIST 17
   - 3.6. SEQUENCE-DATAGRAMS 18

Capítulo 3 Túneles Dinámicos (Internet) 19

1. Descripción 20
   - 1.1. Escenarios/Problemática presentada 20
   - 1.2. Tipos de túneles 21
     - *a) Túneles dinámicos 21*
     - *b) Túneles semidinámicos 21*
     - *c) Túneles promiscuos 22*
   - 1.3. Importancia del RIP 22
2. Escenarios de utilización 24
   - 2.1. Funcionamiento de túneles sin navegar por Internet (Esc. 2/3) 24
     - *a) Mínimo esfuerzo de configuración a costa de RIP 24*
     - *b) Mayor esfuerzo de configuración reduciendo el tráfico RIP 25*
   - 2.2. Túneles y Navegación simultánea (Esc. 1 + 2/3) 25
     - *a) Mayor sobrecarga en la red / Mínimo esfuerzo de configuración 25*
     - *b) Mínima sobrecarga en la red / Mayor esfuerzo de configuración 26*
     - *c) Sobrecarga nula en la red / Mayor esfuerzo de configuración/Control de clientes 26*

3. Seguridad 28
Capítulo 4 Monitorización del Interfaz túnel IP (TNIP) 29
1. Monitorización del interfaz Túnel IP (TNIP) 30
a) LIST 30
- LIST STATE 30
2. Estadísticos del interfaz Túnel IP (TNIP) 31
Capítulo 5 Ejemplos de configuración de Túnel IP 32
1. Túnel IP sobre IP 33
1.1. Pasos a seguir en cada extremo del túnel 33
1.2. Pasos a seguir en los equipos que atraviesa el túnel 33
1.3. Ejemplo 1.a: IP sobre IP con GRE 33
a) Configuración Router1 33
b) Configuración Router2 34
c) Configuración Router3 35
1.4. Ejemplo 1.b: túnel promiscuo 37
a) Configuración CXSEC1 37
b) Configuración ROUTER1 39
c) Configuración ROUTER2 42
d) Resultado final 43
2. Túnel IP sobre SRT 45
2.1. Pasos a seguir en cada extremo del túnel 45
2.2. Pasos a seguir en los equipos que atraviesa el túnel 45
2.3. Ejemplo 2: IP sobre SRT con GRE 45
a) Configuración Router1 46
b) Configuración Router2 y Router 3 48
c) Configuración Router4 48
Capítulo 6 Eventos del interfaz Túnel IP (TNIP) 51
1. Monitorización de eventos del interfaz Túnel IP (TNIP) 52

Capítulo 1
Interfaz túnel IP (TNIP)



1. Descripción

1.1. Introducción

Se denomina *procesado Túnel* (Tunneling) al procedimiento mediante el cual paquetes de diversos protocolos son encapsulados dentro de otro protocolo. Dicha funcionalidad es implementada mediante un interfaz virtual que se denomina: *Interfaz Túnel.* El interfaz Túnel no está ligado de antemano a ningún protocolo de transporte, de encapsulado, o interno fijo, sino que es una arquitectura que proporciona los servicios necesarios para implementar cualquier esquema estándar de encapsulación. Como los túneles son enlaces punto a punto, se deben configurar túneles independientes para cada enlace.

El procesado Túnel posee tres componentes:

- Protocolo Interno, protocolo viajero o protocolo de carga (Payload Protocol): Es el protocolo que está siendo encapsulado (IP o SRT).
- Protocolo Encapsulador (Carrier Protocol): Es el protocolo que se encarga de encapsular.
    - ◊ Generic Routing Encapsulation (GRE).
- Protocolo de transporte, protocolo externo (Delivery Protocol): Es el protocolo que se encarga de transportar el protocolo interno ya encapsulado. (Sólo IP).

1.2. Ventajas del tunneling

Existen diversas situaciones en las que encapsular tráfico de un protocolo en otro es útil:

- Para interconectar redes locales multiprotocolo a través de un backbone con un sólo protocolo.
- Para resolver el problema de interconexión de redes que contienen protocolos con un número limitado de saltos y que sin este procedimiento no podrían llegar a conectarse.
- Para conectar dos subredes discontinuas.
- Para permitir Redes Privadas Virtuales a lo largo de redes WAN.

1.3. Consideraciones especiales

Los siguientes puntos describen consideraciones y precauciones que deben observarse a la hora de configurar túneles:

- El encapsulado y desencapsulado que se produce en los extremos del túnel son operaciones lentas.
- Hay que tener cuidado en las configuraciones y tener en cuenta los posibles problemas de seguridad y topología. Por ejemplo, se podría configurar un túnel cuyo origen y destino no estén restringidos por Firewalls.

- Hay que elegir correctamente los medios a través de los cuales irá el túnel. Podría darse el caso de que atravesara redes Fast FDDI y enlaces lentos de 9600 baudios. Algunos protocolos internos se comportan incorrectamente en redes compuestas de medios mixtos.
- Muchos túneles punto a punto podrían llegar a saturar un enlace con la información de encaminamiento.
- Aquellos protocolos de encaminamiento que deciden el mejor camino basándose únicamente en el número de saltos preferirán el túnel aunque exista un camino mejor sin pasar por él. El túnel siempre aparenta ser un salto aunque realmente su coste sea mayor.
- Un problema aún peor podría ocurrir si la información de encaminamiento de las redes conectadas por el túnel se llegara a mezclar con la de las redes que transportan dicha información. En esos casos, el mejor camino hacia el destino del túnel sería a través del túnel. A este tipo de ruta se le denomina *ruta recursiva* y provoca que el túnel se caiga temporalmente. Para impedir el problema de las rutas recursivas hay que mantener las informaciones de encaminamiento independientes;
    - ◊ Usando un número AS o TAG distinto.
    - ◊ Usando un protocolo de encaminamiento distinto.
    - ◊ Usando rutas estáticas para el primer salto (pero teniendo cuidado con los bucles de rutas).

2. Estructura de la trama encapsulada

En el caso del Túnel IP el protocolo de transporte o externo es IP por tanto la estructura de trama encapsulada sería la siguiente.

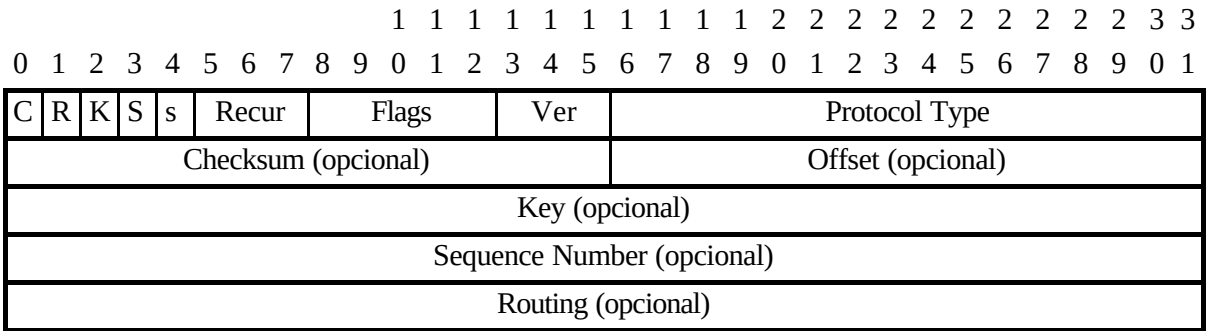
Cabecera del protocolo externo: IP
Cabecera del protocolo encapsulador
Paquete del protocolo interno

2.1. IP sobre IP con GRE

En este caso, el protocolo encargado del encapsulado es GRE. Y el protocolo interno o protocolo que está siendo encapsulado es IP. El protocolo de encapsulado GRE (Generic Routing Encapsulation) está descrito en la RFC1701 y el caso particular de IP sobre IP con GRE en la RFC1702.

Cabecera del protocolo externo: IP
Cabecera del protocolo encapsulador: GRE
Protocolo interno: IP

La cabecera IP está fuera de los límites de este documento, no así la cabecera GRE. La cabecera GRE posee la forma:



Checksum Present (bit 0) (C)

Si está puesto a 1, entonces el campo ***checksum*** está presente y contiene información válida. Si el bit de ***checksum*** o de ***routing*** está presente tanto el campo ***offset*** como el campo ***checksum*** estarán presentes en el paquete.

Routing Present (bit 1) (R)

No usado.

Key Present (bit 2)

Si está a 1, entonces el campo ***key*** (o identificador) está presente en el paquete y tiene valor válido.

Sequence Number Present (bit 3)

Si está a 1, entonces el campo Número de Secuencia (***Sequence Number***) está presente y contiene un valor válido.

Strict Source Route Present (bit 4)

No usado.

Recursion Control (bits 5-7)

Contiene un entero positivo de 3 bits que indica el número de encapsulaciones adicionales que están permitidas. Siempre 0.

Número de versión (bits 13-15)

Siempre 0.

Tipo de protocolo (2 octetos)

Contiene el tipo de protocolo del paquete interno.

Offset (2 octetos)

Indica el desplazamiento en octetos desde el inicio del campo ***routing*** hasta la primera ruta que debe ser examinada.

Checksum (2 octetos)

Contiene el ***checksum*** IP de la cabecera GRE y el paquete interno.

Key (4 octetos)

Identificador del túnel.

Número de secuencia (4 octetos)

Número usado por el receptor para asegurar el correcto orden de llegada de los paquetes.

Routing (longitud variable)

No existirá.

Cuando IP se encapsula en IP usando GRE el TOS y las opciones de seguridad IP son copiadas de la cabecera del protocolo interno (payload protocol) en la cabecera del protocolo externo (delivery protocol). EL TTL sin embargo no se copia sino que se establece al valor por defecto usado para IP con el fin de evitar que los paquetes RIP que viajen a través del túnel expiren antes de llegar al destino.

2.2. IP sobre SRT con GRE

De nuevo el protocolo encargado del encapsulado es GRE. En este caso el protocolo interno o protocolo que está siendo encapsulado es SRT.

Los campos de la cabecera GRE se rellenan e interpretan de la misma manera.

El TTL, TOS y las opciones de seguridad en la cabecera del protocolo externo (delivery protocol) son las usadas por defecto en IP.

3. Paquetes de mantenimiento “keepalive”

El interfaz Túnel IP dispone de un mecanismo “keepalive” para monitorizar la conectividad con el extremo remoto del túnel. Mediante este mecanismo se consigue que el interfaz esté operativo sólo cuando se disponga de conectividad real entre los extremos del túnel, y así poder tomar rutas alternativas (rutas de backup) sin necesidad de emplear protocolos de enrutado (routing) como RIP u OSPF.

La monitorización de la conectividad se realiza enviando paquetes de mantenimiento y comprobando que se recibe la respuesta. En los siguientes apartados se describen los paquetes de mantenimiento “keepalive” empleados en los interfaces Túnel IP.

3.1. Paquete de petición “keepalive”

El paquete que envía el equipo para determinar la conectividad con el extremo remoto se compone de:

1. Cabecera IP con:

   Precedencia (campo TOS) = “Internetwork Control”

   Dirección origen = Dirección de origen del túnel

   Dirección destino = Dirección de destino del túnel
2. Cabecera GRE con los parámetros configurados, y protocolo del paquete interno = IP
3. Cabecera IP con:

   Precedencia (campo TOS) = “Internetwork Control”

   Dirección origen = Dirección de destino del túnel

   Dirección destino = Dirección de origen del túnel
4. Cabecera GRE con los parámetros configurados (excepto número de secuencia), y protocolo del paquete interno = 0x0000.

3.2. Paquete de respuesta “keepalive”

Cuando un paquete de petición de keepalive llega al extremo destino del túnel (dirección IP de destino del túnel) el correspondiente equipo procesa la trama, y desencapsula el paquete interno. Este paquete interno se enruta normalmente, de vuelta al extremo origen del túnel.

Así pues, el paquete de petición “keepalive” es un paquete IP convencional encapsulado en GRE, por lo que el equipo remoto lo enrutará normalmente, aunque no disponga de la funcionalidad “keepalive”.

El paquete de respuesta “keepalive” se distingue porque el campo de protocolo del paquete interno (en la cabecera GRE) tiene el valor 0x0000. El formato completo es el siguiente:

1. Cabecera IP con:

   Precedencia (campo TOS) = “Internetwork Control”

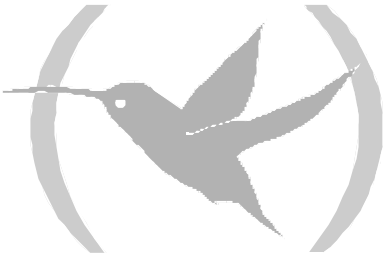
   Dirección origen = Dirección de destino del túnel

   Dirección destino = Dirección de origen del túnel
2. Cabecera GRE con los parámetros configurados (excepto número de secuencia), y protocolo del paquete interno = 0x0000.

4. Referencias

RFC-1701: Generic Routing Encapsulation (GRE), S. Hanks, Octubre-1994
RFC-1702: Generic Routing Encapsulation over IPv4 networks, S. Hanks, Octubre-1994

Capítulo 2
Configuración del Interfaz túnel IP (TNIP)



1. Creación del Interfaz túnel IP (TNIP)

Para crear un interfaz de tipo Túnel IP se debe introducir ***"ADD DEVICE tnip <identificador del túnel>"*** en el menú de configuración global.

```
Config>ADD DEVICE tnip 1
Added TNIP interface tnip1
Config>
```

Para entrar posteriormente en configuración basta con teclear ***"NETWORK tnipX"***, donde **X** representa el identificador del túnel:

```
Config>NETWORK tnip1

-- IP Tunnel Net Configuration --
TNIP config>
```

El protocolo que es soportado sobre el interfaz TNIP es el IP. Es necesario para activar el IP sobre el interfaz TNIP asignar una dirección IP al citado interfaz o configurarlo como interfaz de tipo no numerado.

Ejemplo con dirección IP conocida:

```
*P 4
Config>PROTOCOL IP

-- Internet protocol user configuration --
IP config>ADDRESS tnip1 5.5.5.1 255.255.0.0
IP config>LIST ADDRESSES
IP addresses for each interface:
 ethernet0/0     172.16.200.15   255.255.255.0   NETWORK broadcast,  fill 0
 serial0/0                                       IP disabled on this ifc
 atm0/0                                          IP disabled on this ifc
 bri0/0                                          IP disabled on this ifc
 x25-node                                        IP disabled on this ifc
 tnip1           5.5.5.1         255.255.0.0     NETWORK broadcast,  fill 0
IP config>
```

Ejemplo como interfaz no numerado:

```
*P 4
Config>PROTOCOL IP

-- Internet protocol user configuration --
IP config>ADDRESS tnip1 unnumbered
IP config>LIST ADDRESSES
IP addresses for each interface:
 ethernet0/0     17.16.200.15    255.255.255.0   NETWORK broadcast,  fill 0
 serial0/0                                       IP disabled on this ifc
 atm0/0                                          IP disabled on this ifc
 bri0/0                                          IP disabled on this ifc
 x25-node                                        IP disabled on this ifc
 tnip1           unnumbered      0.0.0.0         NETWORK broadcast,  fill 0
IP config>
```

2. Configuración del interfaz Túnel IP (TNIP)

En este apartado se describen los comandos de configuración del interfaz TNIP. Para acceder al entorno de configuración de TNIP, se debe introducir ***"NETWORK <interfaz TNIP>"***:

```
Config>NETWORK tnip1

-- IP Tunnel Net Configuration --
TNIP config>
```

Los comandos disponibles son los siguientes:

Comando	**Función**
DESTINATION	Configura la dirección IP destino del túnel.
DISABLE	Deshabilita el interfaz túnel.
ENABLE	Habilita el interfaz túnel.
ENCAPSULATION	Accede al menú de configuración del protocolo encapsulador.
KEEPALIVE	Habilita el mantenimiento "keepalive".
LIST	Muestra los parámetros configurados.
MODE	Selecciona el modo de encapsular en el interfaz túnel (protocolo encapsulador).
QOS-PRE-CLASSIFY	Habilita la preclasificación de paquetes de BRS.
SOURCE	Configura la dirección IP origen del túnel.

2.1. DESTINATION

Configura la dirección IP destino del túnel IP. Debe coincidir con la dirección IP configurada como origen del túnel en el router del otro extremo. Si la dirección IP destino del túnel no coincide con la configurada como origen en el otro extremo, los paquetes que se envíen a dicho router serán descartados por no pertenecer al túnel.

Es necesario que exista ruta hacia la dirección IP destino pues si no los paquetes del túnel no pueden ser encaminados. Como precaución dicha ruta debe ser una ruta estática para evitar el problema de recursividad en la tabla de rutas explicado en el capítulo 1.

Ejemplo:

```
TNIP config>DESTINATION 66.187.232.56
TNIP config>
```

2.2. DISABLE

Deshabilita el interfaz túnel. Por defecto el interfaz túnel está deshabilitado.

Ejemplo:

```
TNIP config>DISABLE
TNIP config>
```

2.3. ENABLE

Habilita el interfaz túnel. Por defecto el interfaz túnel no se encuentra activo.

Ejemplo:

```
TNIP config>ENABLE
TNIP config>
```

2.4. ENCAPSULATION

Accede a la configuración del protocolo encapsulador. De momento el único protocolo encapsulador soportado es GRE (Generic Routing Encapsulation).

Ejemplo:

```
TNIP config>ENCAPSULATION

-- GRE Configuration --
GRE config>
```

2.5. KEEPALIVE

Habilita el mantenimiento "keepalive" del Túnel IP. Este mantenimiento consiste en el envío periódico de paquetes de petición "keepalive". Si no se reciben dentro del periodo configurado se determina la pérdida de conectividad del túnel, y se deja el interfaz de Túnel IP inoperativo (estado "down") hasta que se restablezca la conectividad.

El formato de este comando es "**KEEPALIVE** [<*periodo*> [<*intentos*>]]". Los parámetros se describen a continuación:

Parámetro	Descripción
periodo	Número de segundos entre envíos sucesivos de paquetes de petición keepalive. También actúa como tiempo máximo de respuesta, ya que sólo se consideran las respuestas al último paquete de petición keepalive enviado. El rango permitido es de 1 a 32767 segundos, y el valor por defecto 10 segundos.
intentos	Número de paquetes consecutivos de petición keepalive sin respuesta para determinar que se ha perdido la conectividad. El rango permitido es de 1 a 255 envíos sin respuesta, y el valor por defecto 3.

Para deshabilitar el mantenimiento "keepalive" se emplea el comando "**NO KEEPALIVE**".

Ejemplo:

```
TNIP config>KEEPALIVE 30 5
TNIP config>
```

2.6. LIST

Muestra la configuración del túnel IP.

Ejemplo:

```
TNIP config>LIST
Tunnel mode: GRE (enabled)
Tunnel source 212.95.195.132, destination 66.187.232.56
QoS preclassify: disabled
Keepalive enabled with period 10, 3 retries
TNIP config>
```

Tunnel mode: indica el tipo de encapsulado y el estado (habilitado/deshabilitado).

Tunnel source / destination: direcciones IP origen / destino del túnel.

QoS preclassify: indica si se encuentra habilitada la preclasificación de BRS.

Keepalive: muestra la configuración del mantenimiento "keepalive".

2.7. MODE

Selecciona el modo de encapsulación. De momento se soporta únicamente GRE (Generic Routing Encapsulation).

Ejemplo:

```
TNIP config>MODE GRE
TNIP config>
```

2.8. QOS-PRE-CLASSIFY

Habilita la preclasificación de paquetes por parte del BRS. Al habilitar esta opción los paquetes que llegan al túnel son clasificados por el BRS (consultar el manual de BRS, Dm715) antes de ser encapsulados por el túnel. Esto permite distinguir entre distintos tipos de tráfico IP que son enviados a través del túnel. Si esta opción está deshabilitada los paquetes serán clasificados una vez encapsulados, por lo que todo el tráfico que sea procesado por el túnel tendrá la misma cabecera IP (la que pone el túnel) y serán clasificados todos en la misma clase de BRS.

Para deshabilitar este parámetro se utiliza ***"NO QOS-PRE-CLASSIFY"***.

Ejemplo:

```
TNIP config>QOS-PRE-CLASSIFY
TNIP config>
```

2.9. SOURCE

Configura la dirección IP origen del túnel IP. Debe coincidir con la dirección IP de uno de los interfaces configurados en el router (Ethernet, PPP, Loopbak, etc.) **excepto** la del propio túnel. También debe coincidir con la dirección IP configurada como destino en el equipo que sea el otro extremo del túnel.

Si la dirección IP origen del túnel no coincide con ninguno de los interfaces del router los paquetes destinados a esta dirección IP no son considerados por el router como propios y los intentará encaminar hacia otro equipo.

Si la dirección IP configurada como origen no coincide con la configurada como destino en el otro extremo del router, no existirá nunca enlace.

Si el origen del túnel es un interfaz PPP que recibe asignación dinámica de dirección IP (consultar el manual del Interfaz PPP, Dm710), entonces hay que configurar como origen del túnel IP la dirección **"0.0.0.0"**.

Ejemplo:

```
TNIP config>SOURCE 212.95.195.132
TNIP config>
```

3. Configuración del protocolo de encapsulado GRE (Generic Routing Encapsulation)

En este apartado se describen los comandos de configuración del protocolo encapsulador GRE. Para acceder al entorno de configuración de GRE hay que introducir el comando ***"ENCAPSULATION"*** en el menú de configuración del interfaz túnel (con el interfaz configurado en modo de encapsulación GRE).

```
Config>NETWORK tnip1

-- IP Tunnel Net Configuration --
TNIP config>ENCAPSULATION

-- GRE Configuration --
GRE config>
```

Los comandos disponibles son los siguientes:

Comando	**Función**
CHECKSUM	Habilita el checksum extremo-a-extremo (GRE).
CIPHER	Habilita el cifrado RC4 en el túnel GRE.
CIPHER-KEY	Configura la clave del cifrado RC4.
KEY	Configura el identificador de túnel.
LIST	Muestra los parámetros configurados.
SEQUENCE-DATAGRAMS	Descartar datagramas recibidos fuera de orden.

3.1. CHECKSUM

Habilita la opción de envío de checksum en el paquete GRE. Por defecto, el túnel no garantiza la integridad de los paquetes. Habilitando dicha opción el router envía los paquetes GRE con campo checksum. Si se recibe un paquete con el campo checksum siempre se comprueba el checksum, descartando aquellos paquetes que lo tengan inválido, independientemente de que el equipo tenga o no habilitada la opción.

Para deshabilitar el checksum se utiliza ***"NO CHECKSUM"***.

Ejemplo:

```
GRE config>CHECKSUM
GRE config>
```

3.2. CIPHER

Activa el cifrado RC4 de los paquetes encapsulados en el túnel GRE. Por defecto no se encuentra habilitado el cifrado.

Aunque los paquetes de petición keepalive se cifran, los de respuesta no se cifran, ya que realmente no se encapsulan en el túnel.

Para deshabilitar el cifrado RC4 se utiliza ***"NO CIPHER"***.

Ejemplo:

```
GRE config>CIPHER
GRE config>
```

3.3. CIPHER-KEY

Configura la clave de cifrado del interfaz túnel. Dicha clave admite un máximo de 32 caracteres alfanuméricos.

Para restablecer la clave por defecto de cifrado en túneles GRE se utiliza ***"NO CIPHER-KEY"***.

Ejemplo:

```
GRE config>CIPHER-KEY thisIsAnExample
GRE config>
```

3.4. KEY

Habilita la comprobación del identificador del túnel. Al habilitarse esta opción el equipo solicita un identificador para el túnel en cuestión. Dicho identificador de túnel **ha de ser igual en ambos extremos** del túnel. El identificador es un número entero comprendido entre 0 y 4294967295 (32 bits). Por defecto el túnel tiene deshabilitada esta opción.

Cuando el identificador de túnel se encuentra habilitado el router descarta aquellos paquetes recibidos con un identificador distinto al configurado.

Ejemplo:

```
GRE config>KEY 5
GRE config>
```

3.5. LIST

Muestra la configuración del protocolo GRE.

Ejemplo:

```
GRE config>LIST
RC4 Cipher.................: enabled
End-to-End Checksumming....: enabled
Tunnel identification key..: enabled [5]
Drop Out-of-Order Datagrams: disabled
GRE config>
```

RC4 Cipher: indica si se encuentra habilitado el cifrado RC4.

End-toEnd Checksumming: indica si se encuentra habilitado el checksum extremo a extremo.

Tunnnel identification key: identificador de túnel (si se ha habilitado).

Drop Out-of-Order Datagrams: descartar datagramas recibidos fuera de orden..

3.6. SEQUENCE-DATAGRAMS

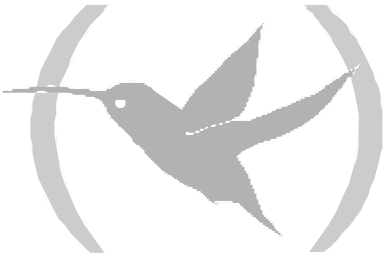
Habilita la opción de asegurar orden en datagramas entrantes. Habilitando esta opción el router comprueba el número de secuencia incluido en la cabecera GRE y descarta aquellos paquetes que lleguen fuera de orden. Por defecto, el túnel GRE tiene deshabilitada esta opción.

Para deshabilitar el número de secuencia se utiliza ***"NO SEQUENCE-DATAGRAMS"***.

Ejemplo:

```
GRE config>SEQUENCE-DATAGRAMS
GRE config>
```

Capítulo 3
Túneles Dinámicos (Internet)



1. Descripción

Si aplicamos la técnica de los túneles a las redes públicas Internet será posible la interconexión de redes locales dispersas de una forma barata y eficaz. A partir de la técnica de túneles vista en los capítulos anteriores podría realizarse, pero nos encontraríamos con algunas dificultades que trataremos en este capítulo, entre ellas las siguientes:

1. Para establecer un túnel entre dos puntos, es imprescindible que ambos conozcan la dirección IP del extremo, por lo que sólo sería realizable con accesos autentificados e IP fijas (lo que es un recurso limitado y caro en Internet).
2. La conexión de *n* redes locales requeriría configurar y establecer en cada uno de ellos *n-1* túneles.

La solución a estos problemas pasa por un equipo central (con una dirección IP fija y conocida) que soporta *n* túneles, y que gestiona el tráfico inter-túnel por lo que el segundo problema queda solucionado automáticamente mientras que la solución al primero consiste en dotar a este equipo de la capacidad de adaptar la configuración de sus túneles para permitir conexiones de equipos cuya IP es distinta cada vez que se conectan. **Esta reconfiguración dinámica del túnel cada nueva conexión es el motivo de que estos túneles reciban el nombre de túneles dinámicos.**

En adelante denominaremos al equipo central como el router de ISP, puesto que su ubicación normal será en un Internet Server Provider, mientras que los routers que se conectan a él los denominaremos los routers clientes.

1.1. Escenarios/Problemática presentada

- **Escenario 1:** Acceso de los equipos de una red local a Internet para servicios de la red (html, ftp, etc.) mediante un router.
- **Escenario 2**: Interconexión de redes locales remotas con la red local del ISP, mediante routers con túneles.
- **Escenario 3:** Interconexión de redes locales remotas entre sí, y con la red local del ISP, mediante routers con túneles.

Si el objetivo es el escenario 2/3, la configuración del los routers clientes es relativamente sencilla, puesto que podemos predecir que cualquier dirección IP que no sea local es accesible mediante el túnel, pero si la intención es permitir simultáneamente el escenario 1, los routers clientes deben distinguir si una determinada dirección destino es accesible por el túnel o fuera de él (dirección de Internet). Esto obliga a que los routers clientes conozcan las redes alcanzables a través del túnel. La solución pasa por configurar todas las rutas posibles en todos los clientes, o configurarlas en el router central ISP, y que éste informe a los clientes mediante un protocolo de routing (RIP).

Además el equipo del ISP debe conocer las redes accesibles a través de routers clientes.

1.2. Tipos de túneles

Se realiza la siguiente clasificación de túneles con el objetivo de analizar los distintos comportamientos en cada caso:

- **Estáticos**: Túneles donde las direcciones origen y destino son fijas. Estos túneles han sido tratados anteriormente.
- **Dinámicos**: Cuando una de las direcciones (origen o destino) del túnel es desconocida antes de efectuarse la conexión, y el equipo que se conectará es desconocido.
- **Semidinámicos**: Se trata de un caso especial de túnel dinámico en el que a pesar de no conocer la dirección (origen o destino) del túnel conocemos el equipo que se conectará, pues el identificador de túnel (campo *key* de GRE) es único.
- **Promiscuos**: Son un caso especial de túnel estático en el que no se conocen las direcciones origen ni destino del túnel. Actúan a modo de "interfaz túnel por defecto" recibiendo el tráfico que no va destinado a ningún otro interfaz túnel, pero no permiten transmitir (encapsular) tráfico.

(Los túneles dinámicos y semidinámicos son el caso normal en Internet cuando la dirección obtenida por el equipo remoto no está preasignada).

a) Túneles dinámicos

Son los túneles más fáciles de configurar y a la vez los más flexibles, por lo que se recomienda su uso frente a los otros tipos.

El equipo de ISP ofrece *n* túneles a los clientes, que éstos utilizan a medida que se conectan; cuando un cliente deja de enviar información, el túnel queda disponible para una nueva conexión.

La potencia de esta configuración radica en el uso de RIP, imprescindible para que el cliente informe al ISP de las redes accesibles a través de él y viceversa.

b) Túneles semidinámicos

Son una extensión de los túneles dinámicos, en los que se discriminan los equipos remotos que se permiten conectar en cada túnel del ISP. Para ello se configura un identificador único en el túnel que debe coincidir con el configurado en el equipo remoto (campo *key* de GRE). Realmente es como configurar un único túnel dinámico para cada equipo remoto.

Estos túneles añaden dos funcionalidades:

- Identificación del equipo remoto.
- Como se conoce a priori el cliente que se conecta a cada túnel, se pueden agregar rutas, y por tanto no es necesario habilitar RIP.

Por lo tanto estos túneles se recomiendan únicamente cuando se quiera prescindir de RIP o la seguridad y el control de acceso sea indispensable, pues requieren mayor esfuerzo de configuración, al tener que relacionar identificadores con equipos remotos.

c) Túneles promiscuos

Son un caso especial de túneles estáticos, en el que no conocemos las direcciones de origen ni destino del túnel.

Estos túneles reciben todo el tráfico que no corresponde a ningún otro interfaz de túnel, siempre que la configuración del identificador de túnel (campo *key* de GRE) se corresponda con el paquete recibido, pero no permiten transmitir (encapsular) ningún tipo de tráfico.

Mediante esta configuración del interfaz de túnel IP se puede recibir tráfico simultáneamente de muchos túneles con un solo interfaz, aunque no podemos transmitir porque el interfaz es estático, y por lo tanto no puede aprender las direcciones del túnel.

Este tipo de túnel es útil en casos muy particulares, como por ejemplo cuando queremos aprender por RIP las redes remotas a las que accedemos por IPSEC.

1.3. Importancia del RIP

Uno de los aspectos más delicados de este tipo de túneles es el control del estado de los mismos; a partir de un estado en el que están *a la espera* pasa a un estado *conectado* con el equipo remoto *cliente* que lo solicite. Mientras el túnel permanece en uso se mantiene en este estado, pero cuando deja de ser utilizado debe volver al estado inicial, en espera de futuros usos.

Uno de los aspectos más críticos cuando se establecen túneles dinámicos es la decisión de si el túnel permanece en uso o no, pues el equipo remoto puede haberse desconectado o apagado sin previo aviso, por lo que es necesario un* diálogo *entre los routers que sostienen los túneles, para lo que se utiliza el protocolo RIP.

Si se quiere evitar la reutilización de un túnel, reservándolo exclusivamente para dar servicio a un equipo cliente, se identificará éste con un* key *único.

Resumen de facilidades que aporta RIP:

a) Control de desconexión para reutilización de túneles.
b) Informar de red(es) accesibles.

Problemas que puede presentar el RIP:

Cuando las direcciones IP de los extremos de un túnel son de redes distintas, puede darse el caso de que un router reciba información de accesibilidad de la red destino del túnel a través del propio interfaz túnel, por lo que se perdería el acceso al extremo remoto. Esto no se dará nunca en Internet cuando la dirección que adquiera el cliente pertenezca a la red del equipo de ISP, pero sí en el resto de casos.

Para solucionarlo bastaría con añadir una ruta estática para acceder al destino, siempre que se conozca a priori.

En cualquier caso esta situación es detectada por los routers, que a su vez informan con eventos y estadísticos.

2. Escenarios de utilización

Es fundamental definir la utilización del router antes de configurar túneles en el mismo, pues el aprovechamiento óptimo de router y línea de comunicaciones está en función de que la configuración encaje al máximo con las necesidades.

La decisión más importante radica en la configuración de rutas estáticas, o delegar esta cuestión en un protocolo de routing, que facilitará la configuración pero disminuirá el rendimiento al utilizar parte del ancho de banda de la línea en intercambiar mensajes entre routers.

Aunque no existe una norma general, sí es posible basarse en unas directrices que ayudarán a tomar la decisión más conveniente, para lo cual es imprescindible definir el escenario en el que trabajará el router:

- **Escenario 2/3 (túnel para interconectar redes locales por Internet):** Todas las direcciones no locales son accesibles a través del túnel, por lo que resulta suficiente configurar éste como ruta por defecto (excepto la dirección del extremo remoto del túnel, que debe configurarse de forma estática).
- **Escenario 1+2/3 (navegar por Internet e interconectar redes locales por Internet mediante túneles):** Las direcciones no locales son accesibles por Internet, pero existen túneles dinámicos o semidinámicos por Internet, configurados en el router de cliente por cada LAN remota.

Además hay que tener en cuenta aspectos vistos en capítulos anteriores:

- Cuando se utilicen túneles *dinámicos* es necesario configurar RIP, debido a la reutilización de los mismos.

2.1. Funcionamiento de túneles sin navegar por Internet (Esc. 2/3)

La configuración en los clientes se basa en definir el túnel como el camino por defecto hacia cualquier destino (excepto destinos locales o ISP), por tanto podemos inhibir el RIP que llega desde el ISP, mejorando el rendimiento de la línea pues el tráfico RIP en este sentido es elevado si el ISP soporta muchos túneles.

a) Mínimo esfuerzo de configuración a costa de RIP

Se consigue con túneles dinámicos (por tanto reutilizables), siendo imprescindible el uso de RIP en el sentido *cliente ® ISP* para conocer las redes accesibles del cliente.

Permite navegar:	No
Tipo de túnel:	Dinámico
Ruta por defecto:	Túnel
RIP:	Cliente ⇒ ISP
Seguridad:	Baja
Dificultad de configuración	Muy baja
Eficiencia	Alta

b) Mayor esfuerzo de configuración reduciendo el tráfico RIP

Mediante túneles dedicados para cada equipo remoto que quiera acceder, pues el cliente queda identificado y no es imprescindible RIP en el sentido *cliente ® ISP*. Por el contrario deben mantenerse los identificadores al configurar túneles en ISP y clientes.

Permite navegar:	No
Tipo de túnel:	Semidinámico
Ruta por defecto:	Túnel
RIP:	No
Seguridad:	Alta
Dificultad de configuración	Media
Eficiencia	Muy Alta

2.2. Túneles y Navegación simultánea (Esc. 1 + 2/3)

En los clientes el camino por defecto hacia cualquier destino será la red extensa (Internet), y por tanto, los destinos hacia redes accesibles por el túnel han de ser conocidos, lo que se puede conseguir mediante configuración estática en cada uno de ellos, o puede ser configurado únicamente en el ISP y que éste informe a los clientes por RIP.

a) Mayor sobrecarga en la red / Mínimo esfuerzo de configuración

En casos en los que el número de clientes no sea muy elevado, puede delegarse el mecanismo de routing en el protocolo RIP, eliminando la necesidad de configurar rutas estáticas en los clientes.

Permite navegar:	No
Tipo de túnel:	Dinámico
Ruta por defecto:	Internet
RIP:	Clientes ⇔ ISP
Seguridad:	Baja
Dificultad de configuración	Baja
Eficiencia	Baja si hay muchos túneles

Esto no excluye algún caso concreto en el que no sea conveniente utilizar RIP con algún cliente particular, pudiendo dedicarle un *key* único.

Este escenario es muy útil cuando el objetivo es la interconexión de unas pocas redes locales remotas entre si.

b) Mínima sobrecarga en la red / Mayor esfuerzo de configuración

Cuando el número de rutas conocidas por el equipo del ISP comienza a ser importante (ya sea porque el número de túneles es elevado o porque las redes locales remotas son complejas), el tráfico RIP puede llegar a afectar seriamente al rendimiento de los túneles, por lo que se debe evaluar la posibilidad de prescindir del protocolo de routing en el sentido *ISP ® Clientes*, lo que obliga a configurar estáticamente en los clientes las redes a las que será posible llegar a través del túnel.

Permite navegar:	Si
Tipo de túnel:	Dinámico
Ruta por defecto:	Internet
RIP:	Clientes ⇒ ISP
Seguridad:	Media
Dificultad de configuración	Media
Eficiencia	Alta

Al igual que en los casos anteriores, cabe la posibilidad de eliminar RIP en el sentido *Cliente ® ISP*, dedicándole un *key* único.

Este escenario es muy útil cuando el objetivo es el acceso a las redes locales remotas desde el ISP, y no la interconexión de redes locales remotas entre si, por ejemplo, cuando una entidad posea ISP propio y desee acceder a las delegaciones desde el mismo.

c) Sobrecarga nula en la red / Mayor esfuerzo de configuración/Control de clientes

Este escenario se basa totalmente en la utilización de *keys* distintos, de forma que cada interfaz túnel queda perfectamente definido, es decir, se conoce el cliente que se conectará a él, y las redes alcanzables a través del mismo.

Permite navegar:	Si
Tipo de túnel:	Semidinámico
Ruta por defecto:	Internet
RIP:	No
Seguridad:	Alta
Dificultad de configuración	Alta
Eficiencia	Máxima

Esto reduce a cero la sobrecarga, pues se elimina totalmente el tráfico RIP a costa de configurar explícitamente todas las rutas alcanzables.

Debido al mayor control de los clientes en este escenario, se hace apropiado cuando el ISP ofrece servicios de interconexión de redes a terceros.

IMPORTANTE: Cuando se ofrece el servicio a terceros se debe evitar duplicidad de direcciones entre las instalaciones de los clientes (no puede haber la misma subred en clientes distintos).

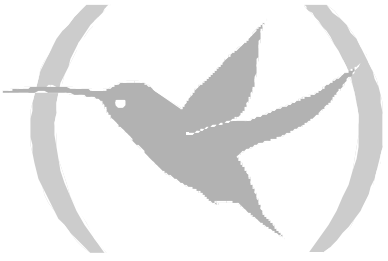
3. Seguridad

Cuando se trata de conectar redes locales a través de una red pública, los aspectos relativos a la seguridad cobran gran importancia. Deben existir mecanismos para autenticar las conexiones, y mecanismos para evitar el tráfico inter-túnel no deseado.

El primer mecanismo de autenticación sería la utilización de direcciones fijas, aunque este es un recursos costoso en Internet. Por tanto cuando no pueda ser utilizado se debe recurrir a otro mecanismo, como pueden ser el identificador del túnel GRE (*key*), que ha de ser conocido por ambos extremos. Además es posible cifrar el contenido del paquete GRE.

Debido a que el tráfico inter-túnel es realizado por el equipo de ISP, se tiene total control sobre el mismo.

Capítulo 4
Monitorización del Interfaz túnel IP (TNIP)



1. Monitorización del interfaz Túnel IP (TNIP)

Para acceder al menú de monitorización del interfaz PPP hay que introducir el comando ***"NETWORK <interfaz TNIP>"*** desde el menú general de monitorización:

```
+NETWORK tnip1
TNIP protocol monitor
TNIP>
```

El comando disponible es el siguiente:

LIST	Muestra la información de monitorización.

a) LIST

Visualiza la información relativa a los túneles.

· *LIST STATE*

Muestra el estado de conexión del túnel (sólo aplica a túneles dinámicos).

Ejemplo:

```
TNIP>LIST STATE

Int Source IP      Dest. IP        Start Time  Conections  Descon.err  Loop
--- ---------      ----------      ----------  ----------  ----------  ----
 7 200.200.200.1   200.200.200.200 00:29                1           0  0

TNIP>
```

Start Time: hora de inicio de la última conexión.
Connections: número de conexiones desde el último arranque del router.
Discon.err: numero de desconexiones por no recibir rutas a través de dicho túnel.
Loop: informa si ha habido un bucle.

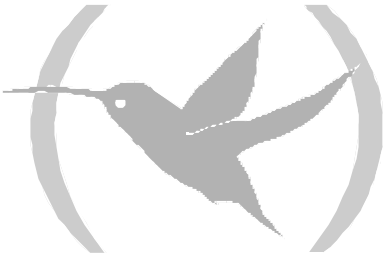
2. Estadísticos del interfaz Túnel IP (TNIP)

Al ejecutar el comando ***"DEVICE <interfaz TNIP>"*** desde el prompt del proceso de monitorización general (+) se muestran todos los estadísticos del interfaz TNIP correspondiente:

```
+DEVICE TNIP1

                                    Auto-test   Auto-test     Maintenance
Interface            CSR    Vect     valids    failures        failures
tnip1                  0       0          2           0               0
Imput Stats
-----------
  Frames ok    12980
  Frames error 0
  ---> Invalid encapsulation    0
  ---> Out-of-Order frames      0
  ---> Checsksum errors         0
  ---> Key errors               0
  ---> Unknown payload protocol 0
  ---> Error in cipher          0
  ---> Internal errors          0
Output Stats
------------
  Frames ok    11545
  Frames error 0
  ---> Invalid encapsulation    0
  ---> Unknown payload protocol 0
+
```

Capítulo 5
Ejemplos de configuración de Túnel IP



1. Túnel IP sobre IP

1.1. Pasos a seguir en cada extremo del túnel

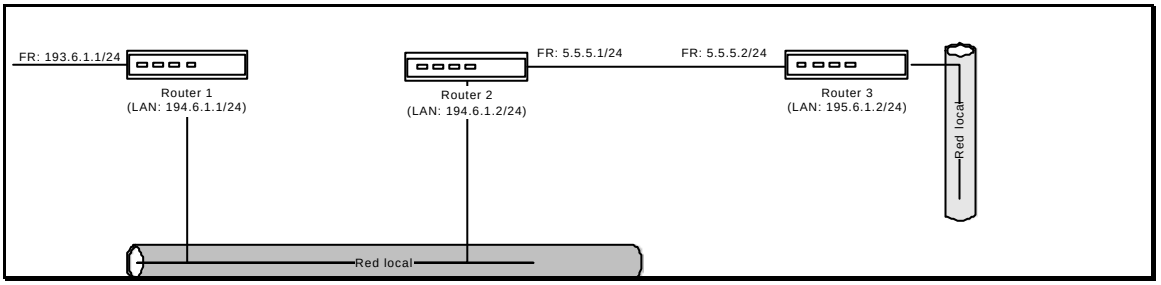
- Crear el interfaz túnel IP.
- Asignar una dirección IP al interfaz túnel o configurarlo como no numerado.
- Configurar el origen del túnel.
- Configurar el destino del túnel. Agregar la ruta IP necesaria para llegar a dicho destino.
- Configurar el protocolo de encapsulado que irá en el túnel (o tipo de túnel).
- Habilitar las opciones deseadas.
- Agregar las rutas IP de aquellas redes que tengan que ser accesibles a través del túnel IP poniendo como siguiente salto el propio interfaz túnel IP.
- Habilitar el túnel, guardar y reiniciar.

1.2. Pasos a seguir en los equipos que atraviesa el túnel

- Agregar las rutas necesarias para que origen y destino del túnel sean accesibles.

1.3. Ejemplo 1.a: IP sobre IP con GRE

Configuración de un túnel con origen Router1 y destino Router3, en el que se puedan comunicar las redes 193.6.1.0/24 y 195.6.1.0/24.



a) Configuración Router1

Se agrega el interfaz Frame Relay y el túnel IP

```
*P 4
Config>SET HOSTNAME Router1
Router1 Config>SET DATA-LINK FRAME-RELAY serial0/0
Router1 Config>ADD DEVICE tnip 1
Added TNIP interface tnip1
Router1 Config>
```

Se configuran las direcciones de los interfaces

```
Router1 Config>PROTOCOL IP

-- Internet protocol user configuration --
Router1 IP config>ADDRESS ethernet0/0 194.6.1.1 255.255.255.0
Router1 IP config>ADDRESS serial0/0 193.6.1.1 255.255.255.0
Router1 IP config>ADDRESS tnip1 unnumbered
Router1 IP config>LIST ADDRESSES
IP addresses for each interface:
 ethernet0/0     194.6.1.1       255.255.255.0   NETWORK broadcast,  fill 0
 serial0/0       193.6.1.1       255.255.255.0   NETWORK broadcast,  fill 0
 serial0/1                                       IP disabled on this ifc
 serial0/2                                       IP disabled on this ifc
 bri0/0                                          IP disabled on this ifc
 x25-node                                        IP disabled on this ifc
 tnip1           unnumbered      0.0.0.0         NETWORK broadcast,  fill 0
Router1 IP config>EXIT
Router1 Config>
```

A continuación se configura el túnel IP

```
Router1 Config>NETWORK tnip1

-- IP Tunnel Net Configuration --
Router1 TNIP config>SOURCE 194.6.1.1
Router1 TNIP config>DESTINATION 5.5.5.2
Router1 TNIP config>ENABLE
Router1 TNIP config>LIST
Tunnel mode: GRE (enabled)
Tunnel source 194.6.1.1, destination 5.5.5.2
QoS preclassify: disabled
Router1 TNIP config>ENCAPSULATION

-- GRE Configuration --
Router1 GRE config>CHECKSUM
Router1 GRE config>KEY 1234
Router1 GRE config>LIST
RC4 Cipher.................: disabled
End-to-End Checksumming....: enabled
Tunnel identification key..: enabled [1234]
Drop Out-of-Order Datagrams: disabled
Router1 GRE config>EXIT
Router1 TNIP config>EXIT
Router1 Config>
```

Se agregan las rutas necesarias

```
Router1 Config>PROTOCOL IP

-- Internet protocol user configuration --
Router1 IP config>ROUTE 5.5.5.2 255.255.255.255 194.6.1.2 1
Router1 IP config>ROUTE 195.6.1.0 255.255.255.0 tnip1 1
Router1 IP config>EXIT
Router1 Config>
```

Una vez realizados todos estos pasos de configuración sólo resta salvar la configuración y reiniciar el equipo.

b) Configuración Router2

Agregamos el interfaz Frame Relay

```
*P 4
Config>SET HOSTNAME Router2
Router2 Config>SET DATA-LINK FRAME-RELAY serial0/0
Router2 Config>
```

Se configura el interfaz Frame Relay

```
Router2 Config>NETWORK serial0/0

-- Frame Relay user configuration --
Router2 FR config>NO LMI
Router2 FR config>PVC 16 default
Router2 FR config>PVC 16 cir 64000
Router2 FR config>PVC 16 bc 16000
Router2 FR config>PROTOCOL-ADDRESS 5.5.5.2 16
Router2 FR config>EXIT
Router2 Config>
```

Se configuran las direcciones de los interfaces

```
Router2 Config>PROTOCOL IP

-- Internet protocol user configuration --
Router2 IP config>ADDRESS ethernet0/0 194.6.1.2 255.255.255.0
Router2 IP config>ADDRESS serial0/0 5.5.5.1 255.255.255.0
Router2 IP config>LIST ADDRESSES
IP addresses for each interface:
 ethernet0/0     194.6.1.2       255.255.255.0   NETWORK broadcast,  fill 0
 serial0/0       5.5.5.1         255.255.255.0   NETWORK broadcast,  fill 0
 serial0/1                                       IP disabled on this ifc
 serial0/2                                       IP disabled on this ifc
 bri0/0                                          IP disabled on this ifc
 x25-node                                        IP disabled on this ifc
Router2 IP config>EXIT
Router2 Config>
```

Una vez realizados todos estos pasos de configuración sólo resta salvar la configuración y reiniciar el equipo.

c) Configuración Router3

Se agrega el interfaz Frame Relay y el túnel IP

```
*P 4
Config>SET HOSTNAME Router3
Router1 Config>SET DATA-LINK FRAME-RELAY serial0/0
Router1 Config>ADD DEVICE tnip 1
Added TNIP interface tnip1
Router1 Config>
```

Se configuran las direcciones de los interfaces

```
Router3 Config>PROTOCOL IP

-- Internet protocol user configuration --
Router3 IP config>ADDRESS ethernet0/0 195.6.1.1 255.255.255.0
Router3 IP config>ADDRESS serial0/0 5.5.5.2 255.255.255.0
Router3 IP config>ADDRESS tnip1 unnumbered
Router3 IP config>LIST ADDRESSES
IP addresses for each interface:
 ethernet0/0     195.6.1.1       255.255.255.0   NETWORK broadcast,  fill 0
 serial0/0       5.5.5.2         255.255.255.0   NETWORK broadcast,  fill 0
 serial0/1                                       IP disabled on this ifc
 serial0/2                                       IP disabled on this ifc
 bri0/0                                          IP disabled on this ifc
 x25-node                                        IP disabled on this ifc
 tnip1           unnumbered      0.0.0.0         NETWORK broadcast,  fill 0
Router3 IP config>EXIT
Router3 Config>
```

Se configura el interfaz Frame Relay

```
Router3 Config>NETWORK serial0/0

-- Frame Relay user configuration --
Router3 FR config>NO LMI
Router3 FR config>PVC 16 default
Router3 FR config>PVC 16 cir 64000
Router3 FR config>PVC 16 bc 16000
Router3 FR config>PROTOCOL-ADDRESS 5.5.5.1 16
Router3 FR config>EXIT
Router3 Config>
```

A continuación se configura el túnel

```
Router3 Config>NETWORK tnip1

-- IP Tunnel Net Configuration --
Router3 TNIP config>ENABLE
Router3 TNIP config>DESTINATION 194.6.1.1
Router3 TNIP config>SOURCE 5.5.5.2
Router3 TNIP config>ENCAPSULATION

-- GRE Configuration --
Router3 GRE config>CHECKSUM
Router3 GRE config>KEY 1234
Router3 GRE config>EXIT
Router3 TNIP config>EXIT
Router3 Config>
```

Se agregan las rutas necesarias

```
Router3 Config>PROTOCOL IP

-- Internet protocol user configuration --
Router3 IP config>ROUTE 194.6.1.1 255.255.255.255 5.5.5.1 1
Router3 IP config>ROUTE 193.6.1.0 255.255.255.0 tnip1 1
Router3 IP config>EXIT
Router3 Config>
```

Sólo resta salvar la configuración y reiniciar el equipo.

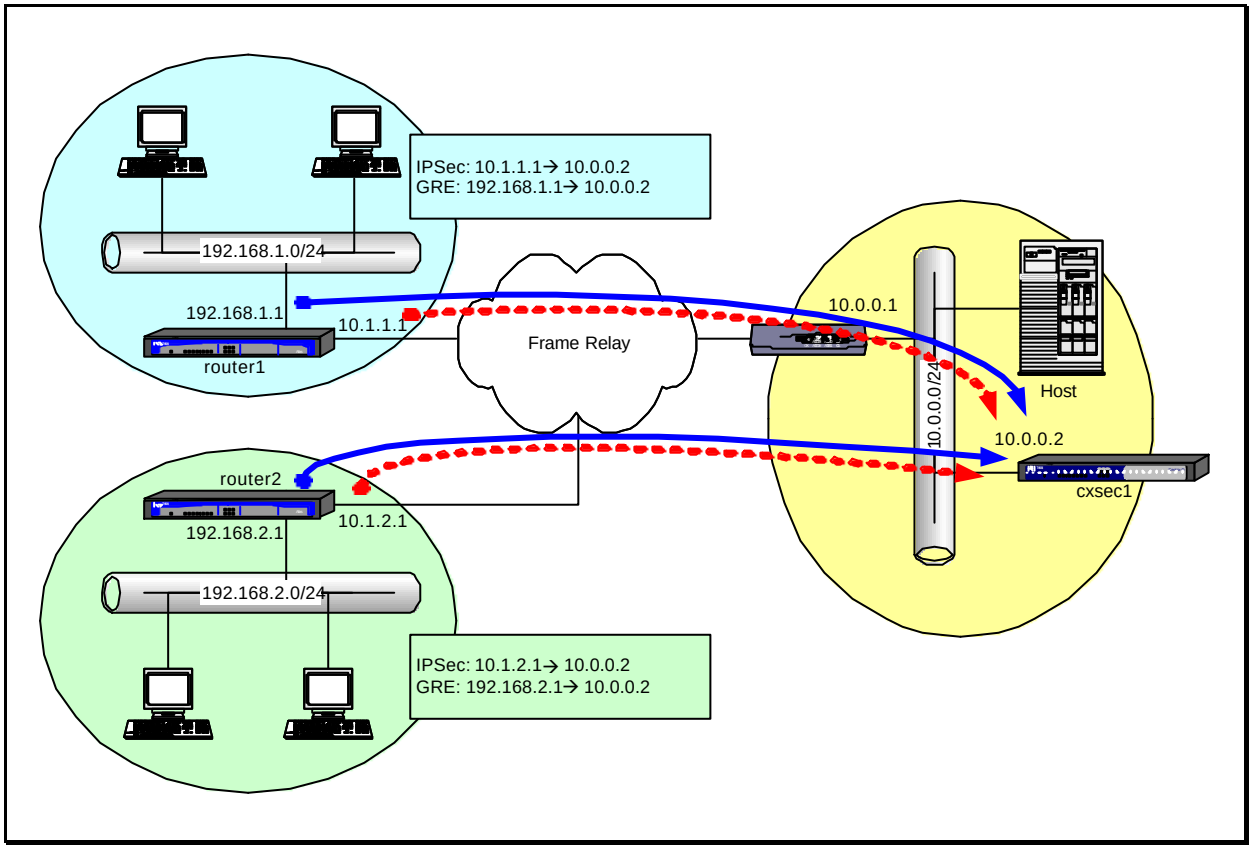
1.4. Ejemplo 1.b: túnel promiscuo

Configuración de un túnel con origen ***ROUTER1*** y destino ***CXSEC1***, para enviar RIP de una delegación al Centrix-Sec.

Este escenario se da cuando queremos transmitir tráfico cifrado con IPSec y se plantea la problemática de enviar RIP por el túnel IPSec. Una solución consiste en configurar un túnel GRE donde se habilita el envío de RIP, y el tráfico encapsulado se envía por el túnel IPSec para que el equipo del otro extremo (posiblemente un Centrix-Sec) reciba tanto los datos cifrados como las redes accesibles a través de dicho túnel.

El problema surge cuando el equipo que recibe la información de RIP da servicio a muchos túneles IPSec, en cuyo caso necesitaría un interfaz TNIP para cada uno. Para evitar esto se puede configurar un solo interfaz TNIP en modo promiscuo, de modo que el mismo interfaz reciba el tráfico RIP encapsulado de todos los túneles.

El esquema de este ejemplo es el siguiente:



Escenario del ejemplo de aplicación de túneles promiscuos.

a) Configuración CXSEC1

Se agrega el interfaz túnel IP:

```
*PROCESS 4
Config>SET HOSTNAME cxsec1
cxsec1 Config>ADD DEVICE tnip 1
Added TNIP interface tnip1
cxsec1 Config>
```

Se configuran las direcciones de los interfaces:

```
cxsec1 Config>PROTOCOL IP

-- Internet protocol user configuration --
cxsec1 IP config>ADDRESS ethernet0/0 10.0.0.2 255.255.255.0
cxsec1 IP config>ADDRESS tnip1 unnumbered 0.0.0.0
cxsec1 IP config>LIST ADDRESSES
IP addresses for each interface:
 ethernet0/0     10.0.0.2        255.255.255.0   NETWORK broadcast,  fill 0
 serial0/0                                       IP disabled on this ifc
 serial0/1                                       IP disabled on this ifc
 serial0/2                                       IP disabled on this ifc
 bri0/0                                          IP disabled on this ifc
 x25-node                                        IP disabled on this ifc
 tnip1           unnumbered      0.0.0.0         NETWORK broadcast,  fill 0
cxsec1 IP config>EXIT
cxsec1 Config>
```

Como se trata de un túnel promiscuo no se configura su dirección de origen ni de destino. En este ejemplo no se emplea ninguna funcionalidad adicional, así que basta con habilitar el interfaz:

```
cxsec1 Config>NETWORK tnip1

-- IP Tunnel Net Configuration --
cxsec1 TNIP config>ENABLE
cxsec1 TNIP config>LIST
Tunnel mode: GRE (enabled)
Tunnel source 0.0.0.0, destination 0.0.0.0
QoS preclassify: disabled
cxsec1 TNIP config>ENCAPSULATION

-- GRE Configuration --
cxsec1 GRE config>LIST
RC4 Cipher.................: disabled
End-to-End Checksumming....: disabled
Tunnel identification key..: disabled
Drop Out-of-Order Datagrams: disabled
cxsec1 GRE config>EXIT
cxsec1 TNIP config>EXIT
cxsec1 Config>
```

Se agregan las rutas necesarias:

```
cxsec1 Config>PROTOCOL IP

-- Internet protocol user configuration --
cxsec1 IP config>ROUTE 0.0.0.0 0.0.0.0 10.0.0.1 1
cxsec1 IP config>LIST ROUTES

route to 0.0.0.0,0.0.0.0 via 10.0.0.1, cost 1
cxsec1 IP config>EXIT
cxsec1 Config>
```

Y se configura el protocolo RIP para que reciba información de rutas por el interfaz TNIP y la envíe a la red local:

```
cxsec1 Config>PROTOCOL RIP

-- RIP protocol user configuration --
cxsec1 RIP config>ENABLE
cxsec1 RIP config>COMPATIBILITY tnip1 send none
cxsec1 RIP config>COMPATIBILITY 10.0.0.2 receive none
cxsec1 RIP config>COMPATIBILITY 10.0.0.2 send rip2-multicast
cxsec1 RIP config>AUTHENTICATION 10.0.0.2 router
cxsec1 RIP config>SENDING 10.0.0.2 no direct-routes
cxsec1 RIP config>EXIT
cxsec1 Config>
```

Finalmente se configura el cifrado IPSec:

```
cxsec1 Config>FEATURE ACCESS-LISTS

-- Access Lists user configuration --
cxsec1 Access Lists config>ACCESS-LIST 100


cxsec1 Extended Access List 100>ENTRY 1 default
cxsec1 Extended Access List 100>ENTRY 1 permit
cxsec1 Extended Access List 100>ENTRY 1 source address 10.0.0.0 255.255.255.0
cxsec1 Extended Access List 100>ENTRY 1 destination address 192.168.0.0
255.255.0.0
cxsec1 Extended Access List 100>EXIT
cxsec1 Access Lists config>EXIT
cxsec1 Config>PROTOCOL IP

-- Internet protocol user configuration --
cxsec1 IP config>IPSEC

-- IPSec user configuration --
cxsec1 IPSec config>ENABLE
cxsec1 IPSec config>ASSIGN-ACCESS-LIST 100
cxsec1 IPSec config>TEMPLATE 1 default
```

```
cxsec1 IPSec config>TEMPLATE 1 isakmp des md5
cxsec1 IPSec config>TEMPLATE 1 ike mode aggressive
cxsec1 IPSec config>TEMPLATE 2 default
cxsec1 IPSec config>TEMPLATE 2 dynamic esp des md5
cxsec1 IPSec config>TEMPLATE 2 source-address 10.0.0.2
cxsec1 IPSec config>MAP-TEMPLATE 100 2
cxsec1 IPSec config>KEY preshared hostname router* plain 0x112233445566

cxsec1 IPSec config>EXIT
cxsec1 IP config>EXIT
cxsec1 Config>
```

Una vez realizados todos estos pasos de configuración sólo resta salvar la configuración y reiniciar el equipo.

b) Configuración ROUTER1

Se agrega el interfaz Frame Relay y el interfaz túnel IP:

```
*PROCESS 4
Config>SET HOSTNAME router1
router1 Config>SET DATA-LINK FRAME-RELAY serial0/0
router1 Config>ADD DEVICE tnip 1
Added TNIP interface tnip1
router1 Config>
```

Se configuran las direcciones de los interfaces y la dirección interna:

```
router1 Config>PROTOCOL IP

-- Internet protocol user configuration --
router1 IP config>ADDRESS ethernet0/0 192.168.1.1 255.255.255.0
router1 IP config>ADDRESS serial0/0 10.1.1.1 255.255.255.252
router1 IP config>ADDRESS tnip1 unnumbered
router1 IP config>INTERNAL-IP-ADDRESS 192.168.1.1
router1 IP config>LIST ADDRESSES
IP addresses for each interface:
 ethernet0/0     192.168.1.1     255.255.255.0   NETWORK broadcast,  fill 0
 serial0/0       10.1.1.1        255.255.255.252 NETWORK broadcast,  fill 0
 serial0/1                                       IP disabled on this ifc
 serial0/2                                       IP disabled on this ifc
 bri0/0                                          IP disabled on this ifc
 x25-node                                        IP disabled on this ifc
 tnip1           unnumbered      0.0.0.0         NETWORK broadcast,  fill 0
Internal IP address: 192.168.1.1
router1 IP config>EXIT
router1 Config>
```

Se configuran las direcciones de origen y destino del túnel y se habilita el interfaz:

```
router1 Config>NETWORK tnip1

-- IP Tunnel Net Configuration --
router1 TNIP config>ENABLE
router1 TNIP config>DESTINATION 10.0.0.2
router1 TNIP config>SOURCE 192.168.1.1
router1 TNIP config>LIST
Tunnel mode: GRE (enabled)
Tunnel source 192.168.1.1, destination 10.0.0.2
QoS preclassify: disabled
router1 TNIP config>EXIT
router1 Config>
```

Se agregan las rutas necesarias:

```
router1 Config>PROTOCOL IP

-- Internet protocol user configuration --
router1 IP config>ROUTE 10.0.0.0 255.255.255.0 10.1.1.2 1
router1 IP config>LIST ROUTES

route to 10.0.0.0,255.255.255.0 via 10.1.1.2, cost 1
router1 IP config>EXIT
router1 Config>
```

Y se configura el protocolo RIP para que envíe por el túnel información de la red local, que es la accesible por IPSec:

```
router1 Config>FEATURE ACCESS-LISTS

-- Access Lists user configuration --
router1 Access Lists config>ACCESS-LIST 1


router1 Standard Access List 1>ENTRY 1 default
router1 Standard Access List 1>ENTRY 1 permit
router1 Standard Access List 1>ENTRY 1 source address 192.168.1.0
255.255.255.0
router1 Standard Access List 1>EXIT
router1 Access Lists config>EXIT
router1 Config>PROTOCOL RIP

-- RIP protocol user configuration --
router1 RIP config>ENABLE
router1 RIP config>COMPATIBILITY 192.168.1.1 send none
router1 RIP config>COMPATIBILITY 192.168.1.1 receive none
router1 RIP config>COMPATIBILITY 10.1.1.1 send none
router1 RIP config>COMPATIBILITY 10.1.1.1 receive none
router1 RIP config>COMPATIBILITY tnip1 receive none
router1 RIP config>SENDING tnip1 distribute-list 1
router1 RIP config>EXIT
router1 Config>
```

Finalmente se configura el interfaz Frame Relay y el cifrado IPSec:

```
router1 Config>NETWORK serial0/0

-- Frame Relay user configuration --
router1 FR config>PVC 21 default
router1 FR config>PROTOCOL-ADDRESS 10.1.1.2 21
router1 FR config>EXIT
router1 Config>FEATURE ACCESS-LISTS

-- Access Lists user configuration --
router1 Access Lists config>ACCESS-LIST 100


router1 Extended Access List 100>ENTRY 1 default
router1 Extended Access List 100>ENTRY 1 permit
router1 Extended Access List 100>ENTRY 1 source address 192.168.1.0
255.255.255.0
router1 Extended Access List 100>ENTRY 1 destination address 10.0.0.0
255.255.255.0
router1 Extended Access List 100>EXIT
router1 Access Lists config>EXIT
router1 Config>PROTOCOL IP

-- Internet protocol user configuration --
router1 IP config>IPSEC

-- IPSec user configuration --
router1 IPSec config>ENABLE
router1 IPSec config>ASSIGN-ACCESS-LIST 100
router1 IPSec config>TEMPLATE 1 default
router1 IPSec config>TEMPLATE 1 isakmp des md5
router1 IPSec config>TEMPLATE 1 destination-address 10.0.0.2
router1 IPSec config>TEMPLATE 1 ike mode aggressive
router1 IPSec config>TEMPLATE 1 ike idtype fqdn
```

```
router1 IPSec config>TEMPLATE 1 keepalive dpd
router1 IPSec config>TEMPLATE 2 default
router1 IPSec config>TEMPLATE 2 dynamic esp des md5
router1 IPSec config>TEMPLATE 2 source-address 10.1.1.1
router1 IPSec config>TEMPLATE 2 destination-address 10.0.0.2
router1 IPSec config>MAP-TEMPLATE 100 2
router1 IPSec config>KEY preshared ip 10.0.0.2 plain 0x112233445566

router1 IPSec config>EXIT
router1 IP config>EXIT
router1 Config>
```

Una vez realizados todos estos pasos de configuración sólo resta salvar la configuración y reiniciar el equipo.

c) Configuración ROUTER2

La configuración del equipo router2 es análoga a la del equipo router1. La diferencia principal son las direcciones IP locales:

```
*PROCESS 4
Config>ADD DEVICE tnip 1
Added TNIP interface tnip1
Config>SET DATA-LINK FRAME-RELAY serial0/0
Config>SET HOSTNAME router2
router2 Config>NETWORK serial0/0

-- Frame Relay user configuration --
router2 FR config>PVC 22 default
router2 FR config>PROTOCOL-ADDRESS 10.1.2.2 22
router2 FR config>EXIT
router2 Config>NETWORK tnip1

-- IP Tunnel Net Configuration --
router2 TNIP config>ENABLE
router2 TNIP config>DESTINATION 10.0.0.2
router2 TNIP config>SOURCE 192.168.2.1
router2 TNIP config>EXIT
router2 Config>FEATURE ACCESS-LISTS

-- Access Lists user configuration --
router2 Access Lists config>ACCESS-LIST 1


router2 Standard Access List 1>ENTRY 1 default
router2 Standard Access List 1>ENTRY 1 permit
router2 Standard Access List 1>ENTRY 1 source address 192.168.2.0
255.255.255.0
router2 Standard Access List 1>EXIT
router2 Access Lists config>ACCESS-LIST 100


router2 Extended Access List 100>ENTRY 1 default
router2 Extended Access List 100>ENTRY 1 permit
router2 Extended Access List 100>ENTRY 1 source address 192.168.2.0
255.255.255.0
router2 Extended Access List 100>ENTRY 1 destination address 10.0.0.0
255.255.255.0
router2 Extended Access List 100>EXIT
router2 Access Lists config>EXIT
```

```
router2 Config>PROTOCOL IP

-- Internet protocol user configuration --
router2 IP config>INTERNAL-IP-ADDRESS 192.168.2.1
router2 IP config>ADDRESS ethernet0/0 192.168.2.1 255.255.255.0
router2 IP config>ADDRESS serial0/0 10.1.2.1 255.255.255.252
router2 IP config>ADDRESS tnip1 unnumbered 0.0.0.0
router2 IP config>ROUTE 10.0.0.0 255.255.255.0 10.1.2.2 1
router2 IP config>IPSEC

-- IPSec user configuration --
router2 IPSec config>ENABLE
router2 IPSec config>ASSIGN-ACCESS-LIST 100
```

```
router2 IPSec config>TEMPLATE 1 default
router2 IPSec config>TEMPLATE 1 isakmp des md5
router2 IPSec config>TEMPLATE 1 destination-address 10.0.0.2
router2 IPSec config>TEMPLATE 1 ike mode aggressive
router2 IPSec config>TEMPLATE 1 ike idtype fqdn
router2 IPSec config>TEMPLATE 1 keepalive dpd
router2 IPSec config>TEMPLATE 2 default
router2 IPSec config>TEMPLATE 2 dynamic esp des md5
router2 IPSec config>TEMPLATE 2 source-address 10.1.2.1
router2 IPSec config>TEMPLATE 2 destination-address 10.0.0.2
router2 IPSec config>MAP-TEMPLATE 100 2
router2 IPSec config>KEY preshared ip 10.0.0.2 plain 0x112233445566

router2 IPSec config>EXIT
router2 IP config>EXIT
router2 Config>PROTOCOL RIP

-- RIP protocol user configuration --
router2 RIP config>ENABLE
router2 RIP config>COMPATIBILITY 192.168.2.1 send none
router2 RIP config>COMPATIBILITY 192.168.2.1 receive none
router2 RIP config>COMPATIBILITY 10.1.2.1 send none
router2 RIP config>COMPATIBILITY 10.1.2.1 receive none
router2 RIP config>COMPATIBILITY tnip1 receive none
router2 RIP config>SENDING tnip1 distribute-list 1
router2 RIP config>EXIT
router2 Config>
```

Sólo resta salvar la configuración y reiniciar el equipo.

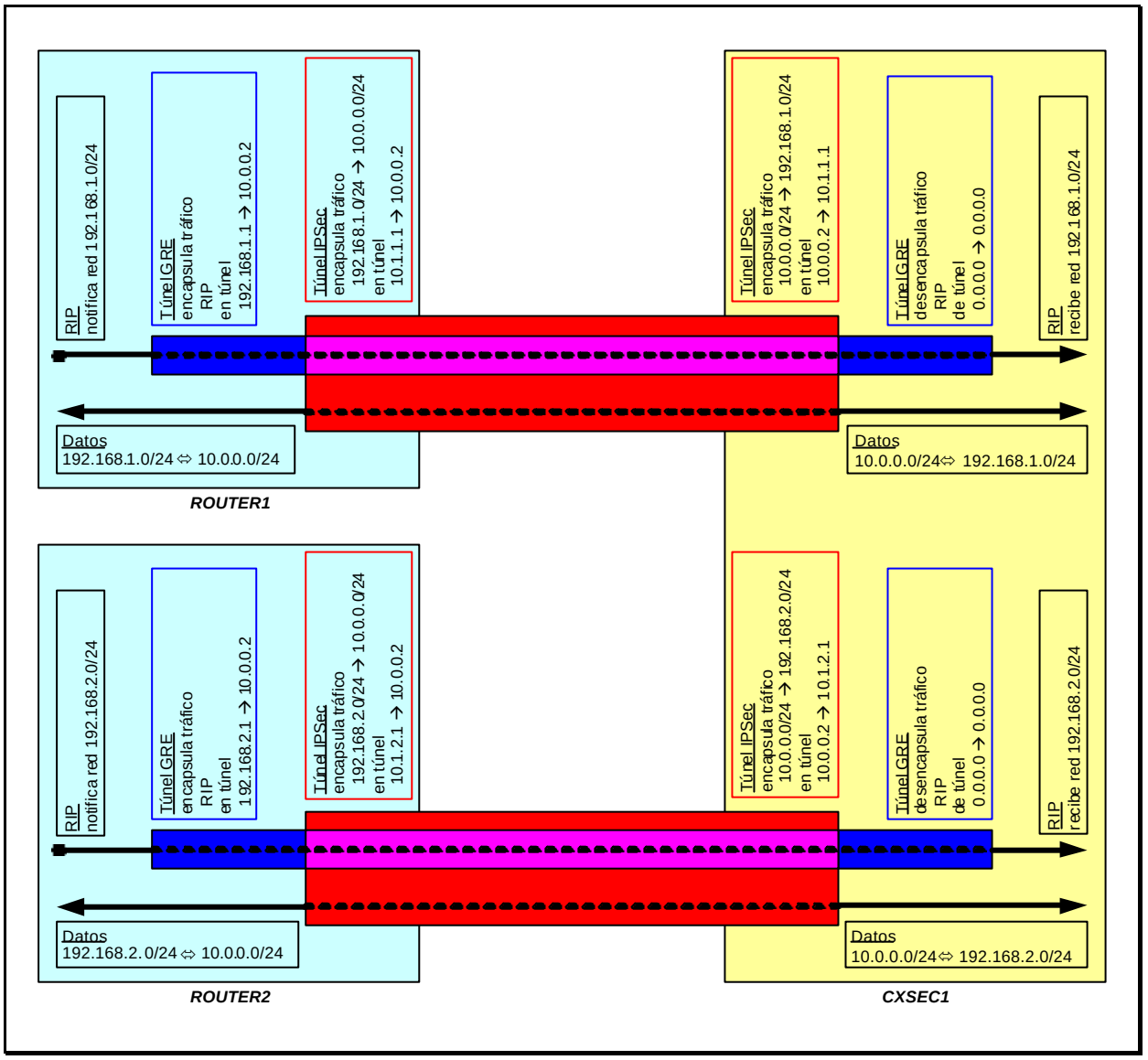
d) Resultado final

El resultado es que tanto ***ROUTER1*** como ***ROUTER2*** envían información de su red local (192.168.1.0/24 y 192.168.2.0/24 respectivamente) mediante RIP, encapsulado en GRE y cifrado con IPSec.

CXSEC1 recibe toda la información de RIP por un mismo interfaz túnel (en modo promiscuo) y la notifica a su red local 10.0.0.0/24.

Así, todos los equipos de la red 10.0.0.0/24 saben que para enviar tráfico a las redes de ***ROUTER1*** y de ***ROUTER2*** deben ir por ***CXSEC1***, que a su vez se encargará de enviarlo cifrado por el túnel IPSec correspondiente.

El siguiente esquema representa el flujo de datos e información RIP entre los equipos ***ROUTER1***, ***ROUTER2*** y ***CXSEC1*** en sus distintos encapsulados:



Encapsulado de los distintos tráficos del escenario.

Como se ve en el esquema, el tráfico RIP del interfaz TNIP de ***ROUTER1*** (flecha negra de arriba) es sólo en un sentido, de ***ROUTER1*** a ***CXSEC1***. Este tráfico se encapsula en el túnel GRE (azul en el esquema), que como tiene origen 192.168.2.1 y destino 10.0.0.2, a su vez se encapsulará en el túnel IPSec (rojo en el esquema).

El equipo ***CXSEC1*** recibe el tráfico por el túnel IPSec (rojo), lo descifra y desencapsula obteniendo tráfico GRE (azul). Como este tráfico está destinado a ***CXSEC1*** (dirección 10.0.0.2) lo procesa en el interfaz TNIP obteniendo así la información de RIP que envió ***ROUTER1***.

El resto de tráfico entre las redes 192.168.1.0/24 y 10.0.0.0/24 se encapsula directamente en el túnel IPSec en ambos sentidos, tal como se ve en el esquema (flecha de abajo).

Gracias al modo promiscuo del interfaz TNIP el equipo ***CXSEC1*** es capaz de recibir la información de RIP de todas las delegaciones por un mismo interfaz, de modo que no nos vemos limitados por el máximo de 15 interfaces TNIP, y así podemos dar servicio a un número elevado de delegaciones.

2. Túnel IP sobre SRT

2.1. Pasos a seguir en cada extremo del túnel

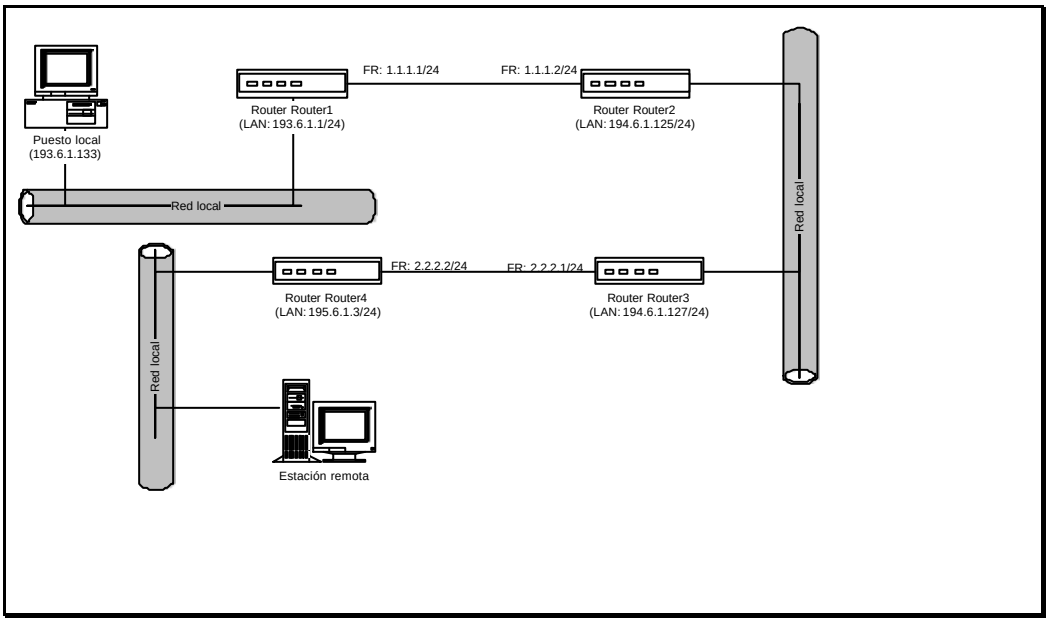
- Crear el interfaz túnel IP.
- Habilitar el bridge.
- Agregar un puerto en el bridge para el interfaz del túnel.
- Configurar el origen del túnel.
- Configurar el destino del túnel. Agregar la ruta IP necesaria para llegar a dicho destino.
- Configurar el protocolo de encapsulado que irá en el túnel (o tipo de túnel).
- Habilitar las opciones deseadas.
- Habilitar el túnel, guardar y reiniciar.

2.2. Pasos a seguir en los equipos que atraviesa el túnel

- Agregar las rutas necesarias para que origen y destino del túnel sean accesibles.

2.3. Ejemplo 2: IP sobre SRT con GRE

Configuración de un túnel con origen Router1 y destino Router4, en el que se puedan comunicar las redes 193.6.1.0/24 y 195.6.1.0/24 mediante tráfico NetBEUI. Para ello se establecerá un túnel IP sobre SRT entre ambas.



a) Configuración Router1

Al igual que en el ejemplo anterior, se añade el interfaz FR y el interfaz túnel IP (TNIP) y se configuran las direcciones IP de los interfaces.

```
*P 4
Config>SET HOSTNAME Router1
Router1 Config>SET DATA-LINK FRAME-RELAY serial0/0
Router1 Config>ADD DEVICE tnip 1
Added TNIP interface tnip1
Router1 Config>PROTOCOL IP

-- Internet protocol user configuration --
Router1 IP config>ADDRESS ethernet0/0 193.6.1.133 255.255.255.0
Router1 IP config>ADDRESS serial0/0 1.1.1.1 255.255.255.0
Router1 IP config>ADDRESS tnip1 unnumbered
Router1 IP config>EXIT
Router1 Config>
```

A continuación se configura el túnel

```
Router1 Config>NETWORK tnip1

-- IP Tunnel Net Configuration --
Router1 TNIP config>ENABLE
Router1 TNIP config>DESTINATION 2.2.2.2
Router1 TNIP config>SOURCE 1.1.1.1
Router1 TNIP config>LIST
Tunnel mode: GRE (enabled)
Tunnel source 1.1.1.1, destination 2.2.2.2
QoS preclassify: disabled
Router1 TNIP config>ENCAPSULATION

-- GRE Configuration --
Router1 GRE config>CHECKSUM
Router1 GRE config>KEY 1234
Router1 GRE config>SEQUENCE-DATAGRAM
Router1 GRE config>LIST
RC4 Cipher.................: disabled
End-to-End Checksumming....: enabled
Tunnel identification key..: enabled [1234]
Drop Out-of-Order Datagrams: enabled
Router1 GRE config>EXIT
Router1 TNIP config>EXIT
Router1 Config>
```

Se configura el interfaz Frame Relay

```
Router1 Config>NETWORK serial0/0

-- Frame Relay user configuration --
Router1 FR config>PVC 16 default
Router1 FR config>PVC 16 CIR 64000
Router1 FR config>PROTOCOL-ADDRESS 1.1.1.2 16
Router1 FR config>NO LMI
Router1 FR config>EXIT
Router1 Config>
```

Se incluyen las rutas necesarias

```
Router1 Config>PROTOCOL IP

-- Internet protocol user configuration --
Router1 IP config>ROUTE 2.2.2.0 255.255.255.0 1.1.1.2 1
Router1 IP config>EXIT
Router1 Config>
```

Por último se configura el bridge.

```
Router1 Config>PROTOCOL ASRT

-- ASRT Bridge user configuration --
Router1 ASRT config>BRIDGE
Router1 ASRT config>PORT ethernet0/0 1
Router1 ASRT config>PORT tnip1 2
Router1 ASRT config>LIST BRIDGE

                Source Routing Transparent Bridge Configuration
              ====================================================

Bridge:    Enabled                              Bridge behavior: STB
                   +----------------------------------------+
-------------------|       SOURCE ROUTING INFORMATION       |---------------
--
                   +----------------------------------------+
Bridge Number:          00                      Segments:          0
Max ARE Hop Cnt:        00                      Max STE Hop cnt:  00
1:N SRB:                Not Active              Internal Segment:  0x000
LF-bit interpret:       Extended
                   +----------------------------------------+
-------------------|            SR-TB INFORMATION           |---------------
--
                   +----------------------------------------+
SR-TB Conversion:       Disabled
TB-Virtual Segment:     0x000                   MTU of TB-Domain:  0
                   +----------------------------------------+
-------------------|    SPANNING TREE PROTOCOL INFORMATION  |---------------
---
                   +----------------------------------------+
Bridge Address:         Default                 Bridge Priority:
32768/0x8000
STP Participation:      Disabled
                   +----------------------------------------+
-------------------|        TRANSLATION INFORMATION         |---------------
---
                   +----------------------------------------+
FA<=>GA Conversion:     Enabled                 UB-Encapsulation:  Disabled
DLS for the bridge:     Disabled
                   +----------------------------------------+
-------------------|            PORT INFORMATION            |---------------
---
                   +----------------------------------------+
Number of ports added: 2
Port:   1    Interface:     ethernet0/0 Behavior:    STB Only   STP: Enabled

Port:   2    Interface:           tnip1 Behavior:    STB Only   STP: Enabled


Router1 ASRT config>EXIT
Router1 Config>
```

Sólo resta salvar la configuración y reiniciar el equipo.

b) Configuración Router2 y Router 3

Se suponen correctamente configurados para dar conectividad IP.

c) Configuración Router4

Al igual que para Router1, se añade el interfaz Frame Relay y el interfaz túnel IP (TNIP)

```
*P 4
Config>SET HOSTNAME Router4
Router4 Config>ADD DEVICE tnip 1
Added TNIP interface tnip1
Router4 Config>SET DATA-LINK FRAME-RELAY serial0/0
Router4 Config>PROTOCOL IP

-- Internet protocol user configuration --
Router4 IP config>ADDRESS ethernet0/0 195.6.1.3 255.255.255.0
Router4 IP config>ADDRESS serial0/0 2.2.2.2 255.255.255.0
Router4 IP config>ADDRESS tnip1 unnumbered
Router4 IP config>LIST ADDRESSES
IP addresses for each interface:
 ethernet0/0     195.6.1.3       255.255.255.0   NETWORK broadcast,  fill 0
 serial0/0       2.2.2.2         255.255.255.0   NETWORK broadcast,  fill 0
 serial0/1                                       IP disabled on this ifc
```

```
 serial0/2                                       IP disabled on this ifc
 bri0/0                                          IP disabled on this ifc
 x25-node                                        IP disabled on this ifc
 tnip1           unnumbered      0.0.0.0         NETWORK broadcast,  fill 0
Router4 IP config>EXIT
Router4 Config>
```

A continuación se configura el túnel

```
Router4 Config>NETWORK tnip1

-- IP Tunnel Net Configuration --
Router4 TNIP config>ENABLE
Router4 TNIP config>DESTINATION 1.1.1.1
Router4 TNIP config>SOURCE 2.2.2.2
Router4 TNIP config>LIST
Tunnel mode: GRE (enabled)
Tunnel source 2.2.2.2, destination 1.1.1.1
QoS preclassify: disabled
Router4 TNIP config>ENCAPSULATION

-- GRE Configuration --
Router4 GRE config>CHECKSUM
Router4 GRE config>KEY 1234
Router4 GRE config>SEQUENCE-DATAGRAM
Router4 GRE config>LIST
RC4 Cipher.................: disabled
End-to-End Checksumming....: enabled
Tunnel identification key..: enabled [1234]
Drop Out-of-Order Datagrams: enabled
Router4 GRE config>EXIT
Router4 TNIP config>EXIT
Router4 Config>
```

Se configura el interfaz Frame Relay

```
Router4 Config>NETWORK serial0/0

-- Frame Relay user configuration --
Router4 FR config>PVC 16 default
Router4 FR config>PVC 16 CIR 64000
Router4 FR config>PROTOCOL-ADDRESS 2.2.2.1 16
Router4 FR config>NO LMI
Router4 FR config>EXIT
Router4 Config>
```

Se incluyen las rutas necesarias

```
Router4 Config>PROTOCOL IP

-- Internet protocol user configuration --
Router4 IP config>ROUTE 1.1.1.0 255.255.255.0 2.2.2.1 1
Router4 IP config>EXIT
Router4 Config>
```

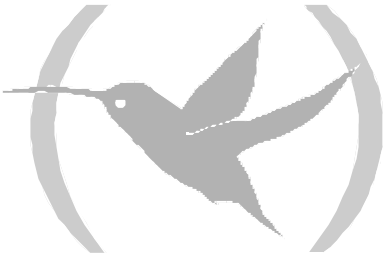
Por último se configura el bridge.

```
Router4 Config>PROTOCOL ASRT

-- ASRT Bridge user configuration --
Router4 ASRT config>BRIDGE
Router4 ASRT config>PORT ethernet0/0 1
Router4 ASRT config>PORT tnip1 2
Router4 ASRT config>NO STP
Router4 ASRT config>EXIT
Router4 Config>
```

Sólo resta salvar la configuración y reiniciar el equipo.

Capítulo 6
Eventos del interfaz Túnel IP (TNIP)



1. Monitorización de eventos del interfaz Túnel IP (TNIP)

Permiten monitorizar en tiempo real los eventos que suceden sobre uno o varios interfaces TNIP cuando esta habilitado el sistema de eventos para ese protocolo.

La forma en que se habilitan desde el menú de configuración es la siguiente:

```
*P 4
Config>EVENT

-- ELS Config --
ELS config>ENABLE TRACE SUBSYSTEM TNIP ALL
ELS config>EXIT
Config>save
Save configuration [n]? yes

Saving configuration...OK on Flash as XXXXX.CFG
Config>                                                    pulsar
<ctrl-p>
*RESTART
Are you sure to restart the system(Yes/No)? yes
```

Así mismo pueden ser habilitados desde el menú de monitorización en cualquier momento sin necesidad de que esté almacenada en la configuración de la siguiente forma:

```
*P 3
Console Operator
+EVENT

-- ELS Monitor --
ELS>ENABLE TRACE SUBSYSTEM TNIP ALL
ELS>EXIT
+
```

El listado de eventos disponibles para el protocolo TNIP es el siguiente:

TNIP.001

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.001 Rec paq *str_protocolo_enc*,prt ext 0x*num_protocolo_externo*, (*dirección_ip_origen->dirección_ip_destino*), int *interfaz*

Sintaxis Larga:

TNIP.001 Recibido paquete con encapsulado *str_protocolo_enc*, prtocolo externo 0x*num_protocolo_externo*, (origen *dirección_ip_origen* y destino *dirección_ip_destino*), en interfaz *interfaz*

Descripción:

Se ha recibido un paquete con un protocolo de encapsulado dado, que viajaba en un protocolo externo con numero dado, y el origen y destino del mismo es el dado. El interfaz encargado de su desencapsulado es el dado.

TNIP.002

Nivel: Error externo anormal, ERROR-AE/UE-ERROR

Sintaxis Corta:

TNIP.002 Rec paq *str_protocolo_enc*, prt ext 0x*num_protocolo_externo*, (*dirección_ip_origen->dirección_ip_destino*), no tunel

Sintaxis Larga:

TNIP.002 Recibido paquete con encapsulado *str_protocolo_enc*, protocolo externo 0x*num_protocolo_externo*, (origen *dirección_ip_origen* y destino *dirección_ip_destino*) no pertenece a ningún tunel

Descripción:

Se ha recibido un paquete con un protocolo de encapsulado dado. El protocolo externo es el dado. Con el origen y destino dados no se ha encontrado tunel que se pueda encargar de su desencapsulado.

Causa:

Un dispositivo externo esta enviando paquetes hacia el router pero no se aceptan por no tener el origen y destino configurados en los interfaces tunel

Acción:

Cambiar configuracion de tuneles para aceptar dichos paquetes si así se desea. Identificar el dispositivo externo y configurarlo para que no mande dichos paquetes.

TNIP.003

Nivel: Error interno anormal, ERROR-AI/UI-ERROR

Sintaxis Corta:

TNIP.003 Paq ent desc int no act, int *interfaz*

Sintaxis Larga:

TNIP.003 Paquete entrante ha sido descartado por estar el interfaz caído, interfaz *interfaz*

Descripción:

La funcion de entrada del interfaz de tunel ha descartado el paquete por estar dicho interfaz inactivo.

TNIP.004

Nivel: Error interno anormal, ERROR-AI/UI-ERROR

Sintaxis Corta:

TNIP.004 Paq ent desc int no conf, int *interfaz*

Sintaxis Larga:

TNIP.004 Paquete entrante ha sido descartado por estar el interfaz desconfigurado, interfaz *interfaz*

Descripción:

La funcion de entrada del interfaz de tunel ha descartado el paquete por estar dicho interfaz desconfigurado.

TNIP.005

Nivel: Error interno anormal, ERROR-AI/UI-ERROR

Sintaxis Corta:

TNIP.005 Paq ent desc no acept prot encapsul, int *interfaz*

Sintaxis Larga:

TNIP.005 Paquete entrante ha sido descartado por estar el interfaz configurado para otro protocolo de encapsulado, interfaz *interfaz*

Descripción:

La funcion de entrada del interfaz de tunel ha descartado el paquete por estar dicho interfaz configurado con un modo (tipo de protocolo de encapsulado esperado) distinto al que ha llegado.

Causa:

Interfaz mal configurado en alguno de los extremos del tunel

Acción:

Configurar el mismo modo de funcionamiento en ambos extremos del tunel.

TNIP.006

Nivel: Error interno anormal, ERROR-AI/UI-ERROR

Sintaxis Corta:

TNIP.006 Err cabec GRE flag routing act, int *interfaz*

Sintaxis Larga:

TNIP.006 Error en cabecera GRE por tener la opcion de routing en flags aciva, interfaz *interfaz*

Descripción:

La funcion de desencacpsulado GRE ha descartado el paquete por tener la opcion de routing activa.

Causa:

El otro extremo del tunel esta enviando paquetes GRE con campo routing. Por ahora no se aceptan este tipo de paquete.

Acción:

Configurar dicho extremo sin routing GRE.

TNIP.007

Nivel: Error externo anormal, ERROR-AE/UE-ERROR

Sintaxis Corta:

TNIP.007 Err cabec GRE chksm 0x*checksum* (esp 0x*checksum_esperado*), int *interfaz*

Sintaxis Larga:

TNIP.007 Error en cabecera GRE, checksum 0x*checksum* (esperado 0x*checksum_esperado*), interfaz *interfaz*

Descripción:

Este mensaje se genera cuando un paquete tiene un checksum invalido. El checksum recibido se muestra junto con el correcto

Causa:

Lo más probable es que sea un paquete dañado. Puede ser que un nodo esté construyendo una cabecera errónea.

Acción:

Si el problema persiste, examinar una traza para determinar donde se daña el paquete.

TNIP.008

Nivel: Error externo anormal, ERROR-AE/UE-ERROR

Sintaxis Corta:

TNIP.008 Desc paq GRE key *key* (esp *key_esperado*), int *interfaz*

Sintaxis Larga:

TNIP.008 Descartado paquete GRE, key *key* (esperado *key_esperado*), interfaz *interfaz*

Descripción:

Este mensaje se genera cuando un paquete tiene un key invalido. El key recibido se muestra junto con el correcto

Causa:

Puede ser que un nodo esté configurado con un key erroneo.

Acción:

Configurar el key correctamente.

TNIP.009

Nivel: Error externo anormal, ERROR-AE/UE-ERROR

Sintaxis Corta:

TNIP.009 Desc paq GRE sin key (esp *key_esperado*), int *interfaz*

Sintaxis Larga:

TNIP.009 Descartado paquete GRE sin key (esperado *key_esperado*), interfaz *interfaz*

Descripción:

Este mensaje se genera cuando un paquete no tiene key y el tunel estaba configurado con comprobacion de identificacion de key.

Causa:

Puede ser que un nodo esté mal configurado.

Acción:

Configurar el key correctamente.

TNIP.010

Nivel: Error externo anormal, ERROR-AE/UE-ERROR

Sintaxis Corta:

TNIP.010 Desc paq GRE con key *key* (esp sin key), int *interfaz*

Sintaxis Larga:

TNIP.010 Descartado paquete GRE, key *key* (no se espera key), interfaz *interfaz*

Descripción:

Este mensaje se genera cuando un paquete tiene key y el tunel esta configurado sin comprobacion de identificacion de key.

Causa:

Puede ser que un nodo esté mal configurado.

Acción:

Configurar el key correctamente.

TNIP.011

Nivel: Error externo anormal, ERROR-AE/UE-ERROR

Sintaxis Corta:

TNIP.011 Desc paq GRE sec *num_sec* menor esp *num_sec_esperado*, int *interfaz*

Sintaxis Larga:

TNIP.011 Descartado paquete GRE, numero secuencia *num_sec* menor que esperado *num_sec_esperado*, interfaz *interfaz*

Descripción:

Este mensaje se genera cuando un paquete llega con un numero de secuencia invalido. El numero de secuencia recibido se muestra junto con el correcto

Causa:

Los extremos estan desincronizados.

Acción:

Eperar a que se sincronicen. Si numeros de secuencia muy dispares y el tiempo de sincronizacion se prevee largo se puede optar por reiniciar los equipos.

TNIP.012

Nivel: Error interno anormal, ERROR-AI/UI-ERROR

Sintaxis Corta:

TNIP.012 Err cabec GRE flag recurs act, int *interfaz*

Sintaxis Larga:

TNIP.012 Error en cabecera GRE por tener la opcion de recursion aciva, interfaz *interfaz*

Descripción:

La funcion de desencacpsulado GRE ha descartado el paquete por tener la opcion de recursion activa. No aceptamos encapsulado multiple.

Causa:

El otro extremo del tunel esta enviando paquetes GRE con encapsulado multiple.

Acción:

Cambiar configuracion para evitar que paquetes GRE entren en el tunel.

TNIP.013

Nivel: Error interno anormal, ERROR-AI/UI-ERROR

Sintaxis Corta:

TNIP.013 Err cabec GRE vrsn *versión* incorrecta, int *interfaz*

Sintaxis Larga:

TNIP.013 Error en cabecera GRE por tener versión *versión* de encapsulado incorrecta, interfaz *interfaz*

Descripción:

La funcion de desencacpsulado GRE ha descartado el paquete por tener una versión no conocida.

Causa:

El otro extremo del tunel esta enviando paquetes GRE con version posterior. O esta construyendo mal la cabecera.

Acción:

Cambiar versión de GRE.

TNIP.014

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.014 Paq GRE desencap, prt inter 0x*num_protocolo*, (*dirección_ip_origen->dirección_ip_destino*), int *interfaz* sec *num_sec*

Sintaxis Larga:

TNIP.014 Paquete GRE desencapsulado con éxito, protocolo interno 0x*num_protocolo* (origen *dirección_ip_origen* y destino *dirección_ip_destino*), interfaz *interfaz* numero de secuencia *num_sec*

Descripción:

Se ha procesado el desencapsulado de un paquete GRE con éxito. Se especifica el protocolo que viajaba en el interior del paquete GRE (el origen y el destino) y el interfaz encargado de su desencapsulado.

TNIP.015

Nivel: Error externo anormal, ERROR-AE/UE-ERROR

Sintaxis Corta:

TNIP.015 Err paq GRE desencap, prt inter 0x*num_protocolo* descono, int *interfaz*

Sintaxis Larga:

TNIP.015 Error en paquete GRE desencapsulado, protocolo interno 0x*num_protocolo* desconocido, interfaz *interfaz*

Descripción:

Se ha procesado el desencapsulado de un paquete GRE y el protocolo que viajaba en el interior del paquete GRE es desconocido.

TNIP.016

Nivel: Error interno anormal, ERROR-AI/UI-ERROR

Sintaxis Corta:

TNIP.016 Paq sal desc int no act, int *interfaz*

Sintaxis Larga:

TNIP.016 Paquete saliente ha sido descartado por estar el interfaz caído, interfaz *interfaz*

Descripción:

La funcion de salida del interfaz de tunel ha descartado el paquete por estar dicho interfaz inactivo.

TNIP.017

Nivel: Error interno anormal, ERROR-AI/UI-ERROR

Sintaxis Corta:

TNIP.017 Paq sal desc int no conf, int *interfaz*

Sintaxis Larga:

TNIP.017 Paquete saliente ha sido descartado por estar el interfaz desconfigurado, interfaz *interfaz*

Descripción:

La funcion de salida del interfaz de tunel ha descartado el paquete por estar dicho interfaz desconfigurado.

TNIP.018

Nivel: Error interno anormal, ERROR-AI/UI-ERROR

Sintaxis Corta:

TNIP.018 Paq sal desc no acept prt inter 0x*num_protocolo*, int *interfaz*

Sintaxis Larga:

TNIP.018 Paquete saliente ha sido descartado por no aceptar protocolo interno 0x*num_protocolo*, interfaz *interfaz*

Descripción:

La funcion de salida del interfaz de tunel ha descartado el paquete por no aceptarse el protocolo interno (protocolo payload) para ser encapsulado.

TNIP.019

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.019 Paq GRE encapsul, prt inter 0x*num_protocolo*, (*dirección_ip_origen->dirección_ip_destino*), int *interfaz*

Sintaxis Larga:

TNIP.019 Paquete GRE encapsulado con éxito, protocolo interno 0x*num_protocolo*, (origen *dirección_ip_origen* y destino *dirección_ip_destino*), interfaz *interfaz*

Descripción:

La funcion de salida del interfaz de tunel ha desencapsulado un paquete GRE con éxito. Se especifica el protocolo del paquete que ha sido encapsulado así como el origen y el destino del mismo.

TNIP.020

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.020 Env paq *str_protocolo_enc*, prt ext 0x*num_protocolo_externo*, (*dirección_ip_origen->dirección_ip_destino*), int *interfaz* sec *num_sec*

Sintaxis Larga:

TNIP.020 Enviado paquete *str_protocolo_enc*, prt ext 0x*num_protocolo_externo*, (origen *dirección_ip_origen* y destino *dirección_ip_destino*), por interfaz *interfaz* numero de secuencia *num_sec*

Descripción:

Se ha enviado un paquete con un protocolo de encapsulado dado, el protocolo externo (delivery protocol) es el dado, también se especifica dirección origen y destino así como el interfaz que se encargó de su encapsulado.

TNIP.021

Nivel: Traza de operación normal, TRAZA-N/C-TRACE

Sintaxis Corta:

TNIP.021 Creado tn int *interfaz* (*dirección_ip_origen->dirección_ip_destino*)

Sintaxis Larga:

TNIP.021 Creado tunel por interfaz *interfaz* con direccion origen *dirección_ip_origen* y destino *dirección_ip_destino*

Descripción:

Se ha usado uno de los tuneles dinamicos libres para crear un nuevo tunel

TNIP.022

Nivel: Error externo anormal, ERROR-AE/UE-ERROR

Sintaxis Corta:

TNIP.022 Err tn lib, int *interfaz*

Sintaxis Larga:

TNIP.022 Error tunel liberado, interfaz *interfaz*

Descripción:

En un tunel dinamico establecido no se han recibido rutas por RIP ni tiene ninguna ruta estatica, o bien no recibe respuestas a los mensajes de mantenimiento keepalive. Por lo tanto se libera para poder ser utilizado de nuevo.

TNIP.023

Nivel: Traza de operación normal, TRAZA-N/C-TRACE

Sintaxis Corta:

TNIP.023 Lib tn int *interfaz* Rx idem IP *dirección_ip*

Sintaxis Larga:

TNIP.023 Liberado tunel interfaz *interfaz* por creacion de otro con misma direccion *dirección_ip*

Descripción:

Se ha creado un nuevo tunel con una direccion IP de un tunel que ya existia y aun no se habian dado las condiciones de liberacion y ahora es el momento de liberarlo

TNIP.024

Nivel: Traza de operación anormal, TRAZA-A/U-TRACE

Sintaxis Corta:

TNIP.024 Err bucle en tunel, int *interfaz*

Sintaxis Larga:

TNIP.024 Error bucle en tunel, interfaz *interfaz*

Descripción:

Se ha detectado en un tunel que se intenta encapsular un paquete que volverá al tunel provocando un bucle infinito

TNIP.025

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.025 Paq GRE cif, prt inter 0x*num_protocolo*, (*dirección_ip_origen->dirección_ip_destino*), int *interfaz*

Sintaxis Larga:

TNIP.025 Paquete GRE cifrado con éxito, protocolo intemo 0x*num_protocolo*, (origen *dirección_ip_origen* y destino *dirección_ip_destino*), interfaz *interfaz*

Descripción:

La funcion de salida del interfaz de tunel ha cifrado con exito el payload de un paquete GRE. Se especifica el protocolo del paquete que ha sido encapsulado así como el origen y el destino del mismo.

TNIP.026

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.026 Paq GRE descif, prt inter 0x*num_protocolo*, (*dirección_ip_origen->dirección_ip_destino*), int *interfaz*

Sintaxis Larga:

TNIP.026 Paquete GRE descifrado con éxito, protocolo interno 0x*num_protocolo* (origen

dirección_ip_origen y destino *dirección_ip_destino*), interfaz *interfaz*

Descripción:

Se ha descifrado con exito el payload de un paquete GRE. Se especifica el protocolo que viajaba en el interior del paquete GRE (el origen y el destino) y el interfaz encargado de su desencapsulado.

TNIP.027

Nivel: Error externo anormal, ERROR-AE/UE-ERROR

Sintaxis Corta:

TNIP.027 Err descif GRE, int *interfaz*

Sintaxis Larga:

TNIP.027 Error descifrando GRE, interfaz *interfaz*

Descripción:

Este mensaje se genera cuando un paquete se descifra y da un checksum de cifrado inválido, o el protocolo no era cifrado.

Causa:

Lo más probable es que la clave de cifrado esté mal, o que el paquete no esté cifrado.

Acción:

Si el problema persiste, comprobar la configuración del cifrado en los dos extremos del túnel.

TNIP.028

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.028 Paq GRE desencap, prt inter 0x*num_protocolo*, int *interfaz* sec *num_sec*

Sintaxis Larga:

TNIP.028 Paquete GRE desencapsulado con éxito, protocolo interno 0x*num_protocolo* , interfaz *interfaz* numero de secuencia *num_sec*

Descripción:

Se ha procesado el desencapsulado de un paquete GRE con éxito. Se especifica el protocolo que viajaba en el interior del paquete GRE y el interfaz encargado de su desencapsulado.

TNIP.029

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.029 Paq GRE cif, prt inter 0x*num_protocolo*, int *interfaz*

Sintaxis Larga:

TNIP.029 Paquete GRE cifrado con éxito, protocolo interno 0x*num_protocolo*, interfaz *interfaz*

Descripción:

La funcion de salida del interfaz de tunel ha cifrado con exito el payload de un paquete GRE. Se especifica el protocolo del paquete que ha sido encapsulado.

TNIP.030

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.030 Paq GRE encapsul, prt inter 0x*num_protocolo,* int *interfaz*

Sintaxis Larga:

TNIP.030 Paquete GRE encapsulado con éxito, protocolo interno 0x*num_protocolo*, interfaz *interfaz*

Descripción:

La funcion de salida del interfaz de tunel ha desencapsulado un paquete GRE con éxito. Se especifica el protocolo del paquete que ha sido encapsulado.

TNIP.031

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.031 Paq GRE desencap, prt inter 0x*num_protocolo*, (*dirección_mac_origen->dirección_mac_destino*), int *interfaz* sec *num_sec*

Sintaxis Larga:

TNIP.031 Paquete GRE desencapsulado con éxito, protocolo interno 0x*num_protocolo* (origen *dirección_mac_origen* y destino *dirección_mac_destino*), interfaz *interfaz* numero de secuencia *num_sec*

Descripción:

Se ha procesado el desencapsulado de un paquete GRE con éxito. Se especifica el protocolo que viajaba en el interior del paquete GRE (el origen y el destino mac) y el interfaz encargado de su desencapsulado.

TNIP.032

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.032 Paq GRE cif, prt inter 0x*num_protocolo*, (*dirección_mac_origen->dirección_mac_destino*), int *interfaz*

Sintaxis Larga:

TNIP.032 Paquete GRE cifrado con éxito, protocolo interno 0x*num_protocolo,* (origen *dirección_mac_origen* y destino *dirección_mac_destino*), interfaz *interfaz*

Descripción:

La funcion de salida del interfaz de tunel ha cifrado con exito el payload de un paquete GRE. Se especifica el protocolo del paquete que ha sido encapsulado así como el origen y el destino mac del mismo.

TNIP.033

Nivel: Traza por paquete, TRAZA-P/P-TRACE

Sintaxis Corta:

TNIP.033 Paq GRE encapsul, prt inter 0x*num_protocolo*, (*dirección_mac_origen->dirección_mac_destino*), int *interfaz*

Sintaxis Larga:

TNIP.033 Paquete GRE encapsulado con éxito, protocolo interno 0x*num_protocolo*, (origen *dirección_mac_origen* y destino *dirección_mac_destino*), interfaz *interfaz*

Descripción:

La funcion de salida del interfaz de tunel ha desencapsulado un paquete GRE con éxito. Se especifica el protocolo del paquete que ha sido encapsulado así como el origen y el destino mac del mismo.

TNIP.034

Nivel: Error interno anormal, ERROR-AI/UI-ERROR

Sintaxis Corta:

TNIP.034 No paq para kal int *interfaz*

Sintaxis Larga:

TNIP.034 No hay paquete para enviar keepalive interfaz *interfaz*

Descripción:

No se ha podido reservar un buffer cuando se necesitaba para enviar un paquete de peticion de keepalive.

Causa:

La causa mas probable es una sobrecarga temporal de trafico.

Acción:

Si el mensaje persiste, contacte con su servicio de atencion al cliente.

TNIP.035

Nivel: Traza de operación normal, TRAZA-N/C-TRACE

Sintaxis Corta:

TNIP.035 Env kal int *interfaz*

Sintaxis Larga:

TNIP.035 Enviando paquete keepalive en interfaz *interfaz*

Descripción:

Se va a enviar un paquete de keepalive por el interfaz indicado para comprobar la conectividad del tunel.

Causa:

La comprobacion de conectividad extremo a extremo del tunel esta habilitada, por lo que se envian paquetes keepalive periodicamente para conocer el estado de dicha conectividad.

TNIP.036

Nivel: Traza de operación anormal, TRAZA-A/U-TRACE

Sintaxis Corta:

TNIP.036 Mal id kal *id_recibido* (esp *id_esperado*) int *interfaz*

Sintaxis Larga:

TNIP.036 Mal numero de identificacion keepalive *id_recibido* (se esperaba *id_esperado*) recibido en interfaz *interfaz*

Descripción:

Se ha recibido una respuesta de keepalive en el interfaz especificado, con un numero de identificacion inesperado. Se muestran los numeros de identificacion recibido y esperado, asi como el interfaz de entrada.

Causa:

La causa mas probable es que se haya configurado un periodo de keepalive mas corto que el retardo que introduce la red.

Acción:

Si el mensaje persiste incremente el periodo de keepalive de acuerdo con el retardo de red.

TNIP.037

Nivel: Traza de operación normal, TRAZA-N/C-TRACE

Sintaxis Corta:

TNIP.037 Rec kal int *interfaz*

Sintaxis Larga:

TNIP.037 Recibido keepalive en interfaz *interfaz*

Descripción:

Se ha recibido una respuesta de keepalive en el interfaz especificado.

Causa:

Esta es la respuesta que se corresponde con el ultimo paquete de peticion keepalive enviado. Este es el comportamiento normal cuando keepalive esta habilitado en el interfaz.

TNIP.038

Nivel: Traza de operación anormal, TRAZA-A/U-TRACE

Sintaxis Corta:

TNIP.038 Desc kal int *interfaz*

Sintaxis Larga:

TNIP.038 Descartado keepalive en interfaz *interfaz*

Descripción:

Se ha recibido una respuesta keepalive inesperada en el interfaz especificado.

Causa:

No se esperaba ninguna respuesta keepalive, posiblemente porque el anterior keepalive vencio sin recibir respuesta y el interfaz se desactivo.

Acción:

Si el mensaje persiste, compruebe el estado del interfaz de tunel, la configuracion de keepalive y el estado de congestion de la red.