

Ejemplo de Configuración de Tunelización de Protocolos Asíncronos en BSTUN

Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Convenciones](#)

[Antecedentes](#)

[Configurar](#)

[Diagrama de la red](#)

[Configuraciones](#)

[Verificación](#)

[Troubleshoot](#)

[Información Relacionada](#)

[Introducción](#)

Los protocolos asíncronos dedicados y nativos no se soportan directamente con ninguna implementación de Cisco. Sin embargo, la tunelización asíncrona de túnel serial de bloque (BSTUN) puede proporcionar una capacidad limitada para tunelizar estos datos.

[Prerequisites](#)

[Requirements](#)

No hay requisitos específicos para este documento.

[Componentes Utilizados](#)

La información que contiene este documento se basa en las versiones de software y hardware.

- Utilice [Feature Navigator II](#) (sólo clientes registrados) y utilice la opción **Search by Feature**.
- Utilice [Software Advisor](#) (sólo clientes [registrados](#)) para buscar la versión mínima de software soportada necesaria para su hardware.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

[Convenciones](#)

Consulte [Convenciones de Consejos Técnicos Cisco para obtener más información sobre las convenciones del documento.](#)

[Antecedentes](#)

Los protocolos asíncronos como el TC500 de Diebold para comunicarse con los cajeros automáticos de dinero o para tunelizar HyperTerminal desde un PC a otro PC no tienen soporte directo ni implementación en el IOS® de Cisco. Como implica el nombre, se trata de una implementación genérica que tiene alguna capacidad para transportar este tipo de datos. Esto se conoce como BSTUN async-generic y requiere el conjunto de características IBM o Enterprise IOS.

BSTUN async-generic se diseñó originalmente para transportar paquetes pequeños unidireccionales desde dispositivos de seguridad a un dispositivo de generación de informes. BSTUN async-generic, sin embargo, puede transportar tráfico interactivo. En esencia, esta implementación se conecta a dispositivos nativos asíncronos y recibe los datos en la interfaz serial y luego en un buffer de memoria. Periódicamente, los datos almacenados en la memoria intermedia se encapsulan en un paquete TCP y se envían al peer BSTUN donde se desencapsula y se envían al dispositivo asíncrono conectado en el sitio remoto.

BSTUN async-generic es una operación simplista. El router no tiene capacidad para ser configurado para tener conocimiento del inicio de la trama (SOF), el fin de la trama (EOF) o el esquema de direccionamiento del protocolo asíncrono. Si la parte de dirección de la trama está en cada trama, tiene un byte de longitud y es el mismo lugar en la trama, entonces el comando **asp address-offset** se puede ejecutar para especificar al router dónde encontrar la dirección en la trama, como se muestra más adelante en este documento. En muchas situaciones, sin embargo, no habrá una parte de dirección contenida en el protocolo. No tener conocimiento de la construcción del protocolo asíncrono significa que el router no puede distinguir paquetes individuales de otros si no están separados por un período de tiempo. Se requieren aproximadamente 40 ms entre tramas a 9600 bits/seg para proporcionar al router una cantidad de tiempo adecuada para distinguir correctamente un paquete de otro. El router simplemente ve un flujo de datos en su interfaz serial y luego envuelve estos datos en TCP. No hay posibilidad de que el router pueda tomar decisiones de ruteo en función de cualquier aspecto individual de la trama entrante. Por lo tanto, BSTUN async-generic debe estar diseñado físicamente de modo que sólo un dispositivo se conecte a la interfaz serial del router. No hay función de reconocimiento local. BSTUN sólo admite Local-ack para el protocolo IBM3270 BISYNC.

[Configurar](#)

En esta sección encontrará la información para configurar las funciones descritas en este documento.

[Diagrama de la red](#)

Este documento utiliza la configuración de red que se muestra en el siguiente diagrama.



Ambos PC utilizan HyperTerminal de Microsoft o en lugar de uno de los PC podría ser una conexión al puerto de la consola de un router de Cisco. Estas configuraciones de ejemplo representan configuraciones implementadas desde routers que no estaban previamente configurados en un escenario de laboratorio, y muestran las partes relevantes de la configuración necesaria. Se configuran suponiendo una conexión de 9600 bits/seg, 8N1.

Configuraciones

Este documento utiliza las configuraciones que se muestran en esta sección.

- Router principal (router Cisco 1700)
- Router remoto (router Cisco 3640)
- Router principal (router Cisco 3600)
- Mando a distancia n.º 1 (router Cisco 1700)
- Mando a distancia n.º 2 (router Cisco 1700)

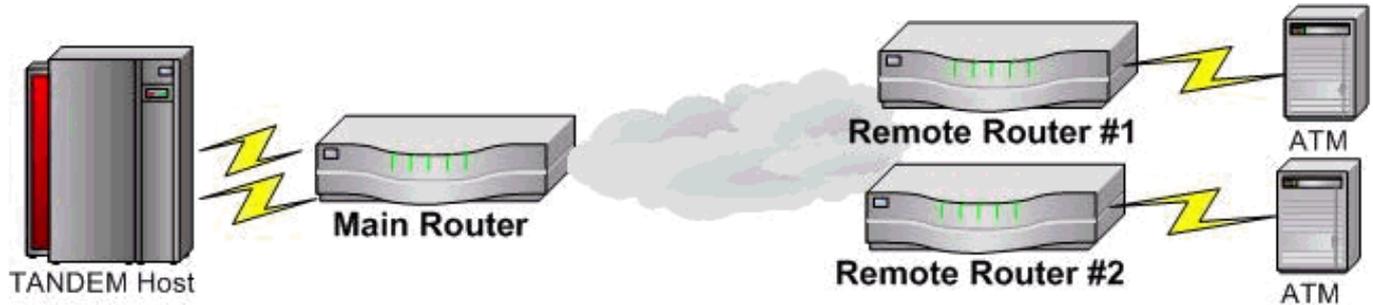
Router principal (router Cisco 1700)

```
main#show running-config
Building configuration...
.
.
.
ip subnet-zero
bstun peer-name 10.1.1.1
bstun protocol-group 1 async-generic
interface loopback0
  ip address 10.1.1.1 255.0.0.0
interface serial0
  physical-layer async
  encapsulation bstun
  asp role secondary
  bstun group 1
  bstun route all tcp 30.1.1.1
interface serial1
  ip address 20.1.1.1 255.0.0.0
ip route 0.0.0.0 0.0.0.0 20.1.1.2
line 1
  speed 9600
  databits 8
  parity none
  stopbits 1
.
.
.
!
```

Router remoto (router Cisco 3640)

Line(s) not in async mode -or- with no hardware support:
1-64, 66-128

En este escenario, un tándem se comunica con dispositivos ATM remotos. En esta configuración de ejemplo, el protocolo asíncrono ejecuta un protocolo 4800 7E2 y el router principal conectado al TANDEM es un router de la serie 3600 a routers de la serie 1700 remotos. Consulte este diagrama de red.



Router principal (router Cisco 3600)

```
main#show running-config
Building configuration...
bstun peer-name 10.1.1.1.
bstun protocol-group 1 async-generic
bstun protocol-group 2 async-generic
interface loopback 0
  ip address 10.1.1.1
interface serial11/0
  encapsulation frame-relay
interface serial 1/0.1 point-to-point
  ip address 20.1.1.1 255.255.255.0
  frame-relay interface-dlci 100
interface serial 1/0.2 point-to-point
  ip address 20.2.1.1 255.255.255.0
  frame-relay interface-dlci 200
interface serial 2/0
  physical-layer async
  encapsulation bstun
  asp role secondary
  bstun group 1
  bstun route all tcp 30.1.1.1

interface serial 2/1
  physical-layer async
  encapsulation bstun
  asp role secondary
  bstun group 2
  bstun route all tcp 30.2.1.1

ip route 30.2.1.0 255.255.0.0 20.2.1.2
ip route 0.0.0.0 0.0.0.0 20.1.1.2
line 65
  speed 4800
  parity even
  databits 7
  stopbits 1
.
line 66
```

```
speed 4800
parity even
databits 7
stopbits 1
.
!
end
```

Mando a distancia n.º 1 (router Cisco 1700)

```
REMOTE1#show running-config
Building configuration...
bstun peer-name 30.1.1.1
bstun protocol-group 1 async-generic
interface loopback0
  ip address 30.1.1.1 255.255.0.0
interface serial0
  physical-layer async
  encapsulation bstun
  asp role primary
  bstun group 1
  bstun route all tcp 10.1.1.1
interface serial1
  encapsulation frame-relay
interface serial1.1 point-to-point
  ip address 20.1.1.2 255.255.255.0
  frame-relay interface-dlci 100
ip route 0.0.0.0 0.0.0.0 20.1.1.1
line 1
  speed 4800
  databits 7
  parity even
  stopbits 2
.
.
.
!
end
```

Mando a distancia n.º 2 (router Cisco 1700)

```
REMOTE2#show running-config
Building configuration...
bstun peer-name 30.2.1.1
bstun protocol-group 2 async-generic
interface loopback0
  ip address 30.2.1.1 255.255.0.0
interface serial0
  physical-layer async
  encapsulation bstun
  asp role primary
  bstun group 2
  bstun route all tcp 10.1.1.1
interface serial1
  encapsulation frame-relay
interface serial1.1 point-to-point
  ip address 20.2.1.2 255.255.255.0
  frame-relay interface-dlci 100
ip route 0.0.0.0 0.0.0.0 20.2.1.1
line 1
  speed 4800
```

```
databits 7
parity even
stopbits 2
.
.
.
!
end
```

Verificación

Actualmente, no hay un procedimiento de verificación disponible para esta configuración.

Troubleshoot

BSTUN recibe un paquete en la interfaz serial, lo encapsula y envía este paquete TCP al router remoto cuando se ejecuta el comando **bstun route all tcp**. El paquete TCP se recibe en el router remoto y se desencapsula. Los datos se envían en la interfaz serial. Si esta conexión no funciona, primero se deben verificar los datos entrantes con el paquete **debug asp**. Los datos recibidos por el router en la interfaz serial. Dado que el router no tiene construcción de protocolo y varía según el protocolo asíncrono, no se proporciona debug de ejemplo. El flujo de datos visto por el router debe coincidir con lo que envía el dispositivo. Si no coincide, lo más probable es que la velocidad, los databits, la paridad o los stopbits no estén configurados para coincidir con el dispositivo. Este también puede ser el caso si no se recibe ningún dato.

Si se reciben datos en la interfaz serial, ejecute el comando **show bstun** para mostrar si la conexión está abierta o cerrada. El estado abierto con sólo paquetes transmitidos indica que el TCP se envía al peer BSTUN remoto. En este punto, una prueba de ping desde la dirección IP del nombre de peer BSTUN local a la dirección IP de nombre de peer BSTUN remota verifica si la IP está configurada y funcionando correctamente. Si la prueba de ping es exitosa, entonces en el mando a distancia, ejecute el comando **debug asp packet** para determinar si el paquete se recibe y se envía a la interfaz serial al dispositivo asíncrono.

Complete estos pasos para resolver el problema.

1. Verifique que los datos se reciban en el router host con el comando **debug asp packet**.
2. Asegúrese de la conectividad IP con ping test sourcing pings de la dirección IP bstun peer-name a la dirección IP remota del nombre de peer BSTUN remoto.
3. En el mando a distancia, verifique que los paquetes se envíen al dispositivo remoto con el comando **debug asp packet**.
4. Si el protocolo asíncrono tiene una dirección contenida en los paquetes enviados al router, puede ser beneficioso ejecutar el comando **asp offset-address** bajo la interfaz con el número de byte apropiado correspondiente al lugar donde la dirección está contenida en el paquete. El valor predeterminado para esto es 0. Por ejemplo, si el paquete es 01C1ABCDEF, donde C1 es la dirección, la interfaz serial se puede configurar con el comando **asp offset-address 01**. En algunos casos, esto permite al router identificar un paquete y aumenta la probabilidad de que el router maneje los datos como un paquete enmarcado y no sólo como un flujo de datos.

Información Relacionada

- [Soporte Técnico STUN \(Túnel Serial\) y BSTUN \(Túnel Serial de Bloqueo\)](#)
- [Soporte Técnico - Cisco Systems](#)