IPSec sobre configurações de exemplo de cabo e depurações

Índice

Introdução
Pré-requisitos
Requisitos
Componentes Utilizados
Convenções
Material de Suporte
Configurar
Diagrama de Rede
Configurações
Verificar
Troubleshooting
Informações Relacionadas

Introdução

A segurança de protocolo do Internet (IPsec) é uma estrutura dos padrões abertos que assegure comunicações privadas seguras sobre redes IP. Baseado nos padrões desenvolvidos pelo Internet Engineering Task Force (IETF), o IPsec assegura a confidencialidade, a integridade, e a autenticidade das comunicações de dados através de uma rede IP pública. O IPsec fornece um componente necessário para um com base em padrões, solução flexível para distribuir uma política de segurança de toda a rede.

Este documento fornece um exemplo de configuração de um IPsec entre dois cable modens Cisco. Esta configuração cria um túnel de criptografia através de uma rede de cabo entre dois roteadores de cable modem do uBR9xx Series de Cisco. Todo o tráfego entre as duas redes é cifrado. Mas o tráfego destinado para outras redes é permitido passar unencrypted. Para usuários do escritório pequeno, escritório home (SOHO), isto permite a criação do Virtual Private Networks (VPNs) através de uma rede de cabo.

Pré-requisitos

Requisitos

Não existem requisitos específicos para este documento.

Componentes Utilizados

O Modems deve conformar-se a estas exigências configurar o IPsec em dois Modems a cabo:

- Cisco uBR904, uBR905, ou uBR924 no modo de roteamento
- Conjunto de recursos do IPsec 56
- Software Release 12.0(5)T ou Mais Recente de Cisco IOS®

Além, você deve ter um cable modem termination system (CMTS), que seja todo o Data-over-Cable Service Interface Specifications (DOCSIS) - roteador de cabo complacente do final do cabeçalho, tal como o uBR7246VXR do Cisco uBR7246, do Cisco uBR7223, ou do Cisco.

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a sua rede estiver ativa, certifique-se de que entende o impacto potencial de qualquer comando.

Convenções

Consulte as Convenções de Dicas Técnicas da Cisco para obter mais informações sobre convenções de documentos.

Material de Suporte

O exemplo neste documento usa um modem a cabo do uBR904, um modem a cabo do uBR924, e um uBR7246VXR CMTS. O Modems a cabo executa o Cisco IOS Software Release 12.1(6), e o CMTS executa o Cisco IOS Software Release 12.1(4)EC.

Nota: Este exemplo é feito com configuração manual no Modems a cabo através da porta de Console. Se um processo automático é executado através do arquivo de configuração DOCSIS (o script ios.cfg está criado com as Listas de acesso da configuração IPSec) então 100 e 101 não pode ser usado. Isto é porque a implementação Cisco da tabela dos docsDevNmAccess do Simple Network Management Protocol (SNMP) usa Listas de acesso do Cisco IOS. Cria uma lista de acessos pela relação. No uBR904, em 924, e em 905, as primeiras duas Listas de acesso são usadas geralmente (100 e 101). Em um modem a cabo que apoie o barramento serial universal (USB), como o CVA120, três Listas de acesso são usados (100, 101, e 102).

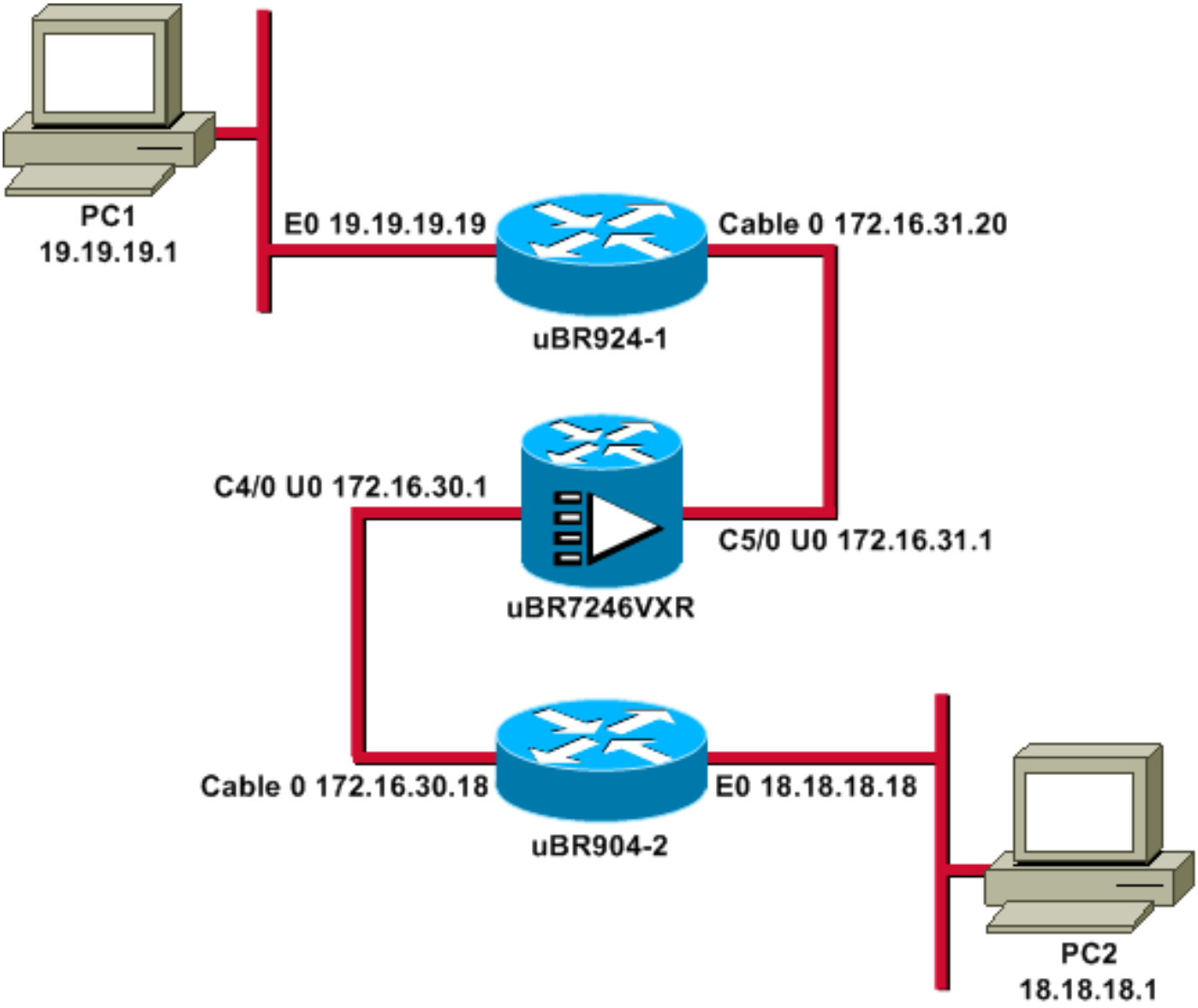
Configurar

Nesta seção, você encontrará informações para configurar os recursos descritos neste documento.

Nota: Use a ferramenta de consulta de comandos (clientes registrados somente) para encontrar a informação adicional sobre os comandos neste documento.

Diagrama de Rede

Este documento utiliza a seguinte configuração de rede:



Nota: Todos os endereços IP de Um ou Mais Servidores Cisco ICM NT neste diagrama têm uma máscara 24-bit.

Configurações

Este documento utiliza as seguintes configurações:

- uBR924-1
- uBR904-2
- uBR7246VXR

uBR924-1

```
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname ubr924-1
!
enable password ww
!
!
!
```

```
!
clock timezone - -8
ip subnet-zero
no ip finger
!
ip audit notify log
ip audit po max-events 100
!
!
crypto isakmp policy 10 !--- Creates an Internet Key
Exchange (IKE) policy with the specified priority !---
number of 10. The range for the priority is 1 to 10000,
where 1 is the !--- highest priority. This command also
enters Internet Security Association !--- and Key
Management Protocol (ISAKMP) policy configuration
command mode. hash md5 !--- Specifies the MD5 (HMAC
variant) hash algorithm for packet authentication.
authentication pre-share !--- Specifies that the
authentication keys are pre-shared, as opposed to !---
dynamically negotiated using Rivest, Shamir, and Adelman
(RSA) public !--- key signatures. group 2 !--- Diffie-
Hellman group for key negotiation. lifetime 3600 !---
Defines how long, in seconds, each security association
should exist before !--- it expires. Its range is 60 to
86400, and in this case, it is 1 hour. crypto isakmp key
mykey address 18.18.18.18 !--- Specifies the pre-shared
key that should be used with the peer at the !---
specific IP address. The key can be any arbitrary
alphanumeric key up to !--- 128 characters. The key is
case-sensitive and must be entered identically !--- on
both routers. In this case, the key is mykey and the
peer is the !--- Ethernet address of uBR904-2 . ! crypto
IPSec transform-set TUNNELSET ah-md5-hmac esp-des !---
Establishes the transform set to use for IPsec
encryption. As many as !--- three transformations can be
specified for a set. Authentication Header !--- and ESP
are in use. Another common transform set used in
industry is !--- esp-des esp-md5-hmac. ! crypto map
MYMAP local-address Ethernet0 !--- Creates the MYMAP
crypto map and applies it to the Ethernet0 interface.
crypto map MYMAP 10 ipsec-isakmp !--- Creates a crypto
map numbered 10 and enters crypto map configuration
mode. set peer 18.18.18.18 !--- Identifies the IP
address for the destination peer router. In this case,
!--- the Ethernet interface of the remote cable modem
(ubr904-2) is used. set transform-set TUNNELSET !---
Sets the crypto map to use the transform set previously
created. match address 101 !--- Sets the crypto map to
use the access list that specifies the type of !---
traffic to be encrypted. !--- Do not use access lists
100, 101, and 102 if the IPsec config is !--- downloaded
through the ios.cfg in the DOCSIS configuration file. !
! ! ! voice-port 0 input gain -2 output attenuation 0 !
voice-port 1 input gain -2 output attenuation 0 ! ! !
interface Ethernet0 ip address 19.19.19.19 255.255.255.0
ip rip send version 2 ip rip receive version 2 no ip
route-cache no ip mroute-cache ! interface cable-modem0
ip rip send version 2 ip rip receive version 2 no ip
route-cache no ip mroute-cache cable-modem downstream
saved channel 525000000 39 1 cable-modem mac-timer t2
40000 no cable-modem compliant bridge crypto map MYMAP
!--- Applies the previously created crypto map to the
cable interface. ! router rip version 2 network 19.0.0.0
network 172.16.0.0 ! ip default-gateway 172.16.31.1 ip
```

```
classless ip http server ! access-list 101 permit ip
19.19.19.0 0.0.0.255 18.18.18.0 0.0.0.255 !--- Access
list that identifies the traffic to be encrypted. In
this case, !--- it is setting traffic from the local
Ethernet network to the remote !--- Ethernet network.
snmp-server manager ! line con 0 transport input none
line vty 0 4 password ww login ! end
```

A configuração do outro modem a cabo é muito similar, a maioria dos comentários na configuração precedente é omitida assim.

uBR904-2

```
version 12.1
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname ubr904-2
!
enable password ww
!
!
!
!
!
clock timezone - -8
ip subnet-zero
no ip finger
!
!
!
crypto isakmp policy 10 hash md5 authentication pre-
share group 2 lifetime 3600 crypto isakmp key mykey
address 19.19.19.19 ! ! crypto IPSec transform-set
TUNNELSET ah-md5-hmac ESP-Des ! crypto map MYMAP local-
address Ethernet0 crypto map MYMAP 10 ipsec-isakmp set
peer 19.19.19.19 !--- Identifies the IP address for the
destination peer router. In this case, !--- the Ethernet
interface of the remote cable modem (uBR924-1) is used.
set transform-set TUNNELSET match address 101 ! ! ! !
interface Ethernet0 ip address 18.18.18.18 255.255.255.0
ip rip send version 2 ip rip receive version 2 !
interface cable-modem0 ip rip send version 2 ip rip
receive version 2 no keepalive cable-modem downstream
saved channel 555000000 42 1 cable-modem Mac-timer t2
40000 no cable-modem compliant bridge crypto map MYMAP !
router rip version 2 network 18.0.0.0 network 172.16.0.0
! ip default-gateway 172.16.30.1 ip classless no ip http
server ! access-list 101 permit ip 18.18.18.0 0.0.0.255
19.19.19.0 0.0.0.255 snmp-server manager ! line con 0
transport input none line vty 0 4 password ww login !
end
```

O uBR7246VXR CMTS igualmente executa a versão 2 do Routing Information Protocol (RIP), de modo que o roteamento trabalhe. Esta é a configuração RIP usada no CMTS:

uBR7246VXR

```
router rip
 version 2
 network 172.16.0.0
```

```
no auto-summary
```

Verificar

Use esta seção para confirmar se a sua configuração funciona corretamente.

A fim verificar que o IPsec trabalha:

- Verifique estas coisas:O Cisco IOS Software apoia o IPsec.A configuração running está correta.As relações estão acima.Distribuindo trabalhos.A lista de acessos definida para cifrar o tráfego está correta.
- Crie o tráfego e olhe a criptografia e o Decrypt, para ver a quantidade que está aumentando.
- Gire debuga sobre para cripto.

A Output Interpreter Tool (apenas para clientes registrados) (OIT) suporta determinados comandos show. Use a OIT para exibir uma análise da saída do comando show.

Emita o **comando show version em** ambo o Modems a cabo.

```
ubr924-1#show version Cisco Internetwork Operating System Software IOS (tm) 920 Software
(UBR920-K1O3SV4Y556I-M), Version 12.1(6), RELEASE SOFTWARE (fc1) Copyright (c) 1986-2000 by
Cisco Systems, Inc. Compiled Wed 27-Dec-00 16:36 by kellythw Image text-base: 0x800100A0, data-
base: 0x806C1C20 ROM: System Bootstrap, Version 12.0(6r)T3, RELEASE SOFTWARE (fc1) ubr924-1
uptime is 1 hour, 47 minutes System returned to ROM by reload at 10:39:05 - Fri Feb 9 2001
System restarted at 10:40:05 - Fri Feb 9 2001 System image file is "flash:ubr920-k1o3sv4y556i-
mz.121-6" cisco uBR920 CM (MPC850) processor (revision 3.e) with 15872K/1024K bytes of memory.
Processor board ID FAA0422Q04F Bridging software. 1 Ethernet/IEEE 802.3 interface(s) 1 Cable
Modem network interface(s) 3968K bytes of processor board System flash (Read/Write) 1536K bytes
of processor board Boot flash (Read/Write) Configuration register is 0x2102
```

O uBR924-1 executa o Cisco IOS Software Release 12.1(6) com o conjunto de recursos do VALUE SMALL OFFICE/VOICE/FW IPSEC 56.

```
ubr904-2#show version Cisco Internetwork Operating System Software IOS (TM) 900 Software
(UBR900-K1OY556I-M), Version 12.1(6), RELEASE SOFTWARE (fc1) Copyright (c) 1986-2000 by cisco
Systems, Inc. Compiled Wed 27-DEC-00 11:06 by kellythw Image text-base: 0x08004000, database:
0x085714DC ROM: System Bootstrap, Version 11.2(19980518:195057), RELEASED SOFTWARE ROM: 900
Software (UBR900-RBOOT-M), Version 11.3(11)NA, EARLY DEPLOYMENT RELEASE SOFTWARE (fc1) ubr904-2
uptime is 1 hour, 48 minutes System returned to ROM by reload at 10:38:44 - Fri Feb 9 2001
System restarted at 10:40:37 - Fri Feb 9 2001 System image file is "flash:ubr900-k1oy556i-
mz.121-6" cisco uBR900 CM (68360) processor (revision D) with 8192K bytes of memory. Processor
board ID FAA0235Q0ZS Bridging software. 1 Ethernet/IEEE 802.3 interface(s) 1 Cable Modem network
interface(s) 4096K bytes of processor board System flash (Read/Write) 2048K bytes of processor
board Boot flash (Read/Write) Configuration register is 0x2102
```

O uBR904-2 executa o Cisco IOS Software Release 12.1(6) com conjunto de recursos PEQUENO do IPsec 56 OFFICE/FW.

```
ubr924-1#show ip interface brief Interface IP-Address OK? Method Status Protocol Ethernet0
19.19.19.19 YES NVRAM up up cable-modem0 172.16.31.20 YES unset up up ubr904-2#show ip interface
brief Interface IP-Address OK? Method Status Protocol Ethernet0 18.18.18.18 YES NVRAM up up
cable-modem0 172.16.30.18 YES unset up up
```

Do último comando, você pode ver que as interfaces Ethernet estão acima. Os endereços IP de Um ou Mais Servidores Cisco ICM NT das interfaces Ethernet foram incorporados manualmente. As interfaces de cabo são igualmente ascendentes e aprenderam seus endereços IP de Um ou Mais Servidores Cisco ICM NT com o DHCP. Porque estes endereços de cabo são atribuídos dinamicamente, não podem ser usados como pares na configuração IPSec.

```
ubr924-1#show ip route Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area N1 - OSPF NSSA external type 1,
N2 - OSPF NSSA external type 2 E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP i -
IS-IS, L1 - ISIS level-1, L2 - ISIS level-2, ia - ISIS inter area * - candidate default, U -
per-user static route, o - ODR P - periodic downloaded static route Gateway of last resort is
172.16.31.1 to network 0.0.0.0 19.0.0.0/24 is subnetted, 1 subnets C 19.19.19.0 is directly
connected, Ethernet0 R 18.0.0.0/8 [120/2] via 172.16.31.1, 00:00:23, cable-modem0 172.16.0.0/16
is variably subnetted, 4 subnets, 3 masks R 172.16.135.0/25 [120/1] via 172.16.31.1, 00:00:23,
cable-modem0 R 172.16.29.0/27 [120/1] via 172.16.31.1, 00:00:23, cable-modem0 R 172.16.30.0/24
[120/1] via 172.16.31.1, 00:00:23, cable-modem0 C 172.16.31.0/24 is directly connected, cable-
modem0 R 192.168.99.0/24 [120/3] via 172.16.31.1, 00:00:24, cable-modem0 10.0.0.0/24 is
subnetted, 2 subnets R 10.10.10.0 [120/2] via 172.16.31.1, 00:00:24, cable-modem0 S* 0.0.0.0/0
[1/0] via 172.16.31.1
```

Você pode ver do este para output que uBR924-1 está aprendendo sobre a rota 18.18.18.0, que é a interface Ethernet de uBR904-2.

```
ubr904-2#show ip route Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area N1 - OSPF NSSA external type 1,
N2 - OSPF NSSA external type 2 E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP i -
ISIS, L1 - ISIS level-1, L2 - ISIS level-2, IA - ISIS inter area * - candidate default, U - per-
user static route, o - ODR P - periodic downloaded static route Gateway of last resort is
172.16.30.1 to network 0.0.0.0 R 19.0.0.0/8 [120/2] via 172.16.30.1, 00:00:17, cable-modem0
18.0.0.0/24 is subnetted, 1 subnets C 18.18.18.0 is directly connected, Ethernet0 172.16.0.0/16
is variably subnetted, 4 subnets, 3 masks R 172.16.135.0/25 [120/1] via 172.16.30.1, 00:00:17,
cable-modem0 R 172.16.29.224/27 [120/1] via 172.16.30.1, 00:00:17, cable-modem0 C 172.16.30.0/24
is directly connected, cable-modem0 R 172.16.31.0/24 [120/1] via 172.16.30.1, 00:00:17, cable-
modem0 R 192.168.99.0/24 [120/3] via 172.16.30.1, 00:00:18, cable-modem0 10.0.0.0/24 is
subnetted, 1 subnets R 10.10.10.0 [120/2] via 172.16.30.1, 00:00:18, cable-modem0 S* 0.0.0.0/0
[1/0] via 172.16.30.1
```

Da tabela de roteamento de uBR904-2, você pode ver que a rede para os Ethernet de uBR924-1 está na tabela de roteamento.

Nota: Pôde haver os casos onde você não pode executar um protocolo de roteamento entre os dois Modems a cabo. Nesses casos, você deve adicionar rotas estáticas no CMTS ao tráfego direto para as interfaces Ethernet do Modems a cabo.

A coisa seguinte a verificar é a certificação da lista de acessos; emita o **comando show access-lists em** ambo o Roteadores.

```
ubr924-1#show access-lists Extended IP access list 101 permit ip 19.19.19.0 0.0.0.255 18.18.18.0
0.0.0.255 (2045 matches) ubr904-2#show access-lists Extended IP access list 101 permit ip
18.18.18.0 0.0.0.255 19.19.19.0 0.0.0.255 (2059 matches)
```

A lista de acessos ajustou a sessão IPSec quando o LAN atrás de uBR924-1 (19.19.19.0) envia o tráfego IP ao LAN atrás de uBR904-2 (18.18.18.0), e vice-versa. Não use “alguns” nas Listas de acesso, porque cria problemas. Refira configurar a Segurança de rede IPSec para mais detalhes.

Não há nenhum tráfego de IPSec. Emita o **comando show crypto engine connection ative**.

```
ubr924-1#show crypto engine connection active ID Interface IP-Address State Algorithm Encrypt
Decrypt 1 set HMAC_MD5+DES_56_CB 0 0 ubr904-2#show crypto engine connection active ID Interface
IP-Address State Algorithm Encrypt Decrypt 1 set HMAC_MD5+DES_56_CB 0 0
```

Não há nenhuma conexão IPSec porque o sem tráfego combinou as Listas de acesso.

Nota: Consulte Informações Importantes sobre Comandos de Depuração antes de usar comandos **debug**.

A próxima etapa é girar sobre algum cripto debuga para gerar o tráfego interessante.

Neste exemplo, estes debugam são girados sobre:

- **debug crypto engine**
- debug crypto ipsec
- **debug crypto key-exchange**
- debug crypto isakmp

Você deve primeiramente gerar algum tráfego interessante para ver que a saída do debuga. Emita um ping estendido da porta Ethernet de uBR904-2 ao PC em uBR924-1 (endereço IP de Um ou Mais Servidores Cisco ICM NT 19.19.19.1).

```
ubr904-2#ping ip Target IP address: 19.19.19.1 !--- IP address of PC1 behind the Ethernet of
uBR924-1. Repeat count [5]: 100 !--- Sends 100 pings. Datagram size [100]: Timeout in seconds
[2]: Extended commands [n]: y Source address or interface: 18.18.18.18 !--- IP address of the
Ethernet behind uBR904-2. Type of service [0]: Set DF bit in IP header? [no]: Validate reply
data? [no]: Data pattern [0xABCD]: Loose, Strict, Record, Timestamp, Verbose[none]: Sweep range
of sizes [n]: Type escape sequence to abort. Sending 100, 100-byte ICMP Echos to 19.19.19.1,
timeout is 2 seconds:
```

O uBR924-2 mostra este resultado do debug:

```
ubr904-2#
01:50:37: IPSec(sa_request): , (key eng. msg.) src= 18.18.18.18, dest= 19.19.19.19, src_proxy=
18.18.18.0/255.255.255.0/0/0 (type=4), dest_proxy= 19.19.19.0/255.255.255.0/0/0 (type=4),
protocol= AH, transform= ah-md5-hmac , lifedur= 3600s and 4608000kb, spi= 0x19911A16(428939798),
conn_id= 0, keysize= 0, flags= 0x4004 01:50:37: IPSec(sa_request): , (key Eng. msg.) src=
18.18.18.18, dest= 19.19.19.19, src_proxy= 18.18.18.0/255.255.255.0/0/0 (type=4), dest_proxy=
19.19.19.0/255.255.255.0/0/0 (type=4), protocol= ESP, transform= ESP-Des , lifedur= 3600s and
4608000kb, spi= 0x7091981(118036865), conn_id= 0, keysize= 0, flags= 0x4004 01:50:37: ISAKMP:
received ke message (1/2) 01:50:37: ISAKMP (0:1): sitting IDLE. Starting QM immediately
(QM_IDLE) 01:50:37: ISAKMP (0:1): beginning Quick Mode exchange, M-ID of 1108017901 01:50:37:
CryptoEngine0: generate hmac context for conn id 1 01:50:37: ISAKMP (1): sending packet to
19.19.19.19 (I) QM_IDLE 01:50:37: ISAKMP (1): received packet from 19.19.19.19 (I) QM_IDLE
01:50:37: CryptoEngine0: generate hmac context for conn id 1 01:50:37: ISAKMP (0:1): processing
SA payload. message ID = 1108017901 01:50:37: ISAKMP (0:1): Checking IPSec proposal 1 01:50:37:
ISAKMP: transform 1, AH_MD5 01:50:37: ISAKMP: attributes in transform:
01:50:3.!!!!!!!!!!!!!!!!!!!!!!!!7: ISAKMP: encaps is 1 01:50:37: ISAKMP: SA life type in seconds
01:50:37: ISAKMP: SA life duration (basic) of 3600 01:50:37: ISAKMP: SA life type in kilobytes
01:50:37: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0 01:50:37: ISAKMP: authenticator is
HMAC-MD5 01:50:37: validate proposal 0 01:50:37: ISAKMP (0:1): atts are acceptable. 01:50:37:
ISAKMP (0:1): Checking IPSec proposal 1 01:50:37: ISAKMP: transform 1, ESP_DES 01:50:37: ISAKMP:
attributes in transform: 01:50:37: ISAKMP: encaps is 1 01:50:37: ISAKMP: SA life type in seconds
01:50:37: ISAKMP: SA life duration (basic) of 3600 01:50:37: ISAKMP: SA life type in kilobytes
01:50:37: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0 01:50:37: validate proposal 0
01:50:37: ISAKMP (0:1): atts are acceptable. 01:50:37: IPSec(validate_proposal_request):
proposal part #1, (key Eng. msg.) dest= 19.19.19.19, src= 18.18.18.18, dest_proxy=
19.19.1!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! !!!!!!!!!!!!!!!!!!!!!!!!!!!! Success
rate is 99 percent (99/100), round-trip min/avg/max = 30/40/70 ms ubr904-2#
```

Observe que o primeiro sibilo falhou. Isto é porque precisa de estabelecer a conexão.

O uBR924-1 mostra este resultado do debug:

```
ubr924-1#
01:50:24: ISAKMP (1): received packet from 18.18.18.18 (R) QM_IDLE 01:50:24: CryptoEngine0:
generate hmac context for conn id 1 01:50:24: ISAKMP (0:1): processing SA payload. Message ID =
1108017901 01:50:24: ISAKMP (0:1): Checking IPSec proposal 1 01:50:24: ISAKMP: transform 1,
AH_MD5 01:50:24: ISAKMP: attributes in transform: 01:50:24: ISAKMP: encaps is 1 01:50:24:
ISAKMP: SA life type in seconds 01:50:24: ISAKMP: SA life duration (basic) of 3600 01:50:24:
ISAKMP: SA life type in kilobytes 01:50:24: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0
01:50:24: ISAKMP: authenticator is HMAC-MD5 01:50:24: validate proposal 0 01:50:24: ISAKMP
(0:1): atts are acceptable. 01:50:24: ISAKMP (0:1): Checking IPSec proposal 1 01:50:24: ISAKMP:
```

transform 1, ESP_DES 01:50:24: ISAKMP: attributes in transform: 01:50:24: ISAKMP: encaps is 1 01:50:24: ISAKMP: SA life type in seconds 01:50:24: ISAKMP: SA life duration (basic) of 3600 01:50:24: ISAKMP: SA life type in kilobytes 01:50:24: ISAKMP: SA life duration (VPI) of 0x0 0x46 0x50 0x0 01:50:24: **validate proposal 0** 01:50:24: ISAKMP (0:1): atts are acceptable. 01:50:24: IPSec**(validate_proposal_request)**: proposal part #1, (key Eng. msg.) **dest= 19.19.19.19, src= 18.18.18.18,** dest_proxy= 19.19.19.0/255.255.255.0/0/0 (type=4), src_proxy= 18.18.18.0/255.255.255.0/0/0 (type=4), **protocol= AH, transform= ah-md5-hmac** , lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4 01:50:24: IPSec(validate_proposal_request): proposal part #2, (key Eng. msg.) **dest= 19.19.19.19, src= 18.18.18.18**, dest_proxy= 19.19.19.0/255.255.255.0/0/0 (type=4), src_proxy= 18.18.18.0/255.255.255.0/0/0 (type=4), **protocol= ESP, transform= ESP-Des** , lifedur= 0s and 0kb, spi= 0x0(0), conn_id= 0, keysize= 0, flags= 0x4 01:50:24: validate proposal request 0 01:50:24: ISAKMP (0:1): processing NONCE payload. Message ID = 1108017901 01:50:24: ISAKMP (0:1): processing ID payload. Message ID = 1108017901 01:50:24: ISAKMP (1): ID_IPV4_ADDR_SUBNET src 18.18.18.0/255.255.255.0 prot 0 Port 0 01:50:24: ISAKMP (0:1): processing ID payload. Message ID = 1108017901 01:50:24: ISAKMP (1): ID_IPV4_ADDR_SUBNET dst 19.19.19.0/255.255.255.0 prot 0 Port 0 01:50:24: **ISAKMP (0:1): asking for 2 spis from IPSec** 01:50:24: IPSec(key_engine): got a queue event... 01:50:24: IPSec(spi_response): getting spi 393021796 for SA from 18.18.18.18 to 19.19.19.19 for prot 2 01:50:24: IPSec(spi_response): getting spi 45686884 for SA from 18.18.18.18 to 19.19.19.19 for prot 3 01:50:24: **ISAKMP: received ke message (2/2)** 01:50:24: CryptoEngine0: generate hmac context for conn id 1 01:50:24: ISAKMP (1): sending packet to 18.18.18.18 (R) QM_IDLE 01:50:24: ISAKMP (1): received packet from 18.18.18.18 (R) QM_IDLE 01:50:24: **CryptoEngine0: generate hmac context for conn id 1** 01:50:24: IPSec allocate flow 0 01:50:24: IPSec allocate flow 0 01:50:24: **ISAKMP (0:1): Creating IPSec SAs** 01:50:24: **inbound SA from 18.18.18.18 to 19.19.19.19 (proxy 18.18.18.0 to 19.19.19.0)** 01:50:24: has spi 393021796 and conn_id 2000 and flags 4 01:50:24: lifetime of 3600 seconds 01:50:24: lifetime of 4608000 kilobytes 01:50:24: **outbound SA from 19.19.19.19 to 18.18.18.18 (proxy 19.19.19.0 to 18.18.18.0)** 01:50:24: has spi 428939798 and conn_id 2001 and flags 4 01:50:24: lifetime of 3600 seconds 01:50:24: lifetime of 4608000 kilobytes 01:50:24: **ISAKMP (0:1): Creating IPSec SAs** 01:50:24: **inbound SA from 18.18.18.18 to 19.19.19.19 (proxy 18.18.18.0 to 19.19.19.0)** 01:50:24: has spi 45686884 and conn_id 2002 and flags 4 01:50:24: lifetime of 3600 seconds 01:50:24: lifetime of 4608000 kilobytes 01:50:24: **outbound SA from 19.19.19.19 to 18.18.18.18 (proxy 19.19.19.0 to 18.18.18.0)** 01:50:24: has spi 118036865 and conn_id 2003 and flags 4 01:50:25: lifetime of 3600 seconds 01:50:25: lifetime of 4608000 kilobytes 01:50:25: ISAKMP (0:1): deleting node 1108017901 error FALSE reason "quick mode done (await()" 01:50:25: **IPSec(key_engine): got a queue event...** 01:50:25: **IPSec(initialize_sas):** , (key Eng. msg.) **dest= 19.19.19.19, src= 18.18.18.18**, dest_proxy= 19.19.19.0/255.255.255.0/0/0 (type=4), src_proxy= 18.18.18.0/255.255.255.0/0/0 (type=4), **protocol= AH, transform= ah-md5-hmac** , lifedur= 3600s and 4608000kb, spi= 0x176D0964(393021796), **conn_id= 2000**, keysize= 0, flags= 0x4 01:50:25: **IPSec(initialize_sas):** , (key Eng. msg.) **src= 19.19.19.19, dest= 18.18.18.18**, src_proxy= 19.19.19.0/255.255.255.0/0/0 (type=4), dest_proxy= 18.18.18.0/255.255.255.0/0/0 (type=4), **protocol= AH, transform= ah-md5-hmac** , lifedur= 3600s and 4608000kb, spi= 0x19911A16(428939798), **conn_id= 2001**, keysize= 0, flags= 0x4 01:50:25: **IPSec(initialize_sas):** , (key Eng. msg.) dest= 19.19.19.19, src= 18.18.18.18, dest_proxy= 19.19.19.0/255.255.255.0/0/0 (type=4), src_proxy= 18.18.18.0/255.255.255.0/0/0 (type=4), **protocol= ESP, transform= ESP-Des** , lifedur= 3600s and 4608000kb, spi= 0x2B92064(45686884), **conn_id= 2002**, keysize= 0, flags= 0x4 01:50:25: **IPSec(initialize_sas):** , (key Eng. msg.) **src= 19.19.19.19, dest= 18.18.18.18**, src_proxy= 19.19.19.0/255.255.255.0/0/0 (type=4), dest_proxy= 18.18.18.0/255.255.255.0/0/0 (type=4), **protocol= ESP, transform= ESP-Des** , lifedur= 3600s and 4608000kb, spi= 0x7091981(118036865), **conn_id= 2003**, keysize= 0, flags= 0x4 01:50:25: IPSec(create_sa): sa created, (sa) sa_dest= 19.19.19.19, sa_prot= 51, sa_spi= 0x176D0964(393021796), sa_trans= ah-md5-hmac , sa_conn_id= 2000 01:50:25: IPSec(create_sa): sa created, (sa) sa_dest= 18.18.18.18, sa_prot= 51, sa_spi= 0x19911A16(428939798), sa_trans= ah-md5-hmac , sa_conn_id= 2001 01:50:25: IPSec(create_sa): sa created, (sa) sa_dest= 19.19.19.19, sa_prot= 50, sa_spi= 0x2B92064(45686884), sa_trans= ESP-Des , sa_conn_id= 2002 01:50:25: IPSec(create_sa): sa created, (sa) sa_dest= 18.18.18.18, sa_prot= 50, sa_spi= 0x7091981(118036865), sa_trans= ESP-Des , sa_conn_id= 2003 ubr924-1#

Uma vez que o túnel de IPsec é criado, você pode ver a conexão e os pacotes criptografado e decriptografado.

ubr924-1#**show crypto engine connection active** ID Interface IP-Address State Algorithm Encrypt Decrypt 1 set HMAC_MD5+DES_56_CB 0 0 2000 cable-modem0 172.16.31.20 set **HMAC_MD5 0 99** 2001 cable-modem0 172.16.31.20 set HMAC_MD5 **99 0** 2002 cable-modem0 172.16.31.20 set **DES_56_CBC 0 99** 2003 cable-modem0 172.16.31.20 set DES_56_CBC **99 0**

A primeira linha 200x mostra os 99 pacotes recebidos. Tem que decifrar os pacotes a fim enviá-los ao PC1. A segunda linha mostra 99 pacotes enviados. Tem que cifrar os pacotes antes que os envie a uBR904-2. As terceiras e quartas linhas fazem o mesmo processo, mas com ESP-DES transformam-no em vez do AH-MD5-HMAC.

Nota: Se a transformação se ajustou que é configurada no modem a cabo é ESP-DES ESP-MD5-HMAC, você considera somente dois sistemas autônomo (AS), ao contrário dos quatro mostrados no **comando show** precedente.

```
ubr904-2#show crypto engine connection active ID Interface IP-Address State Algorithm Encrypt
Decrypt 1 set HMAC_MD5+DES_56_CB 0 0 2000 cable-modem0 172.16.30.18 set HMAC_MD5 0 99 2001
cable-modem0 172.16.30.18 set HMAC_MD5 99 0 2002 cable-modem0 172.16.30.18 set DES_56_CBC 0 99
2003 cable-modem0 172.16.30.18 set DES_56_CBC 99 0
```

Emita um ping estendido ao PC2 de uBR924-1 para ver se os contadores incrementam para os pacotes criptografado e decriptografado.

```
ubr924-1#ping ip Target IP address: 18.18.18.1 Repeat count [5]: 50 Datagram size [100]: Timeout
in seconds [2]: Extended commands [n]: y Source address or interface: 19.19.19.19 Type of
service [0]: Set DF bit in IP header? [no]: Validate reply data? [no]: Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]: Sweep range of sizes [n]: Type escape sequence
to abort. Sending 50, 100-byte ICMP Echos to 18.18.18.1, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! Success rate is 100 percent (50/50), round-
trip min/avg/max = 28/30/33 ms ubr924-1#show crypto engine connection active ID Interface IP-
Address State Algorithm Encrypt Decrypt 1 set HMAC_MD5+DES_56_CB 0 0 2000 cable-modem0
172.16.31.20 set HMAC_MD5 0 149 2001 cable-modem0 172.16.31.20 set HMAC_MD5 149 0 2002 cable-
modem0 172.16.31.20 set DES_56_CBC 0 149 2003 cable-modem0 172.16.31.20 set DES_56_CBC 149 0
ubr904-2#show crypto engine connection active ID Interface IP-Address State Algorithm Encrypt
Decrypt 1 set HMAC_MD5+DES_56_CB 0 0 2000 cable-modem0 172.16.30.18 set HMAC_MD5 0 149 2001
cable-modem0 172.16.30.18 set HMAC_MD5 149 0 2002 cable-modem0 172.16.30.18 set DES_56_CBC 0 149
2003 cable-modem0 172.16.30.18 set DES_56_CBC 149 0
```

Um outro ping estendido pode ser emitido, para considerar que os contadores incrementam outra vez. Esta vez, envia um sibilo 500-packet de uBR904-2 à interface Ethernet de uBR924-1 (19.19.19.19).

```
ubr904-2#ping ip Target IP address: 19.19.19.19 Repeat count [5]: 500 Datagram size [100]: 1000
Timeout in seconds [2]: Extended commands [n]: y Source address or interface: 18.18.18.18 Type
of service [0]: Set DF bit in IP header? [no]: Validate reply data? [no]: Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]: Sweep range of sizes [n]: Type escape sequence
to abort. Sending 500, 1000-byte ICMP Echos to 19.19.19.19, timeout is 2 seconds:
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! 01:59:06: IPSec(encapsulate):
encaps area too small, moving to new buffer: idbtype 0, encaps_size 26, header size 60, avail
84!!!!!!!!! !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! !!!!!!!!!! Success rate
is 100 percent (500/500), round-trip min/avg/max = 98/135/352 ms ubr904-2#show crypto engine
connection active ID Interface IP-Address State Algorithm Encrypt Decrypt 1 set
HMAC_MD5+DES_56_CB 0 0 2000 cable-modem0 172.16.30.18 set HMAC_MD5 0 649 2001 cable-modem0
172.16.30.18 set HMAC_MD5 649 0 2002 cable-modem0 172.16.30.18 set DES_56_CBC 0 649 2003 cable-
modem0 172.16.30.18 set DES_56_CBC 649 0 ubr924-1#show crypto engine connection active ID
Interface IP-Address State Algorithm Encrypt Decrypt 1 set HMAC_MD5+DES_56_CB 0 0 2000 cable-
modem0 172.16.31.20 set HMAC_MD5 0 649 2001 cable-modem0 172.16.31.20 set HMAC_MD5 649 0 2002
cable-modem0 172.16.31.20 set DES_56_CBC 0 649 2003 cable-modem0 172.16.31.20 set DES_56_CBC 649
0
```

Você pode emitir os **comandos clear crypto isakmp e clear crypto sa** cancelar as conexões. Também, se há um sem tráfego através do túnel de IPsec durante o tempo de expiração, o IPsec restaura a conexão automaticamente.

Troubleshooting

Não há atualmente nenhuma informações disponíveis específica para pesquisar defeitos esta configuração.

Informações Relacionadas

- Comandos da Segurança de rede IPSec
- Uma introdução à criptografia do protocolo de segurança IP (IPSEC) - Debugar a informação
- Exemplos da configuração IPSec
- Configurando a Segurança de rede IPSec
- Configurando o Roteadores de acesso por cabo Cisco série uBR900
- Cabo Cisco/transferências de faixa larga (clientes registrados somente)
- Suporte por tecnologia da Banda larga a cabo
- Suporte Técnico e Documentação - Cisco Systems