IPSec over Cable 샘플 컨피그레이션 및 디버그

목차

소개

IPsec(Internet Protocol Security)는 IP 네트워크를 통해 안전한 개인 통신을 보장하는 개방형 표준의 프레임워크입니다.IETF(Internet Engineering Task Force)에서 개발한 표준에 따라 IPsec은 공용 IP 네트워크 전체에서 데이터 통신의 기밀성, 무결성 및 신뢰성을 보장합니다.IPsec은 네트워크 전체의 보안 정책을 구축하기 위한 표준 기반의 유연한 솔루션에 필요한 구성 요소를 제공합니다.

이 문서에서는 두 Cisco 케이블 모뎀 간의 IPsec 구성 예를 제공합니다.이 컨피그레이션은 두 Cisco uBR9xx Series 케이블 모뎀 라우터 간의 케이블 네트워크 전체에서 암호화 터널을 생성합니다.두 네트워크 간의 모든 트래픽은 암호화됩니다.그러나 다른 네트워크로 향하는 트래픽은 암호화되지 않은 상태로 전달될 수 있습니다.소규모 사무실, 홈 오피스(SOHO) 사용자의 경우 케이블 네트워크에 VPN(가상 사설망)을 생성할 수 있습니다.

사전 요구 사항

요구 사항

이 문서에 대한 특정 요건이 없습니다.

사용되는 구성 요소

모뎀은 두 개의 케이블 모뎀에서 IPsec을 구성하려면 다음 요구 사항을 준수해야 합니다.

- 라우팅 모드의 Cisco uBR904, uBR905 또는 uBR924
- IPsec 56 기능 집합
- Cisco IOS® 소프트웨어 릴리스 12.0(5)T 이상

또한 Cisco uBR7246, Cisco uBR7223 또는 Cisco uBR7246VXR 등 DOCSIS(Data-over-Cable Service Interface Specifications) 호환 헤드엔드 케이블 라우터인 CMTS(Cable Modem Termination System)가 있어야 합니다.

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다.이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다.현재 네트워크가 작동 중인 경우, 모든 명령어의 잠재적인 영향을 미리 숙지하시기 바랍니다.

표기 규칙

문서 규칙에 대한 자세한 내용은 Cisco 기술 팁 표기 규칙을 참고하십시오.

배경 이론

이 문서의 예에서는 uBR904 케이블 모뎀, uBR924 케이블 모뎀 및 uBR7246VXR CMTS를 사용합니다.케이블 모뎀은 Cisco IOS Software Release 12.1(6)을 실행하고 CMTS는 Cisco IOS Software Release 12.1(4)EC를 실행합니다.

참고: 이 예는 콘솔 포트를 통해 케이블 모뎀의 수동 컨피그레이션을 수행합니다.DOCSIS 구성 파일(ios.cfg 스크립트는 IPsec 컨피그레이션으로 작성됨)을 통해 자동화된 프로세스를 수행하면 액세스 목록 100 및 101을 사용할 수 없습니다.이는 Cisco가 SNMP(Simple Network Management Protocol) docsDevNmAccess 테이블을 구현하면 Cisco IOS 액세스 목록이 사용되기 때문입니다.인터페이스당 하나의 액세스 목록을 생성합니다.uBR904, 924 및 905에서는 처음 두 개의 액세스 목록이 일반적으로 사용됩니다(100 및 101). CVA120과 같이 범용 직렬 버스(USB)를 지원하는 케이블 모뎀에서 3개의 액세스 목록(100, 101, 102)이 사용됩니다.

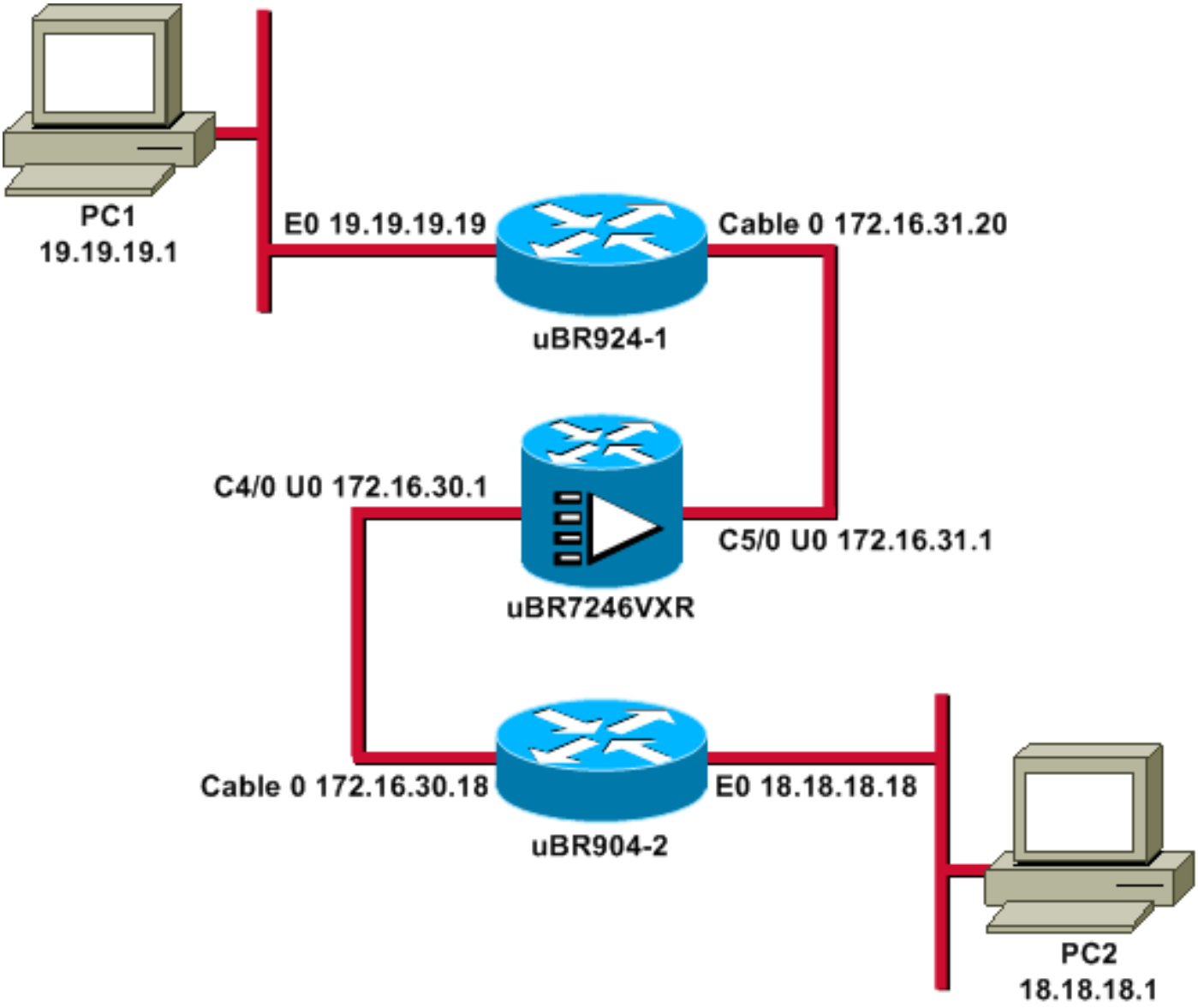
구성

이 섹션에서는 이 문서에 설명된 기능을 구성하는 정보를 제공합니다.

참고: 명령 조회 도구(등록된 고객만 해당)를 사용하여 이 문서의 명령에 대한 추가 정보를 찾습니다.

네트워크 다이어그램

이 문서에서는 다음 네트워크 설정을 사용합니다.



참고: 이 다이어그램의 모든 IP 주소에는 24비트 마스크가 있습니다.

구성

이 문서에서는 다음 구성을 사용합니다.

- 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
```

다른 케이블 모뎀의 컨피그레이션이 매우 비슷하므로 이전 컨피그레이션의 대부분의 코멘트는 생략됩니다.

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
```

CMTS uBR7246VXR는 라우팅 기능이 작동하도록 RIP(Routing Information Protocol) 버전 2도 실행합니다.다음은 CMTS에서 사용되는 RIP 컨피그레이션입니다.

uBR7246VXR
router rip version 2 network 172.16.0.0 no auto-summary

다음을 확인합니다.

이 섹션을 사용하여 컨피그레이션이 제대로 작동하는지 확인합니다.

IPsec이 작동하는지 확인하려면 다음을 수행합니다.

- 다음 사항을 확인합니다.Cisco IOS 소프트웨어는 IPsec을 지원합니다.실행 중인 구성이 올바릅니다.인터페이스가 작동 중입니다.라우팅이 작동합니다.트래픽을 암호화하기 위해 정의된 액세스 목록이 정확합니다.
- 트래픽을 생성하고 Encrypt and Decrypt(암호화 및 암호 해독)를 확인하여 증가하는 양을 확인합니다.
- crypto의 디버그를 켜십시오.

Output Interpreter 도구(등록된 고객만 해당)(OIT)는 특정 **show** 명령을 지원합니다.OIT를 사용하여 **show** 명령 출력의 분석을 봅니다.

두 케이블 모뎀에서 **show version** 명령을 실행합니다.

```
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
```

uBR924-1은 Small OFFICE/VOICE/FW IPSec 56 기능 세트와 함께 Cisco IOS Software 릴리스 12.1(6)을 실행합니다.

```
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
```

uBR904-2는 Cisco IOS Software 릴리스 12.1(6)과 SMALL OFFICE/FW IPSec 56 기능 세트를 실행합니다.

```
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
```

마지막 명령에서 이더넷 인터페이스가 작동 중임을 확인할 수 있습니다.이더넷 인터페이스의 IP 주소를 수동으로 입력했습니다.케이블 인터페이스도 작동하며 DHCP를 통해 IP 주소를 학습했습니다.이러한 케이블 주소는 동적으로 할당되므로 IPSec 컨피그레이션에서 피어로 사용할 수 없습니다.

```
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
```

uBR924-1에서 uBR904-2의 이더넷 인터페이스인 경로 18.18.18.0에 대해 학습하고 있는 것을 볼 수 있습니다.

```
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
```

uBR904-2 라우팅 테이블에서 uBR924-1 이더넷 네트워크가 라우팅 테이블에 있음을 확인할 수 있습니다.

참고: 두 케이블 모뎀 간에 라우팅 프로토콜을 실행할 수 없는 경우가 있을 수 있습니다.이러한 경우 케이블 모뎀의 이더넷 인터페이스에 대한 트래픽을 다이렉트하려면 CMTS에 고정 경로를 추가해야 합니다.

다음은 액세스 목록의 인증입니다.두 라우터에서 **show access-lists** 명령을 실행합니다.

```
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)
```

uBR924-1(19.19.19.0) 뒤의 LAN이 uBR904-2(18.18.18.0) 뒤의 LAN으로 IP 트래픽을 전송할 때 액세스 목록은 IPsec 세션을 설정합니다.액세스 목록에서 "any"를 사용하지 *마십시오*. 문제가 발생하기 때문입니다.자세한 내용은 IPsec 네트워크 보안 구성을 참조하십시오.

IPsec 트래픽이 없습니다.show crypto **engine connection active** 명령을 실행합니다.

```
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
```

액세스 목록과 일치하는 트래픽이 없으므로 IPsec 연결이 없습니다.

참고: debug **명령**을 사용하기 전에 디버그 명령에 대한 중요 정보를 참조하십시오.

다음 단계는 일부 암호화 디버그를 활성화하여 흥미로운 트래픽을 생성하는 것입니다.

이 예에서는 다음 디버그가 켜집니다.

- **디버그 암호화 엔진**
- **디버그 암호화 IPsec**
- **디버그 암호화 키 교환**
- **디버그 암호화 isakmp**

먼저 디버깅 출력을 보려면 흥미로운 트래픽을 생성해야 합니다.uBR904-2의 이더넷 포트에서 uBR924-1(IP 주소 19.19.19.1)의 PC로 확장된 ping을 실행합니다.

```
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:
```

uBR924-2는 다음 디버그 출력을 보여 줍니다.

```
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#
```

첫 번째 ping이 실패했음을 확인합니다.연결을 설정해야 하기 때문입니다.

uBR924-1은 다음 디버그 출력을 보여 줍니다.

```
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#
```

IPsec 터널이 생성되면 연결 및 암호화된/또는 해독된 패킷을 볼 수 있습니다.

```
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
```

첫 번째 200x 행은 수신된 패킷 99개를 보여줍니다.PC1로 전송하려면 패킷을 해독해야 합니다. 두 번째 행은 99개의 전송된 패킷을 보여줍니다.uBR904-2로 전송하기 전에 패킷을 암호화해야 합니다. 세 번째 및 네 번째 행은 동일한 프로세스를 수행하지만 AH-MD5-HMAC 대신 ESP-DES 변환을 사용합니다.

참고: 케이블 모뎀에 구성된 변형 집합이 ESP-DES ESP-MD5-HMAC인 경우 이전 **show** 명령에 표시된 4개와 달리 두 개의 자동 시스템(AS)만 표시됩니다.

```
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
```

uBR924-1에서 PC2에 대한 확장 ping을 실행하여 카운터가 암호화된 패킷과 해독된 패킷에 대해 증가하는지 확인합니다.

```
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
```

카운터가 다시 증가하는지 확인하기 위해 다른 확장 ping을 실행할 수 있습니다.이번에는 uBR904-2에서 uBR924-1(19.19.19.19)의 이더넷 인터페이스로 500-packet ping을 전송합니다.

```
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
```

clear crypto isakmp 및 **clear crypto sa** 명령을 실행하여 연결을 지울 수 있습니다.또한 만료 시간 동안 IPsec 터널을 통해 트래픽이 없는 경우 IPsec은 연결을 자동으로 재설정합니다.

문제 해결

현재 이 컨피그레이션의 문제를 해결하는 데 사용할 수 있는 특정 정보가 없습니다.

관련 정보

- IPsec 네트워크 보안 명령
- IPsec(IP Security) 암호화 소개 - 디버그 정보
- IPsec 컨피그레이션 예
- IPsec 네트워크 보안 구성
- Cisco uBR900 Series Cable Access Router 구성
- Cisco 케이블/광대역 다운로드(등록된 고객만 해당)
- 광대역 케이블 기술 지원
- 기술 지원 및 문서 – Cisco Systems