여러 인증서가 있는 프로파일에 대한 IOS IKEv1 및 IKEv2 패킷 교환 프로세스

목차

소개

이 문서에서는 인증서 인증이 사용되는 경우 IKEv1(Internet Key Exchange Version 1) 및 IKEv2(Internet Key Exchange Version 2) 패킷 교환 프로세스에 대해 설명하고 발생할 수 있는 문제를 설명합니다.

이 문서에 설명된 주제 목록은 다음과 같습니다.

- IKE(Internet Key Exchange) 개시자 및 IKE 응답자에 대한 인증서 선택 기준
- 여러 IKE 프로파일이 일치하는 경우(중복 및 비중복 시나리오의 경우) IKE 프로파일 일치 기준
- IKE 프로파일에서 신뢰 지점을 사용하지 않는 경우 기본 설정 및 동작
- 프로필 및 인증서 선택 기준과 관련하여 IKEv1과 IKEv2 간의 차이점

> **참고:**특정 문제를 해결하는 방법에 대한 자세한 내용은 올바른 섹션을 참조하십시오.또한 이 문서의 끝에 간단한 요약이 제공됩니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- Cisco IOS® VPN 컨피그레이션
- IKEv1 및 IKEv2 프로토콜(패킷 교환)

사용되는 구성 요소

이 문서의 정보는 Cisco IOS Version15.3T를 기반으로 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.

배경 정보

이 문서에서 설명하는 문제는 여러 신뢰 지점 및 여러 IKE 프로파일을 사용할 때 발생합니다.

이 문서에서 사용되는 초기 예에는 각 라우터에 두 개의 신뢰 지점이 있는 IKEv1 LAN-to-LAN 터널이 있습니다.처음에는 컨피그레이션이 올바른 것 같습니다.그러나 VPN 터널은 **ca trust-point** 명령이 ISAKMP(Internet Security Association and Key Management Protocol) 프로필 동작 및 로컬 저장소의 등록된 인증서 순서에 대해 사용되는 방식으로 인해 연결의 한 쪽에서만 시작할 수 있습니다.

라우터가 ISAKMP 개시자인 경우 ISAKMP 프로파일에 대해 **ca trust-point** 명령을 사용하여 다른 동작이 구성됩니다.ISAKMP 이니시에이터가 처음부터 ISAKMP 프로필을 인식하므로 프로필에 대해 구성된 **ca trust-point** 명령이 MM3(Main Mode Packet 3)에서 인증서 요청에 대한 페이로드에 영향을 줄 수 있습니다. 그러나 라우터가 ISAKMP responder인 경우, 바인딩을 생성하는 데 필요한 IKE ID를 포함하는 MM5(Main Mode Packet 5)를 수신한 후 인바운드 트래픽을 특정 ISAKMP 프로필에 바인딩합니다.따라서 MM5보다 먼저 프로파일이 결정되지 않으므로 MM4(Main Mode Packet 4) 패킷에 **ca trust-point** 명령을 적용할 수 없습니다.

MM3 및 MM4에서 인증서 요청 페이로드의 순서 및 전체 협상 프로세스에 미치는 영향에 대해 설명하고, VPN 터널의 한 쪽에서만 연결을 설정할 수 있도록 하는 이유에 대해 설명합니다.

다음은 IKEv1 개시자 및 responder 동작의 요약입니다.

	IKEv1 개시자	IKEv1 응답자
요청 보내기	프로필에 구성된 신뢰 지점에 대해서만 특정 요청을 보냅니다.	사용 가능한 모든 신뢰 지점에 대한 요청을 보냅니다.
요청 검증	프로필에 구성된 특정 신뢰 지점에 대해 검증	프로필에 구성된 특정 신뢰 지점에 대해 검증

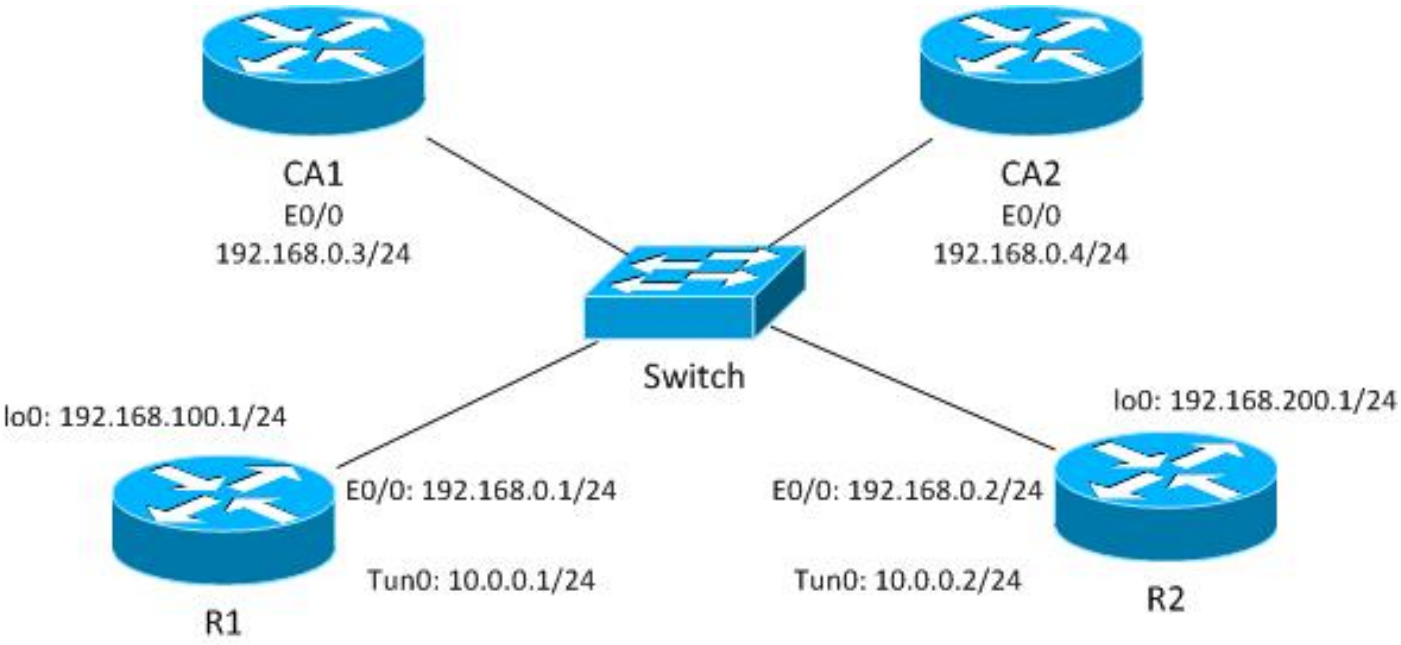
Cisco에서는 여러 ISAKMP 프로파일이 있고 전역적으로 구성된 신뢰 지점을 사용하는 ISAKMP 응답자에 대해 ca **trust-point** 명령을 사용하지 않는 것이 좋습니다.여러 ISAKMP 프로필이 있는 ISAKMP 이니시에이터의 경우 각 프로필에서 ca **trust-point** 명령을 사용하여 인증서 선택 프로세스를 좁히는 것이 좋습니다.

IKEv2 프로토콜은 IKEv1 프로토콜과 동일한 문제를 가지지만 pki trustpoint 명령의 서로 다른 동작을 사용하면 문제가 발생하지 않습니다.이는 **pki trustpoint** 명령이 IKEv2 개시자에 필수이고 **ca trust-point** 명령은 IKEv1 개시자에 대해 선택 사항이기 때문입니다.경우에 따라(한 프로파일의 여러 신뢰 지점) 이전에 설명한 문제가 발생할 수 있습니다.따라서 Cisco에서는 연결의 양쪽에 대칭 신뢰 지점 컨피그레이션을 사용하는 것이 좋습니다(IKEv2 프로파일 둘 다에 구성된 동일한 신뢰 지점).

토폴로지

이 문서의 모든 예에 사용되는 일반 토폴로지입니다.

> **참고:**라우터 1(R1) 및 라우터 2(R2)는 VTI(가상 터널 인터페이스)를 사용하여 루프백에 액세스합니다.이러한 VTI는 IPSec에 의해 보호됩니다.



이 IKEv1 예에서는 각 라우터에 각 CA(Certificate Authority)에 대한 두 개의 신뢰 지점이 있으며 각 신뢰 지점에 대한 인증서가 등록됩니다.

R1이 ISAKMP 개시자인 경우 터널이 올바르게 협상되고 트래픽이 보호됩니다.이는 예상 동작입니다.R2가 ISAKMP 개시자인 경우 1단계 협상이 실패합니다.

참고:이 문서의 IKEv2 예제의 경우 토폴로지 및 주소 지정은 IKEv1 예와 동일합니다.

패킷 교환 프로세스

이 섹션에서는 패킷 교환 프로세스에 사용되는 IKEv1 및 IKEv2 컨피그레이션 변형 및 발생할 수 있는 문제에 대해 설명합니다.

여러 인증서가 있는 IKEv1

다음은 여러 인증서가 있는 IKEv1에 대한 R1 네트워크 및 VPN 컨피그레이션입니다.

```
crypto isakmp policy 10
 encr 3des
 hash md5
 group 2

crypto isakmp profile prof1
   self-identity fqdn
   ca trust-point IOSCA1
   match identity host R2.cisco.com
!
crypto ipsec transform-set TS esp-aes esp-sha256-hmac
 mode tunnel
!
crypto ipsec profile prof1
 set transform-set TS
 set isakmp-profile prof1
!
interface Loopback0
 description Simulate LAN
 ip address 192.168.100.1 255.255.255.0
!
interface Tunnel1
 ip address 10.0.0.1 255.255.255.0
 tunnel source Ethernet0/0
 tunnel destination 192.168.0.2
 tunnel protection ipsec profile prof1
!
interface Ethernet0/0
 ip address 192.168.0.1 255.255.255.0

ip route 192.168.200.0 255.255.255.0 10.0.0.2
```

다음은 여러 인증서가 있는 IKEv1에 대한 R2 네트워크 및 VPN 컨피그레이션입니다.

```
crypto isakmp policy 10
 encr 3des
 hash md5
 group 2

crypto isakmp profile prof1
   self-identity fqdn
   match identity host R1.cisco.com
!
crypto ipsec transform-set TS esp-aes esp-sha256-hmac
 mode tunnel
!
crypto ipsec profile prof1
 set transform-set TS
 set isakmp-profile prof1
!
interface Loopback0
 ip address 192.168.200.1 255.255.255.0
!
interface Tunnel1
 ip address 10.0.0.2 255.255.255.0
 tunnel source Ethernet0/0
 tunnel destination 192.168.0.1
 tunnel protection ipsec profile prof1
!
interface Ethernet0/0
 ip address 192.168.0.2 255.255.255.0

ip route 192.168.100.0 255.255.255.0 10.0.0.1
```

이 예에서 R1에는 두 개의 신뢰 지점이 있습니다.하나는 **IOSCA1을** 사용하고 다른 하나는 IOSCA**2를 사용합니다**.

```
crypto pki trustpoint IOSCA1
 enrollment url http://192.168.0.3:80
 serial-number
 fqdn R1.cisco.com
 ip-address 192.168.0.1
 subject-name CN=R1,OU=IT,O=cisco,O=com
 revocation-check crl
!
crypto pki trustpoint IOSCA2
 enrollment url http://192.168.0.4:80
 serial-number
 fqdn R1.cisco.com
 ip-address 192.168.0.1
 subject-name CN=R1,OU=IT,O=cisco,O=com
 revocation-check crl
```

이 예에서 R2에는 두 개의 신뢰 지점이 있습니다.하나는 **IOSCA1을** 사용하고 다른 하나는 IOSCA**2를 사용합니다**.

```
crypto pki trustpoint IOSCA1
 enrollment url http://192.168.0.3:80
```

```
 serial-number
 fqdn R2.cisco.com
 ip-address 192.168.0.2
 subject-name CN=R2,OU=IT,O=cisco,O=com
 revocation-check crl
!
crypto pki trustpoint IOSCA2
 enrollment url http://192.168.0.4:80
 serial-number
 fqdn R2.cisco.com
 ip-address 192.168.0.2
 subject-name CN=R2,OU=IT,O=cisco,O=com
 revocation-check crl
```

이러한 컨피그레이션의 단일 차이점을 주목해야 합니다.R1 ISAKMP 프로필은 IOSCA1 신뢰 지점에 대해 ca **trust-point** 명령을 사용합니다. 이는 R1이 특정 신뢰 지점에서 검증한 인증서만 신뢰함을 나타냅니다.반면 R2는 전역적으로 정의된 모든 신뢰 지점에서 검증한 모든 인증서를 신뢰합니다.

R1을 IKEv1 Initiator로 사용

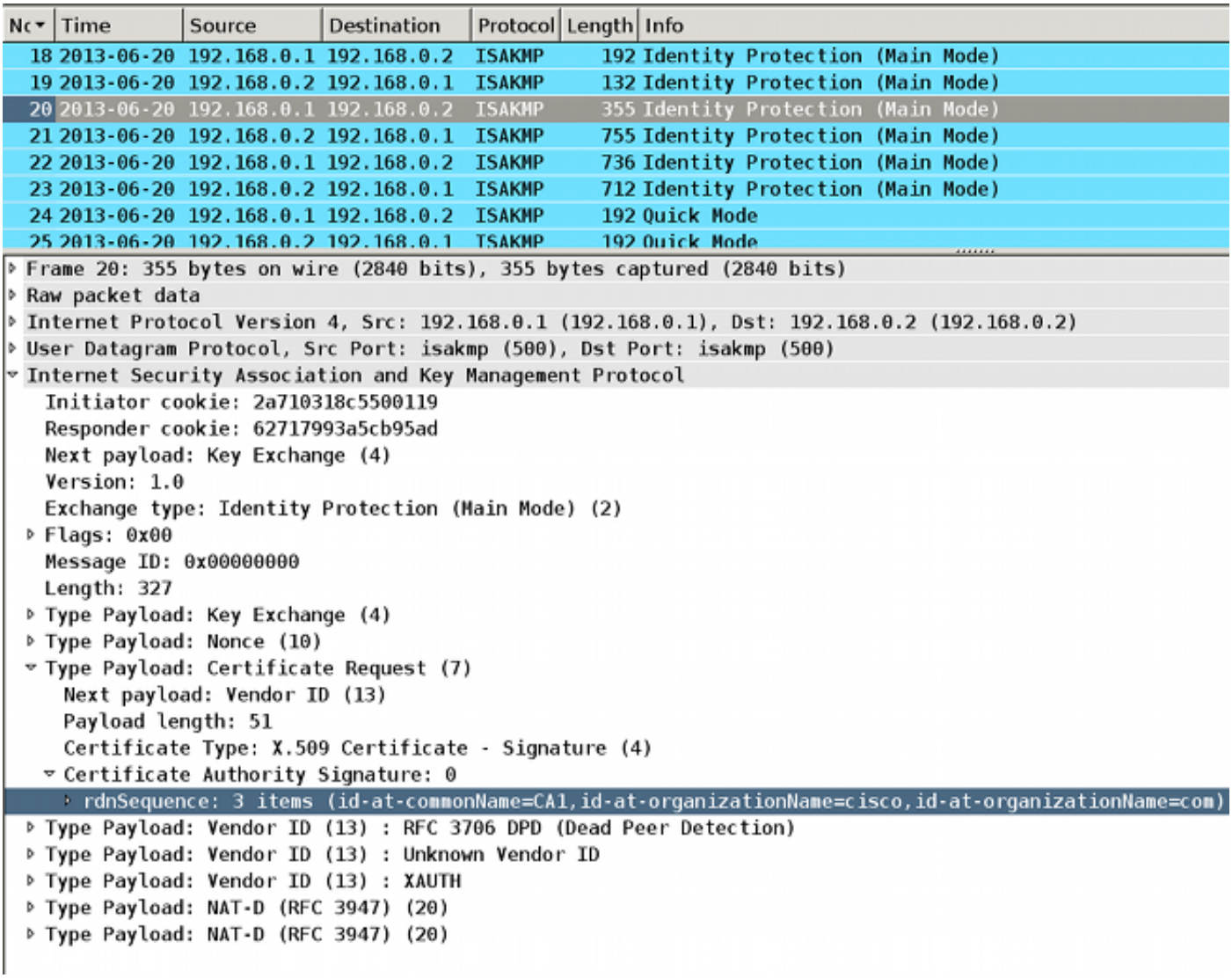
다음은 R1 및 R2에 대한 debugs 명령입니다.

- **R1# 디버그 암호화 isakmp**
- **R1# 디버그 암호화 ipsec**
- **R1# 디버그 암호화 키 유효성 검사**

여기서 R1은 터널을 시작하고 MM3에 대한 인증서를 전송합니다.

```
*Jun 20 13:00:37.609: ISAKMP:(0): SA request profile is prof1
*Jun 20 13:00:37.610: ISAKMP (0): constructing CERT_REQ for issuer
 cn=CA1,o=cisco,o=com
*Jun 20 13:00:37.610: ISAKMP:(0): sending packet to 192.168.0.2
 my_port 500 peer_port 500 (I) MM_SA_SETUP
*Jun 20 13:00:37.610: ISAKMP:(0):Old State = IKE_I_MM2  New State = IKE_I_MM3
```

패킷에 IOSCA1 신뢰 지점에 대해서만 적용되는 인증서 요청이 하나만 포함되어 있다는 **것을** 유의해야 합니다.이는 ISAKMP 프로필(**CN=CA1, O=cisco, O=com**)의 현재 컨피그레이션과 함께 예상되는 동작입니다. 다른 인증서 요청은 전송되지 않으며, Embedded Packet Capture 기능으로 확인할 수 있습니다.



No.	Time	Source	Destination	Protocol	Length	Info
18	2013-06-20	192.168.0.1	192.168.0.2	ISAKMP	192	Identity Protection (Main Mode)
19	2013-06-20	192.168.0.2	192.168.0.1	ISAKMP	132	Identity Protection (Main Mode)
20	2013-06-20	192.168.0.1	192.168.0.2	ISAKMP	355	Identity Protection (Main Mode)
21	2013-06-20	192.168.0.2	192.168.0.1	ISAKMP	755	Identity Protection (Main Mode)
22	2013-06-20	192.168.0.1	192.168.0.2	ISAKMP	736	Identity Protection (Main Mode)
23	2013-06-20	192.168.0.2	192.168.0.1	ISAKMP	712	Identity Protection (Main Mode)
24	2013-06-20	192.168.0.1	192.168.0.2	ISAKMP	192	Quick Mode
25	2013-06-20	192.168.0.2	192.168.0.1	ISAKMP	192	Quick Mode

```
▸ Frame 20: 355 bytes on wire (2840 bits), 355 bytes captured (2840 bits)
▸ Raw packet data
▸ Internet Protocol Version 4, Src: 192.168.0.1 (192.168.0.1), Dst: 192.168.0.2 (192.168.0.2)
▸ User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
▾ Internet Security Association and Key Management Protocol
    Initiator cookie: 2a710318c5500119
    Responder cookie: 62717993a5cb95ad
    Next payload: Key Exchange (4)
    Version: 1.0
    Exchange type: Identity Protection (Main Mode) (2)
  ▸ Flags: 0x00
    Message ID: 0x00000000
    Length: 327
  ▸ Type Payload: Key Exchange (4)
  ▸ Type Payload: Nonce (10)
  ▾ Type Payload: Certificate Request (7)
      Next payload: Vendor ID (13)
      Payload length: 51
      Certificate Type: X.509 Certificate - Signature (4)
    ▾ Certificate Authority Signature: 0
      ▸ rdnSequence: 3 items (id-at-commonName=CA1,id-at-organizationName=cisco,id-at-organizationName=com)
  ▸ Type Payload: Vendor ID (13) : RFC 3706 DPD (Dead Peer Detection)
  ▸ Type Payload: Vendor ID (13) : Unknown Vendor ID
  ▸ Type Payload: Vendor ID (13) : XAUTH
  ▸ Type Payload: NAT-D (RFC 3947) (20)
  ▸ Type Payload: NAT-D (RFC 3947) (20)
```

R2가 패킷을 수신하면 인증서 요청을 처리하기 시작합니다. 그러면 MM5에서 인증에 사용되는 신뢰 지점 및 관련 인증서를 결정하는 일치가 생성됩니다. 프로세스 순서는 ISAKMP 패킷의 인증서 요청 페이로드와 동일합니다.즉, 첫 번째 일치 항목이 사용됩니다.이 시나리오에서는 R1이 특정 신뢰 지점으로 구성되어 신뢰 지점과 연결된 인증서 요청을 하나만 보내기 때문에 일치하는 항목이 하나만 있습니다.

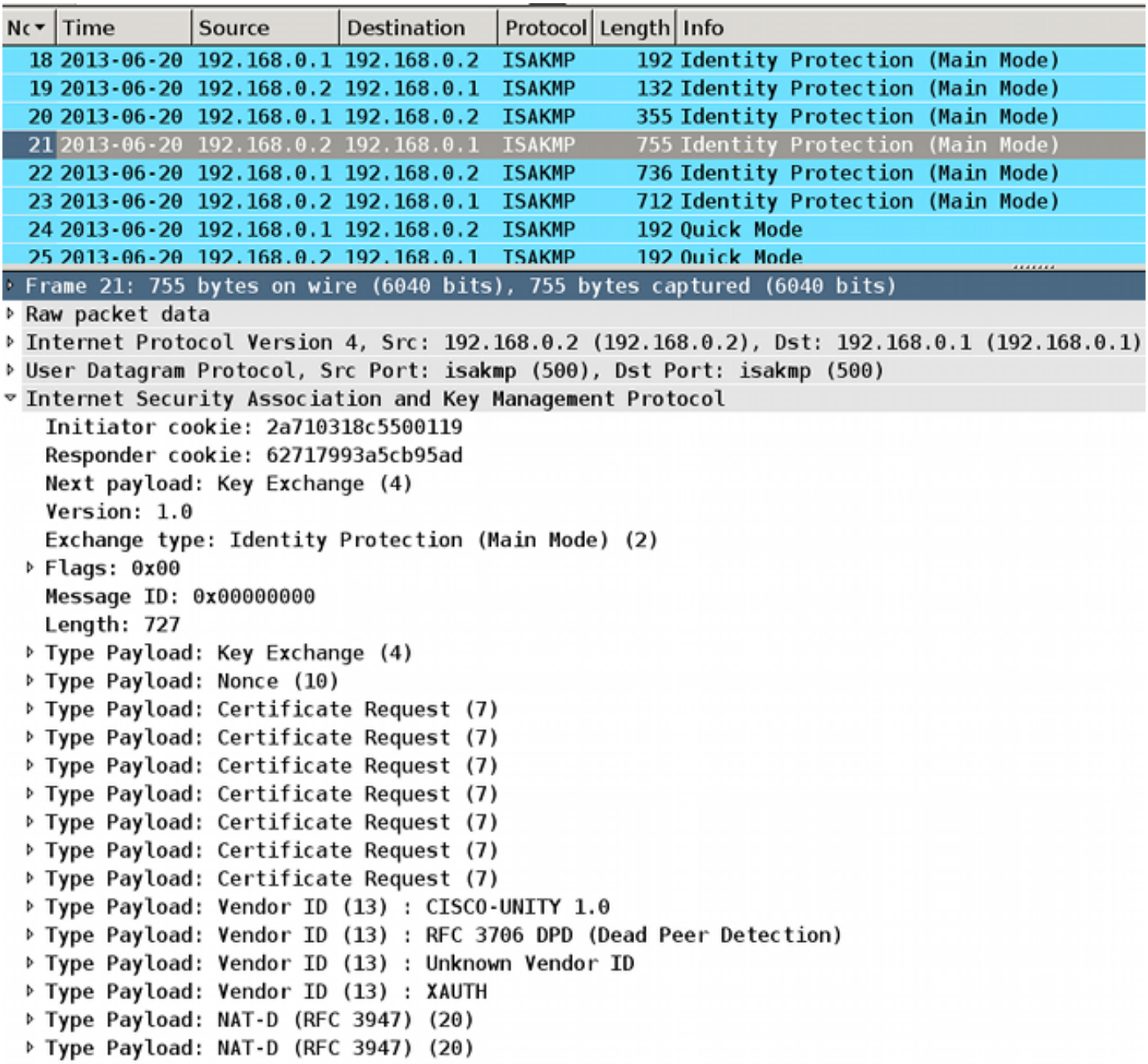
```
*Jun 20 13:00:37.617: ISAKMP:(1010): peer wants a CT_X509_SIGNATURE cert
*Jun 20 13:00:37.617: ISAKMP:(1010): peer wants cert issued
 by cn=CA1,o=cisco,o=com
*Jun 20 13:00:37.617:  Choosing trustpoint IOSCA1 as issuer
```

그런 다음 R2가 MM4를 준비합니다. 이 패킷은 모든 신뢰할 수 있는 신뢰 지점에 대한 인증서 요청을 포함합니다.R2는 ISAKMP responder이므로 전체적으로 정의된 모든 신뢰 지점을 신뢰합니다(ca **trust-point** 컨피그레이션은 선택되지 않음). 신뢰 지점 중 두 개는 수동으로 정의되며(**IOSCA1**과 **IOSCA2**) 나머지는 미리 정의되어 있습니다.

```
*Jun 20 13:00:37.617: ISAKMP (1010): constructing CERT_REQ
 for issuer cn=CA1,o=cisco,o=com
*Jun 20 13:00:37.617: ISAKMP (1010): constructing CERT_REQ
```

```
 for issuer cn=CA2,o=cisco,o=com
*Jun 20 13:00:37.617: ISAKMP (1010): constructing CERT_REQ
 for issuer ou=Class 3 Public Primary Certification Authority,
 o=VeriSign, Inc.,c=US
*Jun 20 13:00:37.617: ISAKMP (1010): constructing CERT_REQ
 for issuer cn=Cisco SSCA2,o=Cisco Systems
*Jun 20 13:00:37.617: ISAKMP (1010): constructing CERT_REQ
 for issuer cn=Cisco Manufacturing CA,o=Cisco Systems
*Jun 20 13:00:37.617: ISAKMP (1010): constructing CERT_REQ
 for issuer cn=Cisco Root CA 2048,o=Cisco Systems
*Jun 20 13:00:37.617: ISAKMP (1010): constructing CERT_REQ
 for issuer cn=Cisco Root CA M1,o=Cisco
*Jun 20 13:00:37.617: ISAKMP:(1010): sending packet to
 192.168.0.1 my_port 500 peer_port 500 (R) MM_KEY_EXCH
*Jun 20 13:00:37.617: ISAKMP:(1010):Sending an IKE IPv4 Packet.
*Jun 20 13:00:37.617: ISAKMP:(1010):Input = IKE_MESG_INTERNAL,
 IKE_PROCESS_COMPLETE
*Jun 20 13:00:37.617: ISAKMP:(1010):Old State = IKE_R_MM3
 New State = IKE_R_MM4
```

Wireshark를 사용하여 패킷을 확인할 수 있습니다.R2의 MM4 패킷에는 7개의 인증서 요청 항목이 포함되어 있습니다.



No.	Time	Source	Destination	Protocol	Length	Info
18	2013-06-20	192.168.0.1	192.168.0.2	ISAKMP	192	Identity Protection (Main Mode)
19	2013-06-20	192.168.0.2	192.168.0.1	ISAKMP	132	Identity Protection (Main Mode)
20	2013-06-20	192.168.0.1	192.168.0.2	ISAKMP	355	Identity Protection (Main Mode)
21	2013-06-20	192.168.0.2	192.168.0.1	ISAKMP	755	Identity Protection (Main Mode)
22	2013-06-20	192.168.0.1	192.168.0.2	ISAKMP	736	Identity Protection (Main Mode)
23	2013-06-20	192.168.0.2	192.168.0.1	ISAKMP	712	Identity Protection (Main Mode)
24	2013-06-20	192.168.0.1	192.168.0.2	ISAKMP	192	Quick Mode
25	2013-06-20	192.168.0.2	192.168.0.1	ISAKMP	192	Quick Mode

그런 다음 R1은 여러 인증서 요청 필드가 있는 R2에서 MM4를 수신합니다.

```
*Jun 20 13:00:37.623: ISAKMP:(1010): processing CERT_REQ payload. message ID = 0
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants a CT_X509_SIGNATURE cert
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants cert issued by
 cn=CA1,o=cisco,o=com
*Jun 20 13:00:37.623: ISAKMP: Examining profile list for trustpoint IOSCA1
*Jun 20 13:00:37.623: ISAKMP: Found matching profile for IOSCA1
*Jun 20 13:00:37.623:  Choosing trustpoint IOSCA1 as issuer
*Jun 20 13:00:37.623: ISAKMP:(1010): processing CERT_REQ payload. message ID = 0
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants a CT_X509_SIGNATURE cert
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants cert issued by
 cn=CA2,o=cisco,o=com
*Jun 20 13:00:37.623: ISAKMP:(1010): processing CERT_REQ payload. message ID = 0
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants a CT_X509_SIGNATURE cert
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants cert issued by ou=Class 3
 Public Primary Certification Authority,o=VeriSign, Inc.,c=US
*Jun 20 13:00:37.623: ISAKMP:(1010): processing CERT_REQ payload. message ID = 0
```

```
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants a CT_X509_SIGNATURE cert
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants cert issued by
 cn=Cisco SSCA2,o=Cisco Systems
*Jun 20 13:00:37.623: ISAKMP:(1010): processing CERT_REQ payload. message ID = 0
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants a CT_X509_SIGNATURE cert
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants cert issued by
 cn=Cisco Manufacturing CA,o=Cisco Systems
*Jun 20 13:00:37.623: ISAKMP:(1010): processing CERT_REQ payload. message ID = 0
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants a CT_X509_SIGNATURE cert
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants cert issued by
 cn=Cisco Root CA 2048,o=Cisco Systems
*Jun 20 13:00:37.623: ISAKMP:(1010): processing CERT_REQ payload. message ID = 0
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants a CT_X509_SIGNATURE cert
*Jun 20 13:00:37.623: ISAKMP:(1010): peer wants cert issued by
 cn=Cisco Root CA M1,o=Cisco
```

R1의 첫 번째 일치 규칙은 IOSCA1 신뢰 지점과 첫 번째 인증서 요청과 일치합니다.R1이 MM5의 인증에 신뢰 지점 IOSCA1과 연결된 인증서를 사용한다고 결정합니다. FQDN(정규화된 도메인 이름)은 IKE ID로 사용됩니다.이는 ISAKMP 프로필의 **자체 ID fqdn** 컨피그레이션 때문입니다.

```
*Jun 20 13:00:37.624: ISAKMP (1010): constructing CERT payload for serialNumber=
 100+ipaddress=192.168.0.1+hostname=R1.cisco.com,cn=R1,ou=IT,o=cisco,o=com
*Jun 20 13:00:37.624: ISAKMP:(1010): using the IOSCA1 trustpoint's
 keypair to sign
```

MM5는 R2에서 수신되고 처리됩니다. 수신된 IKE ID(**R1.cisco.com**)는 ISAKMP 프로필 **prof1**과 일치합니다. 그런 다음 수신된 인증서가 검증되고 인증이 성공적입니다.

```
*Jun 20 13:00:37.625: ISAKMP:(1010): processing ID payload. message ID = 0
*Jun 20 13:00:37.625: ISAKMP (1010): ID payload
       next-payload : 6
       type         : 2
       FQDN name    : R1.cisco.com
       protocol     : 17
       port         : 500
       length       : 20
*Jun 20 13:00:37.625: ISAKMP:(0):: peer matches prof1 profile
..........
*Jun 20 13:00:37.626: CRYPTO_PKI: (A0013) Certificate validation succeeded
..........
*Jun 20 13:00:37.626: ISAKMP:(1010):SA authentication status:
       authenticated
```

그런 다음 R2는 IOSCA1과 연결된 인증서로 MM6을 **준비합니다**.

```
*Jun 20 13:00:37.627: ISAKMP (1010): constructing CERT payload for serialNumber=
 101+ipaddress=192.168.0.2+hostname=R2.cisco.com,cn=R2,ou=IT,o=cisco,o=com
*Jun 20 13:00:37.627: ISAKMP:(1010): using the IOSCA1 trustpoint's keypair to sign
*Jun 20 13:00:37.632: ISAKMP:(1010): sending packet to 192.168.0.1
 my_port 500 peer_port 500 (R) MM_KEY_EXCH
```

패킷이 R1에 의해 수신되고 R1은 인증서와 인증을 확인합니다.

```
*Jun 20 13:00:37.632: ISAKMP (1010): received packet from 192.168.0.2
 dport 500 sport 500 Global (I) MM_KEY_EXCH
*Jun 20 13:00:37.632: ISAKMP:(1010): processing ID payload. message ID = 0
*Jun 20 13:00:37.632: ISAKMP (1010): ID payload
       next-payload : 6
       type         : 2
       FQDN name    : R2.cisco.com
       protocol     : 17
       port         : 500
       length       : 20
....
*Jun 20 13:00:37.632: ISAKMP:(0): Creating CERT validation list: IOSCA1
....
*Jun 20 13:00:37.633: CRYPTO_PKI: (80013) Certificate validation succeeded
....
*Jun 20 13:00:37.637: ISAKMP:(1010):SA authentication status:
       authenticated
*Jun 20 13:00:37.637: ISAKMP:(1010):Old State = IKE_I_MM6
 New State = IKE_P1_COMPLETE
```

1단계가 완료되었습니다. 2단계가 평소와 같이 협상됩니다.터널이 성공적으로 설정되고 트래픽이 보호됩니다.

R2를 IKEv1 Initiator로 사용

이 예에서는 R2가 동일한 IKEv1 터널을 시작하는 프로세스와 이 터널이 설정되지 않은 이유를 설명합니다.

참고:이전 섹션에 제시된 예와 관련된 차이점에만 집중하기 위해 로그의 일부가 제거됩니다.

R2에는 ISAKMP 프로파일과 연결된 신뢰 지점이 없으므로 R2는 7개의 인증서 요청 페이로드와 함께 MM3을 보냅니다(모든 신뢰 지점은 신뢰됨).

```
*Jun 17 18:08:44.321: ISAKMP (0): constructing CERT_REQ for
 issuer cn=CA1,o=cisco,o=com
*Jun 17 18:08:44.321: ISAKMP (0): constructing CERT_REQ for
 issuer cn=CA2,o=cisco,o=com
*Jun 17 18:08:44.321: ISAKMP (0): constructing CERT_REQ for
 issuer ou=Class 3 Public Primary Certification Authority,
 o=VeriSign, Inc.,c=US
*Jun 17 18:08:44.321: ISAKMP (0): constructing CERT_REQ for
 issuer cn=Cisco SSCA2,o=Cisco Systems
*Jun 17 18:08:44.321: ISAKMP (0): constructing CERT_REQ for
 issuer cn=Cisco Manufacturing CA,o=Cisco Systems
*Jun 17 18:08:44.321: ISAKMP (0): constructing CERT_REQ for
 issuer cn=Cisco Root CA 2048,o=Cisco Systems
*Jun 17 18:08:44.321: ISAKMP (0): constructing CERT_REQ for
 issuer cn=Cisco Root CA M1,o=Cisco
*Jun 17 18:08:44.321: ISAKMP (0): sending packet to 192.168.0.1
```

```
my_port 500 peer_port 500 (I) MM_SA_SETUP
```

R1은 R2에서 패킷을 수신하면 인증서 요청을 처리하고 MM6에서 전송되는 인증서를 결정하는 IOSCA1 신뢰 지점을 확인합니다.

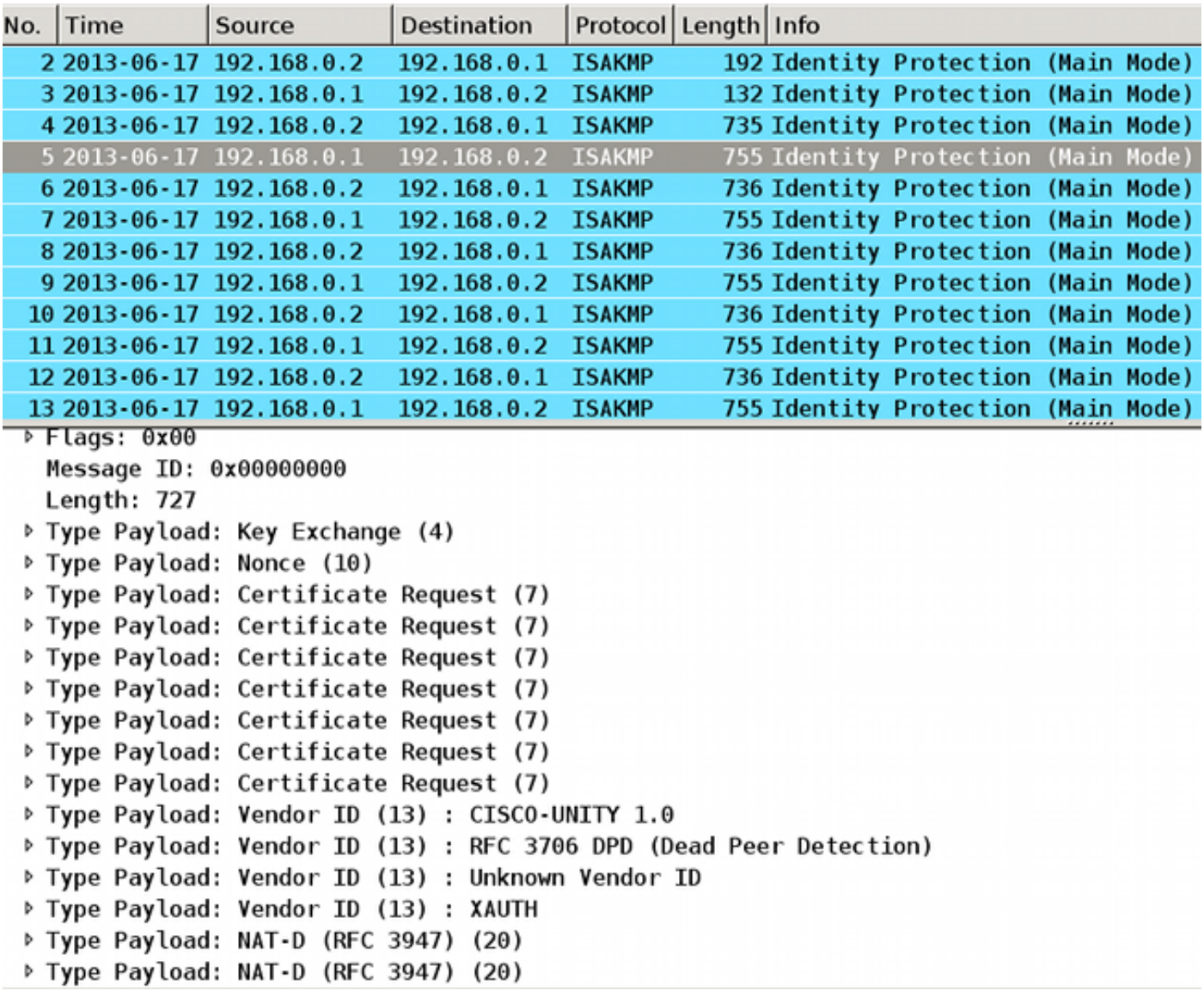
```
*Jun 17 18:08:14.321: ISAKMP:(1099): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants cert issued by
 cn=CA1,o=cisco,o=com
*Jun 17 18:08:14.321:  Choosing trustpoint IOSCA1 as issuer
*Jun 17 18:08:14.321: ISAKMP:(1099): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants cert issued by
 cn=CA2,o=cisco,o=com
*Jun 17 18:08:14.321: ISAKMP:(1099): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants cert issued by
 ou=Class 3 Public Primary Certification Authority,o=VeriSign, Inc.,c=US
*Jun 17 18:08:14.321: ISAKMP:(1099): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants cert issued by
 cn=Cisco SSCA2,o=Cisco Systems
*Jun 17 18:08:14.321: ISAKMP:(1099): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants cert issued by
 cn=Cisco Manufacturing CA,o=Cisco Systems
*Jun 17 18:08:14.321: ISAKMP:(1099): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants cert issued by
 cn=Cisco Root CA 2048,o=Cisco Systems
*Jun 17 18:08:14.321: ISAKMP:(1099): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:14.321: ISAKMP:(1099): peer wants cert issued by
 cn=Cisco Root CA M1,o=Cisco
```

그런 다음 R1은 인증서 요청 페이로드를 사용하여 MM4 패킷을 준비합니다.이제 여러 인증서 요청 페이로드가 있습니다.

```
*Jun 17 18:08:14.321: ISAKMP (1099): constructing CERT_REQ for issuer
 cn=CA2,o=cisco,o=com
*Jun 17 18:08:14.321: ISAKMP (1099): constructing CERT_REQ for issuer
 cn=CA1,o=cisco,o=com
*Jun 17 18:08:14.322: ISAKMP (1099): constructing CERT_REQ for issuer
 ou=Class 3 Public Primary Certification Authority,
 o=VeriSign, Inc.,c=US
*Jun 17 18:08:14.322: ISAKMP (1099): constructing CERT_REQ for issuer
 cn=Cisco SSCA2,o=Cisco Systems
*Jun 17 18:08:14.322: ISAKMP (1099): constructing CERT_REQ for issuer
 cn=Cisco Manufacturing CA,o=Cisco Systems
*Jun 17 18:08:14.322: ISAKMP (1099): constructing CERT_REQ for issuer
```

```
 cn=Cisco Root CA 2048,o=Cisco Systems
*Jun 17 18:08:14.322: ISAKMP (1099): constructing CERT_REQ for issuer
 cn=Cisco Root CA M1,o=Cisco
*Jun 17 18:08:14.322: ISAKMP:(1099): sending packet to 192.168.0.2
 my_port 500 peer_port 500 (R) MM_KEY_EXCH
```

EPC(Embedded Packet Capture) 및 Wireshark를 사용하여 로그를 확인합니다.



No.	Time	Source	Destination	Protocol	Length	Info
2	2013-06-17	192.168.0.2	192.168.0.1	ISAKMP	192	Identity Protection (Main Mode)
3	2013-06-17	192.168.0.1	192.168.0.2	ISAKMP	132	Identity Protection (Main Mode)
4	2013-06-17	192.168.0.2	192.168.0.1	ISAKMP	735	Identity Protection (Main Mode)
5	2013-06-17	192.168.0.1	192.168.0.2	ISAKMP	755	Identity Protection (Main Mode)
6	2013-06-17	192.168.0.2	192.168.0.1	ISAKMP	736	Identity Protection (Main Mode)
7	2013-06-17	192.168.0.1	192.168.0.2	ISAKMP	755	Identity Protection (Main Mode)
8	2013-06-17	192.168.0.2	192.168.0.1	ISAKMP	736	Identity Protection (Main Mode)
9	2013-06-17	192.168.0.1	192.168.0.2	ISAKMP	755	Identity Protection (Main Mode)
10	2013-06-17	192.168.0.2	192.168.0.1	ISAKMP	736	Identity Protection (Main Mode)
11	2013-06-17	192.168.0.1	192.168.0.2	ISAKMP	755	Identity Protection (Main Mode)
12	2013-06-17	192.168.0.2	192.168.0.1	ISAKMP	736	Identity Protection (Main Mode)
13	2013-06-17	192.168.0.1	192.168.0.2	ISAKMP	755	Identity Protection (Main Mode)

R1이 ISAKMP 프로파일의 단일 신뢰 지점(**IOSCA1**)에 대해 구성되었지만 여러 인증서 요청이 전송되었습니다.이는 ISAKMP 프로필의 **ca trust-point** 명령이 인증서 요청 페이로드를 결정하지만 라우터가 ISAKMP 세션의 개시자인 경우에만 발생합니다.라우터가 responder인 경우 R1은 IKE 세션에 사용되는 ISAKMP 프로필을 아직 알지 못하므로 전역적으로 정의된 모든 신뢰 지점에 대해 여러 인증서 요청 페이로드가 있습니다.

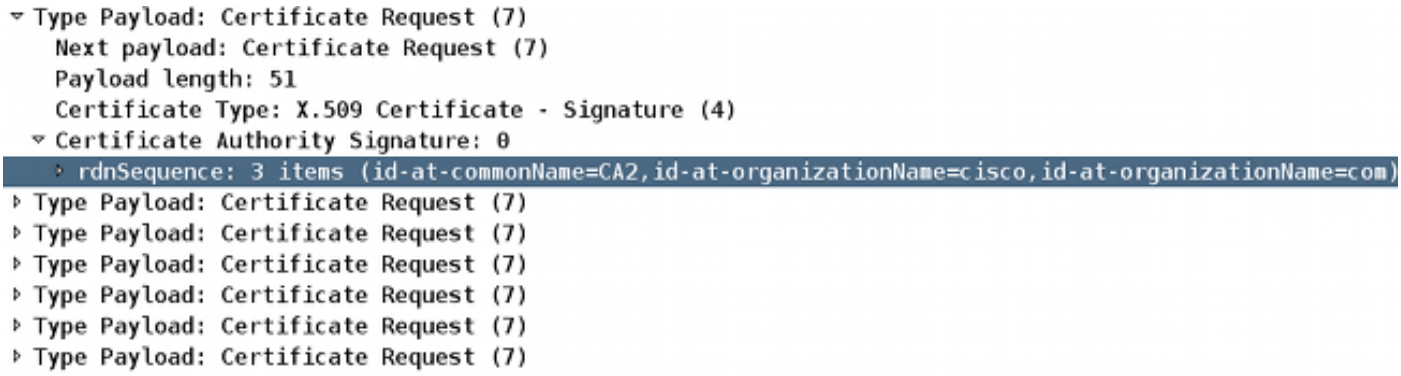
인바운드 IKE 세션은 MM5를 수신한 후 특정 ISAKMP 프로필에 바인딩되며 IKE ID가 포함됩니다.그런 다음 특정 프로필에 대한 match identity 명령이 IKE 세션을 프로파일에 바인딩합니다.그러나 라우터가 이를 결정할 수는 없습니다.각 프로필에 대해 서로 다른 ca **trust-point** 명령을 구성한 여러 ISAKMP 프로필이 있을 수 있습니다.

따라서 R1은 전역으로 구성된 모든 신뢰 지점에 대한 인증서 요청을 보내야 합니다.

ca **trust-point** 명령에 대한 명령 참조를 참조하십시오.

IKE를 시작하는 라우터와 IKE 요청에 응답하는 라우터는 대칭 신뢰 지점 컨피그레이션을 가져야 합니다.예를 들어 RSA 서명 암호화 및 인증을 수행하는 응답 라우터(IKE Main Mode)는 CERT-REQ 페이로드를 보낼 때 전역 컨피그레이션에 정의된 신뢰 지점을 사용할 수 있습니다.그러나 라우터는 인증서 확인을 위해 ISAKMP 프로필에 정의된 제한된 신뢰 지점 목록을 사용할 수 있습니다.피어(IKE 개시자)가 신뢰 지점이 응답 라우터의 전역 목록에 있지만 응답 라우터의 ISAKMP 프로필에 없는 인증서를 사용하도록 구성된 경우 인증서가 거부됩니다.그러나 시작 라우터가 응답 라우터의 전역 컨피그레이션에서 신뢰 지점을 모르는 경우에도 인증서를 인증할 수 있습니다.

이제 첫 번째 인증서 요청 페이로드를 검색하기 위해 MM4 패킷 세부사항을 확인합니다.



R1에서 전송된 MM4 패킷에는 인증서가 설치된 순서로 인해 첫 번째 인증서 요청 페이로드에 IOSCA**2** 신뢰 지점이 포함됩니다.첫 번째 항목은 IOSCA**2** 신뢰 지점에 의해 서명됩니다.

```
R1#sh crypto pki certificates
Certificate
 Status: Available
 Certificate Serial Number (hex): 03
 Certificate Usage: General Purpose
 Issuer:
    cn=CA2
   o=cisco
   o=com
 Subject:
   Name: R1.cisco.com
   IP Address: 192.168.0.1
   Serial Number: 100
   serialNumber=100+ipaddress=192.168.0.1+hostname=R1.cisco.com
   cn=R1
   ou=IT
   o=cisco
   o=com
 Validity Date:
   start date: 13:25:01 CET Jun 17 2013
   end   date: 13:25:01 CET Jun 17 2014
 Associated Trustpoints: IOSCA2
...
<output omitted, 1 more R1 cert signed by CA1, 2 more CA certs>
```

IOSCA**1** 신뢰 지점이 첫 번째 인증서 요청 페이로드에 포함된 경우 R2에서 전송된 MM3 패킷과 비교하십시오.

```
R2#sh crypto pki certificates
Certificate
 Status: Available
 Certificate Serial Number (hex): 02
 Certificate Usage: General Purpose
 Issuer:
   cn=CA1
   o=cisco
   o=com
 Subject:
   Name: R2.cisco.com
   IP Address: 192.168.0.2
   Serial Number: 101
   serialNumber=101+ipaddress=192.168.0.2+hostname=R2.cisco.com
   cn=R2
   ou=IT
   o=cisco
   o=com
 Validity Date:
   start date: 13:23:49 CET Jun 17 2013
   end   date: 13:23:49 CET Jun 17 2014
 Associated Trustpoints: IOSCA1
 Storage: nvram:CA1#2.cer
...
<output omitted, 1 more R2 cert signed by CA2, 2 more CA certs>
```

이제 R2는 R1에서 MM4 패킷을 수신하고 인증서 요청을 처리하기 시작합니다.첫 번째 인증서 요청 페이로드는 IOSCA**2** 신뢰 지점**과** 일치합니다.

```
*Jun 17 18:08:44.335: ISAKMP:(1100): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants cert issued by
 cn=CA2,o=cisco,o=com
*Jun 17 18:08:44.335:  Choosing trustpoint IOSCA2 as issuer
*Jun 17 18:08:44.335: ISAKMP:(1100): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants cert issued by
 cn=CA1,o=cisco,o=com
*Jun 17 18:08:44.335: ISAKMP:(1100): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants cert issued by
 ou=Class 3 Public Primary Certification Authority,o=VeriSign, Inc.,c=US
*Jun 17 18:08:44.335: ISAKMP:(1100): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants cert issued by
 cn=Cisco SSCA2,o=Cisco Systems
*Jun 17 18:08:44.335: ISAKMP:(1100): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants cert issued by
 cn=Cisco Manufacturing CA,o=Cisco Systems
*Jun 17 18:08:44.335: ISAKMP:(1100): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants a CT_X509_SIGNATURE cert
```

```
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants cert issued by
 cn=Cisco Root CA 2048,o=Cisco Systems
*Jun 17 18:08:44.335: ISAKMP:(1100): processing CERT_REQ payload.
 message ID = 0
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants a CT_X509_SIGNATURE cert
*Jun 17 18:08:44.335: ISAKMP:(1100): peer wants cert issued by
 cn=Cisco Root CA M1,o=Cisco
```

R2가 MM5 패킷을 준비할 때 IOSCA2 신뢰 지점과 연결된 인증서를 사용합니다.

```
*Jun 17 18:08:44.335: ISAKMP:(1100):SA is doing RSA signature authentication
 using id type ID_FQDN
*Jun 17 18:08:44.335: ISAKMP (1100): ID payload
       next-payload : 6
       type         : 2
       FQDN name    : R2.cisco.com
       protocol     : 17
       port         : 500
       length       : 20
*Jun 17 18:08:44.335: ISAKMP:(1100):Total payload length: 20
*Jun 17 18:08:44.335: ISAKMP:(1100): IKE->PKI Get CertificateChain to be sent
 to peer state (I) MM_KEY_EXCH (peer 192.168.0.1)
*Jun 17 18:08:44.335: ISAKMP:(1100): PKI->IKE Got CertificateChain to be sent
 to peer state (I) MM_KEY_EXCH (peer 192.168.0.1)
*Jun 17 18:08:44.336: ISAKMP (1100): constructing CERT payload for
 serialNumber=101+ipaddress=192.168.0.2+hostname=R2.cisco.com,cn=R2,
 ou=IT,o=cisco,o=com
R2#
*Jun 17 18:08:44.336: ISAKMP:(1100): using the IOSCA2 trustpoint's
 keypair to sign
*Jun 17 18:08:44.336: ISAKMP:(1100): sending packet to 192.168.0.1
 my_port 500 peer_port 500 (I) MM_KEY_EXCH
*Jun 17 18:08:44.336: ISAKMP:(1100):Sending an IKE IPv4 Packet.
```

MM5 패킷은 R1에서 수신됩니다. R1은 IOSCA1 신뢰 지점(ISAKMP 프로필 prof1의 경우)만 신뢰하므로 인증서 유효성 검사가 실패합니다.

```
*Jun 17 18:08:44.337: ISAKMP (1100): received packet from 192.168.0.2
 dport 500 sport 500 Global (R) MM_KEY_EXCH
*Jun 17 18:08:44.337: ISAKMP:(1100):Input = IKE_MESG_FROM_PEER, IKE_MM_EXCH
*Jun 17 18:08:44.337: ISAKMP:(1100):Old State =IKE_R_MM4  New State = IKE_R_MM5

*Jun 17 18:08:44.337: ISAKMP:(1100): processing ID payload. message ID = 0
*Jun 17 18:08:44.337: ISAKMP (1100): ID payload
       next-payload : 6
       type         : 2
       FQDN name    : R2.cisco.com
       protocol     : 17
       port         : 500
       length       : 20
*Jun 17 18:08:44.337: ISAKMP:(0):: peer matches prof1 profile
*Jun 17 18:08:44.337: ISAKMP:(1100): processing CERT payload. message ID = 0
*Jun 17 18:08:44.337: ISAKMP:(1100): processing a CT_X509_SIGNATURE cert
*Jun 17 18:08:44.337: ISAKMP:(1100): IKE->PKI Add peer's certificate state
 (R) MM_KEY_EXCH (peer 192.168.0.2)
*Jun 17 18:08:44.337: CRYPTO_PKI: (900C5) Adding peer certificate
*Jun 17 18:08:44.337: ISAKMP:(1100): PKI->IKE Added peer's certificate state
```

```
 (R) MM_KEY_EXCH (peer 192.168.0.2)
*Jun 17 18:08:44.337: ISAKMP:(1100): IKE->PKI Get PeerCertificateChain state
 (R) MM_KEY_EXCH (peer 192.168.0.2)
*Jun 17 18:08:44.337: ISAKMP:(1100): PKI->IKE Got PeerCertificateChain state
 (R) MM_KEY_EXCH (peer 192.168.0.2)
*Jun 17 18:08:44.337: ISAKMP:(1100): peer's pubkey isn't cached
*Jun 17 18:08:44.337: ISAKMP:(1100):Profile has no keyring, aborting key search
*Jun 17 18:08:44.337: ISAKMP:(0): Creating CERT validation list: IOSCA1,
*Jun 17 18:08:44.337: ISAKMP:(1100): IKE->PKI Validate certificate chain state
 (R) MM_KEY_EXCH (peer 192.168.0.2)
*Jun 17 18:08:44.337: CRYPTO_PKI:ip-ext-val:IP extension validation not required
*Jun 17 18:08:44.341: CRYPTO_PKI: (900C5) Check for identical certs
*Jun 17 18:08:44.341: CRYPTO_PKI: (900C5) Create a list of suitable trustpoints
*Jun 17 18:08:44.341: CRYPTO_PKI: (900C5) No suitable trustpoints found
*Jun 17 18:08:44.341: ISAKMP:(1100): PKI->IKE Validate certificate chain state
 (R) MM_KEY_EXCH (peer 192.168.0.2)
*Jun 17 18:08:44.341: %CRYPTO-5-IKMP_INVAL_CERT: Certificate received from
 192.168.0.2 is bad: unknown error returned in certificate validation
R1#
*Jun 17 18:08:44.341: ISAKMP:(1100): Unknown error in cert validation, -1
```

이 컨피그레이션은 처음 표시된 인증서가 IOSCA1 신뢰 지점에 의해 서명되었으므로 R1의 인증서 등록 순서가 다른 경우 작동합니다.또한 MM4의 첫 번째 인증서 요청 페이로드는 **IOSCA1** 신뢰 지점이며, 이 신뢰 지점은 R2에 의해 선택되고 MM6의 R1에서 성공적으로 검증됩니다.

프로파일에 ca *trust-point* 명령이 없는 IKEv1

여러 프로파일 및 신뢰 지점이 있지만 프로파일에 특정 신뢰 지점 컨피그레이션이 없는 시나리오의 경우 ca **trust-point** 명령 컨피그레이션에 의해 결정되는 특정 신뢰 지점의 검증이 없으므로 문제가 없습니다.그러나 선택 과정이 명확하지 않을 수도 있다.개시자인 라우터에 따라 인증서 등록 순서와 관련하여 인증 프로세스에 대해 서로 다른 인증서가 선택됩니다.

때로는 x509 버전 1과 같이 연결의 한 쪽에서만 인증서를 지원할 수 있습니다. 이 경우 서명하기 위해 사용되는 일반적인 해시 함수가 아닙니다.VPN 터널은 연결의 한 쪽에서만 설정할 수 있습니다.

IKEv1에 대한 RFC 참조

다음은 RFC4945의 스냅샷입니다.

3.2.7.1. 인증 기관 지정

키 자료의 대역 내 교환을 **요청할** 경우, 구현에서는 지정된 교환 동안 **로컬 정책**이 명시적으로 신뢰한다고 **판단되는** 모든 피어 트러스트 앵커에 대해 CERTREQ를 생성해야 합니다.
RFC가 명확하지 않습니다.**로컬 정책**은 **crypto ISAKMP** 프로파일에 구성된 ca trust-**point** 명령과 관련될 수 있습니다.문제는 프로세스의 MM3 및 MM4 단계에서 ID 및 신뢰 지점에 IP 주소를 사용하지 않는 한 ISAKMP 프로파일을 선택할 수 없다는 것입니다. MM5 및 MM6 단계의 인증이 먼저 발생해야 하기 때문입니다.따라서 **로컬 정책**은 디바이스에 구성된 모든 신뢰 지점과 명시적으로 연결됩니다.

참고:이 정보는 Cisco에 한정되지 않지만 IKEv1에 한정됩니다.

ID가 겹치는 IKEv2 프로파일 선택

IKEv2에 대한 여러 인증서를 설명하려면 모든 프로파일에 대해 충족되는 일치 ID를 사용할 때 프로파일을 선택하는 방법을 알아야 합니다.IKEv2 협상 결과는 여러 요인에 따라 결정되므로 이 시나리오는 권장되지 않습니다.겹치는 프로파일을 사용하는 경우 IKEv1에도 동일한 문제가 있습니다.

다음은 IKEv2 개시자 구성의 예입니다.

```
crypto ikev2 proposal prop-1
 encryption 3des
 integrity md5
 group 2
!
crypto ikev2 policy pol-1
 match fvrf any
 proposal prop-1
!
crypto ikev2 profile profile1
 match identity remote address 192.168.0.2 255.255.255.255
 identity local address 192.168.0.1
 authentication remote rsa-sig
 authentication local rsa-sig
 pki trustpoint TP1

crypto ipsec transform-set trans esp-3des esp-sha-hmac
 mode tunnel
!
crypto ipsec profile profile1
 set transform-set trans
 set ikev2-profile profile1
!
interface Loopback0
 ip address 192.168.100.1 255.255.255.255
!
interface Tunnel1
 ip address 10.0.0.1 255.255.255.0
 tunnel source Ethernet0/0
 tunnel destination 192.168.0.2
 tunnel protection ipsec profile profile1
!
interface Ethernet0/0
 ip address 192.168.0.1 255.255.255.0

ip route 192.168.200.1 255.255.255.255 10.0.0.2
```

ID 유형 주소는 연결의 양쪽에 사용됩니다.이 예에서는 인증서를 통한 인증(사전 공유 키일 수도 있음)이 중요하지 않습니다.응답자는 모두 인바운드 IKEv2 트래픽과 일치하는 여러 프로필을 가지고 있습니다.

```
crypto ikev2 proposal prop-1
```

```
 encryption 3des
 integrity md5
 group 2
!
crypto ikev2 policy pol-1
 match fvrf any
 proposal prop-1
!
crypto ikev2 profile profile1
 match identity remote address 192.168.0.1 255.255.255.255
 identity local address 192.168.0.2
 authentication remote rsa-sig
 authentication local rsa-sig
 pki trustpoint TP1
!
crypto ikev2 profile profile2
 match identity remote address 192.168.0.1 255.255.255.255
 identity local address 192.168.0.2
 authentication remote rsa-sig
 authentication local rsa-sig
 pki trustpoint TP1
!
crypto ikev2 profile profile3
 match identity remote address 192.168.0.1 255.255.255.255
 identity local address 192.168.0.2
 authentication remote rsa-sig
 authentication local rsa-sig
 pki trustpoint TP1

crypto ipsec transform-set trans esp-3des esp-sha-hmac
 mode tunnel
!
crypto ipsec profile profile1
 set transform-set trans
 set ikev2-profile profile1
!
interface Loopback0
 ip address 192.168.200.1 255.255.255.255
!
interface Tunnel1
 ip address 10.0.0.2 255.255.255.0
 tunnel source Ethernet0/0
 tunnel destination 192.168.0.1
 tunnel protection ipsec profile profile1
!
interface Ethernet0/0
 ip address 192.168.0.2 255.255.255.0

ip route 192.168.100.1 255.255.255.255 10.0.0.1
```

개시자는 세 번째 IKEv2 패킷을 전송하고, 응답자는 수신된 ID를 기반으로 프로파일을 선택해야 합니다.ID는 IPv4 주소(192.168.0.1)입니다.

```
IKEv2:(SA ID = 1):Searching policy based on peer's identity '192.168.0.1' of
 type 'IPv4 address'
```

구성된 match identity 명령으로 인해 모든 프로파일이 이 ID를 충족합니다.IOS는 컨피그레이션의 마지막 항목을 선택합니다. 이는 다음 예에서 profile3입니다.

```
IKEv2:found matching IKEv2 profile 'profile3'
```

순서를 확인하려면 show crypto ikev2 profile 명령을 입력합니다.

> **참고:**프로파일에 일반 주소(0.0.0.0)이 있는 경우에도 여전히 선택됩니다.IOS는 최상의 일치 항목을 찾으려고 시도하지 않습니다.첫 번째 일치를 찾으려고 합니다.그러나 이 문제는 모든 프로파일이 동일한 match identity remote 명령을 구성했기 때문에 발생합니다.서로 다른 일치 ID 규칙이 있는 IKEv1 및 IKEv2 프로필의 경우 가장 구체적인 규칙이 항상 사용됩니다 .Cisco에서는 **중복** match identity 명령으로 구성된 프로파일이 없는 것이 좋습니다. 선택한 프로파일을 예측하기가 어렵기 때문입니다.

이 시나리오에서 profile3은 responder에 의해 선택되지만 profile1은 터널 인터페이스에 사용됩니다 .이렇게 하면 프록시 ID가 협상될 때 오류가 표시됩니다.

```
*Jul 17 09:23:48.187: map_db_check_isakmp_profile profile did not match
*Jul 17 09:23:48.187: map_db_find_best did not find matching map
*Jul 17 09:23:48.187: IPSEC(ipsec_process_proposal):
 proxy identities not supported
*Jul 17 09:23:48.187: IKEv2:(SA ID = 1):There was no
 IPSEC policy found for received TS
*Jul 17 09:23:48.187: IKEv2:(SA ID = 1):
*Jul 17 09:23:48.187: IKEv2:(SA ID = 1):Sending TS unacceptable notify
```

인증서 사용 시 IKEv2 흐름

인증하기 위해 IKEv2에 인증서를 사용하는 경우, 개시자는 첫 번째 패킷에서 인증서 요청 페이로드를 보내지 않습니다.

```
IKEv2 IKE_SA_INIT Exchange REQUEST
Payload contents:
 SA KE N VID VID NOTIFY(NAT_DETECTION_SOURCE_IP)
 NOTIFY(NAT_DETECTION_DESTINATION_IP)
```

응답자는 이 단계에서 사용해야 하는 프로파일에 대한 지식이 없기 때문에 인증서 요청 페이로드(두 번째 패킷) 및 모든 CA를 사용하여 응답합니다.정보가 포함된 패킷은 이니시에이터로 전송됩니다.

```
IKEv2 IKE_SA_INIT Exchange RESPONSE
Payload contents:
 SA KE N VID VID NOTIFY(NAT_DETECTION_SOURCE_IP) NOTIFY
 (NAT_DETECTION_DESTINATION_IP) CERTREQ NOTIFY(HTTP_CERT_LOOKUP_SUPPORTED)
```

개시자는 패킷을 처리하고 제안된 CA와 일치하는 신뢰 지점을 선택합니다.

```
IKEv2:(SA ID = 1):[IKEv2 -> PKI] Retrieving trustpoint(s) from
 received certificate hash(es)
IKEv2:(SA ID = 1):[PKI -> IKEv2] Retrieved trustpoint(s): 'TP1'
```

그런 다음 개시자는 인증서 요청 및 인증서 페이로드를 모두 사용하여 세 번째 패킷을 전송합니다 .이 패킷은 DH(Diffie-Hellman) 단계의 키 자료로 이미 암호화되어 있습니다.

```
IKEv2:(SA ID = 1):Building packet for encryption.
Payload contents:
 VID IDi CERT CERTREQ NOTIFY(HTTP_CERT_LOOKUP_SUPPORTED) AUTH CFG SA TSi
 TSr NOTIFY(INITIAL_CONTACT) NOTIFY(SET_WINDOW_SIZE) NOTIFY(ESP_TFC_NO_SUPPORT)
 NOTIFY(NON_FIRST_FRAGS)
```

네 번째 패킷은 responder에서 initiator로 전송되며 인증서 페이로드만 포함합니다.

```
IKEv2 IKE_AUTH Exchange RESPONSE
Payload contents:
 VID IDr CERT AUTH SA TSi TSr NOTIFY(SET_WINDOW_SIZE) NOTIFY(ESP_TFC_NO_SUPPORT)
NOTIFY(NON_FIRST_FRAGS)
```

여기에 설명된 플로우는 IKEv1 플로우와 유사합니다.responder는 사용해야 할 프로파일을 알지 못한 채 인증서 요청 페이로드를 앞에 전송해야 합니다. 이렇게 하면 프로토콜 관점에서 IKEv1에 대해 이전에 설명한 것과 동일한 문제가 발생합니다. 그러나 IOS의 구현은 IKEv1보다 IKEv2에 더 좋습니다.

이니시에이터의 IKEv2 필수 신뢰 지점

다음은 IKEv2 개시자가 인증서 인증과 함께 프로파일을 사용하려고 시도하는데 해당 프로파일 아래에 신뢰 지점이 구성되지 않은 경우의 예입니다.

```
crypto ikev2 profile profile1
 match identity remote address 192.168.0.2 255.255.255.255
 identity local address 192.168.0.1
 authentication remote rsa-sig
 authentication local rsa-sig
```

앞서 설명한 대로 첫 번째 패킷은 인증서 요청 페이로드 없이 전송됩니다.응답자의 응답에는 전역 컨피그레이션 모드에서 정의된 모든 신뢰 지점에 대한 인증서 요청 페이로드가 포함됩니다.개시자가 수신한 메시지:

```
*Jul 17 17:40:43.183: IKEv2:(SA ID = 1):[IKEv2 -> PKI] Retrieving trustpoint(s)
```

```
 from received certificate hash(es)
*Jul 17 17:40:43.183: IKEv2:(SA ID = 1):[PKI -> IKEv2] Retrieved
 trustpoint(s): 'TP1'
*Jul 17 17:40:43.183: CRYPTO_PKI: crypto_pki_get_cert_record_by_subject()
*Jul 17 17:40:43.183: CRYPTO_PKI: Found a subject match
*Jul 17 17:40:43.183: CRYPTO_PKI: crypto_pki_get_cert_record_by_subject()
*Jul 17 17:40:43.183: CRYPTO_PKI: Found a subject match
*Jul 17 17:40:43.183: CRYPTO_PKI: Trust-Point TP1 picked up
*Jul 17 17:40:43.183: CRYPTO_PKI: 1 matching trustpoints found
*Jul 17 17:40:43.183: IKEv2:(SA ID = 1):[IKEv2 -> PKI] Retrieving trustpoint(s)
 from received certificate hash(es)
*Jul 17 17:40:43.183: IKEv2:(SA ID = 1):[PKI -> IKEv2] Retrieved
 trustpoint(s): 'TP2'
*Jul 17 17:40:43.183: CRYPTO_PKI: Trust-Point TP2 picked up
*Jul 17 17:40:43.183: CRYPTO_PKI: crypto_pki_get_cert_record_by_subject()
*Jul 17 17:40:43.183: CRYPTO_PKI: Found a subject match
*Jul 17 17:40:43.183: CRYPTO_PKI: crypto_pki_get_cert_record_by_subject()
*Jul 17 17:40:43.183: CRYPTO_PKI: Found a subject match
*Jul 17 17:40:43.183: CRYPTO_PKI: 1 matching trustpoints found
*Jul 17 17:40:43.183: IKEv2:(SA ID = 1):Failed to build certificate payload
```

개시자가 서명하기 위해 사용해야 하는 신뢰 지점을 알지 못합니다.이는 IKEv2 구현을 IKEv1과 비교할 때 가장 큰 차이입니다. IKEv2 개시자는 IKEv2 개시자 프로필에 구성된 신뢰 지점을 가져야 하지만 IKEv2 응답자에게는 필요하지 않습니다.

명령 참조에서 발췌한 내용은 다음과 같습니다.

IKEv2 프로파일 컨피그레이션에 정의된 신뢰 지점이 없는 경우 기본값은 전역 컨피그레이션에 정의된 모든 신뢰 지점을 사용하여 **인증서**를 검증하는 것입니다
서로 다른 신뢰 지점을 정의할 수 있습니다.서명해야 하며 검증해야 합니다.안타깝게도 IKEv2 프로필에 구성된 필수 신뢰 지점은 모든 문제를 해결하지 않습니다.

IKEv2 개시자로 R2

이 예에서 R2는 IKEv2 개시자입니다.

```
crypto ikev2 profile profile1
 match identity remote address 192.168.0.1 255.255.255.255
 identity local address 192.168.0.2
 authentication remote rsa-sig
 authentication local rsa-sig
 pki trustpoint TP1
 pki trustpoint TP2
```

이 예에서 R1은 IKEv2 responder입니다.

```
crypto ikev2 profile profile1
 match identity remote address 192.168.0.2 255.255.255.255
 identity local address 192.168.0.1
 authentication remote rsa-sig
 authentication local rsa-sig
```

```
 pki trustpoint TP1
```

여기서 R2는 인증서 요청 없이 첫 번째 패킷을 전송합니다.응답자는 구성된 모든 신뢰 지점에 대해 인증서 요청으로 응답합니다.페이로드 순서는 IKEv1과 유사하며 설치된 인증서에 따라 다릅니다.

```
R1#show crypto pki certificates
Certificate
 Status: Available
 Certificate Serial Number (hex): 04
 Certificate Usage: General Purpose
 Issuer:
   cn=CA2
....
 Associated Trustpoints: TP2
```

R1에서 처음 구성된 인증서는 **TP2** 신뢰 지점과 연결되므로 첫 번째 인증서 요청 페이로드는 **TP2** 신뢰 지점과 연결된 CA에 대한 것입니다.따라서 R2는 인증을 위해 선택합니다(첫 번째 일치 규칙).

```
R2#
*Jul 17 18:09:04.542: IKEv2:(SA ID = 1):Processing IKE_SA_INIT message
*Jul 17 18:09:04.542: IKEv2:(SA ID = 1):[IKEv2 -> PKI] Retrieving trustpoint(s)
 from received certificate hash(es)
*Jul 17 18:09:04.542: IKEv2:(SA ID = 1):[PKI -> IKEv2] Retrieved
 trustpoint(s): 'TP2'
*Jul 17 18:09:04.542: IKEv2:(SA ID = 1):[IKEv2 -> PKI] Getting cert chain for
 the trustpoint TP2
*Jul 17 18:09:04.542: IKEv2:(SA ID = 1):[PKI -> IKEv2] Getting of cert chain
 for the trustpoint PASSED
```

그런 다음 R2는 **TP2**와 연결된 인증 요청 페이로드와 함께 응답(패킷 3)을 준비합니다. R1은 **TP1** 신뢰 지점에 대한 유효성 검사를 위해 구성되어 있으므로 인증서를 신뢰할 수 없습니다.

```
*Jul 17 18:09:04.550: IKEv2:(SA ID = 1):[IKEv2 -> PKI] Retrieving trustpoint(s)
 from received certificate hash(es)
*Jul 17 18:09:04.550: IKEv2:(SA ID = 1):[PKI -> IKEv2] Retrieved
 trustpoint(s): 'TP1'
*Jul 17 18:09:04.550: IKEv2:(SA ID = 1):[IKEv2 -> PKI] Getting cert chain for
 the trustpoint TP1
*Jul 17 18:09:04.550: IKEv2:(SA ID = 1):[PKI -> IKEv2] Getting of cert chain
 for the trustpoint PASSED
*Jul 17 18:09:04.550: IKEv2:(SA ID = 1):Get peer's authentication method
*Jul 17 18:09:04.550: IKEv2:(SA ID = 1):Peer's authentication method is 'RSA'
*Jul 17 18:09:04.550: IKEv2:(SA ID = 1):[IKEv2 -> PKI] Validating
 certificate chain
*Jul 17 18:09:04.554: IKEv2:(SA ID = 1):[PKI -> IKEv2] Validation of certificate
 chain FAILED
*Jul 17 18:09:04.554: IKEv2:(SA ID = 1):Verification of peer's authentication
 data FAILED
*Jul 17 18:09:04.554: IKEv2:(SA ID = 1):Sending authentication failure notify
*Jul 17 18:09:04.554: IKEv2:(SA ID = 1):Building packet for encryption.
Payload contents:
 NOTIFY(AUTHENTICATION_FAILED)
```

앞서 언급한 대로, Cisco에서는 하나의 IKEv2 프로필 아래에 여러 신뢰 지점을 사용하지 않는 것이 좋습니다.여러 신뢰 지점을 사용하는 경우 양쪽이 정확히 동일한 신뢰 지점을 신뢰하도록 해야 합니다.예를 들어, R1과 R2 모두 프로파일에 TP1과 TP2가 모두 구성되어 있습니다.

요약

이 섹션에서는 문서에 설명된 정보에 대한 간략한 요약을 제공합니다.

인증서 요청 페이로드 콘텐츠는 구성에 따라 다릅니다.ISAKMP 프로파일에 대해 특정 신뢰 지점이 구성되어 있고 라우터가 ISAKMP 개시자인 경우 MM3의 인증서 요청에는 신뢰 지점과 연결된 CA만 포함됩니다.그러나 동일한 라우터가 ISAKMP responder인 경우 라우터에서 전송하는 MM4 패킷에는 전역적으로 정의된 모든 신뢰 지점에 대한 여러 인증서 요청 페이로드가 포함됩니다(**ca trust-point** 명령을 고려하지 않은 경우). 이는 ISAKMP 응답자가 MM5와 MM4에 포함된 인증서 요청을 받은 후에만 사용해야 하는 ISAKMP 프로파일을 결정할 수 있기 때문입니다.

첫 번째 일치 규칙 때문에 MM3 및 MM4의 인증서 요청 페이로드가 중요합니다.첫 번째 일치 규칙은 MM5 및 MM6에서 인증에 필요한 인증서 선택에 사용되는 신뢰 지점을 결정합니다.

인증서 요청 페이로드의 순서는 설치된 인증서의 순서에 따라 달라집니다.show crypto pki certificate 명령의 출력에 나타나는 첫 번째 인증서**의 발급자가** 먼저 전송됩니다.이 첫 번째 인증서는 마지막으로 등록된 인증서입니다.

ISAKMP 프로필에 대해 여러 신뢰 지점을 구성할 수 있습니다.이 작업을 수행해도 이전 규칙은 모두 적용됩니다.

이 문서에 설명된 모든 문제 및 주의 사항은 IKEv1 프로토콜 설계로 인해 발생합니다.인증 단계는 MM5와 MM6에서 발생하는 반면, 인증(인증서 요청)에 대한 제안서는 사용해야 하는 ISAKMP 프로파일을 알지 못한 채 이전 단계(앞쪽)에서 전송해야 합니다.이는 Cisco 고유의 문제가 아니며 IKEv1 프로토콜 설계의 제한과 관련이 있습니다.

IKEv2 프로토콜은 인증서 협상 프로세스와 관련하여 IKEv1과 유사합니다.그러나 IOS에서 구현하면 이니시에이터에 대해 특정 신뢰 지점을 사용해야 합니다.그렇다고 모든 문제가 해결되지는 않습니다.단일 프로파일에 대해 여러 신뢰 지점이 구성되고 다른 쪽에 단일 신뢰 지점이 구성된 경우에도 인증 문제가 발생할 수 있습니다.Cisco에서는 연결의 양쪽에 대칭 신뢰 지점 컨피그레이션을 사용하는 것이 좋습니다(IKEv2 프로파일 모두에 대해 구성된 동일한 신뢰 지점).

다음은 이 문서에 설명된 정보에 대한 몇 가지 중요한 참고 사항입니다.

- 피어의 IKEv1 프로파일에 대해 비대칭 신뢰 지점 컨피그레이션을 사용할 경우 터널은 터널의 한 쪽에서만 시작할 수 있습니다.IKEv1 프로파일에 대한 신뢰 지점 컨피그레이션은 선택 사항입니다.

- 피어의 IKEv2 프로파일에 대해 비대칭 신뢰 지점 컨피그레이션을 사용할 경우 터널은 터널의 한 쪽에서만 시작할 수 있습니다.IKEv2 프로파일에 대한 신뢰 지점 컨피그레이션은 이니시에이터에 필수입니다.

- 인증서 요청 페이로드 순서는 show crypto pki certificate 명령(첫 번째 일치)의 출력에 나타나는 **인증서**의 순서에 따라 달라집니다.

- 인증서 요청 페이로드 순서는 응답자가 선택한 인증서(첫 번째 일치)를 결정합니다.
- IKEv1 및 IKEv2에 대해 여러 프로필을 사용하고 동일한 일치 ID 규칙이 구성된 경우 결과를 예측하기가 어렵습니다(너무 많은 관련 요소).
- IKEv1 및 IKEv2 모두에 대칭 신뢰 지점 컨피그레이션을 사용하는 것이 좋습니다.

관련 정보

- Internet Key Exchange for IPsec VPNs 컨피그레이션 가이드, Cisco IOS 릴리스 15M&T - 인증서와 ISAKMP 프로파일 매핑
- Cisco IOS Security 명령 참조:명령 A~C - clear eou를 통해 ca trust-point
- 기술 지원 및 문서 – Cisco Systems