FIPS가 활성화된 경우 AnyConnect 암호화 알고리즘 오류 수정

목차

[소개](#)
[배경 정보](#)
[문제](#)
[솔루션](#)

소개

이 문서에서는 사용자가 FIPS 지원 암호화 알고리즘을 지원하는 정책이 있는 ASA(Adaptive Security Appliance)에 FIPS(Federal Information Processing Standard) 지원 클라이언트를 사용하여 연결할 수 없는 이유에 대해 설명합니다.

배경 정보

IKEv2(Internet Key Exchange Version 2) 연결을 설정하는 동안 개시자는 피어에서 어떤 제안을 수락할 수 있는지 알지 못하므로 개시자는 첫 번째 IKE 메시지가 전송될 때 어떤 DH(Diffie-Hellman) 그룹을 사용할지 추측해야 합니다. 이 추측에 사용되는 DH 그룹은 일반적으로 구성된 DH 그룹 목록에서 첫 번째 DH 그룹입니다. 그런 다음 개시자는 추측된 그룹에 대한 키 데이터를 계산하지만 모든 그룹의 전체 목록을 피어에 보냅니다. 그러면 피어가 추측된 그룹이 잘못된 경우 다른 DH 그룹을 선택할 수 있습니다.

클라이언트의 경우 사용자가 구성한 IKE 정책 목록이 없습니다. 대신 클라이언트가 지원하는 사전 구성된 정책 목록이 있습니다. 이러한 이유로, 잘못된 그룹을 사용하여 첫 번째 메시지의 키 데이터를 계산할 때 클라이언트의 계산 로드를 줄이기 위해 DH 그룹 목록이 가장 취약성에서 가장 강한 그룹으로 정렬되었습니다. 따라서 클라이언트는 컴퓨팅 집약도가 가장 낮은 DH를 선택하여 초기 추측에 가장 적은 리소스를 소모하는 그룹을 선택하지만 이후 메시지에서 헤드엔드에 의해 선택된 그룹으로 전환합니다.

> **참고:** 이 동작은 DH 그룹을 가장 강한 것에서 가장 약한 것으로 주문한 AnyConnect 버전 3.0 클라이언트와 다릅니다.

그러나 헤드엔드에서 게이트웨이에 구성된 DH 그룹과 일치하는 클라이언트가 보낸 목록의 첫 번째 DH 그룹은 선택한 그룹입니다. 따라서 ASA가 더 약한 DH 그룹을 구성한 경우, 양쪽 끝에서 더 안전한 DH 그룹을 사용할 수 있음에도 불구하고 클라이언트에서 지원되고 헤드엔드에 구성된 가장 약한 DH 그룹을 사용합니다.

이 동작은 Cisco 버그 ID CSCub92935를 통해 클라이언트에서 수정되었습니다. 이 버그의 수정 사항이 있는 모든 클라이언트 버전은 헤드엔드로 보낼 때 DH 그룹이 나열되는 순서를 반대로 합니다. 그러나 비-Suite B 게이트웨이와의 역호환성 문제를 방지하기 위해 가장 약한 DH 그룹(비-FIPS 모드용 그룹 및 FIPS 모드용 2개)은 목록의 맨 위에 있습니다.

> **참고:** 1차 명단(1차 명단, 2차 명단 등)이 가장 약한 놈부터 가장 강한 놈순으로 나열된다. 이렇

게 하면 타원 곡선 그룹이 먼저(21, 20, 19), MODP(Modular Exponder) 그룹(24, 14, 5, 2)이 그 뒤에 옵니다.

팁: 게이트웨이가 동일한 정책에서 여러 DH 그룹으로 구성되고 그룹 1(또는 FIPS 모드의 2개)이 포함된 경우 ASA는 더 약한 그룹을 수락합니다. 이 문제는 게이트웨이에 구성된 정책에 DH 그룹 1만 포함시키는 것입니다. 하나의 정책에 여러 그룹이 구성되었지만 그룹 1이 포함되지 않은 경우 가장 강력한 그룹이 선택됩니다. 예를 들면 다음과 같습니다.

- IKEv2 정책이 1 2 5 14 24 19 20 21로 설정된 ASA 버전 9.0(제품군 B)에서 **그룹 1이** 예상대로 **선택됩니다**.

- IKEv2 정책이 2 5 14 24 19 20 21로 설정된 ASA 버전 9.0(제품군 B)에서 **그룹 21이** 예상대로 **선택됩니다**.

- IKEv2 정책이 1 2 5 14 24 19 20 21로 설정된 ASA 버전 9.0(suite B)에서 클라이언트가 FIPS 모드로 설정된 경우 **그룹 2가** 예상대로 선택됩니다.

- IKEv2 정책이 5 14 24 19 20 21로 설정된 ASA Version 9.0(Suite B)에서 테스트된 클라이언트가 FIPS 모드로 설정된 경우 **그룹 21이** 예상대로 선택됩니다.

- IKEv2 정책이 1 2 5 14로 설정된 ASA 버전 8.4.4(비B)에서 **그룹 1이** 예상대로 **선택됩니다**.

- IKEv2 정책이 2 5 14로 설정된 ASA 버전 8.4.4(비B)에서 **그룹 14가** 예상대로 **선택됩니다**.

문제

ASA는 다음 IKEv2 정책으로 구성됩니다.

```
crypto ikev2 policy 1
encryption aes-gcm-256
integrity null
group 20
prf sha384 sha
lifetime seconds 86400
crypto ikev2 policy 10
encryption aes-192
integrity sha
group 5 2
prf sha
lifetime seconds 86400
crypto ikev2 policy 20
encryption aes
integrity sha
group 5 2
prf sha
lifetime seconds 86400
```

이 컨피그레이션에서는 모든 FIPS 지원 암호화 알고리즘을 지원하기 위해 정책 1이 명확하게 구성됩니다. 그러나 사용자가 FIPS 지원 클라이언트에서 연결을 시도하면 다음과 같은 오류 메시지와 함께 연결이 실패합니다.

```
The cryptographic algorithms required by the secure gateway do not match those supported by
AnyConnect.
```

```
Please contact your network administrator.
```

그러나 관리자가 20 대신 DH 그룹 2를 사용하도록 policy1을 변경하면 연결이 작동합니다.

솔루션

이 증상에 따라 첫 번째 결론은 FIPS가 활성화되고 다른 어떤 것도 작동하지 않을 때 클라이언트가 DH 그룹 2만 지원한다는 것입니다. 이것은 사실 틀립니다. ASA에서 이 디버그를 활성화하면 클라이언트에서 보낸 제안을 볼 수 있습니다.

debug crypto ikev2 proto 127

연결을 시도하는 동안 첫 번째 디버그 메시지는 다음과 같습니다.

```
IKEv2-PROTO-2: Received Packet [From 192.168.30.5:51896/To 192.168.30.2:500/
VRF i0:f0]
Initiator SPI : 74572B8D1BEC5873 - Responder SPI : 0000000000000000 Message id: 0
IKEv2 IKE_SA_INIT Exchange REQUESTIKEv2-PROTO-3: Next payload: SA, version:
2.0 Exchange type: IKE_SA_INIT, flags: INITIATOR Message id: 0, length: 747
Payload contents:
SA Next payload: KE, reserved: 0x0, length: 316
last proposal: 0x2, reserved: 0x0, length: 140
Proposal: 1, Protocol id: IKE, SPI size: 0, #trans: 15 last transform: 0x3,
reserved: 0x0: length: 12
type: 1, reserved: 0x0, id: AES-GCM
last transform: 0x3, reserved: 0x0: length: 12
type: 1, reserved: 0x0, id: AES-GCM
last transform: 0x3, reserved: 0x0: length: 12
type: 1, reserved: 0x0, id: AES-GCM
last transform: 0x3, reserved: 0x0: length: 8
type: 2, reserved: 0x0, id: SHA512
last transform: 0x3, reserved: 0x0: length: 8
type: 2, reserved: 0x0, id: SHA384
last transform: 0x3, reserved: 0x0: length: 8
type: 2, reserved: 0x0, id: SHA256
last transform: 0x3, reserved: 0x0: length: 8
type: 2, reserved: 0x0, id: SHA1
last transform: 0x3, reserved: 0x0: length: 8
type: 3, reserved: 0x0, id: None
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_1024_MODP/Group 2
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_521_ECP/Group 21
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_384_ECP/Group 20
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_256_ECP/Group 19
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_2048_MODP_256_PRIME/Group 24
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_2048_MODP/Group 14
last transform: 0x0, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_1536_MODP/Group 5
last proposal: 0x0, reserved: 0x0, length: 172
Proposal: 2, Protocol id: IKE, SPI size: 0, #trans: 19 last transform: 0x3,
reserved: 0x0: length: 12
type: 1, reserved: 0x0, id: AES-CBC
last transform: 0x3, reserved: 0x0: length: 12
type: 1, reserved: 0x0, id: AES-CBC
last transform: 0x3, reserved: 0x0: length: 12
```

```
type: 1, reserved: 0x0, id: AES-CBC
last transform: 0x3, reserved: 0x0: length: 8
type: 1, reserved: 0x0, id: 3DES
last transform: 0x3, reserved: 0x0: length: 8
type: 2, reserved: 0x0, id: SHA512
last transform: 0x3, reserved: 0x0: length: 8
type: 2, reserved: 0x0, id: SHA384
last transform: 0x3, reserved: 0x0: length: 8
type: 2, reserved: 0x0, id: SHA256
last transform: 0x3, reserved: 0x0: length: 8
type: 2, reserved: 0x0, id: SHA1
last transform: 0x3, reserved: 0x0: length: 8
type: 3, reserved: 0x0, id: SHA512
last transform: 0x3, reserved: 0x0: length: 8
type: 3, reserved: 0x0, id: SHA384
last transform: 0x3, reserved: 0x0: length: 8
type: 3, reserved: 0x0, id: SHA256
last transform: 0x3, reserved: 0x0: length: 8
type: 3, reserved: 0x0, id: SHA96
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_1024_MODP/Group 2
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_521_ECP/Group 21
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_384_ECP/Group 20
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_256_ECP/Group 19
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_2048_MODP_256_PRIME/Group 24
last transform: 0x3, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_2048_MODP/Group 14
last transform: 0x0, reserved: 0x0: length: 8
type: 4, reserved: 0x0, id: DH_GROUP_1536_MODP/Group 5
KE Next payload: N, reserved: 0x0, length: 136
DH group: 2, Reserved: 0x0

fc c9 90 2b 15 35 31 34 0e 75 88 c0 f9 2a 1e 0a
a5 6b e3 8e e1 73 b9 d1 56 1e 60 9f 82 71 6c 4e
5c 1c a4 bd b5 23 a2 bc 82 f2 11 17 61 28 33 3f
02 c9 e7 cb f7 84 a6 22 4a 64 eb fa d7 84 a1 d9
ad c7 5d 77 cd 2a 65 79 95 9a d4 5c 22 8c 62 ae
0e fc c8 fd bd c8 4d 66 0d c3 69 d3 c4 cb e8 33
72 1a f1 cc 31 5f 08 75 65 6b 77 3b 23 c3 b8 74
02 fa 15 6e e4 7a b2 73 17 8f 08 02 20 7e b8 d7
N Next payload: VID, reserved: 0x0, length: 24

87 4d 63 76 cc 10 30 0e 4c 95 40 24 d3 b3 3b f3
44 be 0f e5
```

따라서 클라이언트가 그룹 2,21,20,19,24,14 및 5(이러한 FIPS 호환 그룹)를 보냈지만 헤드엔드는 여전히 이전 컨피그레이션의 정책 1에서 그룹 2-enabled만 연결합니다. 이 문제는 디버그에서 더 명확하게 나타납니다.

```
IKEv2 received all requested SPIs from CTM to respond to a tunnel request.
IKEv2-PROTO-5: (64): SM Trace-> SA: I_SPI=74572B8D1BEC5873 R_SPI=E4160C492A824B5F
(R) MsgID = 00000006 CurState: R_VERIFY_AUTH Event: EV_OK_RECD_IPSEC_RESP
IKEv2-PROTO-2: (64): Processing IKE_AUTH message
IKEv2-PROTO-1: Tunnel Rejected: Selected IKEv2 encryption algorithm (AES-CBC-192)
is not strong enough to secure proposed IPsec encryption algorithm (AES-GCM-256).
IKEv2-PROTO-1: (64): Failed to find a matching policy
IKEv2-PROTO-1: (64): Received Policies:
ESP: Proposal 1: AES-GCM-256 AES-GCM-192 AES-GCM-128 None Don't use ESN
```

```
ESP: Proposal 2: AES-CBC-256 AES-CBC-192 AES-CBC-128 3DES SHA512 SHA384 SHA256 SHA96
Don't use ESN

IKEv2-PROTO-1: (64): Failed to find a matching policy
IKEv2-PROTO-1: (64): Expected Policies:
ESP: Proposal 0: AES-GCM-256 SHA384 Don't use ESN

IKEv2-PROTO-5: (64): Failed to verify the proposed policies
IKEv2-PROTO-1: (64): Failed to find a matching policy
```

다음과 같은 요인의 조합으로 인해 연결이 실패합니다.

1. FIPS가 활성화된 경우, 클라이언트는 특정 정책만 전송하며 해당 정책은 일치해야 합니다. 이러한 정책 중에서 키 크기가 256보다 크거나 같은 AES(Advanced Encryption Standard) 암호화만 제안합니다.

2. ASA는 여러 IKEv2 정책으로 구성되며, 이 중 2개는 그룹 2가 활성화되었습니다. 앞에서 설명한 것처럼 이 시나리오에서는 그룹 2가 활성화된 정책이 연결에 사용됩니다. 그러나 두 정책의 암호화 알고리즘은 키 크기 192를 사용하며, 이는 FIPS 지원 클라이언트에 비해 너무 낮습니다.

따라서 이 경우 ASA와 클라이언트는 컨피그레이션에 따라 작동합니다. FIPS 지원 클라이언트에 대해 이 문제를 해결하는 방법에는 세 가지가 있습니다.

1. 원하는 정확한 제안으로 정책을 하나만 구성합니다.

2. 여러 제안이 필요한 경우 그룹 2로 제안서를 구성하지 마십시오. 그렇지 않으면 항상 선택됩니다.

3. 그룹 2를 활성화해야 하는 경우 올바른 암호화 알고리즘(Aes-256 또는 aes-gcm-256)이 구성되어 있는지 확인합니다.