Correzione dell'errore degli algoritmi di crittografia AnyConnect con FIPS abilitato

Sommario

Introduzione

In questo documento viene descritto il motivo per cui gli utenti potrebbero non essere in grado di connettersi a un'appliance ASA (Adaptive Security Appliance) con una policy che supporta gli algoritmi di crittografia FIPS.

Premesse

Durante la configurazione di una connessione IKEv2 (Internet Key Exchange versione 2), l'iniziatore non è mai a conoscenza di quali proposte siano accettabili dal peer, pertanto deve individuare il gruppo Diffie-Hellman (DH) da utilizzare quando viene inviato il primo messaggio IKE. Il gruppo DH utilizzato per questa ipotesi è in genere il primo gruppo DH nell'elenco dei gruppi DH configurati. L'iniziatore calcola quindi i dati chiave per i gruppi indovinati, ma invia anche un elenco completo di tutti i gruppi al peer, che consente al peer di selezionare un gruppo DH diverso se il gruppo indovinato è errato.

Nel caso di un client, non è disponibile un elenco di criteri IKE configurato dall'utente. Esiste invece un elenco preconfigurato di criteri supportati dal client. Per questo motivo, per ridurre il carico di calcolo sul client quando si calcolano i dati chiave per il primo messaggio con un gruppo che probabilmente è quello sbagliato, l'elenco dei gruppi DH è stato ordinato dal più debole al più forte. Pertanto, il client sceglie il DH con il minor utilizzo di risorse computazionali e quindi il gruppo con il minor utilizzo di risorse per la stima iniziale, ma passa al gruppo scelto dall'headend nei messaggi successivi.

> **Nota:** Questo comportamento è diverso dai client AnyConnect versione 3.0 che hanno ordinato i gruppi DH dal più forte al più debole.

Tuttavia, nell'headend, il primo gruppo DH dell'elenco inviato dal client che corrisponde a un gruppo DH configurato sul gateway è il gruppo selezionato. Pertanto, se anche l'ASA ha gruppi DH più deboli configurati, viene usato il gruppo DH più debole supportato dal client e configurato sull'headend, nonostante la disponibilità di un gruppo DH più sicuro su entrambi i lati.

Questo comportamento è stato risolto sul client con l'ID bug Cisco CSCub92935. Tutte le versioni del client in cui il bug è stato risolto invertono l'ordine in cui i gruppi DH vengono elencati quando vengono inviati all'headend. Tuttavia, per evitare problemi di compatibilità con i gateway non Suite B, il gruppo DH più debole (uno per la modalità non FIPS e due per la modalità FIPS) rimane in

cima all'elenco.

> **Nota:** Dopo la prima voce dell'elenco (gruppo 1 o 2), i gruppi sono elencati in ordine di grandezza da forte a debole. Questo mette i gruppi di curve ellittiche per primi (21, 20, 19), seguiti dai gruppi modulari esponenziali (MODP) (24, 14, 5, 2).

> **Suggerimento**: Se il gateway è configurato con più gruppi DH nello stesso criterio e il gruppo 1 (o 2 in modalità FIPS) è incluso, l'ASA accetta il gruppo più debole. Per risolvere il problema, includere solo il gruppo DH 1 in un criterio configurato sul gateway. Se in un criterio sono configurati più gruppi ma il gruppo 1 non è incluso, viene selezionato il gruppo più forte. Ad esempio:
>
> - Su ASA versione 9.0 (suite B) con criteri IKEv2 impostati su 1 2 5 14 24 19 20 21, il **gruppo 1 viene selezionato** come previsto.
>
> - Su ASA versione 9.0 (suite B) con criteri IKEv2 impostati su 2 5 14 24 19 20 21, il **gruppo 21 viene selezionato** come previsto.
>
> - Se il client è in modalità FIPS su ASA versione 9.0 (suite B) con i criteri IKEv2 impostati su 1 2 5 14 24 19 20 21, il **gruppo 2 viene selezionato** come previsto.
>
> - Se il client testato è in modalità FIPS su ASA versione 9.0 (suite B) con la policy IKEv2 impostata su 5 14 24 19 20 21, il **gruppo 21 viene selezionato** come previsto.
>
> - Su ASA versione 8.4.4 (non suite B) con criteri IKEv2 impostati su 1 2 5 14, **viene selezionato il gruppo 1** come previsto.
>
> - Su ASA versione 8.4.4 (non suite B) con criteri IKEv2 impostati su 2.5.14, **viene selezionato il gruppo 14** come previsto.

Problema

L'ASA è configurata con i seguenti criteri IKEv2:

```
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
```

In questa configurazione, la policy 1 è configurata in modo chiaro per supportare tutti gli algoritmi di crittografia abilitati per FIPS. Tuttavia, quando un utente tenta di connettersi da un client abilitato per FIPS, la connessione non riesce e viene visualizzato il messaggio di errore:

```
The cryptographic algorithms required by the secure gateway do not match those supported by
AnyConnect.
Please contact your network administrator.
```

Tuttavia, se l'amministratore modifica policy1 in modo che utilizzi il gruppo DH 2 anziché 20, la connessione funziona.

Soluzione

In base ai sintomi, la prima conclusione è che il client supporta il gruppo DH 2 solo quando FIPS è abilitato e nessuno degli altri funziona. Questo in realtà non è corretto. Se si abilita il debug sull'appliance ASA, sarà possibile visualizzare le proposte inviate dal client:

```
debug crypto ikev2 proto 127
```

Durante un tentativo di connessione, il primo messaggio di debug è:

```
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
```

Pertanto, nonostante il client abbia inviato ai gruppi 2,21,20,19,24,14 e 5 (questi gruppi compatibili con FIPS), l'headend connette solo il gruppo 2 abilitato nel criterio 1 della configurazione precedente. Questo problema diventa evidente più avanti nei debug:

```
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
```

La connessione non riesce a causa di una combinazione di fattori:

1. Se FIPS è abilitato, il client invia solo criteri specifici che devono corrispondere. Tra queste, si propone solo la crittografia AES (Advanced Encryption Standard) con una dimensione della chiave pari o superiore a 256.

2. L'ASA è configurata con più criteri IKEv2, due dei quali con il gruppo 2 abilitato. Come descritto in precedenza, in questo scenario per la connessione viene utilizzato il criterio con il gruppo 2 abilitato. Tuttavia, l'algoritmo di crittografia di entrambi i criteri utilizza una dimensione della chiave di 192, che è troppo bassa per un client abilitato per FIPS.

Pertanto, in questo caso, l'ASA e il client si comportano secondo la configurazione. Per risolvere il problema relativo ai client abilitati per FIPS, è possibile procedere in tre modi:

1. Configurare un solo criterio con le proposte esatte desiderate.

2. Se sono necessarie più proposte, non configurarne una con il gruppo 2; altrimenti è sempre selezionato.

3. Se è necessario abilitare il gruppo 2, verificare che disponga dell'algoritmo di crittografia corretto configurato (Aes-256 o aes-gcm-256).