Risoluzione dei problemi di IPSec per i tunnel di servizio sui bordi con IKEv2

Sommario

Introduzione

In questo documento viene descritto come risolvere i problemi più comuni dei tunnel IPsec (Internet Protocol Security) di dispositivi di terze parti con Internet Key Exchange versione 2 (IKEv2) configurato. Nella documentazione di Cisco SD-WAN, i tunnel di servizio/trasporto vengono comunemente chiamati tunnel. Questo documento spiega anche come abilitare e leggere i debug IKE e associarli allo scambio di pacchetti per capire il punto di errore su una negoziazione IPsec.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- IKEv2
- Negoziazione IPSec
- Cisco SD-WAN

Componenti usati

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

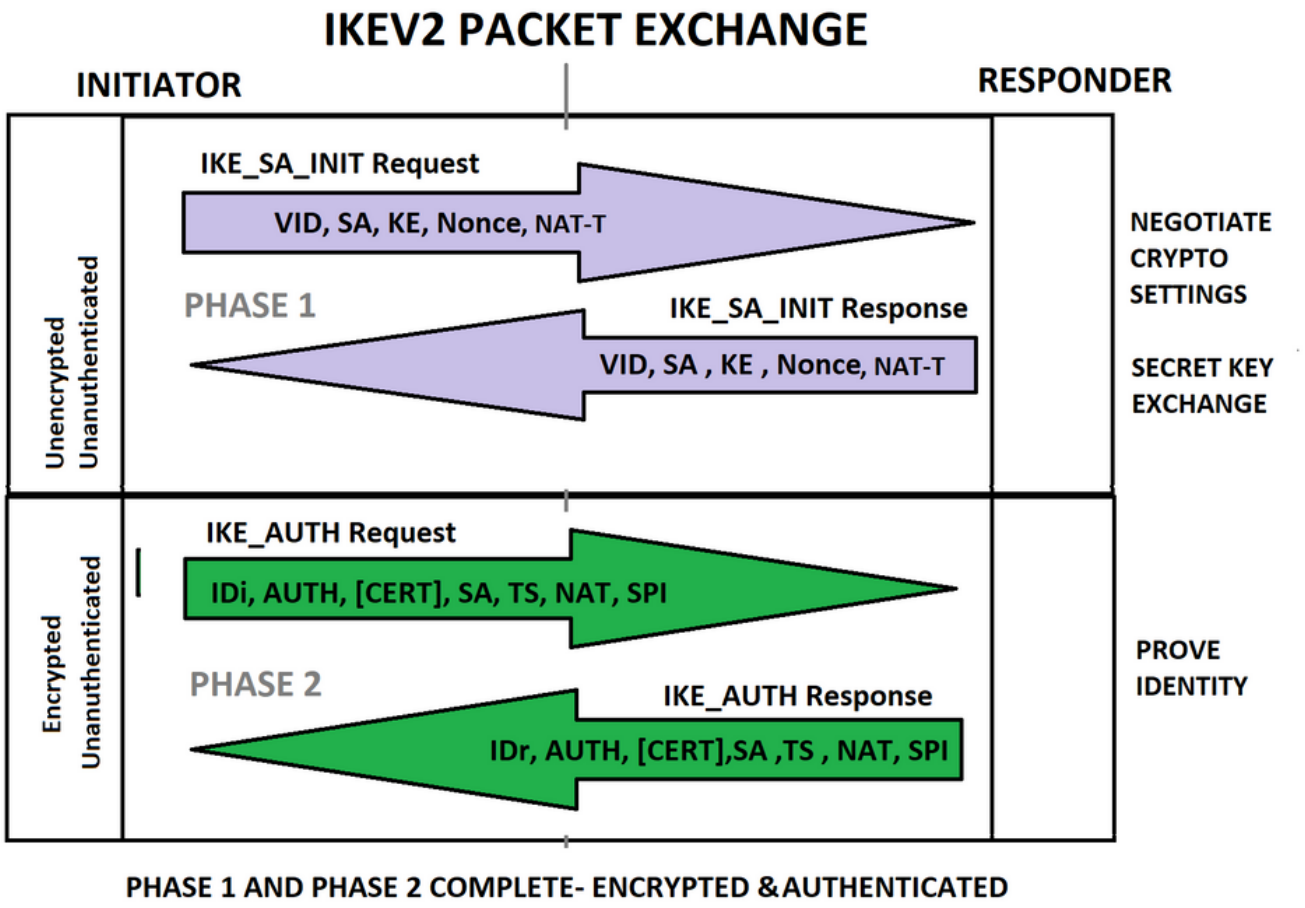
Premesse

Glossario IKE

- **IPSec (Internet Protocol Security)** è una suite standard di protocolli tra 2 punti di comunicazione sulla rete IP che forniscono autenticazione, integrità e riservatezza dei dati.
- **IKEv2 (Internet Key Exchange versione 2) è il protocollo utilizzato per configurare un'associazione di sicurezza (SA, Security Association) nella suite di protocolli IPsec.**
- Un'**associazione di sicurezza** (**SA**) è la definizione di attributi di sicurezza condivisi tra due entità di rete per supportare la comunicazione protetta. Un'associazione di protezione può includere attributi quali l'algoritmo e la modalità di crittografia; chiave di crittografia del traffico; e i parametri per i dati di rete da passare attraverso la connessione.
- Gli **ID dei fornitori** (**VID**) vengono usati per identificare i dispositivi peer con la stessa implementazione del fornitore e supportare così le funzionalità specifiche del fornitore.
- **Nessuna**: valori casuali creati nello scambio per aggiungere casualità e prevenire attacchi di tipo replay.
- Informazioni **KE (Key-Exchange)** per il processo di scambio sicuro delle chiavi Diffie-Hellman (DH).
- **Identity Initiator/responder (IDi/IDr.)** viene utilizzato per inviare informazioni di autenticazione al peer. Queste informazioni vengono trasmesse sotto la protezione del segreto condiviso comune.
- La chiave condivisa IPSec può essere derivata utilizzando nuovamente DH per garantire il **PFS (Perfect Forward Secrecy)** o aggiornando il segreto condiviso derivato dallo scambio DH originale.
- **Lo scambio di chiavi Diffie-Hellman (DH) è un metodo per lo scambio sicuro di algoritmi crittografici su un canale pubblico.**
- **I selettori del traffico (TS, Traffic Selectors)** sono le identità proxy o il traffico scambiato nella negoziazione IPsec per passare attraverso il tunnel crittografato.

Packet Exchange IKEv2

Ogni pacchetto IKE contiene le informazioni sul payload per la definizione del tunnel. Il glossario IKE spiega le abbreviazioni mostrate in questa immagine come parte del contenuto del payload per lo scambio di pacchetti.

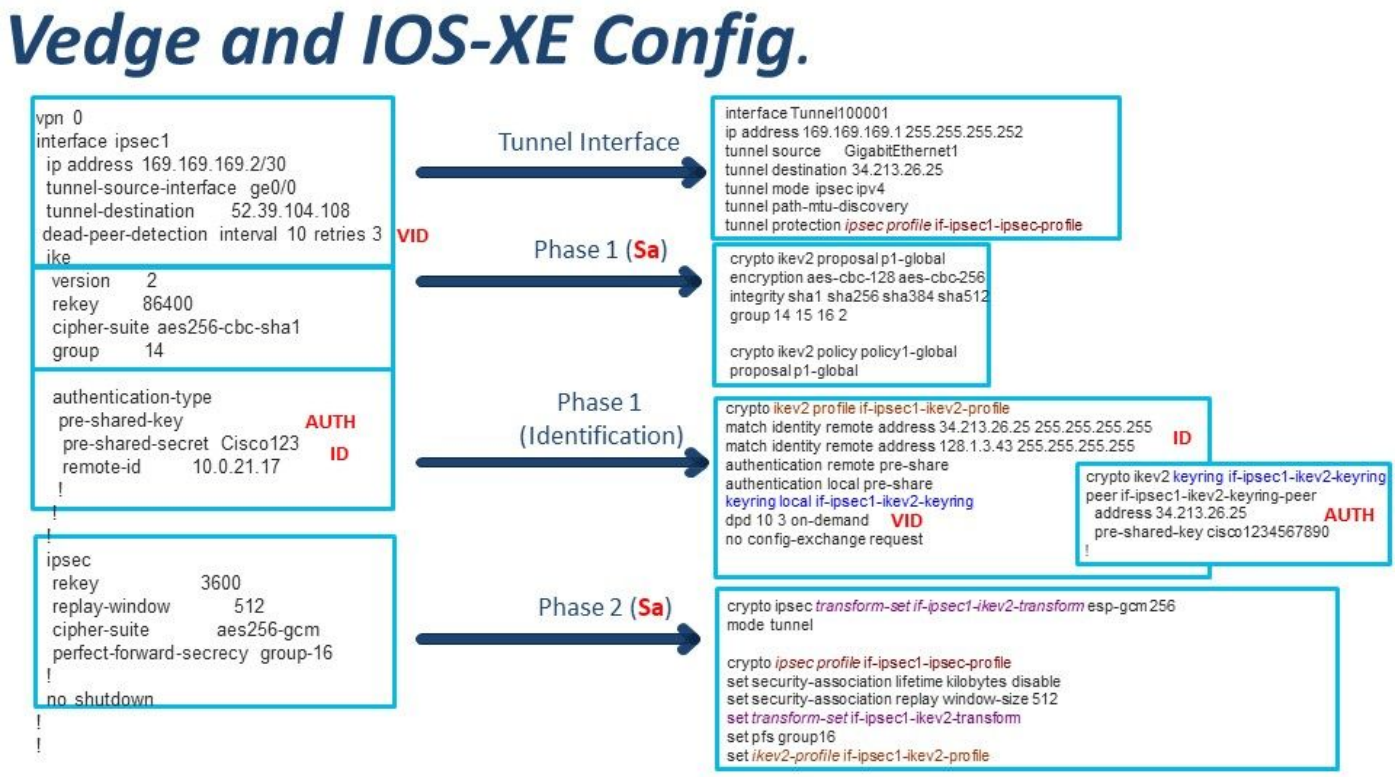


IKEV2-Exchange

> **Nota:** È importante verificare su quale scambio di pacchetti della negoziazione IKE il tunnel IPsec non riesce ad analizzare rapidamente la configurazione interessata per risolvere efficacemente il problema.

> **Nota:** In questo documento non viene fornita una descrizione più dettagliata dello scambio di pacchetti IKEv2. Per ulteriori riferimenti, passare a Packet Exchange IKEv2 e debug a livello di protocollo

Ènecessario per correlare la configurazione vEdge alla configurazione Cisco IOS® XE. Inoltre, è utile far corrispondere i concetti di IPSec e il contenuto del payload per gli scambi di pacchetti IKEv2, come mostrato nell'immagine.



Nota: Ogni parte della configurazione modifica un aspetto dello scambio di negoziazione IKE. È importante correlare i comandi alla negoziazione del protocollo di IPSec.

Risoluzione dei problemi

Abilita debug IKE

On Edge **debug iked** abilita le informazioni a livello di debug per IKEv1 o IKEv2.

```
debug iked misc high
debug iked event high
```

È possibile visualizzare le informazioni di debug correnti in **vshell** ed eseguire il comando **tail -f <percorso di debug>**.

```
vshell
tail -f /var/log/message
```

Nella CLI è anche possibile visualizzare le informazioni correnti di log/debug per il percorso specificato.

```
monitor start /var/log/messages
```

Suggerimenti per avviare il processo di risoluzione dei problemi di IPSec

Èpossibile separare tre diversi scenari IPsec. È un buon punto di riferimento identificare il sintomo per avere un approccio migliore per sapere come iniziare.

1. Tunnel IPsec non stabilito.

2. Il tunnel IPsec è crollato e si è ristabilito da solo. (Lampeggiato)
3. Il tunnel IPsec è stato interrotto e rimane in stato di inattività.

Poiché il tunnel IPsec non stabilisce i sintomi, è necessario eseguire il debug in tempo reale per verificare il comportamento corrente nella negoziazione IKE.

Per il tunnel IPsec è stato interrotto e ristabilito sui propri sintomi, più comunemente noti come tunnel Flapped ed è necessaria l'analisi della causa principale (RCA). È indispensabile conoscere la data e l'ora in cui il tunnel è crollato o avere un tempo stimato per esaminare i debug.

Se il tunnel IPsec è crollato e continua a presentare sintomi di downstate, significa che il tunnel ha funzionato prima ma, per qualsiasi motivo, è crollato ed è necessario conoscere il motivo della disinstallazione e il comportamento corrente che impedisce la riattivazione del tunnel.

Identificare i punti prima dell'inizio della risoluzione dei problemi:

1. Tunnel IPsec (numero) con problemi e configurazione.
2. Timestamp dell'interruzione del tunnel (se applicabile).
3. Indirizzo IP peer IPsec (destinazione tunnel).

Tutti i debug e i log vengono salvati sui file **/var/log/messages**, per i log correnti, vengono salvati sui file di messaggi, ma per questo sintomo specifico il flap potrebbe essere identificato ore/giorni dopo il problema, molto probabilmente i debug correlati sarebbero sui messaggi 1,2,3..etc. È importante conoscere il timestamp per esaminare il file di messaggi corretto e analizzare i debug (charon) per la negoziazione IKE dei dati correlati al tunnel IPsec.

La maggior parte dei debug non stampa il numero del tunnel IPsec. Il modo più frequente per identificare la negoziazione e i pacchetti è tramite l'indirizzo IP del peer remoto e l'indirizzo IP da cui il tunnel ha origine sul vedge. Alcuni esempi di debug IKE stampati:

```
Jun 18 00:31:22 vedge01 charon: 09[CFG] vici initiate 'child_IPsec2_1'
Jun 18 00:31:22 vedge01 charon: 16[IKE] initiating IKE_SA ipsec2_1[223798] to 10.10.10.1
Jun 18 00:31:22 vedge01 charon: 16[IKE] initiating IKE_SA ipsec2_1[223798] to 10.10.10.1
```

I debug per la negoziazione INIT IKE mostrano il numero del tunnel IPsec. Tuttavia, le informazioni successive per lo scambio di pacchetti utilizzano solo gli indirizzi IP del tunnel IPsec.

```
Jun 18 00:31:22 vedge01 charon: 09[CFG] vici initiate 'child_ipsec2_1'
Jun 18 00:31:22 vedge01 charon: 16[IKE] initiating IKE_SA ipsec2_1[223798] to 10.10.10.1
Jun 18 00:31:22 vedge01 charon: 16[IKE] initiating IKE_SA ipsec2_1[223798] to 10.10.10.1
Jun 18 00:31:22 vedge01 charon: 16[ENC] generating IKE_SA_INIT request 0 [ SA KE No N(NATD_S_IP)
N(NATD_D_IP) N(FRAG_SUP) N(HASH_ALG) N(REDIR_SUP) ]
Jun 18 00:31:22 vedge01 charon: 16[NET] sending packet: from 10.132.3.92[500] to 10.10.10.1[500]
(464 bytes)
Jun 18 00:31:22 vedge01 charon: 12[NET] received packet: from 10.10.10.1[500] to
10.132.3.92[500] (468 bytes)
Jun 18 00:31:22 vedge01 charon: 12[ENC] parsed IKE_SA_INIT response 0 [ SA KE No N(NATD_S_IP)
N(NATD_D_IP) N(HTTP_CERT_LOOK) N(FRAG_SUP) V ]
Jun 18 00:31:22 vedge01 charon: 12[ENC] received unknown vendor ID:
4f:85:58:17:1d:21:a0:8d:69:cb:5f:60:9b:3c:06:00
Jun 18 00:31:22 vedge01 charon: 12[IKE] local host is behind NAT, sending keep alives
```

Configurazione tunnel IPsec:

```
interface ipsec2 ip address 192.168.1.9/30 tunnel-source 10.132.3.92 tunnel-destination
10.10.10.1 dead-peer-detection interval 30 ike version 2 rekey 86400 cipher-suite aes256-cbc-
sha1 group 14 authentication-type pre-shared-key pre-shared-secret
$8$wgrs/Cw6tX0na34yF4Fga0B62mGBpHFdOzFaRmoYfnBioWVO3s3efFPBbkaZqvoN ! ! ! ipsec rekey 3600
replay-window 512 cipher-suite aes256-gcm perfect-forward-secrecy group-14 !
```

Sintomo 1. Il tunnel IPsec non viene stabilito

Poiché il problema può essere la prima implementazione del tunnel, non è ancora stato risolto e i debug IKE sono l'opzione migliore.

Sintomo 2. Il tunnel IPsec è stato chiuso ed è stato ristabilito da solo

Come accennato in precedenza, questo sintomo viene in genere affrontato per identificare la causa all'origine del problema. Se l'analisi della causa principale è nota, talvolta l'amministratore della rete evita ulteriori problemi.

Identificare i punti prima dell'inizio della risoluzione dei problemi:

1. Tunnel IPsec (numero) con problemi e configurazione.
2. Timestamp dell'interruzione del tunnel.
3. Indirizzo IP peer IPsec (destinazione tunnel)

Ritrasmissioni DPD

In questo esempio, il tunnel scese il 18 giugno alle 00:31:17.

```
Jun 18 00:31:17 vedge01 FTMD[1472]: %Viptela-vedge01-FTMD-6-INFO-1000001: VPN 1 Interface ipsec2
DOWN
Jun 18 00:31:17 vedge01 FTMD[1472]: %Viptela-vedge01-ftmd-6-INFO-1400002: Notification:
interface-state-change severity-level:major host-name:"vedge01" system-ip:4.0.5.1 vpn-id:1 if-
name:"ipsec2" new-state:down
```

> **Nota:** I log per l'inattività del tunnel IPsec non fanno parte dei debug di tipo Iked ma sono log *FTMD*. Pertanto, né *charon* né *IKE* verrebbero stampati.

> **Nota:** I registri correlati non vengono in genere stampati insieme, ma contengono più informazioni non correlate allo stesso processo.

Passaggio 1. Dopo aver identificato l'indicatore orario e aver correlato l'ora e i registri, iniziare a esaminare i registri dal basso verso l'alto.

```
Jun 18 00:31:17 vedge01 charon: 11[IKE] giving up after 3 retransmits

Jun 18 00:28:22 vedge01 charon: 08[IKE] retransmit 3 of request with message ID 543 (tries=3,
timeout=30, exchange=37, state=2)
Jun 18 00:28:22 vedge01 charon: 08[NET] sending packet: from 10.132.3.92[4500] to
10.10.10.1[4500] (76 bytes)

Jun 18 00:26:45 vedge01 charon: 06[IKE] retransmit 2 of request with message ID 543 (tries=3,
```

```
timeout=30, exchange=37, state=2)
Jun 18 00:26:45 vedge01 charon: 06[NET] sending packet: from 10.132.3.92[4500] to
10.10.10.1[4500] (76 bytes)

Jun 18 00:25:21 vedge01 charon: 08[IKE] sending DPD request
Jun 18 00:25:21 vedge01 charon: 08[ENC] generating INFORMATIONAL request 543 [ ]
Jun 18 00:25:21 vedge01 charon: 08[NET] sending packet: from 10.132.3.92[4500] to
10.10.10.1[4500] (76 bytes)
Jun 18 00:25:51 vedge01 charon: 05[IKE] retransmit 1 of request with message ID 543 (tries=3,
timeout=30, exchange=37, state=2)
Jun 18 00:25:51 vedge01 charon: 05[NET] sending packet: from 10.132.3.92[4500] to
10.10.10.1[4500] (76 bytes)
```

L'ultimo scambio di pacchetti DPD riuscito è descritto come richiesta n. 542.

```
Jun 18 00:24:08 vedge01 charon: 11[ENC] generating INFORMATIONAL request 542 [ ]
Jun 18 00:24:08 vedge01 charon: 11[NET] sending packet: from 10.132.3.92[4500] to
10.10.10.1[4500] (76 bytes)
Jun 18 00:24:08 vedge01 charon: 07[NET] received packet: from 13.51.17.190[4500] to
10.10.10.1[4500] (76 bytes)
Jun 18 00:24:08 vedge01 charon: 07[ENC] parsed INFORMATIONAL response 542 [ ]
```

Passaggio 2. Mettere insieme tutte le informazioni nell'ordine corretto:

```
Jun 18 00:24:08 vedge01 charon: 11[ENC] generating INFORMATIONAL request 542 [ ]
Jun 18 00:24:08 vedge01 charon: 11[NET] sending packet: from 10.132.3.92[4500] to
10.10.10.1[4500] (76 bytes)
Jun 18 00:24:08 vedge01 charon: 07[NET] received packet: from 10.10.10.1[4500] to
10.132.3.92[4500] (76 bytes)
Jun 18 00:24:08 vedge01 charon: 07[ENC] parsed INFORMATIONAL response 542 [ ]

Jun 18 00:25:21 vedge01 charon: 08[IKE] sending DPD request
Jun 18 00:25:21 vedge01 charon: 08[ENC] generating INFORMATIONAL request 543 [ ]
Jun 18 00:25:21 vedge01 charon: 08[NET] sending packet: from 10.132.3.92[4500] to
10.10.10.1[4500] (76 bytes)
Jun 18 00:25:51 vedge01 charon: 05[IKE] retransmit 1 of request with message ID 543 (tries=3,
timeout=30, exchange=37, state=2)
Jun 18 00:25:51 vedge01 charon: 05[NET] sending packet: from 10.132.3.92[4500] to
10.10.10.1[4500] (76 bytes)

Jun 18 00:26:45 vedge01 charon: 06[IKE] retransmit 2 of request with message ID 543 (tries=3,
timeout=30, exchange=37, state=2)
Jun 18 00:26:45 vedge01 charon: 06[NET] sending packet: from 10.132.3.92[4500] to
10.10.10.1[4500] (76 bytes)

Jun 18 00:28:22 vedge01 charon: 08[IKE] retransmit 3 of request with message ID 543 (tries=3,
timeout=30, exchange=37, state=2)
Jun 18 00:28:22 Lvedge01 charon: 08[NET] sending packet: from 10.132.3.92[4500] to
10.10.10.1[4500] (76 bytes)

Jun 18 00:31:17 vedge01 charon: 11[IKE] giving up after 3 retransmits
Jun 18 00:31:17 vedge01 FTMD[1472]: %Viptela-LONDSR01-FTMD-6-INFO-1000001: VPN 1 Interface
ipsec2 DOWN
Jun 18 00:31:17 vedge01 FTMD[1472]: %Viptela-LONDSR01-ftmd-6-INFO-1400002: Notification:
interface-state-change severity-level:major host-name:"LONDSR01" system-ip:4.0.5.1 vpn-id:1 if-
name:"ipsec2" new-state:down
```

Per l'esempio descritto, il tunnel scade perché vEdge01 non riceve i pacchetti DPD da 10.10.10.1. È previsto che dopo 3 ritrasmissioni DPD il peer IPsec venga impostato come "perso" e il tunnel scada. Le cause di questo comportamento sono molteplici e in genere sono correlate all'ISP dove i pacchetti vengono persi o scartati nel percorso. Se il problema si verifica una volta, non è possibile tenere traccia del traffico perso. Se il problema persiste, è possibile tenere traccia del

pacchetto usando le acquisizioni su vEdge, il peer IPSec remoto e l'ISP.

Sintomo 3. Il tunnel IPsec è stato interrotto e rimane in stato di inattività

Come accennato in precedenza, il tunnel funzionava bene ma per qualsiasi motivo è crollato e non è stato possibile ristabilirlo. Questo scenario ha un impatto sulla rete.

identificare i punti prima dell'inizio della risoluzione dei problemi:

1. Tunnel IPsec (numero) con problemi e configurazione.
2. Timestamp dell'interruzione del tunnel.
3. Indirizzo IP peer IPsec (destinazione tunnel)

PFS non corrispondente

Nell'esempio, la risoluzione dei problemi non inizia con l'indicatore orario quando il tunnel diventa inattivo. Poiché il problema persiste, i debug IKE sono la soluzione migliore.

```
interface ipsec1 description VWAN_VPN ip address 192.168.0.101/30 tunnel-source-interface ge0/0
tunnel-destination 10.10.10.1 ike version 2 rekey 28800 cipher-suite aes256-cbc-sha1 group 2
authentication-type pre-shared-key pre-shared-secret
"$8$njK2pLLjgKWNQu0KecNtY3+fo3hbTs0/7iJy6unNtersmCGjGB38kIPjsoqqXZdVmtizLu79\naQdjt2POM242Yw=="
! ! ! ipsec rekey 3600 replay-window 512 cipher-suite aes256-cbc-sha1 perfect-forward-secrecy
group-16 ! mtu 1400 no shutdown
```

Il collegamento debug è abilitato e viene visualizzata la negoziazione.

```
daemon.info: Apr 27 05:12:56 vedge01 charon: 16[NET] received packet: from 10.10.10.1[4500] to
172.28.0.36[4500] (508 bytes)
daemon.info: Apr 27 05:12:56 vedge01 charon: 16[ENC] parsed CREATE_CHILD_SA request 557 [ SA No
TSi TSr ]
daemon.info: Apr 27 05:12:56 vedge01 charon: 16[CFG] received proposals:
ESP:AES_GCM_16_256/NO_EXT_SEQ, ESP:AES_CBC_256/HMAC_SHA1_96/NO_EXT_SEQ,
ESP:3DES_CBC/HMAC_SHA1_96/NO_EXT_SEQ, ESP:AES_CBC_256/HMAC_SHA2_256_128/NO_EXT_SEQ,
ESP:AES_CBC_128/HMAC_SHA1_96/NO_EXT_SEQ, ESP:3DES_CBC/HMAC_SHA2_256_128/NO_EXT_SEQ
daemon.info: Apr 27 05:12:56 vedge01 charon: 16[CFG] configured proposals:
ESP:AES_CBC_256/HMAC_SHA1_96/MODP_4096/NO_EXT_SEQ
daemon.info: Apr 27 05:12:56 vedge01 charon: 16[IKE] no acceptable proposal found
daemon.info: Apr 27 05:12:56 vedge01 charon: 16[IKE] failed to establish CHILD_SA, keeping
IKE_SA
daemon.info: Apr 27 05:12:56 vedge01 charon: 16[ENC] generating CREATE_CHILD_SA response 557 [
N(NO_PROP) ]
daemon.info: Apr 27 05:12:56 vedge01 charon: 16[NET] sending packet: from 172.28.0.36[4500] to
10.10.10.1[4500] (76 bytes)

daemon.info: Apr 27 05:12:57 vedge01 charon: 08[NET] received packet: from 10.10.10.1[4500] to
172.28.0.36[4500] (76 bytes)
daemon.info: Apr 27 05:12:57 vedge01 charon: 08[ENC] parsed INFORMATIONAL request 558 [ ]
daemon.info: Apr 27 05:12:57 vedge01 charon: 08[ENC] generating INFORMATIONAL response 558 [ ]
daemon.info: Apr 27 05:12:57 vedge01 charon: 08[NET] sending packet: from 172.28.0.36[4500] to
10.10.10.1[4500] (76 bytes)
daemon.info: Apr 27 05:12:58 vedge01 charon: 07[NET] received packet: from 10.10.10.1[4500] to
172.28.0.36[4500] (396 bytes)
daemon.info: Apr 27 05:12:58 vedge01 charon: 07[ENC] parsed CREATE_CHILD_SA request 559 [ SA No
TSi TSr ]
daemon.info: Apr 27 05:12:58 vedge01 charon: 07[CFG] received proposals:
```

```
ESP:AES_GCM_16_256/NO_EXT_SEQ, ESP:AES_CBC_256/HMAC_SHA1_96/NO_EXT_SEQ,
ESP:3DES_CBC/HMAC_SHA1_96/NO_EXT_SEQ, ESP:AES_CBC_256/HMAC_SHA2_256_128/NO_EXT_SEQ,
ESP:AES_CBC_128/HMAC_SHA1_96/NO_EXT_SEQ, ESP:3DES_CBC/HMAC_SHA2_256_128/NO_EXT_SEQ
daemon.info: Apr 27 05:12:58 vedge01 charon: 07[CFG] configured proposals:
ESP:AES_CBC_256/HMAC_SHA1_96/MODP_4096/NO_EXT_SEQ
daemon.info: Apr 27 05:12:58 Avedge01 charon: 07[IKE] no acceptable proposal found
daemon.info: Apr 27 05:12:58 vedge01 charon: 07[IKE] failed to establish CHILD_SA, keeping
IKE_SA
```

> **Nota:** I pacchetti CREATE_CHILD_SA vengono scambiati per ogni nuova chiave o associazione di protezione. Per ulteriori riferimenti, vedere Informazioni su IKEv2 Packet Exchange

I debug IKE mostrano lo stesso comportamento e vengono ripetuti continuamente, quindi è possibile prendere parte alle informazioni e analizzarle:

CREATE_CHILD_SA indica una nuova chiave, con lo scopo di generare e scambiare il nuovo SPIS tra gli endpoint IPSec.

- Il vedge riceve il pacchetto di richiesta CREATE_CHILD_SA da 10.10.10.1.
- Il vedge elabora la richiesta e verifica le proposte (SA) inviate dal peer 10.10.10.1
- Il vedge confronta la proposta ricevuta inviata dal peer con le proposte configurate.
- Lo scambio di CREATE_CHILD_SA ha esito negativo e non è stata trovata alcuna proposta accettabile.

A questo punto, la domanda è: **Perché la configurazione non corrisponde se il tunnel funzionava in precedenza e non sono state apportate modifiche?**

Analizza in dettaglio: nelle proposte configurate è presente un campo aggiuntivo che il peer non sta inviando.

proposte configurate: ESP: AES_CBC_256/HMAC_SHA1_96/MODP_4096/NO_EXT_SEQ

Proposte ricevute:
ESP: AES_GCM_16_256/NO_EXT_SEQ
ESP:AES_CBC_256/HMAC_SHA1_96/NO_EXT_SEQ
ESP:3DES_CBC/HMAC_SHA1_96/NO_EXT_SEQ
ESP:AES_CBC_256/HMAC_SHA2_256_128/NO_EXT_SEQ
ESP:AES_CBC_128/HMAC_SHA1_96/NO_EXT_SEQ
ESP:3DES_CBC/HMAC_SHA2_256_128/NO_EXT_SEQ

MODP_4096 è il gruppo DH 16, configurato per PFS (Perfect-Forward-Secrecy) nella fase 2 (sezione IPsec).

PFS è l'unica configurazione non corrispondente in cui è possibile stabilire o meno il tunnel in base all'iniziatore o al risponditore nella negoziazione IKE. Tuttavia, quando la chiave viene avviata, il tunnel non può continuare e questo sintomo può essere presentato o correlato.

Impossibile riavviare il tunnel IPSec/Ikev2 vEdge dopo l'eliminazione a causa di un evento DELETE

Per ulteriori informazioni su questo comportamento, vedere l'ID bug Cisco CSCvx86427.

Poiché il problema persiste, i debug IKE sono le opzioni migliori. Tuttavia, per questo particolare

bug, se i debug sono abilitati, non vengono visualizzate informazioni né sul terminale né sul file dei messaggi.

Per circoscrivere il problema e verificare se vEdge incontra l'ID bug Cisco CSCvx86427, è necessario individuare il momento in cui il tunnel diventa inattivo.

identificare i punti prima dell'inizio della risoluzione dei problemi:

1. Tunnel IPsec (numero) con problemi e configurazione.
2. Timestamp dell'interruzione del tunnel.
3. Indirizzo IP peer IPsec (destinazione tunnel)

Dopo aver identificato l'indicatore orario e aver correlato l'ora e i log, esaminare i log subito prima che il tunnel si blocchi.

```
Apr 13 22:05:21 vedge01 charon: 12[IKE] received DELETE for IKE_SA ipsec1_1[217]
Apr 13 22:05:21 vedge01 charon: 12[IKE] deleting IKE_SA ipsec1_1[217] between
10.16.0.5[10.16.0.5]...10.10.10.1[10.10.10.1]
Apr 13 22:05:21 vedge01 charon: 12[IKE] deleting IKE_SA ipsec1_1[217] between
10.16.0.5[10.16.0.5]...10.10.10.1[10.10.10.1]
Apr 13 22:05:21 vedge01 charon: 12[IKE] IKE_SA deleted
Apr 13 22:05:21 vedge01 charon: 12[IKE] IKE_SA deleted
Apr 13 22:05:21 vedge01 charon: 12[ENC] generating INFORMATIONAL response 4586 [ ]
Apr 13 22:05:21 vedge01 charon: 12[NET] sending packet: from 10.16.0.5[4500] to 10.10.10.1[4500]
(80 bytes)
Apr 13 22:05:21 vedge01 charon: 12[KNL] Deleting SAD entry with SPI 00000e77
Apr 13 22:05:21 vedge01 FTMD[1269]: %Viptela-AZGDSR01-FTMD-6-INFO-1000001: VPN 1 Interface
ipsec1 DOWN
Apr 13 22:05:21 vedge01 FTMD[1269]: %Viptela-AZGDSR01-ftmd-6-INFO-1400002: Notification:
interface-state-change severity-level:major host-name:"vedge01" system-ip:4.1.0.1 vpn-id:1 if-
name:"ipsec1" new-state:down
```

> **Nota:** In una negoziazione IPSec sono presenti più pacchetti DELETES e l'istruzione DELETE per CHILD_SA è un'istruzione DELETE prevista per un processo REKEY. Questo problema si verifica quando si riceve un pacchetto DELETE IKE_SA puro senza una particolare negoziazione IPSec. L'istruzione DELETE rimuove tutti i tunnel IPsec/IKE.

Informazioni correlate

- Debug a livello di protocollo e scambio pacchetti KEv2
- IKE (Internet Key Exchange) - RFC 2409
- IKEv2 - RFC 7296
- IPSec da LAN a LAN da sito a sito tra vEdge e Cisco IOS
- Documentazione e supporto tecnico – Cisco Systems