Configuration du chiffrement ASR1000 sur la monodiffusion OTV

Table des matières

Introduction

Conditions préalables

- Exigences
- Composants utilisés

Informations générales

Configurer

- Diagramme du réseau
- Configurations

Vérifier

Dépannage

Introduction

Ce document décrit l'ensemble de base des configurations qui sont utilisées pour activer la virtualisation de transport de superposition (OTV) avec le cryptage IPSec. Le chiffrement sur OTV ne nécessite aucune configuration supplémentaire du côté OTV. Il vous suffit de comprendre comment OTV et IPSEC coexistent.

Pour ajouter le cryptage sur OTV, vous devez ajouter un en-tête ESP (Encapsulating Security Payload) au-dessus de l'unité de données de protocole OTV. Vous pouvez chiffrer les périphériques ASR1000 Edge Devices (ED) de deux façons : i) IPSec ii) GETVPN.

Conditions préalables

Exigences

Aucune exigence spécifique n'est associée à ce document.

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Routeurs ASR1000 pour périphériques de périphérie (ED)
- Principal (cloud ISP)
- Commutateurs Catalyst 2960 comme commutateur d'accès sur l'un ou l'autre site

The information in this document was created from the devices in a specific lab environment. All of

the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Informations générales

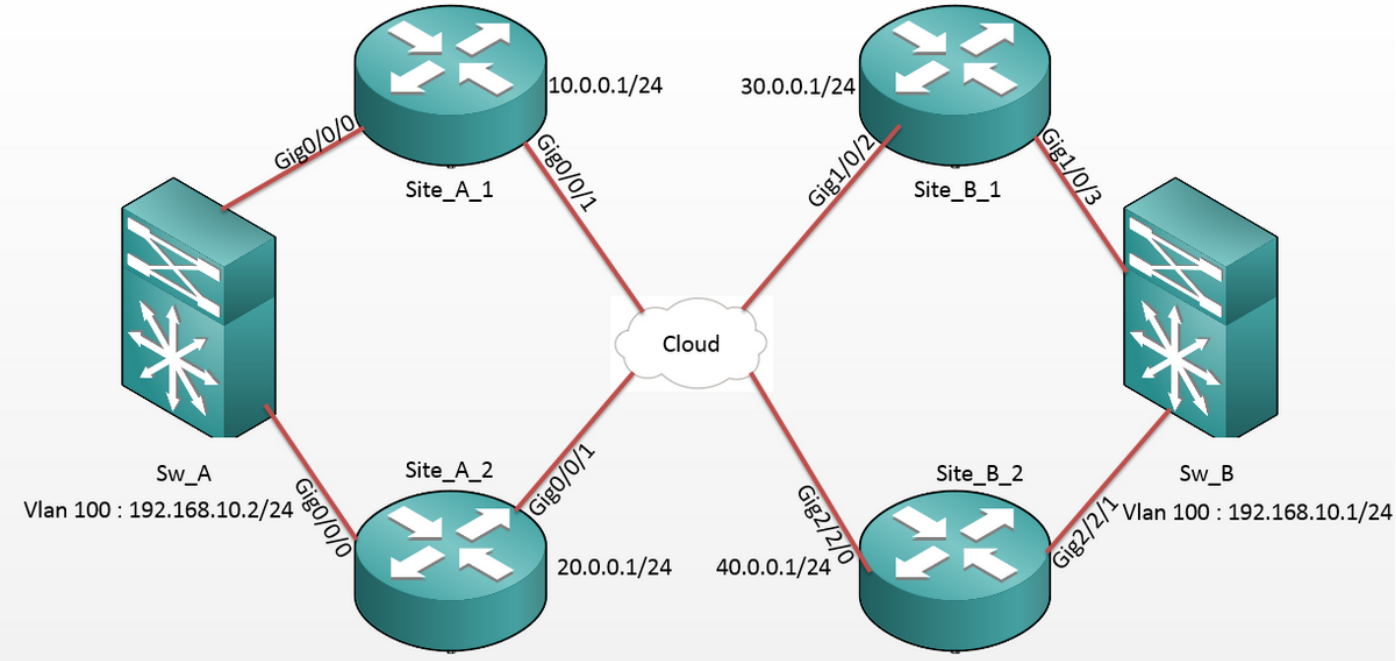
Les fonctionnalités et configurations de base d'OTV sont censées être connues des utilisateurs de ce document.

Vous pouvez également suivre ces documents pour le même :

- Configuration de monodiffusion OTV
- Configuration de multidiffusion OTV

Configurer

Diagramme du réseau



Configurations

Site A : Configurations ED :

Site_A_1#show run Building configuration... otv site bridge-domain 99 ! otv site-identifier 0000.0000.0001	Site_A_2#show run Building configuration... otv site bridge-domain 99 ! otv site-identifier 0000.0000.0001

```
crypto isakmp policy 10
 hash md5
 authentication pre-share
crypto isakmp key cisco address 30.0.0.1
crypto isakmp key cisco address 40.0.0.1
!
crypto ipsec transform-set tset esp-aes esp-md5-hmac
 mode tunnel
!
crypto map cmap 1 ipsec-isakmp
 set peer 30.0.0.1
 set transform-set tset
 match address cryptoacl
crypto map cmap 3 ipsec-isakmp
 set peer 40.0.0.1
 set transform-set tset
 match address cryptoacl3
!
interface Overlay99
 no ip address
 otv join-interface GigabitEthernet0/0/1
 otv adjacency-server unicast-only
 service instance 100 ethernet
  encapsulation dot1q 100
  bridge-domain 100
 !
 service instance 101 ethernet
  encapsulation dot1q 101
  bridge-domain 101
 !
!
```

```
crypto isakmp policy 10
 hash md5
 authentication pre-share
crypto isakmp key cisco address 30.0.0.1
crypto isakmp key cisco address 40.0.0.1
!
crypto ipsec transform-set tset esp-aes esp-md5-hm
 mode tunnel
!
crypto map cmap 2 ipsec-isakmp
 set peer 30.0.0.1
 set transform-set tset
 match address cryptoacl2
crypto map cmap 3 ipsec-isakmp
 set peer 40.0.0.1
 set transform-set tset
 match address cryptoacl3
!
interface Overlay99
 no ip address
 otv join-interface GigabitEthernet0/0/1
 otv use-adjacency-server 10.0.0.1 30.0.0.1 unicas
 service instance 100 ethernet
  encapsulation dot1q 100
  bridge-domain 100
 !
 service instance 101 ethernet
  encapsulation dot1q 101
  bridge-domain 101
 !
!
```

```
interface GigabitEthernet0/0/0
 no ip address
service instance 99 ethernet
  encapsulation dot1q 99
  bridge-domain 99
 !
 service instance 100 ethernet
  encapsulation dot1q 100
  bridge-domain 100
 !
 service instance 101 ethernet
  encapsulation dot1q 101
  bridge-domain 101
 !
!
interface GigabitEthernet0/0/1
 ip address 10.0.0.1 255.255.255.0
 crypto map cmap
!
ip access-list extended cryptoacl
 permit gre host 10.0.0.1 host 30.0.0.1
ip access-list extended cryptoacl3
 permit gre host 10.0.0.1 host 40.0.0.1
```

```
interface GigabitEthernet0/0/0
 no ip address
service instance 99 ethernet
  encapsulation dot1q 99
  bridge-domain 99
 !
 service instance 100 ethernet
  encapsulation dot1q 100
  bridge-domain 100
 !
 service instance 101 ethernet
  encapsulation dot1q 101
  bridge-domain 101
 !
!
interface GigabitEthernet0/0/1
 ip address 20.0.0.1 255.255.255.0
 crypto map cmap
!
ip access-list extended cryptoacl2
 permit gre host 20.0.0.1 host 30.0.0.1
ip access-list extended cryptoacl3
 permit gre host 20.0.0.1 host 40.0.0.1
```

Site B : Configurations ED :

```
Site_B_1#sh run
Building configuration...
otv site bridge-domain 99
!
```

```
Site_B_2#sh run
Building configuration...
otv site bridge-domain 99
!
```

```
otv site-identifier 0000.0000.0002
crypto isakmp policy 10
 hash md5
 authentication pre-share
crypto isakmp key cisco address 10.0.0.1
crypto isakmp key cisco address 20.0.0.1
!
crypto ipsec transform-set tset esp-aes esp-md5-hmac
 mode tunnel
!
crypto map cmap 1 ipsec-isakmp
 set peer 10.0.0.1
 set transform-set tset
 match address cryptoacl
crypto map cmap 2 ipsec-isakmp
 set peer 20.0.0.1
 set transform-set tset
 match address cryptoacl2
!
interface Overlay99
 no ip address
 otv join-interface GigabitEthernet1/0/2
 otv use-adjacency-server 10.0.0.1 unicast-only
 otv adjacency-server unicast-only
 service instance 100 ethernet
  encapsulation dot1q 100
  bridge-domain 100
 !
 service instance 101 ethernet
  encapsulation dot1q 101
  bridge-domain 101
```

```
otv site-identifier 0000.0000.0002
crypto isakmp policy 10
 hash md5
 authentication pre-share
crypto isakmp key cisco address 10.0.0.1
crypto isakmp key cisco address 20.0.0.1
!
crypto ipsec transform-set tset esp-aes esp-md5-hm
 mode tunnel
!
crypto map cmap 1 ipsec-isakmp
 set peer 10.0.0.1
 set transform-set tset
 match address cryptoacl
crypto map cmap 2 ipsec-isakmp
 set peer 20.0.0.1
 set transform-set tset
 match address cryptoacl2
!
interface Overlay99
 no ip address
 otv join-interface GigabitEthernet2/2/0
 otv use-adjacency-server 10.0.0.1 30.0.0.1 unicas
 service instance 100 ethernet
  encapsulation dot1q 100
  bridge-domain 100
 !
 service instance 101 ethernet
  encapsulation dot1q 101
  bridge-domain 101
 !
```

```
 !
!
interface GigabitEthernet1/0/3
 no ip address
service instance 99 ethernet
  encapsulation dot1q 99
  bridge-domain 99
 !
 service instance 100 ethernet
  encapsulation dot1q 100
  bridge-domain 100
 !
 service instance 101 ethernet
  encapsulation dot1q 101
  bridge-domain 101
 !
!
interface GigabitEthernet1/0/2
 ip address 30.0.0.1 255.255.255.0
crypto map cmap
!
ip access-list extended cryptoacl
 permit gre host 30.0.0.1 host 10.0.0.1
ip access-list extended cryptoacl2
 permit gre host 30.0.0.1 host 20.0.0.1
```

```
!
interface GigabitEthernet2/2/1
 no ip address
 service instance 99 ethernet
  encapsulation dot1q 99
  bridge-domain 99
 !
 service instance 100 ethernet
  encapsulation dot1q 100
  bridge-domain 100
 !
 service instance 101 ethernet
  encapsulation dot1q 101
  bridge-domain 101
 !
!
interface GigabitEthernet2/2/0
 ip address 40.0.0.1 255.255.255.0
 crypto map cmap
!
ip access-list extended cryptoacl
 permit gre host 40.0.0.1 host 10.0.0.1
ip access-list extended cryptoacl2
 permit gre host 40.0.0.1 host 20.0.0.1
```

Vérifier

Utilisez cette section pour confirmer que votre configuration fonctionne correctement.

1. Vérifiez si l'adresse MAC de l'hôte VLAN interne (dans ce cas, l'interface SVI sur le

commutateur Catalyst 2960) a été apprise sur les tables de routage OTV.
2. Vérifiez si les encapsulations et les décodages de chiffrement sont effectués pour le trafic de superposition (trafic OTV).

Une fois que l'OTV s'affiche après que vous avez configuré la carte de chiffrement sur l'interface de jointure, vérifiez le redirecteur actif pour les VLAN locaux (dans ce cas VLAN 100 et 101). Ceci montre que Site_A_1 et Site_B_2 sont les redirecteurs actifs pour les VLAN pairs puisque vous allez tester le cryptage du trafic pour les requêtes ping initiées depuis VLAN 100 sur Site A vers VLAN 100 sur Site B :

<#root>

```
Site_A_1#show otv vlan

Key:  SI - Service Instance, NA - Non AED, NFC - Not Forward Capable.

Overlay 99 VLAN Configuration Information

 Inst VLAN BD   Auth ED              State                Site If(s)

 0    100  100  *Site_A_1           active              Gi0/0/0:SI100

 0    101  101   Site_A_2           inactive(NA)        Gi0/0/0:SI101

 0    200  200  *Site_A_1           active              Gi0/0/0:SI200

 0    201  201   Site_A_2           inactive(NA)        Gi0/0/0:SI201

 Total VLAN(s): 4
```

<#root>

```
Site_B_2#show otv vlan

Key:  SI - Service Instance, NA - Non AED, NFC - Not Forward Capable.


Overlay 99 VLAN Configuration Information

 Inst VLAN BD   Auth ED              State                Site If(s)

0    100  100  *Site_B_2           active              Gi2/2/1:SI100

 0    101  101   Site_B_1           inactive(NA)        Gi2/2/1:SI101

 0    200  200  *Site_B_2           active              Gi2/2/1:SI200
```

```
 0    201  201   Site_B_1            inactive(NA)        Gi2/2/1:SI201

 Total VLAN(s): 4
```

Afin de vérifier si les paquets sont effectivement encapsulés et décapsulés sur l'un ou l'autre ED, vous devez vérifier si la session IPSec est active et les valeurs de compteur dans les sessions de chiffrement afin de confirmer que les paquets sont effectivement chiffrés et déchiffrés. Afin de vérifier si la session IPSec est active, puisqu'elle devient active seulement si du trafic circule, vérifiez le résultat de show crypto isakmp sa. Ici, seules les sorties des redirecteurs actifs sont vérifiées, mais cela devrait montrer l'état actif sur tous les ED pour qu'OTV sur le cryptage fonctionne.

```
Site_A_1#show crypto isakmp sa

IPv4 Crypto ISAKMP SA

dst             src             state          conn-id status

10.0.0.1        30.0.0.1        QM_IDLE           1008 ACTIVE

10.0.0.1        40.0.0.1        QM_IDLE           1007 ACTIVE

Site_B_2#sh crypto isakmp sa

IPv4 Crypto ISAKMP SA

dst             src             state          conn-id status

20.0.0.1        40.0.0.1        QM_IDLE           1007 ACTIVE

10.0.0.1        40.0.0.1        QM_IDLE           1006 ACTIVE
```

Maintenant, afin de confirmer si les paquets sont chiffrés et déchiffrés, vous devez d'abord savoir à quoi vous attendre dans les sorties de show crypto session detail. Ainsi, lorsque vous lancez le paquet d'écho ICMP du commutateur Sw_A vers le commutateur Sw_B, ceci est attendu :

- Pendant que l'écho ICMP quitte l'ED Site_A_1, qui est le redirecteur actif pour le VLAN 100, il devra encapsuler la charge utile OTV (Écho ICMP + MPLS + GRE)
- Ensuite, une fois que l'écho ICMP atteint l'ED Site_B_2 qui est le redirecteur actif pour VLAN 100, il doit décapsuler la charge utile OTV (Écho ICMP + MPLS + GRE)
- Maintenant, une fois que le ED Site_B_2 reçoit la réponse d'écho ICMP de Sw_B, il doit encapsuler à nouveau la charge utile OTV (Écho ICMP + MPLS + GRE)
- Et une fois que la réponse d'écho ICMP atteint l'ED Site_A_1, je devrais à nouveau décapsuler la charge utile OTV (ICMP Echo + MPLS + GRE)

Après les requêtes ping réussies de Sw_A vers Sw_B, attendez-vous à voir un incrément de 5 compteurs sous les sections « enc » et « dec » de la sortie show crypto session detail sur les deux

ED du redirecteur actif.

Maintenant, vérifiez la même chose à partir des ED :

<#root>

```
Site_A_1(config-if)#do show crypto session detail | section enc

K - Keepalives, N - NAT-traversal, T - cTCP encapsulation

        Outbound: #pkts enc'ed 0 drop 0 life (KB/Sec) 4608000/3345
```

Outbound: #pkts enc'ed 10 drop 0 life (KB/Sec) 4607998/3291

<<<< 10 counter before ping

<#root>

```
Site_A_1(config-if)#do show crypto session detail | section dec

        Inbound:  #pkts dec'ed 0 drop 0 life (KB/Sec) 4608000/3343
```

Inbound: #pkts dec'ed 18 drop 0 life (KB/Sec) 4607997/3289

<<<< 18 counter before ping

<#root>

```
Site_B_2(config-if)#do show crypto session detail | section enc

K - Keepalives, N - NAT-traversal, T - cTCP encapsulation
```

Outbound: #pkts enc'ed 18 drop 0 life (KB/Sec) 4607997/3295

<<<< 18 counter before ping

```
        Outbound: #pkts enc'ed 9 drop 0 life (KB/Sec) 4607999/3295
```

<#root>

```
Site_B_2(config-if)#do show crypto session detail | section dec
```

Inbound: #pkts dec'ed 10 drop 0 life (KB/Sec) 4607998/3293

<<<< 10 counter before ping

```
        Inbound:  #pkts dec'ed 1 drop 0 life (KB/Sec) 4607999/3293


Sw_A(config)#do ping 192.168.10.1 source vlan 100

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.10.2
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/3/10 ms
Sw_A(config)#
```

<#root>

```
Site_A_1(config-if)#do show crypto session detail | section enc
K - Keepalives, N - NAT-traversal, T - cTCP encapsulation
        Outbound: #pkts enc'ed 0 drop 0 life (KB/Sec) 4608000/3339

Outbound: #pkts enc'ed 15 drop 0 life (KB/Sec) 4607997/3284
 <<<< 15 counter after ping (After ICMP Echo)
```

<#root>

```
Site_A_1(config-if)#do show crypto session detail | section dec
        Inbound:  #pkts dec'ed 0 drop 0 life (KB/Sec) 4608000/3338

Inbound:  #pkts dec'ed 23 drop 0 life (KB/Sec) 4607997/3283
 <<<< 23 counter after ping (After ICMP Echo Reply)
```

<#root>

```
Site_B_2(config-if)#do show crypto session detail | section enc
K - Keepalives, N - NAT-traversal, T - cTCP encapsulation

Outbound: #pkts enc'ed 23 drop 0 life (KB/Sec) 4607997/3282
```

```
 <<<< 23 counter after ping (After ICMP Echo Reply)

        Outbound: #pkts enc'ed 9 drop 0 life (KB/Sec) 4607999/3282
```

<#root>

```
Site_B_2(config-if)#do show crypto session detail | section dec
```

Inbound: #pkts dec'ed 15 drop 0 life (KB/Sec) 4607997/3281

```
 <<<< 15 counter after ping (After ICMP Echo)

        Inbound:  #pkts dec'ed 1 drop 0 life (KB/Sec) 4607999/3281
```

Ce guide de configuration est capable de transmettre les détails de configuration requis avec l'utilisation d'IPSec pour la configuration à double résidence de coeur de monodiffusion.

Dépannage

Il n'existe actuellement aucune information de dépannage spécifique pour cette configuration.

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.