配置基于OTV单播的ASR1000加密

目录

[简介](#)

[先决条件](#)

- [要求](#)
- [使用的组件](#)

[背景信息](#)

[配置](#)

- [网络图](#)
- [配置](#)

[验证](#)

[故障排除](#)

简介

本文档介绍用于启用具有IPSec加密的重叠传输虚拟化(OTV)的基本配置集。OTV加密不需要从OTV端进行任何额外配置。您只需要了解OTV和IPSEC如何共存。

要通过OTV添加加密，您需要在OTV PDU之上添加封装安全负载(ESP)报头。您可以通过两种方式在ASR1000边缘设备(ED)上实现加密：(i)IPSec(ii)GETVPN

先决条件

要求

本文档没有任何特定的要求。

使用的组件

本文档中的信息基于以下软件和硬件版本：

- 用于边缘设备的ASR1000路由器(ED)
- 核心（ISP云）
- Catalyst 2960交换机作为任一站点的接入交换机

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

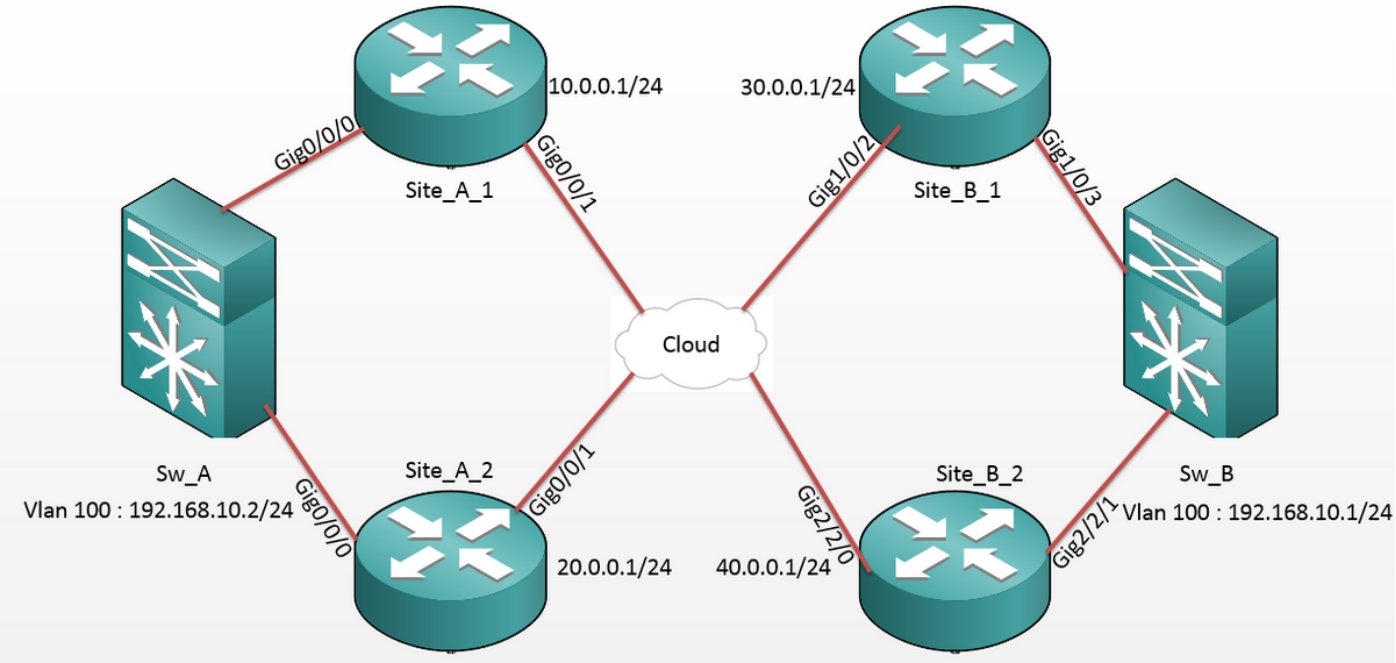
OTV的基本功能和配置假定为本文档的用户所知。

您也可以遵循以下文档进行操作：

- OTV单播配置
- OTV组播配置

配置

网络图



配置

站点 A：ED配置：

```
Site_A_1#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
```

```
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
 !
!
interface GigabitEthernet0/0/0
 no ip address
service instance 99 ethernet
  encapsulation dot1q 99
  bridge-domain 99
```

```
!
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
```

```
 !
 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
```

站点 B:ED配置：

```
Site_B_1#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
```

```
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
 !
!
interface GigabitEthernet1/0/3
 no ip address
service instance 99 ethernet
  encapsulation dot1q 99
```

```
!
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 GigabitEthernet2/2/1
 no ip address
 service instance 99 ethernet
  encapsulation dot1q 99
  bridge-domain 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
```

```
!
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
```

验证

使用本部分可确认配置能否正常运行。

1. 检查是否已在OTV路由表中获取内部VLAN主机（在本例中为2960 catalyst交换机上的SVI）的MAC地址。
2. 检查是否对重叠（OTV流量）流量执行加密封装和decap。

在加入接口上配置加密映射后，一旦OTV启动，请检查本地VLAN的活动转发器（本例中为VLAN 100和101）。 这表明Site_A_1和Site_B_2是偶数VLAN的活动转发器，因为您将测试从站点A的VLAN 100到站点B的VLAN 100发起的ping的流量加密：

<#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
```

为了检查数据包是否确实被封装和解封，您应该检查IPSec会话是否处于活动状态，以及加密会话中的计数器值，以确认数据包确实被加密和解密。要检查IPSec会话是否处于活动状态，因为只有经过任何流量时它才会变为活动状态，请检查show crypto isakmp sa的输出。在这里，只检查活动转发器的输出，但是这应该显示所有ED上的活动状态以使OTV加密正常工作。

```
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
```

现在，为了确认数据包是否被加密和解密，您首先需要知道show crypto session detail的输出中将会显示什么内容。因此，当您从Sw_A交换机向Sw_B发起ICMP回应数据包时，预期会出现以下情况：

- 当ICMP回应从Site_A_1 ED（VLAN 100的活动转发器）离开时，它必须封装OTV负载（ICMP回应+ MPLS + GRE）
- 然后，一旦ICMP回应到达Site_B_2 ED（VLAN 100的活动转发器），它就必须解封OTV负载（ICMP回应+ MPLS + GRE）
- 现在，一旦Site_B_2 ED收到来自Sw_B的ICMP回应应答，它将必须重新封装OTV负载（ICMP回应+ MPLS + GRE）
- 一旦ICMP回应应答到达Site_A_1 ED，我将再次解封OTV负载（ICMP回应+ MPLS + GRE）

从Sw_A ping Sw_B成功后，预计两个活动转发器ED上的show crypto session detail输出的“enc”和“dec”部分下会增加5个计数器。

现在，从ED检查同样的内容：

<#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
```

本配置指南能够使用IPSec传递单播核心双宿主设置所需的配置详细信息。

故障排除

目前没有针对此配置的故障排除信息。

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。