Configurar um túnel VPN site a site com ASA e Strongswan

Contents

Introduction

Este documento descreve como configurar o túnel de Internet Key Exchange Versão 1 de IPSec Site a Site através da CLI entre um ASA e um servidor strongSwan.

Prerequisites

Requirements

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Cisco Adaptive Security Appliance (ASA)
- Comandos Linux Básicos
- Conceitos gerais de IPSec

Componentes Utilizados

As informações neste documento são baseadas nestas versões:

- Cisco ASAv executando 9.12(3)9
- Ubuntu 20.04 executando strongSwan U5.8.2

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. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

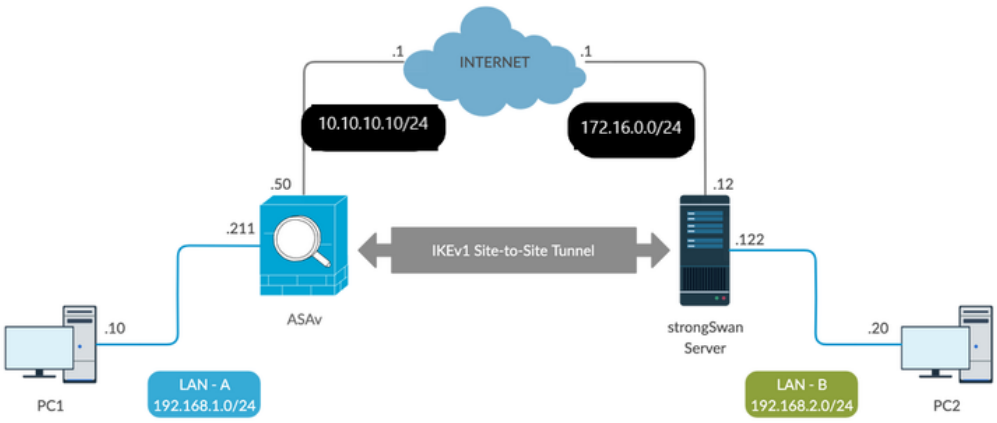
Configurar

Esta seção descreve como concluir as configurações do ASA e do strongSwan.

Cenário

Nesta configuração, o PC1 na LAN-A deseja se comunicar com o PC2 na LAN-B. Esse tráfego precisa ser criptografado e enviado por um túnel Internet Key Exchange Version 1 (IKEv1) entre o ASA e o servidor stongSwan. Os dois peers se autenticam com uma chave pré-compartilhada (PSK).

Diagrama de Rede



Note: Verifique se há conectividade com as redes internas e externas e, especialmente, com o peer remoto usado para estabelecer um túnel VPN site a site. Você pode usar um ping para verificar a conectividade básica.

Configuração do ASA

```
!Configure the ASA interfaces
!
interface GigabitEthernet0/0
nameif inside
security-level 100
ip address 192.168.1.211 255.255.255.0
```

```
!
interface GigabitEthernet0/1
nameif outside
security-level 0
ip address 10.10.10.10 255.255.255.0
!
!Configure the ACL for the VPN traffic of interest
!
object-group network local-network
network-object 192.168.1.0 255.255.255.0
!
object-group network remote-network
network-object 192.168.2.0 255.255.255.0
!
access-list asa-strongswan-vpn extended permit ip object-group local-network object-group
remote-network
!
!Enable IKEv1 on the 'Outside' interface
!
crypto ikev1 enable outside
!
!Configure how ASA identifies itself to the peer
!
crypto isakmp identity address
!
!Configure the IKEv1 policy
!
crypto ikev1 policy 10
authentication pre-share
encryption aes-256
hash sha
group 5
lifetime 3600
!
!Configure the IKEv1 transform-set
!
crypto ipsec ikev1 transform-set tset esp-aes-256 esp-sha-hmac
!
!Configure a crypto map and apply it to outside interface
!
crypto map outside_map 10 match address asa-strongswan-vpn
crypto map outside_map 10 set peer 172.16.0.0
crypto map outside_map 10 set ikev1 transform-set tset
crypto map outside_map 10 set security-association lifetime seconds 28800
crypto map outside_map interface outside
!
!Configure the Tunnel group (LAN-to-LAN connection profile)
!
tunnel-group 172.16.0.0 type ipsec-l2l
tunnel-group 172.16.0.0 ipsec-attributes
ikev1 pre-shared-key cisco
!
```

Note: Existe uma correspondência de política IKEv1 quando ambas as políticas dos dois pares contêm os mesmos valores de parâmetro de autenticação, criptografia, hash e Diffie-Hellman. Para IKEv1, a política de peer remoto também deve especificar um tempo de vida menor ou igual ao tempo de vida na política que o iniciador envia. Se os tempos de vida não forem idênticos, o ASA usará um tempo de vida menor. Além disso, se você não especificar um valor para um determinado parâmetro de política, o valor padrão será aplicado.

Observação: uma ACL para tráfego VPN usa os endereços IP origem e destino após a

conversão de endereço de rede (NAT).

Isenção NAT (opcional):

Normalmente, não deve haver NAT executado no tráfego VPN. Para isentar esse tráfego, você deve criar uma regra de NAT de identidade. A regra NAT de identidade simplesmente converte um endereço no mesmo endereço.

```
nat (inside,outside) source static local-network local-network destination static remote-network
remote-network no-proxy-arp route-lookup
```

Configuração de strongSwan

No Ubuntu, você modificaria esses dois arquivos com parâmetros de configuração a serem usados no túnel IPsec. Você pode usar seu editor favorito para editá-los.

/etc/ipsec.conf

/etc/ipsec.secrets

```
# /etc/ipsec.conf - strongSwan IPsec configuration file

# basic configuration

config setup
             strictcrlpolicy=no
             uniqueids = yes
             charondebug = "all"

# VPN to ASA

conn vpn-to-asa
             authby=secret
             left=%defaultroute
             leftid=172.16.0.0
             leftsubnet=192.168.2.0/24
             right=10.10.10.10
             rightid=10.10.10.10
             rightsubnet=192.168.1.0/24
             ike=aes256-sha1-modp1536
             esp=aes256-sha1
             keyingtries=%forever
             leftauth=psk
             rightauth=psk
             keyexchange=ikev1
             ikelifetime=1h
             lifetime=8h
             dpddelay=30
             dpdtimeout=120
             dpdaction=restart
             auto=start


# config setup - Defines general configuration parameters.
# strictcrlpolicy - Defines if a fresh CRL must be available in order for the peer
authentication based on RSA
signatures to succeed.
```

 # uniqueids - Defines whether a particular participant ID must be kept unique, with any new IKE_SA using an ID
deemed to replace all old ones using that ID.
 # charondebug - Defines how much charon debugging output must be logged.
 # conn

- Defines a connection.
 # authby - Defines how the peers must authenticate; acceptable values are secret or psk, pubkey, rsasig, ecdsasig.
 # left - Defines the IP address of the strongSwan's interface paricipating in the tunnel.
 # lefid - Defines the identity payload for the strongSwan.
 # leftsubnet - Defines the private subnet behind the strongSwan, expressed as network/netmask.
 # right - Defines the public IP address of the VPN peer.
 # rightid - Defines the identity payload for the VPN peer.
 # rightsubnet - Defines the private subnet behind the VPN peer, expressed as network/netmask.
 # ike - Defines the IKE/ISAKMP SA encryption/authentication algorithms. You can add a comma-separated list.
 # esp - Defines the ESP encryption/authentication algorithms. You can add a comma-separated list.
 # keyingtries - Defines the number of attempts that must be made to negotiate a connection.
 # keyexchange - Defines the method of key exchange, whether IKEv1 or IKEv2.
 # ikelifetime - Defines the duration of an established phase-1 connection.
 # lifetime - Defines the duration of an established phase-2 connection.
 # dpddelay - Defines the time interval with which R_U_THERE messages/INFORMATIONAL exchanges are sent to the peer.
These are only sent if no other traffic is received.
 # dpdtimeout - Defines the timeout interval, after which all connections to a peer are deleted in case of inactivity.
 # dpdaction - Defines what action needs to be performed on DPD timeout. Takes three values as paramters : **clear**, **hold**, and **restart.**
With **clear** the connection is closed with no further actions taken, **hold** installs a trap policy, which catches
matching traffic and tries to re-negotiate the connection on demand and **restart** immediately triggers an attempt
to re-negotiate the connection. The default is **none** which disables the active sending of DPD messages.
 # auto - Defines what operation, if any, must be done automatically at IPsec startup (**start** loads a connection and brings
it up immediately).

/etc/ipsec.secrets - This file holds shared secrets or RSA private keys for authentication.

 # RSA private key for this host, authenticating it to any other host which knows the public part.

172.16.0.0 10.10.10.10 : PSK "cisco"

Comandos úteis (strongswan)

Iniciar / Parar / Status:

$ sudo ipsec up <nome-da-conexão>

$ sudo ipsec up vpn-to-asa

generating QUICK_MODE request 656867907 [HASH SA No ID ID]

```
sending packet: from 172.16.0.0[500] to 10.10.10.10[500] (204 bytes)
received packet: from 10.10.10.10[500] to 172.16.0.0[500] (188 bytes)
parsed QUICK_MODE response 656867907 [ HASH SA No ID ID N((24576)) ]
selected proposal: ESP:AES_CBC_256/HMAC_SHA1_96/NO_EXT_SEQ
detected rekeying of CHILD_SA vpn-to-asa{2}
CHILD_SA vpn-to-asa{3} established with SPIs c9080c93_i 3f570a23_o and TS 192.168.2.0/24 ===
192.168.1.0/24
connection 'vpn-to-asa' established successfully
```

$ sudo ipsec down <nome-conexão>

```
$ sudo ipsec down vpn-to-asa

generating QUICK_MODE request 656867907 [ HASH SA No ID ID ]
sending packet: from 172.16.0.0[500] to 10.10.10.10[500] (204 bytes)
received packet: from 10.10.10.10[500] to 172.16.0.0[500] (188 bytes)
parsed QUICK_MODE response 656867907 [ HASH SA No ID ID N((24576)) ]
selected proposal: ESP:AES_CBC_256/HMAC_SHA1_96/NO_EXT_SEQ
detected rekeying of CHILD_SA vpn-to-asa{2}
CHILD_SA vpn-to-asa{3} established with SPIs c9080c93_i 3f570a23_o and TS 192.168.2.0/24 ===
192.168.1.0/24
connection 'vpn-to-asa' established successfully
anurag@strongswan214:~$ sudo ipsec down vpn-to-asa
closing CHILD_SA vpn-to-asa{3} with SPIs c9080c93_i (0 bytes) 3f570a23_o (0 bytes) and TS
192.168.2.0/24 === 192.168.1.0/24
sending DELETE for ESP CHILD_SA with SPI c9080c93
generating INFORMATIONAL_V1 request 3465984663 [ HASH D ]
sending packet: from 172.16.0.0[500] to 10.10.10.10[500] (76 bytes)
deleting IKE_SA vpn-to-asa[2] between 172.16.0.0[172.16.0.0]...10.10.10.10[10.10.10.10]
sending DELETE for IKE_SA vpn-to-asa[2]
generating INFORMATIONAL_V1 request 2614622058 [ HASH D ]
sending packet: from 172.16.0.0[500] to 10.10.10.10[500] (92 bytes)
IKE_SA [2] closed successfully
```

$ sudo ipsec restart

```
Stopping strongSwan IPsec...
Starting strongSwan 5.8.2 IPsec [starter]...
```

$ sudo ipsec status

```
Security Associations (1 up, 0 connecting):
vpn-to-asa[1]: ESTABLISHED 35 seconds ago, 172.16.0.0[172.16.0.0]...10.10.10.10[10.10.10.10]
vpn-to-asa{1}: REKEYED, TUNNEL, reqid 1, expires in 7 hours
vpn-to-asa{1}: 192.168.2.0/24 === 192.168.1.0/24
vpn-to-asa{2}: INSTALLED, TUNNEL, reqid 1, ESP SPIs: c0d93265_i 599b4d60_o
vpn-to-asa{2}: 192.168.2.0/24 === 192.168.1.0/24
```

$ sudo ipsec status all

```
Status of IKE charon daemon (strongSwan 5.8.2, Linux 5.4.0-37-generic, x86_64):
uptime: 2 minutes, since Jun 27 07:15:14 2020
malloc: sbrk 2703360, mmap 0, used 694432, free 2008928
worker threads: 11 of 16 idle, 5/0/0/0 working, job queue: 0/0/0/0, scheduled: 3
loaded plugins: charon aesni aes rc2 sha2 sha1 md5 mgf1 random nonce x509 revocation constraints
pubkey pkcs1 pkcs7 pkcs8 pkcs12 pgp dnskey sshkey pem openssl fips-prf gmp agent xcbc hmac gcm
```

```
drbg attr kernel-netlink resolve socket-default connmark stroke updown eap-mschapv2 xauth-
generic counters
Listening IP addresses:
172.16.0.0
192.168.2.122
Connections:
vpn-to-asa: %any...10.10.10.10 IKEv1, dpddelay=30s
vpn-to-asa: local: [172.16.0.0] uses pre-shared key authentication
vpn-to-asa: remote: [10.10.10.10] uses pre-shared key authentication
vpn-to-asa: child: 192.168.2.0/24 === 192.168.1.0/24 TUNNEL, dpdaction=restart
Security Associations (1 up, 0 connecting):
vpn-to-asa[1]: ESTABLISHED 2 minutes ago, 172.16.0.0[172.16.0.0]...10.10.10.10[10.10.10.10]
vpn-to-asa[1]: IKEv1 SPIs: 57e24d839bf05f95_i* 6a4824492f289747_r, pre-shared key
reauthentication in 40 minutes
vpn-to-asa[1]: IKE proposal: AES_CBC_256/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_1536
vpn-to-asa{2}: INSTALLED, TUNNEL, reqid 1, ESP SPIs: c0d93265_i 599b4d60_o
vpn-to-asa{2}: AES_CBC_256/HMAC_SHA1_96, 0 bytes_i, 0 bytes_o, rekeying in 7 hours
vpn-to-asa{2}: 192.168.2.0/24 === 192.168.1.0/24
```

Obtenha as políticas e os estados do túnel IPsec:

$ sudo ip xfrm state

```
src 172.16.0.0 dst 10.10.10.10
proto esp spi 0x599b4d60 reqid 1 mode tunnel
replay-window 0 flag af-unspec
auth-trunc hmac(sha1) 0x52c84359280868491a37e966384e4c6db05384c8 96
enc cbc(aes) 0x99e00f0989fec6baa7bd4ea1c7fbefdf37f04153e721a060568629e603e23e7a
anti-replay context: seq 0x0, oseq 0x0, bitmap 0x00000000
src 10.10.10.10 dst 172.16.0.0
proto esp spi 0xc0d93265 reqid 1 mode tunnel
replay-window 32 flag af-unspec
auth-trunc hmac(sha1) 0x374d9654436a4c4fe973a54da044d8814184861e 96
enc cbc(aes) 0xf51a4887281551a246a73c3518d938fd4918928088a54e2abc5253bd2de30fd6
anti-replay context: seq 0x0, oseq 0x0, bitmap 0x00000000
```

$ sudo ip xfrm policy

```
src 192.168.2.0/24 dst 192.168.1.0/24
dir out priority 375423
tmpl src 172.16.0.0 dst 10.10.10.10
proto esp spi 0x599b4d60 reqid 1 mode tunnel
src 192.168.1.0/24 dst 192.168.2.0/24
dir fwd priority 375423
tmpl src 10.10.10.10 dst 172.16.0.0
proto esp reqid 1 mode tunnel
src 192.168.1.0/24 dst 192.168.2.0/24
dir in priority 375423
tmpl src 10.10.10.10 dst 172.16.0.0
proto esp reqid 1 mode tunnel
src 0.0.0.0/0 dst 0.0.0.0/0
socket in priority 0
src 0.0.0.0/0 dst 0.0.0.0/0
socket out priority 0
src 0.0.0.0/0 dst 0.0.0.0/0
socket in priority 0
src 0.0.0.0/0 dst 0.0.0.0/0
socket out priority 0
src ::/0 dst ::/0
```

```
socket in priority 0
src ::/0 dst ::/0
socket out priority 0
src ::/0 dst ::/0
socket in priority 0
src ::/0 dst ::/0
socket out priority 0
```

Recarregue os segredos enquanto o serviço estiver em execução:

$ sudo ipsec rereadsecrets

Verifique se o tráfego flui pelo túnel:

$ sudo tcpdump esp

```
09:30:27.788533 IP 172.16.0.0 > 10.10.10.10: ESP(spi=0x599b4d60,seq=0x1e45), length 132
09:30:27.788779 IP 172.16.0.0 > 10.10.10.10: ESP(spi=0x599b4d60,seq=0x1e45), length 132
09:30:27.790348 IP 10.10.10.10 > 172.16.0.0: ESP(spi=0xc0d93265,seq=0x11), length 132
09:30:27.790512 IP 10.10.10.10 > 172.16.0.0: ESP(spi=0xc0d93265,seq=0x11), length 132
09:30:28.788946 IP 172.16.0.0 > 10.10.10.10: ESP(spi=0x599b4d60,seq=0x1e46), length 132
09:30:28.789201 IP 172.16.0.0 > 10.10.10.10: ESP(spi=0x599b4d60,seq=0x1e46), length 132
09:30:28.790116 IP 10.10.10.10 > 172.16.0.0: ESP(spi=0xc0d93265,seq=0x12), length 132
09:30:28.790328 IP 10.10.10.10 > 172.16.0.0: ESP(spi=0xc0d93265,seq=0x12), length 132
```

Verificar

Antes de verificar se o túnel está ativo e se passa o tráfego, você deve garantir que o "tráfego de interesse" seja enviado para o ASA ou para o servidor strongSwan.

> **Observação**: **no ASA, a ferramenta de rastreamento de pacotes que corresponde ao tráfego de interesse pode ser usada para iniciar o túnel IPSec (como entrada de rastreamento de pacotes dentro do tcp 192.168.1.100 12345 192.168.2.200 80 detalhado, por exemplo).**

No ASA

Fase 1 Verificação

Para verificar se a Fase 1 do IKEv1 está ativa no ASA, insira o **comando show crypto ikev1 sa (**ou **show crypto isakmp sa)**. A saída esperada é ver **oMM_**ATIVEstate:

```
ASAv# show crypto ikev1 sa

IKEv1 SAs:

Active SA: 1
Rekey SA: 0 (A tunnel will report 1 Active and 1 Rekey SA during rekey)
Total IKE SA: 1

1 IKE Peer: 172.16.0.0
Type : L2L Role : responder
Rekey : no State : MM_ACTIVE
```

Fase 2 Verificação

Para verificar se a Fase 2 do IKEv1 está ativa no ASA, insira o comando **show crypto ipsec sa** comando. A saída esperada é ver o SPI (Índice de Parâmetros de Segurança) de entrada e de saída. Se o tráfego passar pelo túnel, você deverá ver o incremento dos contadores encaps/decaps.

> **Observação**: para cada entrada de ACL, há uma SA de entrada/saída separada criada, que pode resultar em uma saída longa do comando show crypto ipsec sa (dependente do número de entradas ACE na ACL de criptografia).

```
ASAv# show crypto ipsec sa peer 172.16.0.0
interface: outside
Crypto map tag: outside_map, seq num: 10, local addr: 10.10.10.10

access-list asa-strongswan-vpn extended permit ip 192.168.1.0 255.255.255.0 192.168.2.0
255.255.255.0
local ident (addr/mask/prot/port): (192.168.1.0/255.255.255.0/0/0)
remote ident (addr/mask/prot/port): (192.168.2.0/255.255.255.0/0/0)
current_peer: 172.16.0.0


#pkts encaps: 37, #pkts encrypt: 37, #pkts digest: 37
#pkts decaps: 37, #pkts decrypt: 37, #pkts verify: 37
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 37, #pkts comp failed: 0, #pkts decomp failed: 0
#pre-frag successes: 0, #pre-frag failures: 0, #fragments created: 0
#PMTUs sent: 0, #PMTUs rcvd: 0, #decapsulated frgs needing reassembly: 0
#TFC rcvd: 0, #TFC sent: 0
#Valid ICMP Errors rcvd: 0, #Invalid ICMP Errors rcvd: 0
#send errors: 0, #recv errors: 0

local crypto endpt.: 10.10.10.10/0, remote crypto endpt.: 172.16.0.0/0
path mtu 1500, ipsec overhead 74(44), media mtu 1500
PMTU time remaining (sec): 0, DF policy: copy-df
ICMP error validation: disabled, TFC packets: disabled
current outbound spi: C8F1BFAB
current inbound spi : 3D64961A

inbound esp sas:
spi: 0x3D64961A (1030002202)
SA State: active
transform: esp-aes-256 esp-sha-hmac no compression
in use settings ={L2L, Tunnel, IKEv1, }
slot: 0, conn_id: 31, crypto-map: outside_map
sa timing: remaining key lifetime (kB/sec): (4373997/27316)
IV size: 16 bytes
replay detection support: Y
Anti replay bitmap:
0x000001FF 0xFFFFFFFF
outbound esp sas:
spi: 0xC8F1BFAB (3371286443)
SA State: active
transform: esp-aes-256 esp-sha-hmac no compression
in use settings ={L2L, Tunnel, IKEv1, }
slot: 0, conn_id: 31, crypto-map: outside_map
sa timing: remaining key lifetime (kB/sec): (4373997/27316)
IV size: 16 bytes
replay detection support: Y
```

```
Anti replay bitmap:
0x00000000 0x00000001
```

Como alternativa, você pode usar o comando **show vpn-sessiondb** para verificar os detalhes das Fases 1 e 2 juntos.

```
ASAv# show vpn-sessiondb detail l2l filter ipaddress 172.16.0.0

Session Type: LAN-to-LAN Detailed

Connection :172.16.0.0
Index : 3 IP Addr : 172.16.0.0
Protocol : IKEv1 IPsec
Encryption : IKEv1: (1)AES256 IPsec: (1)AES256
Hashing : IKEv1: (1)SHA1 IPsec: (1)SHA1
Bytes Tx : 536548 Bytes Rx : 536592
Login Time : 12:45:14 IST Sat Jun 27 2020
Duration : 1h:51m:57s

IKEv1 Tunnels: 1
IPsec Tunnels: 1

IKEv1:
Tunnel ID : 3.1
UDP Src Port : 500 UDP Dst Port : 500
IKE Neg Mode : Main Auth Mode : preSharedKeys
Encryption : AES256 Hashing : SHA1
Rekey Int (T): 3600 Seconds Rekey Left(T): 2172 Seconds
D/H Group : 5
Filter Name :

IPsec:
Tunnel ID : 3.2
Local Addr : 192.168.1.0/255.255.255.0/0/0
Remote Addr : 192.168.2.0/255.255.255.0/0/0
Encryption : AES256 Hashing : SHA1
Encapsulation: Tunnel
Rekey Int (T): 28800 Seconds Rekey Left(T): 22099 Seconds
Rekey Int (D): 4608000 K-Bytes Rekey Left(D): 4607476 K-Bytes
Idle Time Out: 30 Minutes Idle TO Left : 30 Minutes
Bytes Tx : 536638 Bytes Rx : 536676
Pkts Tx : 6356 Pkts Rx : 6389
```

Em strongSwan

```
# sudo ipsec statusall

Status of IKE charon daemon (strongSwan 5.8.2, Linux 5.4.0-37-generic, x86_64):
uptime: 2 minutes, since Jun 27 07:15:14 2020
malloc: sbrk 2703360, mmap 0, used 694432, free 2008928
worker threads: 11 of 16 idle, 5/0/0/0 working, job queue: 0/0/0/0, scheduled: 3
loaded plugins: charon aesni aes rc2 sha2 sha1 md5 mgf1 random nonce x509 revocation constraints
pubkey pkcs1 pkcs7 pkcs8 pkcs12 pgp dnskey sshkey pem openssl fips-prf gmp agent xcbc hmac gcm
drbg attr kernel-netlink resolve socket-default connmark stroke updown eap-mschapv2 xauth-
generic counters
Listening IP addresses:
172.16.0.0
192.168.2.122
Connections:
```

```
vpn-to-asa: %any...10.10.10.10 IKEv1, dpddelay=30s
vpn-to-asa: local: [172.16.0.0] uses pre-shared key authentication
vpn-to-asa: remote: [10.10.10.10] uses pre-shared key authentication
vpn-to-asa: child: 192.168.2.0/24 === 192.168.1.0/24 TUNNEL, dpdaction=restart
Security Associations (1 up, 0 connecting):
vpn-to-asa[1]: ESTABLISHED 2 minutes ago, 172.16.0.0[172.16.0.0]...10.10.10.10[10.10.10.10]
vpn-to-asa[1]: IKEv1 SPIs: 57e24d839bf05f95_i* 6a4824492f289747_r, pre-shared key
reauthentication in 40 minutes
vpn-to-asa[1]: IKE proposal: AES_CBC_256/HMAC_SHA1_96/PRF_HMAC_SHA1/MODP_1536
vpn-to-asa{2}: INSTALLED, TUNNEL, reqid 1, ESP SPIs: c0d93265_i 599b4d60_o
vpn-to-asa{2}: AES_CBC_256/HMAC_SHA1_96, 0 bytes_i, 0 bytes_o, rekeying in 7 hours
vpn-to-asa{2}: 192.168.2.0/24 === 192.168.1.0/24
```

Troubleshoot

Depurações ASA

Para solucionar problemas de negociação de túnel IPSec IKEv1 em um firewall ASA, você pode usar estes comandos debug:

> **Caution**: No ASA, você pode definir vários níveis de depuração; por padrão, o nível 1 é usado. Se você alterar o nível de depuração, o detalhamento das depurações poderá aumentar. No, esse nível de caso 127 fornece detalhes suficientes para solucionar problemas. Faça isso com cuidado, especialmente em ambientes de produção.

```
debug crypto ipsec 127
debug crypto isakmp 127
debug ike-common 10
```

> **Observação**: se houver vários túneis VPN no ASA, é recomendável usar depurações condicionais (debug crypto condition peer A.B.C.D), a fim de limitar as saídas de depuração para incluir apenas o peer especificado.

Depurações strongSwan

Certifique-se de que a depuração charon esteja habilitada no arquivo ipsec.conf:

```
charondebug = "all"
```

O destino final das mensagens de log depende de como o syslog é configurado no sistema. Os locais comuns são **/var/log/daemon**, **/var/log/syslog** ou **/var/log/messages**.

Informações Relacionadas

- Documentação do usuário do strongSwan
- Exemplo de configuração de IKEv1/IKEv2 entre Cisco IOS e strongSwan
- Configurar um túnel IPSec IKEv1 site a site entre um ASA e um roteador Cisco IOS

Sobre esta tradução
A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês (link fornecido) seja sempre consultado.