

Configuración de un túnel VPN de sitio a sitio con ASA y Strongswan

Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Configurar](#)

[Situación](#)

[Configuración de ASA](#)

[Configuración strongSwan](#)

[Comandos útiles \(strongswan\)](#)

[Verificación](#)

[En ASA](#)

[Fase 1 Verificación](#)

[Fase 2 Verificación](#)

[En strongSwan](#)

[Troubleshoot](#)

[Depuraciones de ASA](#)

[Depuraciones strongSwan](#)

[Información Relacionada](#)

Introducción

Este documento describe cómo configurar el túnel IPsec Internet Key Exchange Version 1 de sitio a sitio a través de CLI entre un ASA y un servidor strongSwan.

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Dispositivo de seguridad adaptable de Cisco (ASA)
- Comandos básicos de Linux
- Conceptos generales de IPsec

Componentes Utilizados

La información de este documento se basa en estas versiones:

- Cisco ASA v con 9.12(3)9
- Ubuntu 20.04 con 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. Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

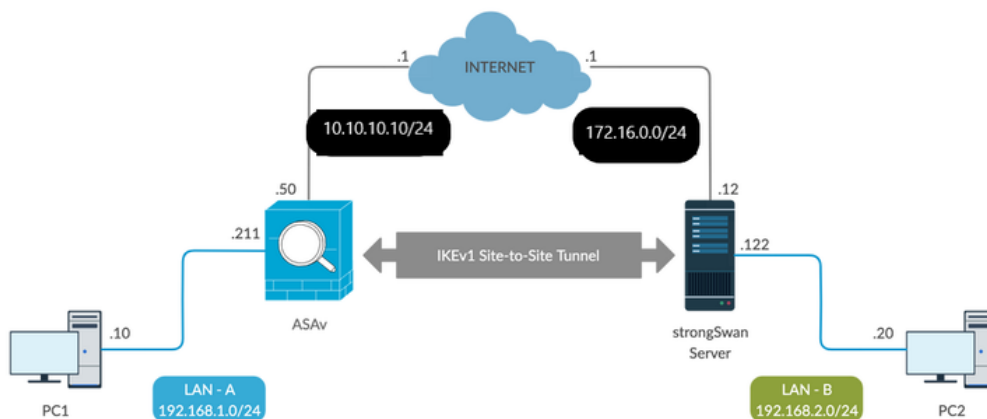
Configurar

En esta sección se describe cómo completar las configuraciones ASA y strongSwan.

Situación

En esta configuración, PC1 en LAN-A quiere comunicarse con PC2 en LAN-B. Este tráfico debe cifrarse y enviarse a través de un túnel de Intercambio de claves de Internet versión 1 (IKEv1) entre ASA y el servidor StrongSwan. Ambos peers se autentican mutuamente con una clave previamente compartida (PSK).

Diagrama de la red



Nota: Asegúrese de que haya conectividad tanto con las redes internas como externas, y especialmente con el peer remoto que se utiliza para establecer un túnel VPN de sitio a sitio. Puede utilizar un ping para verificar la conectividad básica.

Configuración de 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
!

```

Nota: Existe una coincidencia de política IKEv1 cuando ambas políticas de los dos peers contienen los mismos valores de parámetro de autenticación, cifrado, hash y Diffie-Hellman. Para IKEv1, la política de peer remoto también debe especificar una duración menor o igual a la duración en la política que envía el iniciador. Si las duraciones no son idénticas, el ASA utiliza una duración más corta. Además, si no especifica un valor para un parámetro de directiva determinado, se aplica el valor predeterminado.

Nota: Una ACL para el tráfico VPN utiliza las direcciones IP de origen y de destino después

de la traducción de direcciones de red (NAT).

Exención de NAT (opcional):

Normalmente, no debe realizarse ninguna NAT en el tráfico VPN. Para eximir ese tráfico, debe crear una regla de identidad NAT. La regla de identidad NAT simplemente traduce una dirección a la misma dirección.

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

Configuración strongSwan

En Ubuntu, modificaría estos dos archivos con los parámetros de configuración que se utilizarán en el túnel IPsec. Puede utilizar su editor favorito para editarlos.

/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-shal-modp1536
```

```
    esp=aes256-shal
```

```
    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 participating 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 útiles (strongswan)

Inicio / Parada / Estado:

```

$ sudo ipsec up <connection-name>

```

```

$ 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 <connection-name>

```
$ 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 shal md5 mgf1 random nonce x509 revocation constraints
pubkey pkcs1 pkcs7 pkcs8 pkcs12 ppp 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
```

Obtener las directivas y los estados del 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) 0x99e00f0989fec6baa7bd4ealc7fbefdf37f04153e721a060568629e603e23e7a
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
```

Recargue los secretos mientras se ejecuta el servicio:

```
$ sudo ipsec rereadsecret
```

Compruebe si el tráfico fluye a través del 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
```

Verificación

Antes de verificar si el túnel está activo y que pasa el tráfico, debe asegurarse de que el 'tráfico de interés' se envía hacia el ASA o el servidor strongSwan.

Nota: En ASA, la herramienta de seguimiento de paquetes que coincide con el tráfico de interés se puede utilizar para iniciar el túnel IPsec (como la **entrada de seguimiento de paquetes dentro de tcp 192.168.1.100 12345 192.168.2.200 80** detallada por ejemplo).

En ASA

Fase 1 Verificación

Para verificar si IKEv1 Phase 1 está activo en el ASA, ingrese el comando **show crypto ikev1 sa** (o, **show crypto isakmp sa**). El resultado esperado es ver el estado **MM_ACTIVE**:

```
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 Verificación

Para verificar si IKEv1 Phase 2 está activo en el ASA, ingrese el **show crypto ipsec sa** comando. El resultado esperado es ver el Índice de parámetros de seguridad (SPI) entrante y saliente. Si el tráfico pasa a través del túnel, debe ver el incremento de los contadores encaps/decaps.

Nota: Para cada entrada ACL hay una SA entrante/saliente separada creada, que puede resultar en una salida del comando long **show crypto ipsec sa** (dependiente del número de entradas ACE en la ACL crypto).

```
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

Alternativamente, puede utilizar el comando **show vpn-sessiondb** para verificar los detalles de las fases 1 y 2, juntas.

```
ASAv# show vpn-sessiondb detail 121 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
```

En 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 pgg 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

Depuraciones de ASA

Para resolver problemas de negociación de túnel IKEv1 IPsec en un firewall ASA, puede utilizar estos comandos debug:

Precaución: En ASA, puede establecer varios niveles de depuración; de forma predeterminada, se utiliza el nivel 1. Si cambia el nivel de depuración, puede aumentar el nivel de detalle de los depuradores. En, este nivel de caso 127 proporciona suficientes detalles para solucionar problemas. Hágalo con precaución, especialmente en entornos de producción.

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

Nota: Si hay varios túneles VPN en el ASA, se recomienda utilizar debugs condicionales (debug crypto condition peer A.B.C.D), para limitar los resultados de debug de modo que incluyan solamente el peer especificado.

Depuraciones strongSwan

Asegúrese de que charon debug esté habilitado en el archivo ipsec.conf:

```
charondebug = "all"
```

El lugar en el que terminan los mensajes de registro depende de la configuración de syslog en su sistema. Los lugares comunes son `/var/log/daemon`, `/var/log/syslog` o `/var/log/messages`.

Información Relacionada

- [Documentación del usuario de strongSwan](#)
- [Ejemplo de Configuración de IKEv1/IKEv2 entre Cisco IOS y strongSwan](#)
- [Configuración de un túnel IPsec de IKEv1 de sitio a sitio entre ASA y un router Cisco IOS](#)

Acerca de esta traducción

Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).