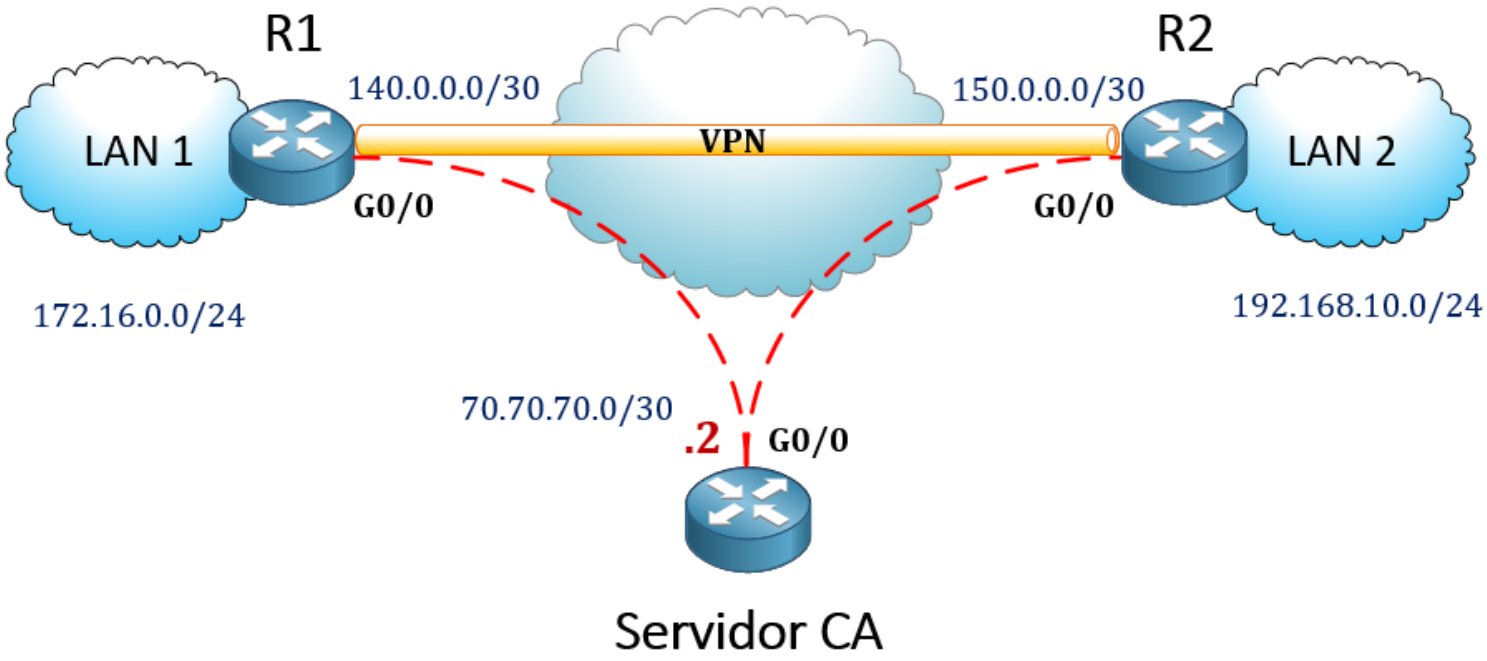
Cisco VPN Sitio-a-Sitio con Certificado Digital



Las VPNs o Virtual Private Networks han sido soluciones o conexiones que hemos venido utilizando por mucho tiempo, para transmitir y proteger información a través de redes propias o externas que se encuentran entre sitios remotos, y también con el fin de evitar ciertos tipos de ataques como por ejemplo Man-In-The-Middle.

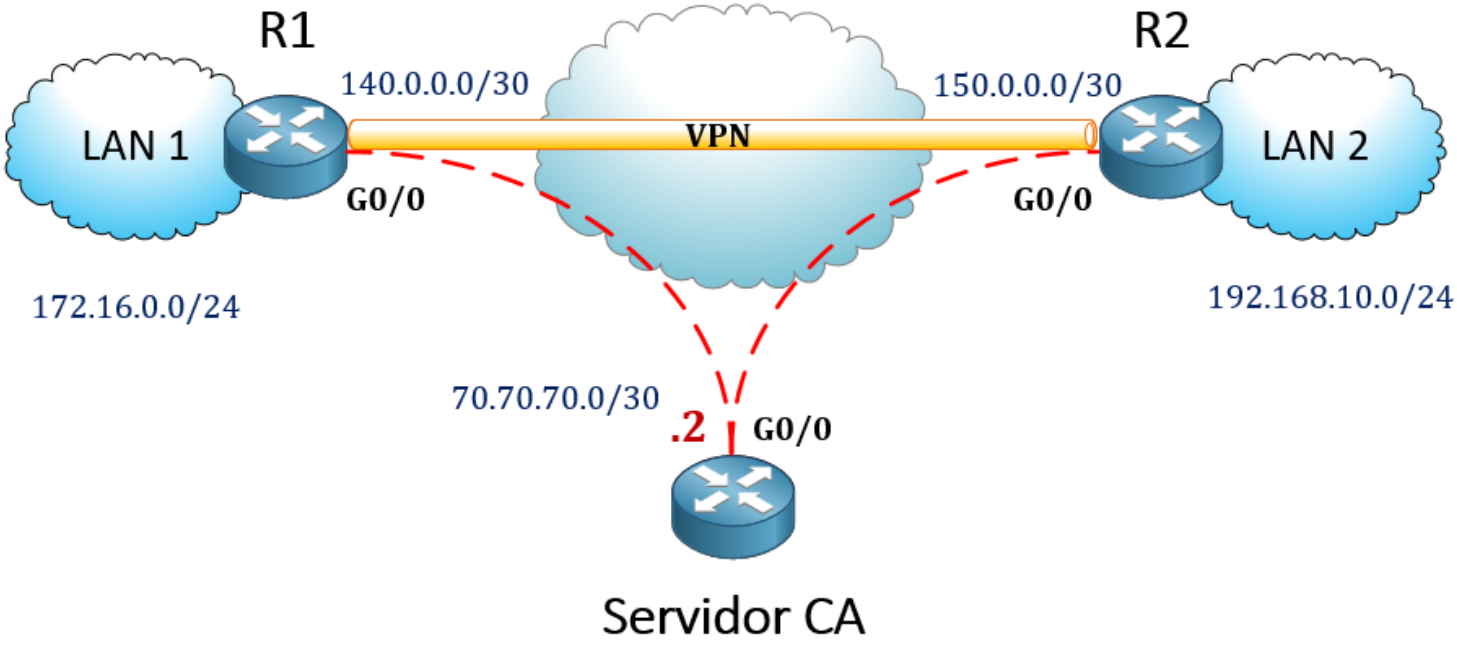
Existen muchos tipos de VPNs: Sitio-a-Sitio, GETVPN, EZVPN, RA VPN, FlexVPN, etc. Todos los tipos de VPN funcionan muy bien y se les puede aplicar métodos de autenticación para fortalecer la protección de los datos que pasan a través de sus túneles; estos métodos son aplicados junto con *IPSec*:

- Pre-shared-key
- Certificado digital

Pre-Shared-Key es un tipo de autenticación comúnmente utilizada, pero cuenta con poca escalabilidad, ya que se deben configurar manualmente la llave en cada dispositivo que participara en la conexión VPN. Aplicar este tipo de autenticación es menos complicado y puede ser sugerido para pocas conexiones de VPN.

Certificados digitales es tipo de autenticación escalable, cuenta con un mecanismo centralizado donde los certificados se encuentran en un dispositivo conocido como: autoridad certificada (CA), este puede ser un servidor, firewall o incluso un router que cuente con el licenciamiento o capacidades correctas. Los dispositivos que conforman la conexión VPN deben poder tener conectividad con el servidor CA para obtener a través de los certificados toda la información para autenticación, a estos certificados se les puede aplicar tiempo de caducidad para optimizar la protección.

Basado en lo que hemos conversado previamente, procederemos a realizar nuestra configuración basada en la siguiente topología:



La topología cuenta una nube central que si se desea simular puede ser un router que interconecte a R1, R2 y el servidor CA.

R1 y R2 serán los routers que crearán la conexión VPN y protegerán la comunicación entre la LAN1 y LAN2.

Ambos routers tendrán conectividad al servidor CA para poder obtener el certificado con la información sobre la autenticación a utilizar en la fase 1.

CONFIGURACIONES

Configuración del servidor CA y R1 – R2

Asumiendo los routers se encuentran ya pre-configurados, iniciaremos con la configuración del servidor CA y R1 – R2. Las loopbacks 0 representaran las redes LAN.

Pre configuraciones:

R1

```
interface Loopback0
 ip address 172.16.0.1 255.255.255.0

interface Ethernet0/0
 ip address 140.0.0.2 255.255.255.252


ip route 0.0.0.0 0.0.0.0 140.0.0.1
```

R2

```
interface Loopback0
 ip address 192.168.10.1 255.255.255.0

interface Ethernet0/0
 ip address 150.0.0.2 255.255.255.252


ip route 0.0.0.0 0.0.0.0 150.0.0.1
```

Servidor-CA

```
interface Ethernet0/0

 ip address 70.70.70.2 255.255.255.252


ip route 0.0.0.0 0.0.0.0 70.70.70.1
```

CONFIGURACIONES PARA EL CERTIFICADO DIGITAL

Las configuraciones son las siguientes para estos dispositivos.

Servidor-CA

Paso 1) Habilitar la capacidad de http en el router

```
Ip http server
```

Paso 2) Crear una llave RSA, en nuestro caso será: c1Sc0 con tamaño de 1024 bits.

```
Crypto key generate rsa label c1Sc0 modulus 1024
```

Paso 3) Importante que todos los routers mantengan una misma zona horaria y el tiempo se encuentre sincronizado, esto más que todo cuando se utilizan certificados que tienen caducidad. En nuestro ejercicio configuraremos el servidor CA como el NTP server con ID de llave 100 y llave CISCO.

```
ntp authentication-key 100 md5 106D202A2638 7
ntp authenticate
ntp trusted-key 100
ntp master 2
```

La llave se encriptará una vez aplicada, en nuestro caso el valor encriptado es: 106D202A2638

Paso 4) Habilitar al router como un servidor de certificado. Bajo el comando principal pueden aplicarse varias configuraciones dependiendo el contexto, en este ejemplo aplicaremos configuraciones básicas, el nombre que le colocaremos será: SERVIDOR, contraseña para almacenar: Cisco123.

```
crypto pki server SERVIDOR
 database level complete
 database archive pem password 7 05280F1C22431F5B4A
 grant auto
 shutdown

crypto pki trustpoint SERVIDOR
 revocation-check crl
 rsakeypair c1Sc0
```

Tomar en consideración que el servidor por defecto está apagado, si se activa antes de configurar el truspoint, podrían generarse inconvenientes donde la solución puede ser remover el servidor y volver a crearlo.

Una vez la configuración se ha verificado y todo está correcto, se aplica el comando no shutdown para habilitar el servidor.

```
crypto pki server SERVIDOR
 database level complete
 database archive pem password 7 032752180500701E1D
 grant auto
no shutdown
```

Si solo configuramos la primera parte (server) automáticamente creara una llave con el mismo nombre que le colocamos al servidor.

Para verificar que el servidor se encuentra activo, podemos ejecutar lo siguiente:

Show crypto pki server

```
SERVER-CA#
SERVER-CA#sh crypto pki server
Certificate Server SERVIDOR:
    Status: enabled
    State: enabled
    Server's configuration is locked  (enter "shut" to unlock it)
    Issuer name: CN=SERVIDOR
    CA cert fingerprint: 8DD393EE 069D7062 7152B66F 64E4E781
    Granting mode is: auto
    Last certificate issued serial number (hex): 1
    CA certificate expiration timer: 19:53:16 EET Jul 25 2022
    CRL NextUpdate timer: 01:53:16 EET Jul 27 2019
    Current primary storage dir: nvram:
    Database Level: Complete - all issued certs written as <serialnum>.cer
SERVER-CA#
```

Para verificar la llave, podemos ejecutar el siguiente comando:

Show crypto key mypubkey rsa

```
% Key pair was generated at: 19:53:16 EET Jul 26 2019
Key name: c1Sc0
Key type: RSA KEYS
 Storage Device: not specified
 Usage: General Purpose Key
 Key is not exportable.
 Key Data:
  30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00C5BAEA
  3FC6427C 7CCA0EC0 28DA1533 C6960331 3AC18374 8E7C520F 55DE704C 758AEAD0
  55F8C580 3AC0EAD4 EBBB7494 CAC26EFB 433FF716 45C02970 986345BD 88761BE0
  7394FB1C C146F599 7EFB6FBB 6F1BC7B7 18276E55 66B665D7 DAD611CD CFE89873
  4A86D0EA 59D1A824 DA4AF0A2 4FB40B3F F44E817D 10002BF3 10E92152 FD020301
  0001
% Key pair was generated at: 19:53:17 EET Jul 26 2019
Key name: c1Sc0.server
Key type: RSA KEYS
Temporary key
 Usage: Encryption Key
 Key is not exportable.
 Key Data:
  307C300D 06092A86 4886F70D 01010105 00036B00 30680261 0099F280 6B0099CB
  AD36C1B5 CEC97E77 42D1E90D 97D363CF EF9D7CE5 4FFF22B0 2DC18EE9 E8FEC255
  9EDD1A68 217233BD E8E8AF05 DB72FA79 AE48ADC4 4A42C708 724ECB70 66C8E803
  C507A05A E683DC84 5827826C 3DE4C5F8 344FF114 45BD80A6 69020301 0001
SERVER-CA#
```

R1 y R2

Paso 1) Podemos configurar NTP si es requerido o si los certificados tienen caducidad.

ntp authentication-key 100 md5 CISCO
ntp authenticate
ntp trusted-key 100
ntp server 70.70.70.2 key 100

Paso 2) Crear una llave RSA, puede ser la misma utilizada en el servidor o una nueva, utilizaremos c1Sc0 con tamaño de 1024 bits.

Crypto key generate rsa label c1Sc0 modulus 1024

Paso 3) Realizar la configuración del trustpoint donde nos enrolaremos con el servidor CA, la configuración es igual para ambos routers, en ocasiones podemos realizar cambios específicos, como subject-name, organizaciones, etc.

crypto pki trustpoint SERVIDOR
enrollment url http://70.70.70.2:80 ← Nos enrolamos con el servidor CA
subject-name cn=r1.cisco.com
revocation-check crl
rsakeypair c1Sc0 ← La llave que creamos previamente.

Paso 4) Aplicar la autenticación. Se colocará el nombre del trustpoint que creamos en el paso 3.

crypto pki authenticate SERVIDOR

```
R1(config)#crypto pki authenticate SERVIDOR
Certificate has the following attributes:
       Fingerprint MD5: 8DD393EE 069D7062 7152B66F 64E4E781
      Fingerprint SHA1: AB9C6ACA CB75E9A4 5A63F45E B20CAC89 106451AF

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
R1(config)#
```

Paso 5) Enrolarse con el trustpoint. La contraseña que aplica puede ser cualquiera que nosotros deseemos colocar, una vez completando las preguntas, aceptamos el certificado desde el Servidor CA.

Crypto pki enroll **SERVIDOR**

```
R1(config)#crypto pki enroll SERVIDOR
%
% Start certificate enrollment ..
% Create a challenge password. You will need to verbally provide this
   password to the CA Administrator in order to revoke your certificate.
   For security reasons your password will not be saved in the configuration.
   Please make a note of it.

Password:
Re-enter password:

% The subject name in the certificate will include: cn=r1.cisco.com
% The subject name in the certificate will include: R1
% Include the router serial number in the subject name? [yes/no]: no
% Include an IP address in the subject name? [no]: no
Request certificate from CA? [yes/no]: yes
% Certificate request sent to Certificate Authority
% The 'show crypto pki certificate verbose SERVIDOR' commandwill show the fingerprint.

R1(config)#
*Jul 26 23:55:13.993: CRYPTO_PKI:  Certificate Request Fingerprint MD5: 5993120E F5B4B0CA 3AFFFB9A B839BC55
*Jul 26 23:55:13.993: CRYPTO_PKI:  Certificate Request Fingerprint SHA1: 4C21460D 867CC935 8DA34AC3 D1F19AC4 4596D3B9
R1(config)#
*Jul 26 23:55:14.029: %PKI-6-CERTRET: Certificate received from Certificate Authority
R1(config)#
R1(config)#
```

Una vez hemos completado la configuración en ambos routers, podemos verificar el trustpoint configurado localmente.

Show crypto pki trustpoints

```
R1#sh crypto pki trustpoints
Trustpoint SERVIDOR:
    Subject Name:
    cn=SERVIDOR
          Serial Number (hex): 01
    Certificate configured.
    SCEP URL: http://70.70.70.2:80/cgi-bin
```

Al ejecutar el comando: show running-config y visualizaremos el certificado

Show running-config

```
crypto pki certificate chain SERVIDOR
 certificate 02
  30820202 3082016B A0030201 02020102 300D0609 2A864886 F70D0101 05050030
  13311130 0F060355 04031308 53455256 49444F52 301E170D 31393037 32363233
  35353134 5A170D32 30303732 35323335 3531345A 302A3115 30130603 55040313
  0C72312E 63697363 6F2E636F 6D311130 0F06092A 864886F7 0D010902 16025231
  30819F30 0D06092A 864886F7 0D010101 05000381 8D003081 89028181 00A16190
  1FB2FEBB 2F537B16 C327FD1D DC1E9B1F 42B6ADB8 BF5C9D6D C2A0FA11 152E3E42
  FD6D30BE 883A9F3A 43851510 39A26287 0CED96A6 61A822EC F4A7B48D B42864E7
  AFD69BA8 D380C899 3AA2199A BF78830B 66050312 7BC8E6AE B78CECCB 3237468E
  C2901090 F8427065 255FD5E2 2B514139 B4F636A2 AC27BC40 C8B746DF 29020301
  0001A34F 304D300B 0603551D 0F040403 0205A030 1F060355 1D230418 30168014
  543AFC0F B340E20C 45129A58 DC432FE8 515851B0 301D0603 551D0E04 1604141B
  1DF25987 E7B8B3D4 212B1FB0 6553BC67 B002B530 0D06092A 864886F7 0D010105
  05000381 8100302C BD862D95 6EDA545A AE3702DB D2CDA2EA A3BD0A59 2520AFB0
  69F3E666 98DC559B FB0BCA56 D197B5BF 59413C63 D9911FCE 2D4B8AD1 FDBF1662
  C502A839 A00BF97F E47C14B7 28AC38B6 D502D93B 2CD13652 D3F3B4C5 4E6D95E8
  B3004FC0 4FC52E16 E345D0F3 E72AF0C2 35859C93 71BD9BAB 4E964AD6 DA9AAC4A
  081D6C0D 48B7
        quit
 certificate ca 01
  308201FF 30820168 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  13311130 0F060355 04031308 53455256 49444F52 301E170D 31393037 32363137
  35333136 5A170D32 32303732 35313735 3331365A 30133111 300F0603 55040313
  08534552 5649444F 5230819F 300D0609 2A864886 F70D0101 01050003 818D0030
  81890281 8100C5BA EA3FC642 7C7CCA0E C028DA15 33C69603 313AC183 748E7C52
  0F55DE70 4C758AEA D055F8C5 803AC0EA D4EBBB74 94CAC26E FB433FF7 1645C029
  70986345 BD88761B E07394FB 1CC146F5 997EFB6F BB6F1BC7 B718276E 5566B665
  D7DAD611 CDCFE898 734A86D0 EA59D1A8 24DA4AF0 A24FB40B 3FF44E81 7D10002B
  F310E921 52FD0203 010001A3 63306130 0F060355 1D130101 FF040530 030101FF
  300E0603 551D0F01 01FF0404 03020186 301F0603 551D2304 18301680 14543AFC
  0FB340E2 0C45129A 58DC432F E8515851 B0301D06 03551D0E 04160414 543AFC0F
  B340E20C 45129A58 DC432FE8 515851B0 300D0609 2A864886 F70D0101 04050003
  81810049 9D7C2198 580AF357 33CB822A ACE1B584 D65E0E58 F2A15B74 B51E584C
  F04F9A6A DC6C65A5 3308F879 B3D26CA6 2C259A54 4FA03D59 207CF426 B4B700DC
 --More--
```

CONFIGURACIONES PARA LA VPN SITIO A SITIO

Las configuraciones son las siguientes para estos dispositivos.

R1 y R2

Paso 1) Crear una ACL para habilitar la comunicación entre las redes que deseamos puedan transferir información.

R1

```
Ip Access-list extended R1-A-R2
Permit ip 172.16.0.0 0.0.0.255 192.168.10.0 0.0.0.255
```

R2

```
Ip Access-list extended R2-A-R1
Permit ip 192.168.10.0 0.0.0.255 172.16.0.0 0.0.0.255
```

Paso 2) Configurar la fase 1 para que se autentique utilizando la información de autenticación recibida a través del certificado proporcionado por el servidor CA. Los valores de encriptación grupo u otros bajo esta fase deben ser los mismos entre los routers.

Para R1 y R2

```
crypto isakmp policy 1   ← el valor de 1 es un identificador cualquiera.
 encr 3des
 hash md5
 group 5
```

En este ejemplo utilice encriptación 3DES, Hash MD5 y el Grupo de Diffie hellman 5.

Paso 3) Implementamos la fase 2, aplicando los parámetros de autenticación y encriptación de la fase 2 para proteger los datos que fluirán a través de la VPN.

Para R1 y R2

Para esta sección utilizaremos 3DES para el algoritmo de encriptación y SHA para la autenticación, el modo por defecto es **tunnel**.

```
crypto ipsec transform-set FASE2-IPSEC esp-3des esp-sha-hmac
 mode tunnel
```

Paso 4) Creamos el crypto map, el cual incluirá la ACL que explícitamente indica lo que está permitido, los parámetros de IPSec y la dirección IP de la interface NBMA del router vecino donde haremos la conexión de la VPN. Nuestra secuencia iniciara con el valor 10, este valor puede ser cualquier numero entero y básicamente indica una secuencia, ya que podrían existir otros MAP aplicados en otras interfaces.

En nuestro ejemplo el MAP, se llamará VPN en ambos routers.

R1

```
crypto map VPN 10 ipsec-isakmp
 set peer 150.0.0.2  ← Dirección IP del router vecino
 set transform-set FASE2-IPSEC  ← la información de IPSEC
 match address R1-A-R2  ← la ACL creada previamente
```

R2

```
crypto map VPN 10 ipsec-isakmp
 set peer 140.0.0.2
 set transform-set FASE2-IPSEC
 match address R2-A-R1
```

Paso 5) Aplicar el MAP llamado VPN en las interfaces NBMA que donde se generara la comunicación hacia el exterior. En nuestro caso tanto R1 y R2 utilizan las interfaces G0/0

En R1 y R2

interface GigaEthernet0/0
crypto map VPN

Verificación

Para poder verificar que la VPN funciona correctamente, se debe generar trafico interesante, este se genera a través de lo que nosotros estamos permitiendo a través de la ACL, ejemplo R1:

Ping 192.168.10.1 source 172.16.0.1

```
R1#PING 192.168.10.1 SOURCE 172.16.0.1 REpeat 10
Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:
Packet sent with a source address of 172.16.0.1

Jul 27 00:37:32.250: %PKI-6-PKI_CRL_DOWNLOADED: CRL download notification sent for Issuer = cn=SERVIDOR
Success rate is 90 percent (9/10), round-trip min/avg/max = 5/5/6 ms
R1#
R1#
R1#PING 192.168.10.1 SOURCE 172.16.0.1 REpeat 10
Type escape sequence to abort.
Sending 10, 100-byte ICMP Echos to 192.168.10.1, timeout is 2 seconds:
Packet sent with a source address of 172.16.0.1
!!!!!!!!!!
Success rate is 100 percent (10/10), round-trip min/avg/max = 5/5/7 ms
R1#
```

Revisemos la fase 1

show crypto isakmp sa, donde nos muestra que se encuentra activa y funcionando correctamente.

```
R1#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
150.0.0.2       140.0.0.2       QM_IDLE           1001 ACTIVE

IPv6 Crypto ISAKMP SA

R1#
```

Revisemos la fase 2

Show crypto ipsec sa, donde el valor que se muestra a continuación, indica los pings que nosotros ejecutamos previamente.

```
R1#show crypto ipsec sa

interface: Ethernet0/0
    Crypto map tag: VPN, local addr 140.0.0.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (172.16.0.0/255.255.255.0/0/0)
   remote ident (addr/mask/prot/port): (192.168.10.0/255.255.255.0/0/0)
   current_peer 150.0.0.2 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 19, #pkts encrypt: 19, #pkts digest: 19
    #pkts decaps: 19, #pkts decrypt: 19, #pkts verify: 19
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 140.0.0.2, remote crypto endpt.: 150.0.0.2
     plaintext mtu 1446, path mtu 1500, ip mtu 1500, ip mtu idb Ethernet0/0
     current outbound spi: 0x9B10C2A(162597930)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0x167FEF8D(377483149)
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 1, flow_id: SW:1, sibling_flags 80004040, crypto map: VPN
        sa timing: remaining key lifetime (k/sec): (4187816/3404)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

R1#
```

Podemos apreciar que la VPN encapsula y des encapsula los datos, debemos recordar que los paquetes ICMP son recíprocos, por ende, al enviar una petición debemos obtener una respuesta, por esto vemos ambos valores bajo la fase2.

Para finalizar podemos indicar que nuestra VPN funciona correctamente.

Creado por: Julio E. Moisa