IKEv2 mit TrustSec SGT-Inline-Tagging und SGT-basiertem zonenbasiertem Firewall-Konfigurationsbeispiel

Inhalt

Einleitung

In diesem Dokument wird die Verwendung von Internet Key Exchange Version 2 (IKEv2) und eines Security Group Tags (SGT) zum Kennzeichnen von Paketen beschrieben, die an einen VPN-Tunnel gesendet wurden. Die Beschreibung umfasst einen typischen Bereitstellungs- und Anwendungsfall. In diesem Dokument wird auch eine SGT-fähige zonenbasierte Firewall (ZBF) erläutert. Außerdem werden zwei Szenarien vorgestellt:

- Eine ZBF, die auf vom IKEv2-Tunnel empfangenen SGT-Tags basiert
- Eine ZBF, die auf SGT eXchange Protocol (SXP)-Zuordnung basiert

Alle Beispiele umfassen Debugging auf Paketebene, um zu überprüfen, wie der SGT-Tag übertragen wird.

Voraussetzungen

Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- Grundkenntnisse der TrustSec-Komponenten
- Grundkenntnisse der Konfiguration von Cisco Catalyst Switches über die Kommandozeile (CLI)
- Erfahrung bei der Konfiguration einer Cisco Identity Services Engine (ISE)
- Grundkenntnisse der zonenbasierten Firewall
- Grundkenntnisse von IKEv2

Verwendete Komponenten

Die Informationen in diesem Dokument basierend auf folgenden Software- und Hardware-Versionen:

- Microsoft Windows 7 und Microsoft Windows XP
- Cisco Catalyst 3750-X Softwareversion 15.0 und höher
- Cisco Identity Services Engine Software Version 1.1.4 und höher
- Cisco 2901 Integrated Services Router (ISR) mit Softwareversion 15.3(2)T oder höher

Hinweis: IKEv2 wird nur auf Plattformen der ISR-Generation 2 (G2) unterstützt.

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netz Live ist, überprüfen Sie, ob Sie die mögliche Auswirkung jedes möglichen Befehls verstehen.

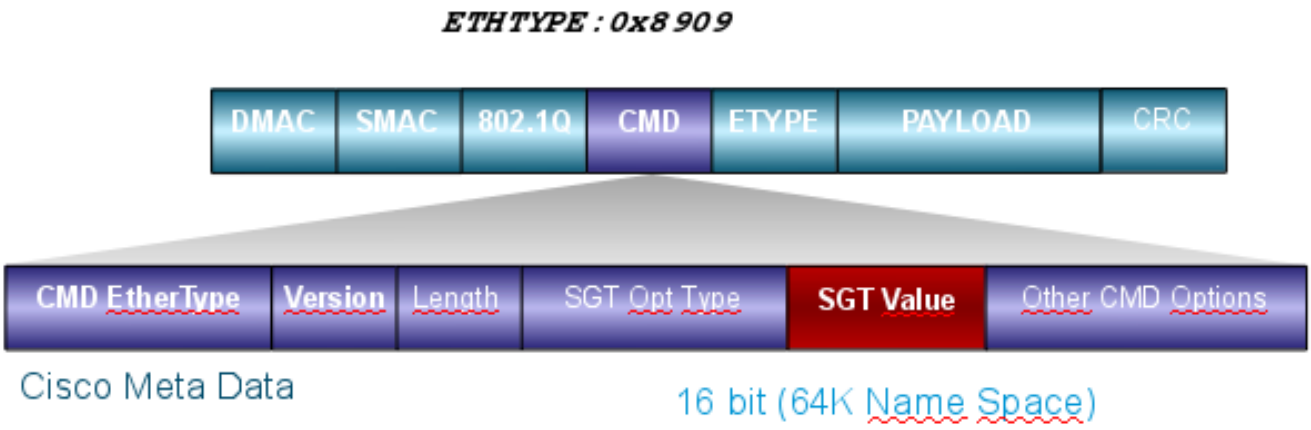
Security Group Tag (SGT)

Das SGT ist Teil der Cisco TrustSec-Lösungsarchitektur, die auf die Verwendung flexibler Sicherheitsrichtlinien ausgelegt ist, die nicht auf IP-Adressen basieren.

Der Datenverkehr in der TrustSec-Cloud wird klassifiziert und mit einem SGT-Tag markiert. Sie können Sicherheitsrichtlinien erstellen, die den Datenverkehr anhand dieses Tags filtern. Alle Richtlinien werden zentral über die ISE verwaltet und auf allen Geräten in der TrustSec-Cloud bereitgestellt.

Um die Informationen zum SGT-Tag weiterzugeben, hat Cisco den Ethernet-Frame ähnlich wie die 802.1q-Tags geändert. Der geänderte Ethernet-Frame kann nur von ausgewählten Cisco

Geräten verstanden werden. Das folgende geänderte Format:



Das Cisco Meta Data (CMD)-Feld wird direkt nach dem Quell-MAC-Adressfeld (SMAC) oder, falls verwendet, dem 802.1q-Feld eingefügt (wie in diesem Beispiel).

Um TrustSec-Clouds über das VPN zu verbinden, wurde eine Erweiterung für die IKE- und IPsec-Protokolle erstellt. Die als IPsec-Inline-Tagging bezeichnete Erweiterung ermöglicht das Senden von SGT-Tags in ESP-Paketen (Encapsulating Security Payload). Die ESP-Nutzlast wird so modifiziert, dass sie ein 8-Byte-CMD-Feld unmittelbar vor der Nutzlast des Pakets enthält. Das verschlüsselte Internet Control Message Protocol (ICMP)-Paket, das über das Internet gesendet wird, enthält beispielsweise [IP][ESP][CMD][IP][ICMP][DATA].

Detaillierte Informationen finden Sie im zweiten Teil des Artikels.

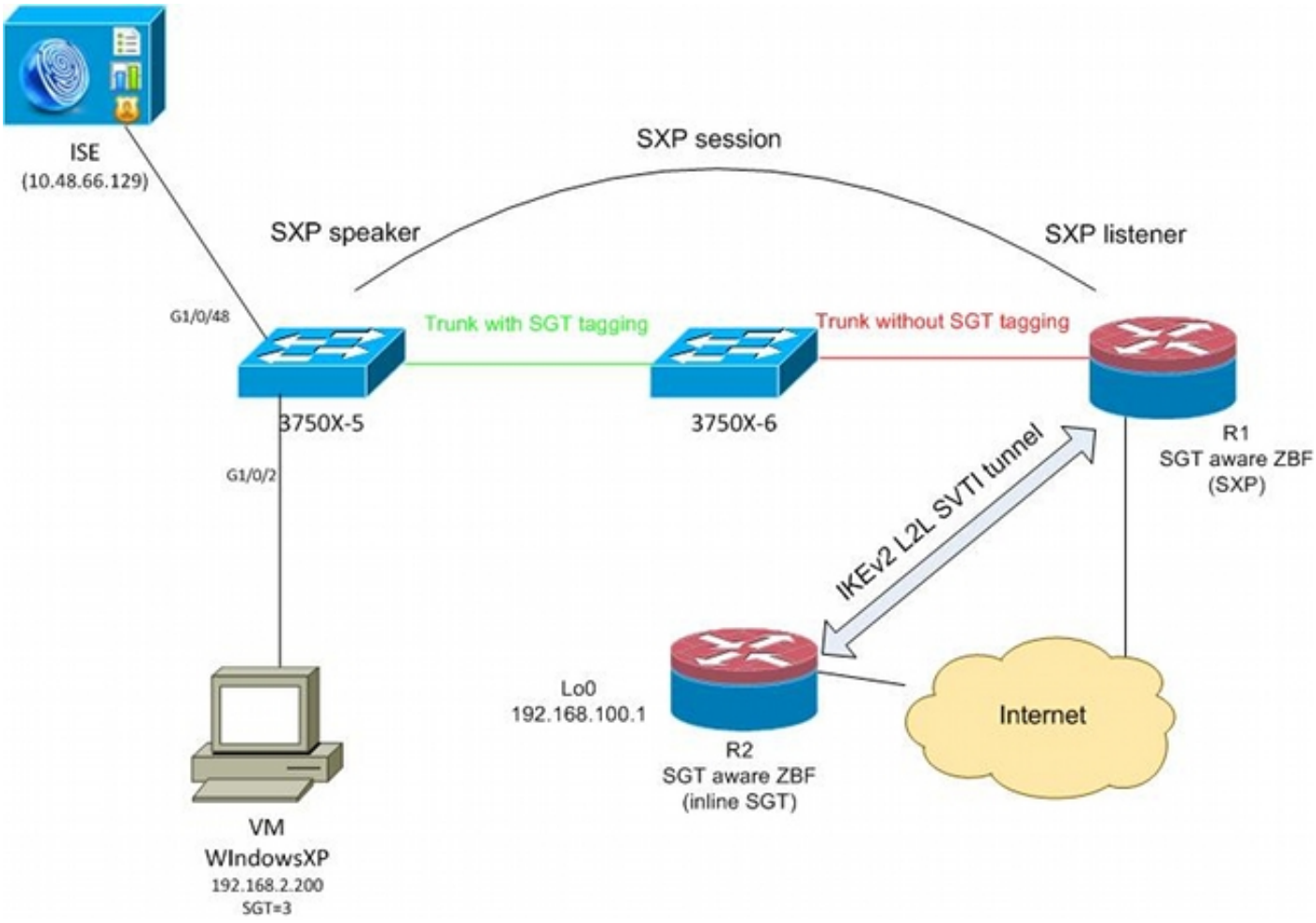
Konfigurieren

Hinweise:

Das Output Interpreter-Tool (nur registrierte Kunden) unterstützt bestimmte **show**-Befehle. Verwenden Sie das Output Interpreter-Tool, um eine Analyse der **show**-Befehlsausgabe anzuzeigen.

Lesen Sie den Artikel Wichtige Informationen zu Debug-Befehlen, bevor Sie **debug**-Befehle verwenden.

Netzwerkdiagramm



Datenverkehrsfluss

In diesem Netzwerk sind 3750X-5 und 3750X-6 Catalyst Switches innerhalb der TrustSec-Cloud. Beide Switches verwenden die automatische Bereitstellung von Protected Access Credentials (PACs), um Teil der Cloud zu werden. Der 3750X-5 wurde als Seed und der 3750X-6 als Non-Seed-Gerät verwendet. Der Datenverkehr zwischen beiden Switches wird mit MACsec verschlüsselt und richtig gekennzeichnet.

WindowsXP verwendet 802.1x, um auf das Netzwerk zuzugreifen. Nach erfolgreicher Authentifizierung gibt die ISE das SGT-Tag-Attribut zurück, das für diese Sitzung angewendet wird. Der gesamte von diesem PC stammende Datenverkehr wird mit SGT=3 gekennzeichnet.

Router 1 (R1) und Router 2 (R2) sind 2901 ISR. Da ISR G2 derzeit kein SGT-Tagging unterstützt, befinden sich R1 und R2 außerhalb der TrustSec-Cloud und verstehen nicht die Ethernet-Frames, die mit CMD-Feldern modifiziert wurden, um die SGT-Tags zu übergeben. Daher wird SXP verwendet, um Informationen über die IP/SGT-Zuordnung von 3750X-5 an R1 weiterzuleiten.

R1 verfügt über einen IKEv2-Tunnel, der so konfiguriert ist, dass er den an einen Remote-Standort (192.168.100.1) gerichteten Datenverkehr schützt, und in dem Inline-Tagging aktiviert ist. Nach der IKEv2-Aushandlung beginnt R1 mit der Kennzeichnung der an R2 gesendeten ESP-Pakete. Das Tagging basiert auf den SXP-Daten, die von 3750X-5 empfangen wurden.

R2 kann diesen Datenverkehr empfangen und basierend auf dem empfangenen SGT-Tag spezifische, von der ZBF definierte Aktionen ausführen.

Dasselbe gilt für R1. Durch die SXP-Zuordnung kann R1 ein vom LAN empfangenes Paket basierend auf einem SGT-Tag verwerfen, selbst wenn SGT-Frames nicht unterstützt werden.

TrustSec-Cloud-Konfiguration

Der erste Schritt bei der Konfiguration ist der Aufbau einer TrustSec-Cloud. Beide Switches der Serie 3750 benötigen Folgendes:

- Rufen Sie eine PAC ab, die für die Authentifizierung an der TrustSec-Cloud (ISE) verwendet wird.
- Authentifizierung und Weiterleitung des Network Device Admission Control (NDAC)-Prozesses
- Verwenden Sie das Security Association Protocol (SAP) für die MACsec-Aushandlung für eine Verbindung.

Dieser Schritt ist für diesen Anwendungsfall erforderlich, ist jedoch für das ordnungsgemäße Funktionieren des SXP-Protokolls nicht erforderlich. R1 benötigt keine PAC oder Umgebungsdaten von der ISE, um SXP-Zuordnung und IKEv2-Inline-Tagging durchzuführen.

Verifizierung

Die Verbindung zwischen 3750X-5 und 3750X-6 verwendet die von 802.1x ausgehandelte MACsec-Verschlüsselung. Beide Switches vertrauen und akzeptieren die vom Peer empfangenen SGT-Tags:

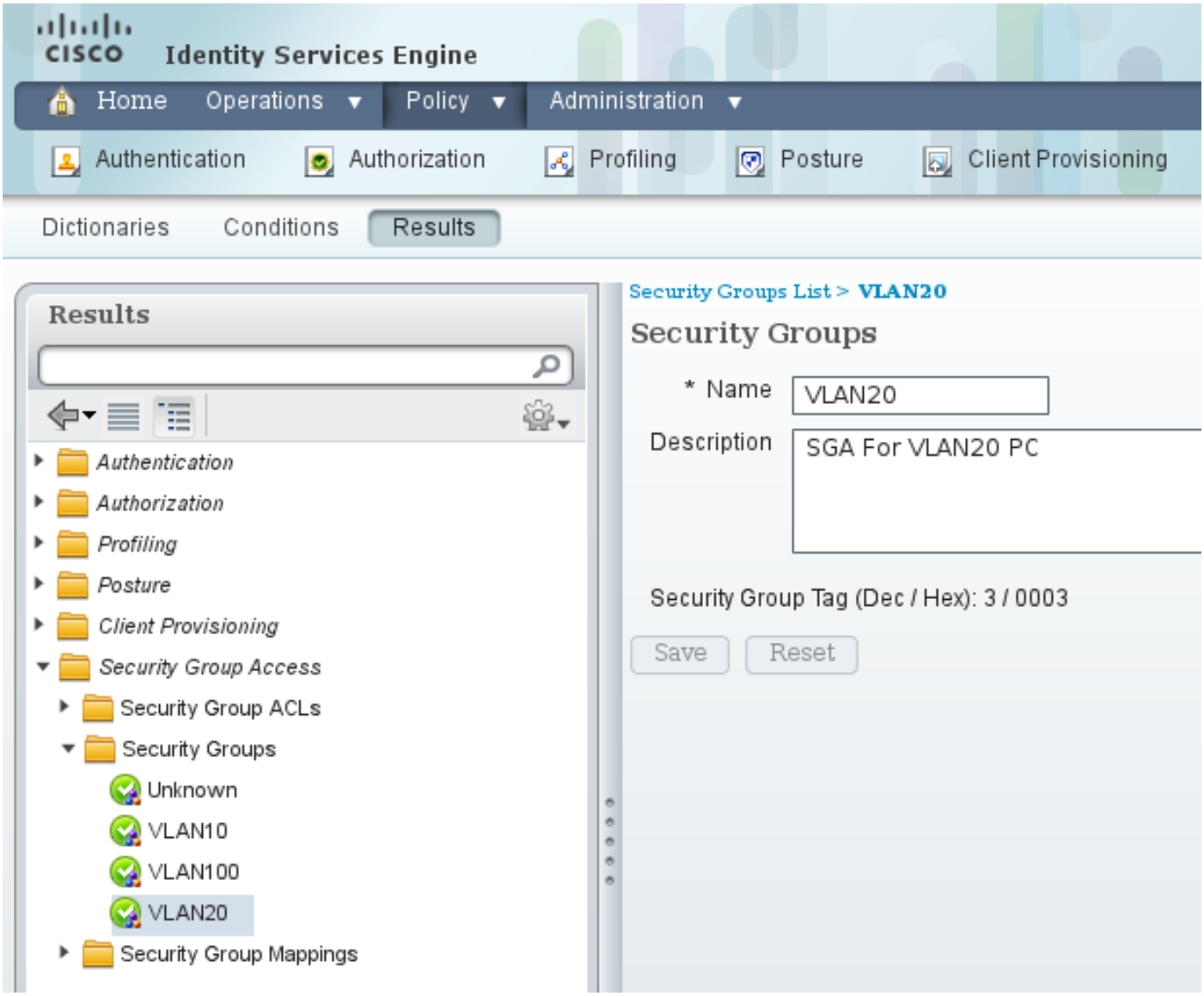
```
bsns-3750-5#show cts interface
Global Dot1x feature is Enabled
Interface GigabitEthernet1/0/20:
   CTS is enabled, mode:    DOT1X
   IFC state:               OPEN
   Authentication Status:   SUCCEEDED
       Peer identity:       "3750X6"
       Peer's advertised capabilities: "sap"
       802.1X role:         Supplicant
       Reauth period applied to link:  Not applicable to Supplicant role
    Authorization Status:    SUCCEEDED
       Peer SGT:            0:Unknown
       Peer SGT assignment: Trusted
   SAP Status:              SUCCEEDED
       Version:             2
        Configured pairwise ciphers:
            gcm-encrypt

       Replay protection:      enabled
       Replay protection mode: STRICT

       Selected cipher:        gcm-encrypt

   Propagate SGT:           Enabled
   Cache Info:
       Cache applied to link : NONE

   Statistics:
       authc success:              32
```

```
authc reject:              1543
authc failure:             0
authc no response:         0
authc logoff:              2
sap success:               32
sap fail:                  0
authz success:             50
authz fail:                0
port auth fail:            0
```

Eine rollenbasierte Zugriffskontrollliste (RBACL) kann nicht direkt auf Switches angewendet werden. Diese Richtlinien werden auf der ISE konfiguriert und automatisch auf die Switches heruntergeladen.

Client-Konfiguration

Der Client kann 802.1x, MAC Authentication Bypass (MAB) oder Webauthentifizierung verwenden. Denken Sie daran, die ISE so zu konfigurieren, dass die richtige Sicherheitsgruppe für die Autorisierungsregel zurückgegeben wird:



Verifizierung

Überprüfen Sie die Client-Konfiguration:

```
bsns-3750-5#show authentication sessions interface g1/0/2
           Interface:  GigabitEthernet1/0/2
         MAC Address:  0050.5699.4ea1
           IP Address:  192.168.2.200
           User-Name:  cisco
              Status:  Authz Success
              Domain:  DATA
     Security Policy:  Should Secure
     Security Status:  Unsecure
      Oper host mode:  multi-auth
    Oper control dir:  both
       Authorized By:  Authentication Server
         Vlan Policy:  20
                 SGT:  0003-0
     Session timeout:  N/A
        Idle timeout:  N/A
   Common Session ID:  C0A80001000006367BE96D54
     Acct Session ID:  0x00000998
              Handle:  0x8B000637

Runnable methods list:
      Method   State
      dot1x    Authc Success
      mab      Not run
```

Ab diesem Zeitpunkt wird der von 3750X-5 an andere Switches in der TrustSec-Cloud gesendete Client-Datenverkehr mit SGT=3 gekennzeichnet.

Ein Beispiel für Autorisierungsregeln finden Sie im Konfigurationsbeispiel und im Leitfaden zur Fehlerbehebung für Switches der Serie ASA und Catalyst 3750X.

SGT-Austauschprotokoll zwischen 3750X-5 und R1

R1 kann nicht der TrustSec-Cloud beitreten, da es sich um einen 2901 ISR G2-Router handelt, der keine Ethernet-Frames mit CMD-Feldern versteht. SXP ist also auf dem 3750X-5 konfiguriert:

```
bsns-3750-5#show run | i sxp
cts sxp enable
cts sxp default source-ip 192.168.1.10
cts sxp default password cisco
cts sxp connection peer 192.168.1.20 password default mode local
```

SXP ist auch auf R1 konfiguriert:

```
BSNS-2901-1#show run | i sxp
cts sxp enable
cts sxp default source-ip 192.168.1.20
cts sxp default password cisco
cts sxp connection peer 192.168.1.10 password default mode local listener
hold-time 0 0
```

Verifizierung

Stellen Sie sicher, dass R1 die IP/SGT-Zuordnungsinformationen empfängt:

```
BSNS-2901-1#show cts sxp sgt-map
SXP Node ID(generated):0xC0A80214(192.168.2.20)
IP-SGT Mappings as follows:
IPv4,SGT: <192.168.2.200 , 3>
source  : SXP;
Peer IP : 192.168.1.10;
Ins Num : 1;
Status  : Active;
Seq Num : 1
Peer Seq: 0
```

R1 weiß jetzt, dass der gesamte von 192.168.2.200 empfangene Datenverkehr so behandelt werden sollte, als wäre er mit SGT=3 gekennzeichnet.

IKEv2-Konfiguration zwischen R1 und R2

Hierbei handelt es sich um ein einfaches SVTI-basiertes Szenario (Static Virtual Tunnel Interfaces) mit intelligenten IKEv2-Standardeinstellungen. Pre-Shared Keys werden für die Authentifizierung verwendet, und Nullverschlüsselung wird für die einfache ESP-Paketanalyse verwendet. Der gesamte Datenverkehr zu 192.168.100.0/24 wird über die Tunnel1-Schnittstelle gesendet.

Dies ist die Konfiguration auf R1:

```
crypto ikev2 keyring ikev2-keyring
 peer 192.168.1.21
 address 192.168.1.21
 pre-shared-key cisco
 !
crypto ikev2 profile ikev2-profile
 match identity remote address 192.168.1.21 255.255.255.255
 authentication remote pre-share
 authentication local pre-share
 keyring local ikev2-keyring

crypto ipsec transform-set tset esp-null esp-sha-hmac
 mode tunnel
!
crypto ipsec profile ipsec-profile
 set transform-set tset
 set ikev2-profile ikev2-profile

interface Tunnel1
 ip address 172.16.1.1 255.255.255.0
 tunnel source GigabitEthernet0/1.10
 tunnel mode ipsec ipv4
 tunnel destination 192.168.1.21
 tunnel protection ipsec profile ipsec-profile

interface GigabitEthernet0/1.10
 encapsulation dot1Q 10
 ip address 192.168.1.20 255.255.255.0

ip route 192.168.100.0 255.255.255.0 172.16.1.2
```

Auf R2 wird der gesamte Rückverkehr zum Netzwerk 192.168.2.0/24 über die Tunnel1-

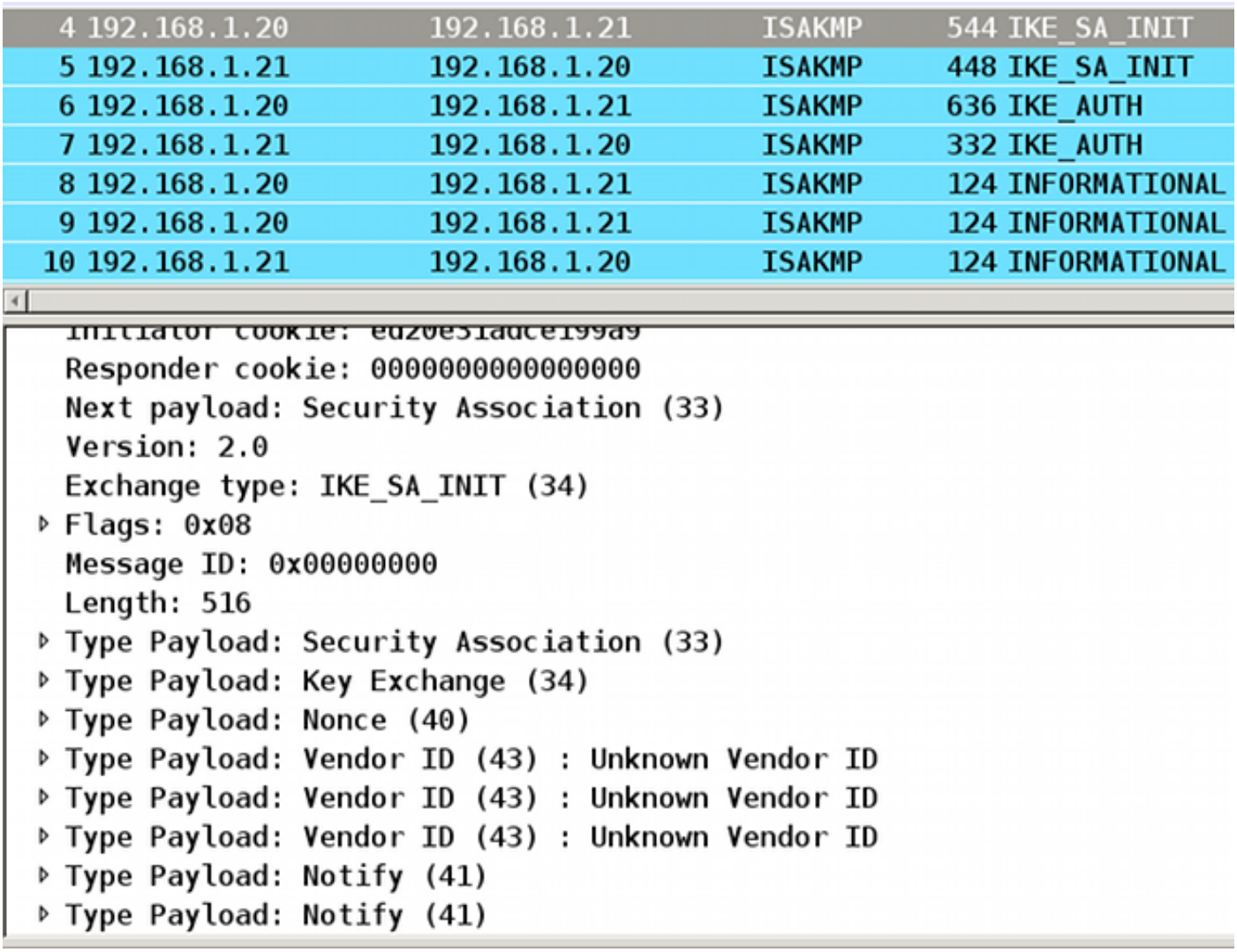
Schnittstelle gesendet:

```
crypto ikev2 keyring ikev2-keyring
 peer 192.168.1.20
 address 192.168.1.20
 pre-shared-key cisco

crypto ikev2 profile ikev2-profile
 match identity remote address 192.168.1.20 255.255.255.255
 authentication remote pre-share
 authentication local pre-share
 keyring local ikev2-keyring

crypto ipsec transform-set tset esp-null esp-sha-hmac
 mode tunnel

crypto ipsec profile ipsec-profile
 set transform-set tset
 set ikev2-profile ikev2-profile

interface Loopback0
description Protected Network
 ip address 192.168.100.1 255.255.255.0

interface Tunnel1
 ip address 172.16.1.2 255.255.255.0
 tunnel source GigabitEthernet0/1.10
 tunnel mode ipsec ipv4
 tunnel destination 192.168.1.20
 tunnel protection ipsec profile ipsec-profile

interface GigabitEthernet0/1.10
 encapsulation dot1Q 10
 ip address 192.168.1.21 255.255.255.0

ip route 192.168.2.0 255.255.255.0 172.16.1.1
```

Auf beiden Routern ist nur ein Befehl erforderlich, um das Inline-Tagging zu aktivieren: der Befehl **crypto ikev2 cts sgt**.

Verifizierung

Die Inline-Kennzeichnung muss ausgehandelt werden. Im ersten und zweiten IKEv2-Paket wird eine bestimmte Anbieter-ID gesendet:



Es gibt drei Anbieter-IDs (VIDs), die Wireshark nicht bekannt sind. Sie betreffen:

- DELETE-REASON, unterstützt von Cisco
- FlexVPN, unterstützt von Cisco
- SGT-Inline-Kennzeichnung

Dies wird durch die Debugs überprüft. R1, ein IKEv2-Initiator, sendet:

```
debug crypto ikev2 internal

*Jul 25 07:58:10.633: IKEv2:Construct Vendor Specific Payload: DELETE-REASON
*Jul 25 07:58:10.633: IKEv2:(1): Sending custom vendor id : CISCO-CTS-SGT

*Jul 25 07:58:10.633: IKEv2:Construct Vendor Specific Payload: (CUSTOM)
*Jul 25 07:58:10.633: IKEv2:Construct Vendor Specific Payload: (CUSTOM)
```

R1 empfängt ein zweites IKEv2-Paket und dieselbe VID:

```
*Jul 25 07:58:10.721: IKEv2:Parse Vendor Specific Payload: CISCO-DELETE-REASON VID
*Jul 25 07:58:10.721: IKEv2:Parse Vendor Specific Payload: (CUSTOM) VID
*Jul 25 07:58:10.721: IKEv2:Parse Vendor Specific Payload: (CUSTOM) VID
*Jul 25 07:58:10.721: IKEv2:Parse Notify Payload: NAT_DETECTION_SOURCE_IP
NOTIFY(NAT_DETECTION_SOURCE_IP)
*Jul 25 07:58:10.725: IKEv2:Parse Notify Payload: NAT_DETECTION_DESTINATION_IP
NOTIFY(NAT_DETECTION_DESTINATION_IP)
```

```
*Jul 25 07:58:10.725: IKEv2:(1): Received custom vendor id : CISCO-CTS-SGT
```

Beide Seiten stimmen daher zu, CMD-Daten an den Anfang der ESP-Nutzlast zu setzen.

Überprüfen Sie die IKEv2-Sicherheitszuordnung (Security Association, SA), um diese Vereinbarung zu überprüfen:

```
BSNS-2901-1#show crypto ikev2 sa detailed
 IPv4 Crypto IKEv2  SA

Tunnel-id Local                 Remote                fvrf/ivrf            Status
1         192.168.1.20/500      192.168.1.21/500      none/none            READY
     Encr: AES-CBC, keysize: 256, Hash: SHA512, DH Grp:5, Auth sign: PSK, Auth
verify: PSK
     Life/Active Time: 86400/225 sec
     CE id: 1019, Session-id: 13
     Status Description: Negotiation done
     Local spi: 1A4E0F7D5093D2B8       Remote spi: 08756042603C42F9
     Local id: 192.168.1.20
     Remote id: 192.168.1.21
     Local req msg id:  2              Remote req msg id:  0
     Local next msg id: 2              Remote next msg id: 0
     Local req queued:  2              Remote req queued:  0
     Local window:      5              Remote window:      5
     DPD configured for 0 seconds, retry 0
     Fragmentation not configured.
     Extended Authentication not configured.
     NAT-T is not detected
     Cisco Trust Security SGT is enabled
     Initiator of SA : Yes

 IPv6 Crypto IKEv2 SA
```

Nachdem der Datenverkehr vom Windows-Client an 192.168.100.1 gesendet wurde, zeigt R1 Folgendes an:

```
BSNS-2901-1#sh crypto session detail
Crypto session current status

Code: C - IKE Configuration mode, D - Dead Peer Detection
K - Keepalives, N - NAT-traversal, T - cTCP encapsulation
X - IKE Extended Authentication, F - IKE Fragmentation

Interface: Tunnel1
Uptime: 00:01:17
Session status: UP-ACTIVE
Peer: 192.168.1.21 port 500 fvrf: (none) ivrf: (none)
     Phase1_id: 192.168.1.21
     Desc: (none)
 IKEv2 SA: local 192.168.1.20/500 remote 192.168.1.21/500 Active
         Capabilities:(none) connid:1 lifetime:23:58:43
 IPSEC FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
       Active SAs: 2, origin: crypto map
       Inbound:  #pkts dec'ed 4 drop 0 life (KB/Sec) 4227036/3522
       Outbound: #pkts enc'ed 9 drop 0 life (KB/Sec) 4227035/3522


BSNS-2901-1#show crypto ipsec sa detail

interface: Tunnel1
   Crypto map tag: Tunnel1-head-0, local addr 192.168.1.20
```

```
  protected vrf: (none)
  local  ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
  remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
  current_peer 192.168.1.21 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 9, #pkts encrypt: 9, #pkts digest: 9
   #pkts decaps: 4, #pkts decrypt: 4, #pkts verify: 4
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #pkts no sa (send) 0, #pkts invalid sa (rcv) 0
   #pkts encaps failed (send) 0, #pkts decaps failed (rcv) 0
   #pkts invalid prot (recv) 0, #pkts verify failed: 0
   #pkts invalid identity (recv) 0, #pkts invalid len (rcv) 0
   #pkts replay rollover (send): 0, #pkts replay rollover (rcv) 0
   ##pkts replay failed (rcv): 0
   #pkts tagged (send): 9, #pkts untagged (rcv): 4
    #pkts not tagged (send): 0, #pkts not untagged (rcv): 0
   #pkts internal err (send): 0, #pkts internal err (recv) 0
   #send dummy packets 9, #recv dummy packets 0

    local crypto endpt.: 192.168.1.20, remote crypto endpt.: 192.168.1.21
    plaintext mtu 1454, path mtu 1500, ip mtu 1500, ip mtu idb
GigabitEthernet0/1.10
    current outbound spi: 0x9D788FE1(2641924065)
    PFS (Y/N): N, DH group: none

    inbound esp sas:
     spi: 0xDE3D2D21(3728551201)
       transform: esp-null esp-sha-hmac ,
       in use settings ={Tunnel, }
       conn id: 2020, flow_id: Onboard VPN:20, sibling_flags 80000040,
crypto map: Tunnel1-head-0
       sa timing: remaining key lifetime (k/sec): (4227036/3515)
       IV size: 0 bytes
       replay detection support: Y
       Status: ACTIVE(ACTIVE)

    inbound ah sas:

    inbound pcp sas:

    outbound esp sas:
     spi: 0x9D788FE1(2641924065)
       transform: esp-null esp-sha-hmac ,
       in use settings ={Tunnel, }
       conn id: 2019, flow_id: Onboard VPN:19, sibling_flags 80000040,
crypto map: Tunnel1-head-0
       sa timing: remaining key lifetime (k/sec): (4227035/3515)
       IV size: 0 bytes
       replay detection support: Y
       Status: ACTIVE(ACTIVE)

    outbound ah sas:

    outbound pcp sas:
BSNS-2901-1#
```

Beachten Sie, dass getaggte Pakete gesendet wurden.

Wenn R1 Datenverkehr, der vom Windows-Client an R2 gesendet wird, mit Tags kennzeichnen muss, stellen Sie sicher, dass das ESP-Paket korrekt mit SGT=3 gekennzeichnet wurde:

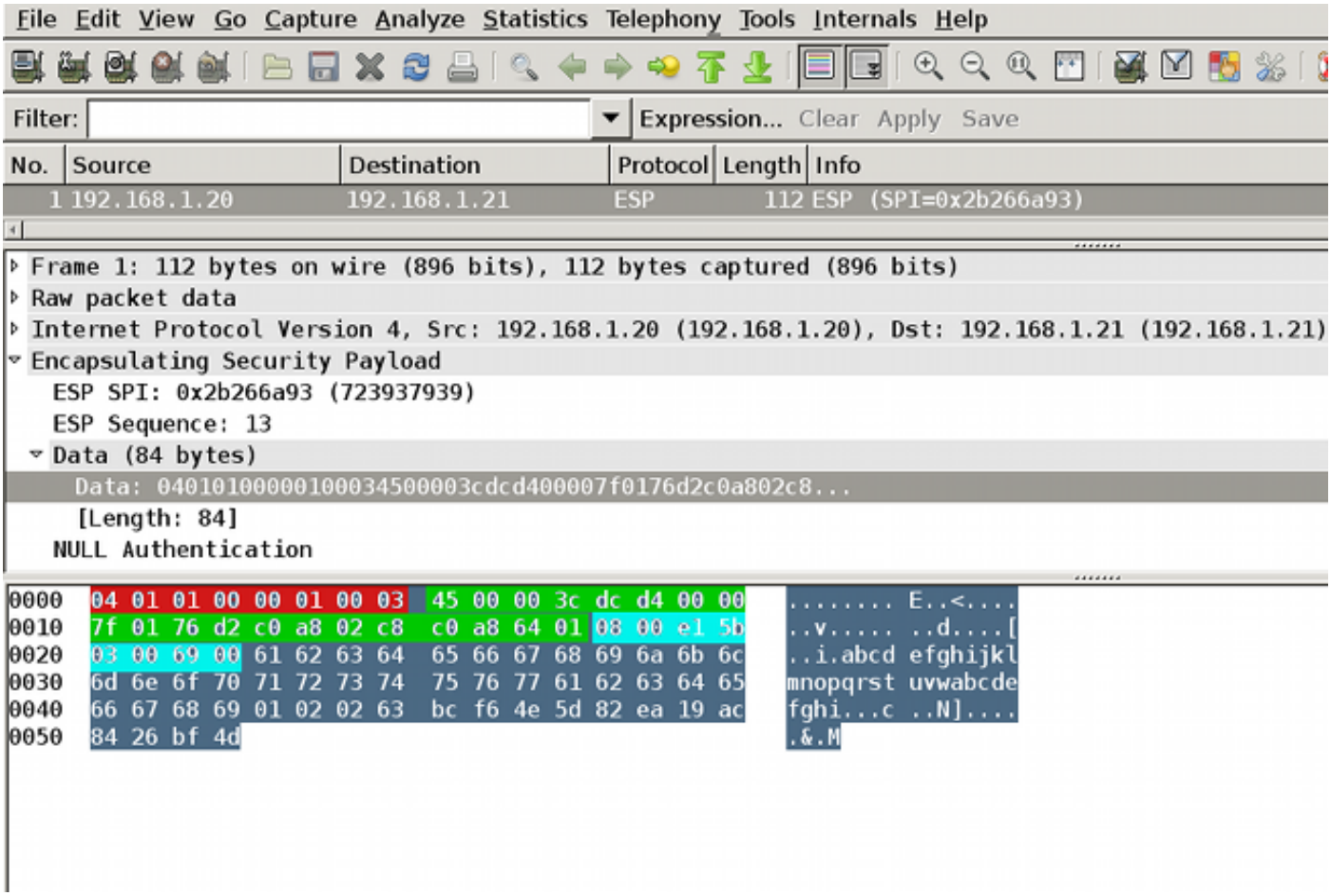
```
debug crypto ipsc metadata sgt
*Jul 23 19:01:08.590: IPsec SGT:: inserted SGT = 3 for src ip 192.168.2.200
```

Der restliche Datenverkehr vom selben VLAN, der vom Switch stammt, lautet standardmäßig SGT=0:

```
*Jul 23 19:43:08.590: IPsec SGT:: inserted SGT = 0 for src ip 192.168.2.10
```

Überprüfung auf ESP-Paketebene

Verwenden Sie Embedded Packet Capture (EPC), um den ESP-Datenverkehr von R1 nach R2 zu überprüfen, wie in der folgenden Abbildung dargestellt:



Wireshark wurde zum Decodieren der Nullverschlüsselung für den Sicherheitsparameterindex (SPI) verwendet. Im IPv4-Header sind die Quell- und Ziel-IP-Adresse die Internet-IP-Adressen der Router (die als Tunnelquelle und Ziel verwendet werden).

Die ESP-Nutzlast umfasst das 8-Byte-CMD-Feld, das rot hervorgehoben ist:

- 0x04 - Nächster Header, der IP ist
- 0x01 - Länge (4 Bytes nach dem Header, 8 Bytes mit dem Header)
- 0x01 - Version 01
- 0x00 - Reserviert
- 0x00 - SGT-Länge (insgesamt 4 Byte)
- 0x01 - SGT-Typ
- 0x0003 - SGT-Tag (die letzten beiden Oktetts sind 00 und 03; SGT wird für den Windows-

Client verwendet)

Da für die Tunnelschnittstelle der IPsec-IPv4-Modus verwendet wurde, ist der nächste Header "IP", der grün markiert ist. Die Quell-IP lautet c0 a8 02 c8 (192.168.2.200), und die Ziel-IP lautet c0 a8 64 01 (192.168.100.1). Die Protokollnummer lautet 1, also ICMP.

Der letzte Header ist ICMP, blau hervorgehoben, mit Typ 08 und Code 8 (Echo Request).

Die nächste ICMP-Nutzlast ist 32 Byte lang (d. h. Buchstaben von a bis i). Die Nutzlast in der Abbildung ist typisch für einen Windows-Client.

Die restlichen ESP-Header folgen der ICMP-Payload:

- 0x01 0x02 - Polsterung.
- 0x02 - Schrittlänge.
- 0x63 - Nächster Header, der auf das Protokoll 0x63 verweist, das "Any private encryption scheme" lautet. Dies zeigt an, dass das nächste Feld (das erste Feld in den ESP-Daten) das SGT-Tag ist.
- 12 Byte Integritätsprüfungswert.

Das CMD-Feld befindet sich innerhalb der ESP-Nutzlast, die normalerweise verschlüsselt ist.

IKEv2-Fehlfunktionen: GRE- oder IPsec-Modus

Bisher wurde in diesen Beispielen der Tunnelmodus IPsec IPv4 verwendet. Was passiert, wenn der Generic Routing Encapsulation (GRE)-Modus verwendet wird?

Wenn der Router ein Übertragungs-IP-Paket in GRE kapselt, betrachtet TrustSec das Paket als lokal generiert, d. h., die Quelle des GRE-Pakets ist der Router und nicht der Windows-Client. Wenn das CMD-Feld hinzugefügt wird, wird immer das Standard-Tag (SGT=0) anstelle eines bestimmten Tags verwendet.

Wenn Datenverkehr vom Windows-Client (192.168.2.200) im IPsec-Modus IPv4 gesendet wird, wird SGT=3 angezeigt:

```
debug crypto ipsc metadata sgt
*Jul 23 19:01:08.590: IPsec SGT:: inserted SGT = 3 for src ip 192.168.2.200
```

Nachdem der Tunnelmodus jedoch für den gleichen Datenverkehr in GRE geändert wurde, sehen Sie, dass SGT=0 ist. In diesem Beispiel ist 192.168.1.20 die Tunnelquellen-IP:

```
*Jul 25 20:34:08.577: IPsec SGT:: inserted SGT = 0 for src ip 192.168.1.20
```

Hinweis: Daher ist es sehr wichtig, **GRE nicht zu verwenden**.

Siehe Cisco Bug-ID CSCuj25890, IOS IPSec-Inline-Tagging für den GRE-Modus: Einfügen des Router-SGT. Dieser Fehler wurde erstellt, um eine ordnungsgemäße SGT-Propagierung zu ermöglichen, wenn Sie GRE verwenden. SGT over DMVPN wird von Cisco IOS® XE 3.13^{S} unterstützt

ZBF basierend auf SGT-Tags von IKEv2

Dies ist eine Beispielkonfiguration für ZBF auf R2. Der VPN-Datenverkehr mit SGT=3 kann identifiziert werden, da alle vom IKEv2-Tunnel empfangenen Pakete markiert sind (d. h., sie enthalten das CMD-Feld). So kann der VPN-Datenverkehr verworfen und protokolliert werden:

```
class-map type inspect match-all TAG_3
 match security-group source tag 3
class-map type inspect match-all TAG_ANY
 match security-group source tag 0
!
policy-map type inspect FROM_VPN
 class type inspect TAG_3
 drop log
 class type inspect TAG_ANY
 pass log
 class class-default
 drop
!
zone security vpn
zone security inside
zone-pair security ZP source vpn destination self
 service-policy type inspect FROM_VPN

interface Tunnel1
 ip address 172.16.1.2 255.255.255.0
 zone-member security vpn
```

Verifizierung

Wenn ein Ping an 192.168.100.1 vom Windows-Client stammt (SGT=3), zeigen die Debugs Folgendes an:

```
*Jul 23 20:05:18.822: %FW-6-DROP_PKT: Dropping icmp session
192.168.2.200:0 192.168.100.1:0 on zone-pair ZP class TAG_3 due to
DROP action found in policy-map with ip ident 0
```

Für einen Ping, der von einem Switch stammt (SGT=0), zeigen die Debugs Folgendes:

```
*Jul 23 20:05:39.486: %FW-6-PASS_PKT: (target:class)-(ZP:TAG_ANY)
Passing icmp pkt 192.168.2.10:0 => 192.168.100.1:0 with ip ident 0
```

Die Firewall-Statistiken von R2 sind:

```
BSNS-2901-2#show policy-firewall stats all
Global Stats:
       Session creations since subsystem startup or last reset 0
       Current session counts (estab/half-open/terminating) [0:0:0]
       Maxever session counts (estab/half-open/terminating) [0:0:0]
       Last session created never
       Last statistic reset never
       Last session creation rate 0
       Maxever session creation rate 0
       Last half-open session total 0

policy exists on zp ZP
 Zone-pair: ZP

 Service-policy inspect : FROM_VPN
```

```
   Class-map: TAG_3 (match-all)
     Match: security-group source tag 3
     Drop
        4 packets, 160 bytes

   Class-map: TAG_ANY (match-all)
     Match: security-group source tag 0
      Pass
        5 packets, 400 bytes

   Class-map: class-default (match-any)
     Match: any
     Drop
       0 packets, 0 bytes
```

Es gibt vier Drops (Standardanzahl der von Windows gesendeten ICMP-Echos) und fünf Akzepte (Standardanzahl für den Switch).

ZBF-basiert auf SGT-Zuordnung über SXP

Es ist möglich, SGT-fähiges ZBF auf R1 auszuführen und den vom LAN empfangenen Datenverkehr zu filtern. Obwohl dieser Datenverkehr nicht mit einem SGT markiert ist, verfügt R1 über SXP-Zuordnungsinformationen und kann diesen Datenverkehr als markiert behandeln.

In diesem Beispiel wird eine Richtlinie zwischen dem LAN und den VPN-Zonen verwendet:

```
class-map type inspect match-all TAG_3
 match security-group source tag 3
class-map type inspect match-all TAG_ANY
 match security-group source tag 0
!
policy-map type inspect FROM_LAN
 class type inspect TAG_3
  drop log
 class type inspect TAG_ANY
  pass log
 class class-default
 drop
!
zone security lan
zone security vpn
zone-pair security ZP source lan destination vpn
 service-policy type inspect FROM_LAN

interface Tunnel1
 zone-member security vpn

interface GigabitEthernet0/1.20
 zone-member security lan
```

Verifizierung

Wenn ICMP Echo vom Windows-Client gesendet wird, werden die Drops angezeigt:

```
*Jul 25 09:22:07.380: %FW-6-DROP_PKT: Dropping icmp session 192.168.2.200:0
192.168.100.1:0 on zone-pair ZP class TAG_3 due to  DROP action found in
```

```
policy-map with ip ident 0

BSNS-2901-1#show policy-firewall stats all
Global Stats:
       Session creations since subsystem startup or last reset 0
       Current session counts (estab/half-open/terminating) [0:0:0]
       Maxever session counts (estab/half-open/terminating) [0:0:0]
       Last session created never
       Last statistic reset never
       Last session creation rate 0
       Maxever session creation rate 0
       Last half-open session total 0

policy exists on zp ZP
 Zone-pair: ZP

 Service-policy inspect : FROM_LAN

   Class-map: TAG_3 (match-all)
     Match: security-group source tag 3
      Drop
        4 packets, 160 bytes

   Class-map: TAG_ANY (match-all)
     Match: security-group source tag 0
      Pass
        5 packets, 400 bytes

   Class-map: class-default (match-any)
     Match: any
     Drop
       0 packets, 0 bytes
```

Da die SXP-Sitzung auf TCP basiert, können Sie auch eine SXP-Sitzung über einen IKEv2-Tunnel zwischen 3750X-5 und R2 erstellen und ZBF-Richtlinien basierend auf den Tags auf R2 ohne Inline-Tagging anwenden.

Roadmap

GET-VPN-Inline-Tagging wird auch auf dem ISR G2 und den Aggregation Services Routern der Cisco Serie ASR 1000 unterstützt. Das ESP-Paket enthält weitere 8 Byte für das CMD-Feld.

Unterstützung für Dynamic Multipoint VPN (DMVPN) ist ebenfalls geplant.

Weitere Informationen finden Sie in der Roadmap für die Cisco TrustSec-fähige Infrastruktur.

Überprüfung

Die Verifizierungsverfahren sind in den Konfigurationsbeispielen enthalten.

Fehlerbehebung

Für diese Konfiguration sind derzeit keine spezifischen Informationen zur Fehlerbehebung verfügbar.

Zugehörige Informationen

- Konfigurationsleitfaden für Cisco TrustSec-Switches: Erläuterungen zu Cisco TrustSec
- Buch 1: Cisco ASA Series General Operations CLI Configuration Guide, 9.1: Configuring the ASA to Integrate with Cisco TrustSec
- Versionshinweise für Cisco TrustSec - Allgemeine Verfügbarkeit: Versionshinweise für Cisco TrustSec 3.0 - Allgemeine Einsatzbereitschaft 2013
- Konfigurieren des IPsec-Inline-Taggings für TrustSec
- Cisco Group Encrypted Transport VPN Configuration Guide, Cisco IOS XE Release 3S: GET VPN-Unterstützung für IPsec-Inline-Tagging für Cisco TrustSec
- Technischer Support und Dokumentation für Cisco Systeme

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.