Understand Crypto ACL Counters within Policy Based VPN Tunnels

Contents

Introduction

This document describes the behavior of crypto Access Control List (ACL) counters within policy based VPN tunnels.

Prerequisites

Requirements

Cisco recommends knowledge of these topics:

- Policy Based Site to Site VPN on Cisco IOS® /Cisco IOS® XE platform
- Access control lists on Cisco IOS/Cisco IOS XE Platform

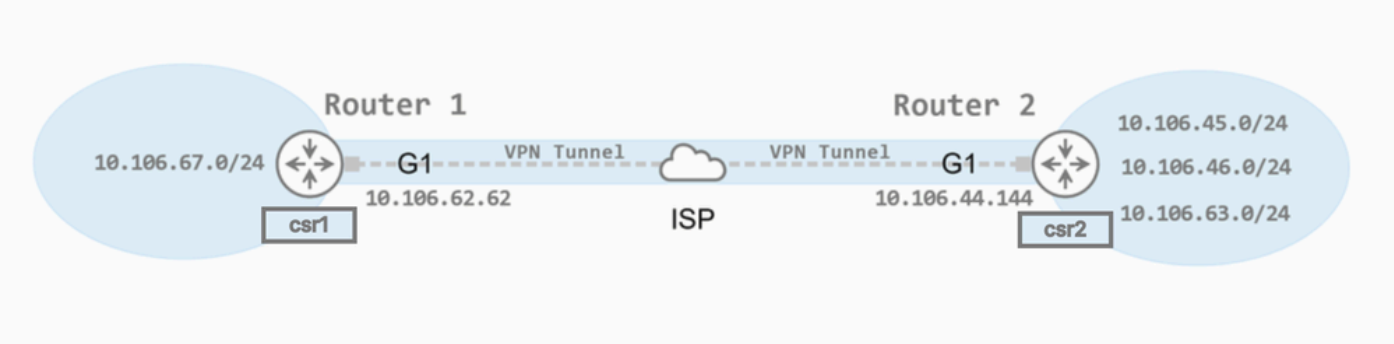
Components Used

The information in this document is based on these software and hardware versions:

- Cisco C8kv, Version 17.12.04(MD)

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. If your network is live, ensure that you understand the potential impact of any command.

Topology



Topology

Scenarios

By examining two distinct scenarios, we aim to understand how ACL hit counts are affected when traffic is initiated from different peers and when tunnels are reset.

1. **Scenario One: Traffic Initiated from Router1 while the VPN Tunnel is Inactive**

   In this scenario, the changes in ACL hit counts are analyzed when the VPN tunnel is initially down, and traffic is initiated from Router1. This analysis helps understand the initial setup and how the crypto ACL counters react to the first traffic flow attempt.

2. **Scenario Two: Traffic Initiated from Router2 while the VPN Tunnel is Active**

   In this scenario, the VPN tunnel is already established and traffic is initiated from Router2 is explored. This scenario provides insights into how ACL counters behave when the tunnel is active and traffic is introduced from a different peer.

By comparing these scenarios, we can gain a comprehensive understanding of the dynamics of ACL counters in VPN tunnels under varying conditions.

Configuration

We have configured a policy-based site-to-site VPN tunnel between two Cisco C8kv routers, designated as peers. Router1 is named "csr1" and Router2 is named "csr2".

Crypto Configuraton on Router1

```
csr1#sh ip int br
Interface              IP-Address      OK?   Method   Status  Protocol
GigabitEthernet1       10.106.62.62    YES   NVRAM    up      up
GigabitEthernet2       10.106.67.27    YES   NVRAM    up      up
```

```
csr1#sh run | sec crypto map
```

```
crypto map nigarapa_map 100 ipsec-isakmp
   set peer 10.106.44.144
   set transform-set new_ts
   set ikev2-profile new_profile
   match address new_acl
```

```
csr1#sh ip access-lists new_acl
Extended IP access list new_acl
   10 permit ip 10.106.67.0 0.0.0.255 10.106.45.0 0.0.0.255 log
   20 permit ip 10.106.67.0 0.0.0.255 10.106.46.0 0.0.0.255
   30 permit ip 10.106.67.0 0.0.0.255 10.106.63.0 0.0.0.255 log
```

```
csr1#sh run int GigabitEthernet1
Building configuration...

Current configuration : 162 bytes
!
interface GigabitEthernet1
ip address 10.106.62.62 255.255.255.0
ip nat outside
negotiation auto
no mop enabled
no mop sysid
crypto map nigarapa_map
end
```

Crypto Configuraton on Router2

```
csr2#sh ip int br
Interface              IP-Address        OK?     Method       Status     Protocol
GigabitEthernet1       10.106.44.144     YES     NVRAM        up         up
GigabitEthernet2       10.106.45.145     YES     NVRAM        up         up
GigabitEthernet3       10.106.46.146     YES     NVRAM        up         up
GigabitEthernet4       10.106.63.13      YES     NVRAM        up         up
```

```
csr2#sh run | sec crypto map
crypto map nigarapa_map 100 ipsec-isakmp
   set peer 10.106.62.62
   set transform-set new_ts
   set ikev2-profile new_profile
   match address new_acl
```

```
csr2#sh ip access-lists new_acl
Extended IP access list new_acl
   10 permit ip 10.106.45.0 0.0.0.255 10.106.67.0 0.0.0.255
   20 permit ip 10.106.46.0 0.0.0.255 10.106.67.0 0.0.0.255
   30 permit ip 10.106.63.0 0.0.0.255 10.106.67.0 0.0.0.255
```

```
csr2#sh run int GigabitEthernet1
Building configuration...

Current configuration : 163 bytes
!
interface GigabitEthernet1
ip address 10.106.44.144 255.255.255.0
ip nat outside
negotiation auto
no mop enabled
no mop sysid
crypto map nigarapa_map
end
```

Behavioral Analysis of Crypto Access Control List Counters within VPN Tunnels

Initially, both devices have an ACL hit count of zero on their respective crypto access lists.

Access Control List hit count of zero on their respective crypto access lists on both the peer devices.

Scenario One: Traffic Initiated from Router1 while the VPN Tunnel is Inactive

Initial State:

The VPN tunnel connecting Router1 (IP: 10.106.67.27) and Router2 (IP: 10.106.45.145) is currently inactive.

Action Taken:

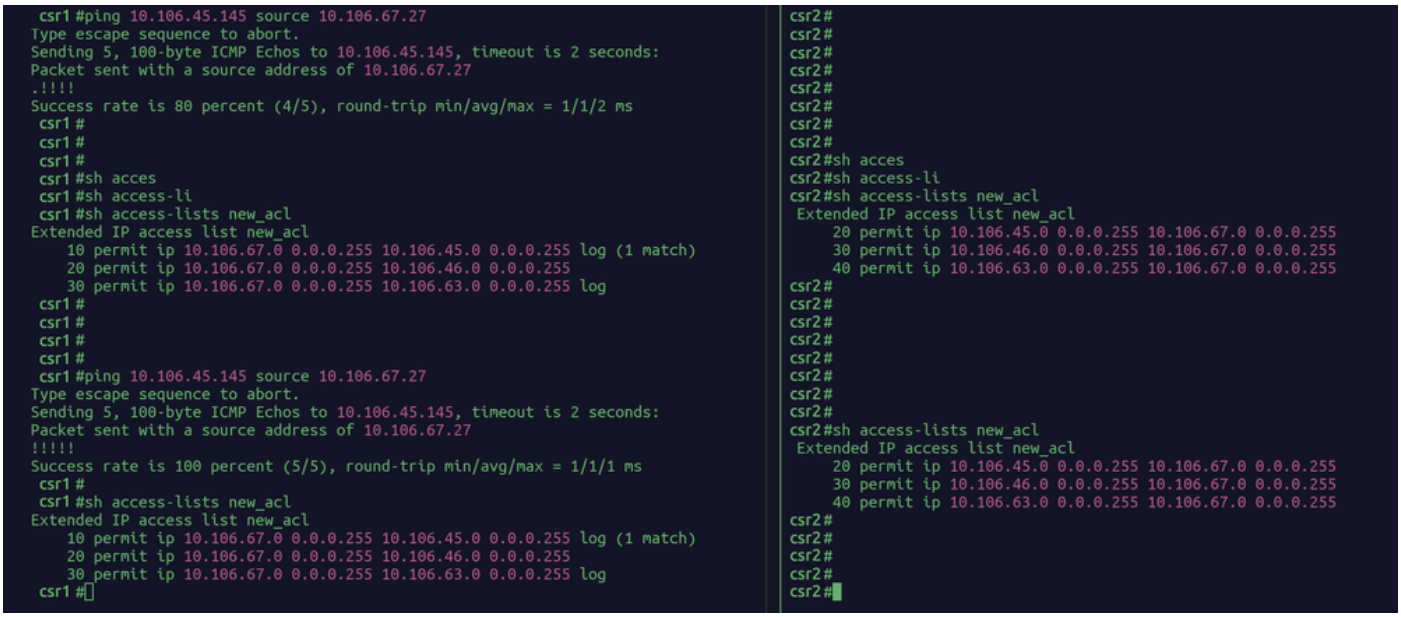
Traffic is initiated from Router1, intended to establish communication with Router2.

Observations:

1. ACL Counter Behavior:
   a. Upon initiating traffic from Router1, there is a noticeable increment in the Access Control List (ACL) counter on Router1. This increase occurs only once at the moment the tunnel attempts to establish.
   b. The rise in the ACL counter is exclusively observed on the initiating router, which is Router1 in this scenario. Router2 does not reflect any changes in its ACL counter at this stage.
2. Tunnel Establishment:
   a. After the initial increment corresponding to the traffic initiation, the tunnel between first and Router2 successfully establishes.
   b. Post-establishment of the tunnel, the ACL counter on Router1 stabilizes and exhibits no further increments, indicating that the ACL rule has been matched and is now consistently allowing traffic through the established tunnel.
3. Tunnel Re-Initiation:
   The ACL counter on Router1 experiences another increment only if the tunnel drops and requires re-

establishment. This suggests that the ACL rule is triggered by the initial traffic initiation which attempts to establish the tunnel, rather than by ongoing data transfer once the tunnel is active.

In summary, this scenario demonstrates that the ACL counter on Router1 is sensitive to the initial traffic attempts for tunnel creation but remains static once the VPN tunnel is up and operational.



Scenaria 1

Scenario Two:Traffic Initiated from Router2 while the VPN Tunnel is Active

Initial State:

The VPN tunnel connecting Router1 (IP: 10.106.67.27) and Router2 (IP: 10.106.45.145) is currently active and operational.

Action Taken:

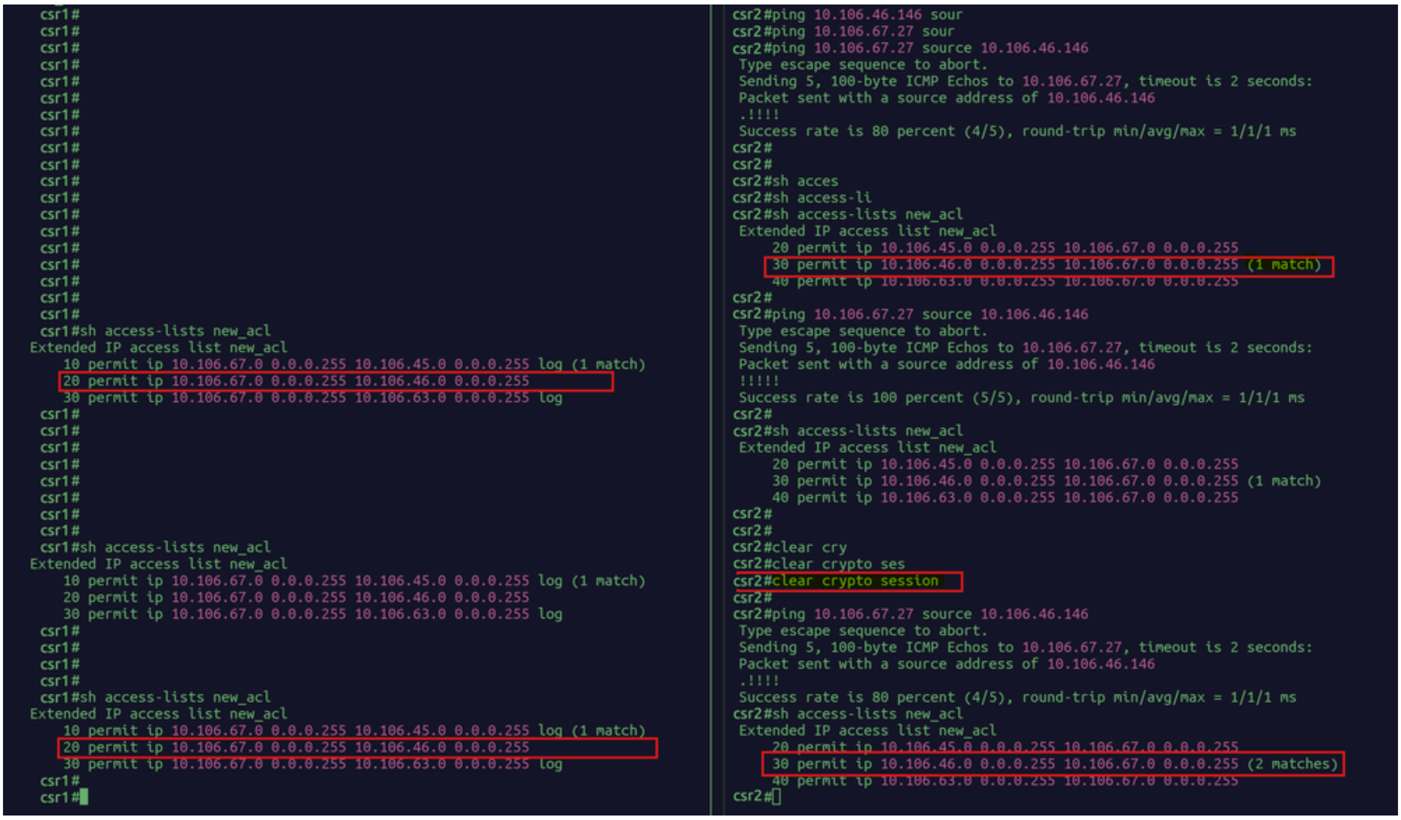
1. Traffic is initiated from Router2 towards Router1 while the tunnel is up.
2. Subsequently, the tunnel is deliberately cleared (or reset).
3. After the tunnel is cleared, Router2 initiates traffic again to re-establish the connection.

Observations:

1. Initial Traffic Initiation:
   a. When traffic is first initiated from Router2 while the tunnel is already established, there is no immediate change in the Access Control List (ACL) counter.
   b. This indicates that ongoing traffic within an already established tunnel does not trigger the ACL counter increment.
2. Tunnel Clearing and Re-Initiation:
   a. Upon clearing the tunnel, the established connection between the first and Router2 is temporarily disrupted. This necessitates a re-establishment process for any subsequent traffic.
   b. When traffic is re-initiated from Router2 after the tunnel has been cleared, there is an observable increment in the ACL counter on Router2. This increment signifies that the ACL rules are being engaged once more to facilitate the creation of the tunnel.
3. ACL Counter Specificity:
   The increment in the ACL counter occurs solely on the side initiating the traffic, which in this case is Router2. This behavior highlights the role of the ACL in monitoring and controlling traffic initiation

processes on the originating side, while Router1's ACL counter remains unaffected during this phase.

In summary, this scenario illustrates that the ACL counter on Router2 is responsive to the initiation of traffic when re-establishing a VPN tunnel. The counter does not increase with regular traffic flow within an active tunnel but reacts to the need for tunnel re-establishment, ensuring precise tracking of tunnel initiation events.



Scenario 2

Conclusion:

The behavior of the crypto ACL counters reveals that they register hit counts exclusively during the initiation phase of the VPN tunnel.

Initiator-Specific Increment: When the tunnel is initiated from Router1, the increase in the hit count is observed solely on Router1. Similarly, if the initiation occurs from Router2, the hit count rises only on Router2. This highlights the role of the ACL in monitoring the traffic initiation process at the source.

Stability Post-Establishment: Once the tunnel is successfully established, the ACL counters on both peers remain unchanged, indicating no further hits. This stability persists until the tunnel is either cleared or reset, and traffic initiation is attempted again.
This behavior underscores the functionality of Access Control Lists in tracking and controlling the initial phase of tunnel creation, ensuring that subsequent data flow within the established tunnel does not affect the hit counts.

Key Takeaways:

ACL Counter Behavior: The ACL counters register increments exclusively on the initiator side during the tunnel initiation process. This indicates that the counters are designed to monitor the initial traffic that triggers the tunnel establishment.

Static Counters Post-Establishment: Once the tunnel is active and established, the ACL counters remain unchanged. They do not reflect any further activity unless the tunnel is reset and needs to be re-initiated, underscoring the focus on initial traffic events.

Traffic Initiation Specificity: The ACL hit counts are specific to the peer initiating the tunnel. This specificity ensures precise tracking of which side is responsible for initiating the VPN connection, allowing for accurate monitoring and control.