BGP Support for TCP Authentication Option

This document describes how to configure Message Digest5 (MD5) authentication on a Transmission Control Protocol (TCP) connection between two BGP peers.

BGP Support for TCP AO Overview

On a secure control plane, BGP uses Message Digest 5 (MD5) algorithm as the authentication mechanism. It uses the TCP API to configure the keychain on a TCP connection. When authentication is enabled, any Transmission Control Protocol (TCP) segments belonging to BGP are exchanged between peers, verified and then accepted only if authentication is successful. BGP application use the TCP API to configure the keychain on a TCP connection. It owns the configuration to associate a TCP-AO keychain name with a neighbor, a peer-group, or a peer-session template.

You can validate the authentication configuration per neighbor/peer-group/peer-session template. Authentication Option is supported for BGP dynamic neighbor, BGP Non-stop forwarding (NSF) and Non-stop routing (NSR). Routing protocols support a different set of cryptographic algorithms, however, BGP supports only MD5. For example, if BGP is configured with the TCP MD5 key (md5-key), it will not allow to configure TCP-AO and vice versa.There are two options to configure BGP:

- **include-tcp-options** - option to specify if the TCP option headers (other than TCP AO option) will be included while computing the MAC digest of the packets.
- **accept-ao-mismatch-connections** - option to accept the connection as non-TCP AO connection when receives a connection from peer without TCP AO option. Similarly, if the connection is initiated from one side, the peer acknowledges with TCP AO, it accepts the ACK and continues the connection.

Restrictions

- Configuring and deconfiguring TCP AO for a certain neighbor or peer-group or peer-session causes existing established BGP session(s) to flap.
- Do not change the configuration of an existing TCP key chain because existing BGP sessions may break.

- TCP OA must be configured between peers with compatible versions, either both running 17.6.2 or later, or both running releases earlier than 17.6.2.
- TCP AO picks up the most valid key under the key chain. The most valid key is the one which has the longest send lifetime. If there are two keys with the same send lifetime, the first best key is selected.
- In a configuration, where one of the devices is configured with the TCP MD5 option and the other with the TCP-AO option not supported, BGP session is not established between the devices until you correct the configuration.
- After a session is established using a specific key chain, if you modify the key chain, the session ends, and an attempt is made to renegotiate the session based on the modified key chain.

Interoperability between Cisco IOS XE and IOS XR

- The cryptographic algorithm specified by the **aes-128-cmac** keyword in IOS XE is functionally equivalent to the AES-128-CMAC-96 algorithm in IOS XR. Both platforms implement the same 96-bit truncated AES-based HMAC algorithm as described in RFC 4494.
- Consequently, an IOS XE BGP peer configured with **aes-128-cmac** can successfully establish a TCP-AO session with an IOS XR peer configured with AES-128-CMAC-96.

Table 1: IOS XE and IOS XR Algorithm Mapping

Platform	Command Keyword	Algorithm (RFC 4494)
IOS XE	aes-128-cmac	AES-128-CMAC-96 (truncated)
IOS XR	ES-128-CMAC-96	AES-128-CMAC-96 (truncated)

How to Configure BGP Using TCP AO

The BGP application must be configured on both the devices. To establish a peer connection with TCP-AO, you must configure the following:

Configuring TCP Key Chain and Keys

Before you begin

TCP-AO key chain and keys must be configured on both the peers communicating through a TCP connection.

- Ensure that the key-string, send-lifetimes, and ids of keys match on both peers.
- Ensure that the send-id on a router matches the receiver-id on the peer router. Also, use the same ID for the parameters.
- The send-id and recv-id of a key cannot be reused for another key in the same key chain.
- Do not modify a key that is in use. Disassociate the key from the TCP connection before modifying the key.

- The **include-tcp-options** and **accept-ao-mismatch** commands are not supported when configured under a key chain for BGP. To enable these options for a BGP neighbor or peer group, configure them directly using the **neighbor <address | peer-group> ao <key-chain> [include-tcp-options] [accept-ao-mismatch]** command in BGP configuration mode.

Procedure

Step 1 enable

Example:

```
Router# enable
```

Enables privileged EXEC mode. Enter your password if prompted.

Step 2 configure terminal

Example:

```
Router# configure terminal
```

Enters global configuration mode.

Step 3 key chain *key-chain-name* tcp

Example:

```
Router(config)# key chain kc1 tcp
```

Creates a TCP-AO key chain with the specified name and enters the TCP-AO key chain configuration mode.

The key chain name can have a maximum of 256 characters.

Step 4 key *key-id*

Example:

```
Router(config-keychain-tcp)# key 10
```

Creates a key with the specified key-id and enters the the TCP-AO key chain key configuration mode.

The key-id must be in the range 0 to 2147483647.

Step 5 send-id *send-identifier*

Example:

```
Router(config-keychain-tcp-key)# send-id 218
```

Specifies the send identifier for the key.

The send-identifier must be in the range 0 to 255.

Step 6 recv-id *receive-identifier*

Example:

```
Router(config-keychain-tcp-key)# recv-id 218
```

Specifies the receive identifier for the key.

The receive-identifier must be in the range 0 to 255.

Step 7 cryptographic-algorithm {aes-128-cmac | hmac-sha-1 | hmac-sha-256}

Example:

```
Router(config-keychain-tcp-key)# cryptographic-algorithm hmac-sha-1
```

Specifies the algorithm to be used to compute MACs for TCP segments.

Step 8 include-tcp-options

Example:

```
Router(config-keychain-tcp-key)# include-tcp-options
```

(Optional)

This flag indicates whether TCP options other than TCP-AO must be used to calculate MACs.

With the flag enabled, the content of all options, in the order present, is included in the MAC and TCP-AO's MAC field is filled with zeroes.

When the flag is disabled, all options other than TCP-AO are excluded from MAC calculations.

This flag is disabled by default.

Step 9 send-lifetime [local] *start-time duration*

Example:

```
Router(config-keychain-tcp-key)# send-lifetime local 12:00:00 28 Feb 2018 duration 20
```

Specifies the time for which the key is valid to be used for TCP-AO authentication.

Use the local keyword to specify the start-time in the local time zone. By default, the start-time corresponds to UTC time.

Step 10 accept-lifetime [local] *start-time duration*

Example:

```
Router(config-keychain-tcp-key)# accept-lifetime local 12:00:00 28 Feb 2018 duration 20
```

Specifies the time for which the key is valid to be used for TCP-AO authentication.

Use the local keyword to specify the start-time in the local time zone. By default, the start-time corresponds to UTC time.

Step 11 key-string *master-key*

Example:

```
Router(config-keychain-tcp-key)# key-string master-key
```

Specifies the master-key for deriving traffic keys.

The master-key must have 32 or 64 hexadecimal digits.

The master-keys must be identical on both the peers. If the master-keys do not match, authentication fails and segments may be rejected by the receiver.

Step 12 accept-ao-mismatch

Example:

```
Router(config-keychain-tcp-key)# accept-ao-mismatch
```

(Optional) This flag indicates whether the receiver should accept segments for which the MAC in the incoming TCP AO does not match the MAC generated on the receiver.

Step 13 end

Example:

```
Router(config-keychain-tcp-key)# end
```

Exits TCP-AO key chain key configuration mode and returns to privileged EXEC mode.

Example

A simple key chain configuration show on 2 end points A and B of the TCP AO enabled connection:

```
R1:

key chain kc1 tcp
key 7890
  send-id 215
  recv-id 215
  key-string klomn
  cryptographic-algorithm hmac-sha-1
 include-tcp-options

R2:

key chain kc1 tcp
key 7890
  send-id 215
  recv-id 215
  key-string klomn
  cryptographic-algorithm hmac-sha-1
  include-tcp-options
```

Configuring BGP Peer- group and Peer-session

BGP neighbor configuration

To configure a BGP neighbor using TCP AO:

```
Router(config)# router bgp <own-AS>
Router(config-router)# neighbor <peer-IP-address|peer-IPv6-address> ao <keychain-name>
[include-tcp-options] [accept-ao-mismatch-connections]
```

You can also configure BGP dynamic neighbor using the above command. Use the no form of the commands to deconfigure BGP neighbor.

BGP peer-group configuration

To configure a BGP peer-group using TCP AO:

```
Router(config-router)# neighbor <peer-group-name> ao <keychain-name>
[include-tcp-options] [accept-ao-mismatch-connections]
```

You can also configure BGP dynamic neighbor using the above command. Use the no form of the commands to deconfigure BGP peer-group.

BGP peer-session configuration

To configure a BGP peer-session template using TCP AO:

```
Router(config-router)# template peer-session <session-name>
Router(config-router-stmp)# ao <keychain-name> [include-tcp-options]
[accept-ao-mismatch-connections]
```

Use the no form of the command to deconfigure BGP peer-session.

Verifying TCP-AO Key Chain and Key Configuration

Use the **show key chain key-chain-name** command in the privileged exec mode to display information about a TCP-AO key chain and keys.

```
Router# show key chain key-chain-name
```

```
Router1# show key chain kc2
Key-chain kc2:
TCP key chain
key 7893 -- text "abcde"
cryptographic-algorithm: hmac-sha-1
accept lifetime (12:32:00 IST Nov 9 2018) - (10:30:00 IST Dec 30 2019) [valid now]
send lifetime (13:05:00 IST Jan 12 2019) - (10:31:00 IST Dec 30 2019) [valid now]
send-id - 218
recv-id - 218
include-tcp-options
MKT ready - true
MKT preferred - true
MKT in-use - true
MKT id - 7893
MKT send-id - 218
MKT recv-id - 218
MKT alive (send) - true
MKT alive (recv) - true
MKT include TCP options - true
MKT accept AO mismatch - false
TCB - 0x7FBD68361838
curr key - 7893
next key - 7893
```

Verifying TCP-AO Key Chain Information in the TCB

Use the **show tcp tcb address-of-tcb** command in the priviledged exec mode to display information about TCP-AO in the Transmission Control Block. Obtain address-of-tcb (the hexadecimal address of the TCB) from the output of the **show key chain key-chain-name** command.

```
Router# show tcp tcb address-of-tcb
```

```
Router1# sh tcp tcb 7FBD68361838
Connection state is ESTAB, I/O status: 1, unread input bytes: 0
Connection is ECN Disabled, Mininum incoming TTL 0, Outgoing TTL 255
Local host: 1.0.2.1, Local port: 40125
Foreign host: 1.0.2.2, Foreign port: 5555
Connection tableid (VRF): 0
Maximum output segment queue size: 50

Enqueued packets for retransmit: 0, input: 0  mis-ordered: 0 (0 bytes)
```

```
Event Timers (current time is 0x2818B07):
Timer          Starts    Wakeups            Next
Retrans             1          0             0x0
TimeWait            0          0             0x0
AckHold             1          0             0x0
SendWnd             0          0             0x0
KeepAlive        6651          0       0x281AC36
GiveUp              0          0             0x0
PmtuAger            0          0             0x0
DeadWait            0          0             0x0
Linger              0          0             0x0
ProcessQ            0          0             0x0

iss: 3307331702  snduna: 3307331703  sndnxt: 3307331703
irs:  725047078  rcvnxt:  725047079

sndwnd:   4128  scale:      0  maxrcvwnd:   4128
rcvwnd:   4128  scale:      0  delrcvwnd:      0

SRTT: 125 ms, RTTO: 2625 ms, RTV: 2500 ms, KRTT: 0 ms
minRTT: 15 ms, maxRTT: 1000 ms, ACK hold: 200 ms
uptime: 40996359 ms, Sent idletime: 6505 ms, Receive idletime: 6505 ms
Status Flags: active open
Option Flags: keepalive running, nagle, Retrans timeout
IP Precedence value : 0

TCP AO Key chain: kc2

TCP AO Current Key:
 Id: 7893, Send-Id: 218, Recv-Id: 218
 Include TCP Options: Yes*
 Accept AO Mismatch: No*

TCP AO Next Key:
 Id: 7893, Send-Id: 218, Recv-Id: 218
 Include TCP Options: Yes*
 Accept AO Mismatch: No*

Datagrams (max data segment is 1460 bytes):
Rcvd: 4372 (out of order: 0), with data: 0, total data bytes: 0
Sent: 4372 (retransmit: 0, fastretransmit: 0, partialack: 0, Second Congestion: 0), with
data: 0, total data bytes: 0

 Packets received in fast path: 0, fast processed: 0, slow path: 0
 fast lock acquisition failures: 0, slow path: 0
TCP Semaphore      0x7FBD6801B2E0  FREE

* - Derived from Key
```

Example: Verifying BGP Configuration

Use the **show ip bgp peer-group peer-group-name** and **show ip bgp template peer-session peer-session-name** commands in the privileged exec mode to display information about BGP configuration on peer groups and peer sessions.

```
Router# show ip bgp peer-group ABC

BGP peer-group is ABC,  remote AS 100
  BGP version 4
```

```
  …
  AO keychain <keychain-name> include-tcp-options accept-ao-mismatch-connections
  …


Router# show ip bgp template peer-session ABC

Template:ABC,  index:1
Local policies:<hex-value>, Inherited polices:<hex-value>
Locally configured session commands:
  …
  AO keychain <keychain-name> include-tcp-options accept-ao-mismatch-connections
  …
Inherited session commands:
  …
  AO keychain <keychain-name> include-tcp-options accept-ao-mismatch-connections
```