T Commands

This chapter describes the Cisco NX-OS Border Gateway Protocol (BGP) commands that begin with T.

template (BGP)

To create a peer template and enter a peer template configuration mode, use the **template** command. To remove a peer template, use the **no** form of this command.

template {**peer** *name* | **peer-policy** *name* | **peer-session** *name*}

no template {**peer** *name* | **peer-policy** *name* | **peer-session** *name*}

Syntax Description

peer *name*	Specifies the name of the neighbor template.
peer-policy *name*	Specifies the name of the peer-policy template.
peer-session *name*	Specifies the name of the peer-session template.

Command Default

This command has no default settings.

Command Modes

Neighbor address-family configuration mode
Router bgp configuration mode

Command History

Release	Modification
6.0(2)N1(1)	This command was introduced.

Usage Guidelines

The **template** command allows you to enable a set of predefined attributes that a neighbor inherits.

Note A Border Gateway Protocol (BGP) neighbor cannot be configured to work with both peer groups and peer templates. A BGP neighbor can be configured to belong to a peer group or to inherit policies from peer templates only.

Peer templates support only general policy commands. BGP policy configuration commands that are configured only for specific address families or NLRI configuration modes are configured with peer templates.

When you enter the peer-policy template configuration mode, the following commands are available:

- **suppress-inactive**—Advertises the active routes to the peer only. See the **suppress-inactive** command for additional information.
- **exit**—Exits current configuration mode.
- **filter-list** *name* {**in** | **out**}—Creates the AS-PATH filter-list on the inbound and the outbound BGP routes. To remove the entry, use the **no** form of this command.
  - **in**—Applies the access list to incoming routes.
  - **out**—Applies the access list to outgoing routes.

- **inherit peer-policy** *policy-name seq-num*—Configures a peer-policy template to inherit the configuration from another peer-policy template. To remove an inherited statement from a peer-policy template, use the **no** form of this command. Range: 1 to 65535. Default: No inherit statements are configured.

  The sequence number specifies the order in which the peer policy template is evaluated. Like a route-map sequence number, the lowest sequence number is evaluated first. Peer policy templates support inheritance and a peer can directly and indirectly inherit up to seven peer policy templates. Inherited peer policy templates are configured with sequence numbers like route maps. An inherited peer policy template, like a route map, is evaluated starting with the inherit statement with the lowest sequence number. However, peer policy templates do not fall through. Every sequence is evaluated. If a BGP policy command is reapplied with a different value, it overwrites any previous value from a lower sequence number.

Note A Border Gateway Protocol (BGP) routing process cannot be configured to be a member of a peer group and to use peer templates for group configurations. You must use one method or the other. We recommend peer templates because they provide improved performance and scalability.

- **maximum-prefix** *max*—Specifies the maximum number of prefixes from this neighbor. Range: 1 to 300000. Default: This command is disabled by default. Peering sessions are disabled when the maximum number of prefixes is exceeded. See the **maximum-prefix** command for additional information.
- **next-hop-self**—Configures the router as the next hop for a Border Gateway Protocol (BGP) neighbor or peer group. To disable this feature, use the **no** form of this command. Default: Disabled.
- **next-hop-third-party**—Computes a third-party next hop if possible.
- **no**—Negates a command or sets its defaults.
- **prefix-list** *name* {**in** | **out**}—Specifies the route type to apply the prefix list. To remove the entry, use the **no** form of this command.
  - **in**—Applies the prefix list to incoming routes.
  - **out**—Applies the prefix list to outgoing routes.
- **route-map** *name* {**in** | **out**}—Specifies the route map name to apply the route type to apply to the neighbor.
  - **in**—Applies the route map to incoming routes.
  - **out**—Applies the route map to outgoing routes.
- **route-reflector-client**—Configures the router as a BGP route reflector and configures the specified neighbor as its client. To indicate that the neighbor is not a client, use the **no** form of this command. Default: There is no route reflector in the autonomous system.

  By default, all internal BGP (iBGP) speakers in an autonomous system must be fully meshed, and neighbors do not readvertise iBGP learned routes to neighbors, which prevents a routing information loop. When all the clients are disabled, the local router is no longer a route reflector.

  If you use route reflectors, all iBGP speakers need not be fully meshed. In the route reflector model, an Interior BGP peer is configured to be a route reflector responsible for passing iBGP learned routes to iBGP neighbors. This scheme eliminates the need for each router to talk to every other router.

  All the neighbors configured with this command are members of the client group and the remaining iBGP peers are members of the nonclient group for the local route reflector.

- **send-community**—Specifies that a community attribute be sent to a BGP neighbor. To remove the entry, use the **no** form of this command.
- **soft-reconfiguration**—Configures the Cisco NX-OS software to start storing updates. To not store received updates, use the **no** form of this command. Default: Disabled. Entering this command starts the storage of updates, which is required to do inbound soft reconfiguration. Outbound BGP soft reconfiguration does not require inbound soft reconfiguration to be enabled.

  To use soft reconfiguration, or a soft reset, without preconfiguration, both BGP peers must support the soft route refresh capability, which is advertised in the open message sent when the peers establish a TCP session. Clearing the BGP session using the **soft-reconfiguration** command has a negative effect on network operations and should only be used as a last resort.

  To determine whether a BGP router supports this capability, use the **show ip bgp neighbors** command. If a router supports the route refresh capability, the following message appears:

  "Received route refresh capability from peer."

  If you specify a BGP peer group by using the peer-group-name argument, all the members of the peer group inherit the characteristic configured with this command.

When you enter the peer-session template configuration mode, the following commands are available:

- **description** *description*—Configures a description to be displayed by the local or a peer router. You can enter up to 80 characters including spaces.
- **disable-connected-check**—Disables connection verification for eBGP peers no more than one hop away when the eBGP peer is configured with a loopback interface.
- **ebgp-multihop**—Accepts and attempts BGP connections to external peers that reside on networks that are not directly connected.

Note You should enter this command under the guidance of Cisco technical support staff only.

- **exit**—Exits current configuration mode.
- **inherit peer-session** *session-name*—Configures a peer-session template. To inherit the configuration from another peer-session template, use the **peer-session** keywords. To remove an inherit statement from a peer-session template, use the **no** form of this command.
- **local-as**—Allows you to customize the autonomous system number for eBGP peer groupings.
- **neighbor inherit peer-session**—Configures a router to send a peer session template to a neighbor so that the neighbor can inherit the configuration.
- **neighbor translate-update**—Upgrades a router running BGP in the NLRI format to support multiprotocol BGP.
- **password**—Enables MD5 authentication on a TCP connection between two BGP peers. The following configuration tools are available:
  - **0** *password*—Specifies an unencrypted neighbor password.
  - **3** *password*—Specifies an 3DES encrypted neighbor password
  - *password*—Specifies an unencrypted (cleartext) neighbor password
- **remote-private-as**—Removes the private AS number from outbound updates.
- **show ip bgp template peer-policy**—Displays the locally configured peer policy templates.
- **show ip bgp template peer-session**—Displays the locally configured peer session templates.
- **shutdown**—Disables a neighbor or peer group.

- **timers** *keepalive-time*—Configures keepalive and hold timers in seconds. Range: 0 to 3600. Default: 60.
- **update-source** {**ethernet** *mod/port* | **loopback** *virtual-interface* | **port-channel** *number*[**.***sub-interface*]}—Specifies the source of the BGP session and updates. Range: *virtual-interface* is 0 to 1023; *number* is 0 to 4096; (optional) **.***sub-interface* is 1 to 4093.

General session commands can be configured once in a peer-session template and then applied to many neighbors through the direct application of a peer-session template or through indirect inheritance from a peer-session template. The configuration of peer-session templates simplify the configuration of general session commands that are commonly applied to all neighbors within an autonomous system.

This command requires the LAN Enterprise Services license.

Examples

This example shows how to create a peer-session template named CORE1. This example inherits the configuration of the peer-session template named INTERNAL-BGP.

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router)# template peer-session CORE1
switch(config-router-stmp)#
```

This example shows how to create and configure a peer-policy template named CUSTOMER-A:

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router)# template peer-policy CUSTOMER-A
switch(config-router-ptmp)# exit
switch(config-router)# route-map SET-COMMUNITY in
switch(config-router)# filter-list 20 in
switch(config-router)# inherit peer-policy PRIMARY-IN 20
switch(config-router)# inherit peer-policy GLOBAL 10
switch(config-router)# exit-peer-policy
switch(config-router)#
```

This example shows that the maximum prefixes that are accepted from the 192.168.1.1 neighbor is set to 1000:

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router) network 192.168.0.0
switch(config-router)# maximum-prefix 1000
switch(config-router)#
```

This example shows that the maximum number of prefixes that are accepted from the 192.168.2.2 neighbor is set to 5000. The router is also configured to display warning messages when 50 percent of the maximum-prefix limit (2500 prefixes) has been reached.

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router) network 192.168.0.0
switch(config-router)# maximum-prefix 5000 50
switch(config-router)#
```

This example shows that the maximum number of prefixes that are accepted from the 192.168.3.3 neighbor is set to 2000. The router is also configured to reestablish a disabled peering session after 30 minutes.

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router) network 192.168.0.0
```

```
switch(config-router)# neighbor 192.168.3.3 maximum-prefix 2000 restart 30
switch(config-router)#
```

This example shows that the warning messages are displayed when the maximum-prefix limit (500) for the 192.168.4.4 neighbor is exceeded:

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router)# network 192.168.0.0
switch(config-router)# maximum-prefix 500 warning-only
switch(config-router)#
```

This example forces all updates destined for 10.108.1.1 to advertise this router as the next hop:

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router)# next-hop-self
switch(config-router)#
```

This example shows that the router belongs to autonomous system 109 and is configured to send the communities attribute to its neighbor at IP address 172.16.70.23:

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router)# send-community
switch(config-router)#
```

This example shows that the router belongs to autonomous system 109 and is configured to send the communities attribute to its neighbor at IP address 172.16.70.23:

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router)# address-family ipv4 multicast
switch(config-router-af)# send-community
switch(config-router-af)#
```

This example enables inbound soft reconfiguration for the neighbor 10.108.1.1. All the updates received from this neighbor are stored unmodified, regardless of the inbound policy. When inbound soft reconfiguration is done later, the stored information is used to generate a new set of inbound updates.

```
switch# configure terminal
switch(config)# router bgp 64496
switch(config-router)# soft-reconfiguration inbound
switch(config-router)#
```

Related Commands

Command	Description
address-family	Enters the address family mode for the Border Gateway Protocol (BGP).
password (BGP)	Configures a MD5 password for two BGP peers.
router bgp	Enters the assign an autonomous system (AS) number to a router and enters the router BGP configuration mode.