Enterprise SONiC Distribution by Dell Technologies

Management Framework CLI Reference Guide Release 3.0

April 2020
Rev. A01

DELL EMC

Notes, cautions, and warnings

NOTE: A NOTE indicates important information that helps you make better use of your product.

CAUTION: A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

WARNING: A WARNING indicates a potential for property damage, personal injury, or death.

Contents

1

Revision history

Table 1. Revision history

Release	Revision	Description
3.0.1	A01 (April 2020)	Updated CLI command syntaxes and examples
3.0.0	A00 (March 2020)	Initial release

Additional documentation

Table 2. Additional documentation

Document	Description
Enterprise SONiC Distribution by Dell Technologies, User Guide Release 3.0	User guide
Enterprise SONiC Distribution by Dell Technologies, Quick Start Guide Release 3.0	Installation and initial configuration
Enterprise SONiC Distribution by Dell Technologies, SONiC CLI Reference Guide Release 3.0	SONiC CLI command syntaxes and examples
Enterprise SONiC Distribution by Dell Technologies, Release Notes Release 3.0	Requirements, known issues

2

A commands

Topics:

aaa authentication failthrough

Enables or disables the authentication, authorization, and accounting (AAA) failthrough option.

Command `aaa authentication failthrough {enable | disable}`

Options
- `enable` — Allows AAA to process with local authentication if remote authentication fails
- `disable` — Disallows AAA to proceed further if remote authentication fails

Command mode CONFIGURATION

Usage Use this command if you have configured more than one TACACS+ or RADIUS server, and have also enabled TACACS+ or RADIUS authentication. When authentication request to the first server fails, this configuration continues the request to the next server. When this configuration is enabled, the authentication process continues through all servers configured. When this is disabled and if the authentication request fails on the first server, the authentication process stops and the login is disallowed. The `no` version of this command removes the configuration.

Example

```
sonic(config)# aaa authentication failthrough enable
```

Releases 3.0 or later

aaa authentication login-method

Configures AAA to use the local, remote RADIUS, or remote TACACS+ databases for authentication.

Command

```
aaa authentication login-method {{[local] [tacacs+] [radius]} | {[tacacs+] [local]} | {[radius] [local]}}
```

Options

- *tacacs+* — (Optional) Enables remote authentication based on TACACS+
- *radius* — (Optional) Enables remote authentication based on RADIUS
- *local* — (Optional) Disables remote authentication and uses local authentication (default)

Command mode

CONFIGURATION

Usage

Use this command to configure the AAA preferred login method to authenticate users. A SONiC switch uses a list of authentication methods to define types of authentication, and the sequence in which they apply. By default, AAA uses only `local` to authenticate users with a local user database. The authentication method list runs in the order configured. You can configure one or more TACACS+ or RADIUS remote server. You must configure the server settings correctly to ensure connectivity is available through the Management interface. If you configure remote authentication using a server, all user logins are authenticated by the TACACS+ or RADIUS server. If the authentication fails, AAA checks the fail-through configuration and authenticates the user based on the local database if fail-through is enabled. The `no` version of this command removes all configured methods and defaults using local authentication.

Examples

```
sonic(config)# aaa authentication login-method tacacs+ local
```

```
sonic(config)# aaa authentication login-method local tacacs+
```

Releases

3.0 or later

activate

Enables or activates a specific address-family for a BGP neighbor or peer-group.

Command

```
activate
```

Options

None

Command modes

- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage

Use this command to enable route advertisement for a specified address-family with BGP neighbor or peer-group members. The `no` version of this command disables the address-family exchange.

Examples

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 20.20.20.2
sonic(config-router-bgp-neighbor)# remote-as 300
sonic(config-router-bgp-neighbor)# address-family ipv4 unicast
sonic(config-router-bgp-neighbor-af)# activate
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 20.20.20.2
sonic(config-router-bgp-neighbor)# remote-as 300
sonic(config-router-bgp-neighbor)# address-family l2vpn evpn
sonic(config-router-bgp-neighbor-af)# activate
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Ext
sonic(config-router-bgp-pg)# remote-as 300
```

```
sonic(config-router-bgp-pg)# address-family ipv4 unicast
sonic(config-router-bgp-pg-af)# activate
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Ext
sonic(config-router-bgp-pg)# remote-as 300
sonic(config-router-bgp-pg)# address-family l2vpn evpn
sonic(config-router-bgp-pg-af)# activate
```

Releases 3.0 or later

addpath-tx-all-paths

Enables BGP to advertise all paths to neighbors in a peer-group.

Command `addpath-tx-all-paths`

Options None

Command modes
- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to enable advertisement of all BGP routes in a BGP peer-group. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 20.20.20.2
sonic(config-router-bgp-neighbor)# remote-as 300
sonic(config-router-bgp-neighbor)# address-family ipv4 unicast
sonic(config-router-bgp-neighbor-af)# addpath-tx-all-paths
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Ext
sonic(config-router-bgp-pg)# address-family ipv4 unicast
sonic(config-router-bgp-pg-af)# addpath-tx-all-paths
```

Releases 3.0 or later

addpath-tx-bestpath-per-AS

Enables BGP to advertise the best-path to neighbors in a peer-group.

Command `addpath-tx-bestpath-per-AS`

Options None

Command mode PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to enable advertisement of only the best-path to each AS in a BGP peer-group. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Ext
sonic(config-router-bgp-pg)# address-family ipv4 unicast
sonic(config-router-bgp-pg-af)# addpath-tx-bestpath-per-AS
```

Releases 3.0 or later

address-family ipv4

Enters into IPv4 unicast address-family configuration mode.

Command `address-family ipv4 unicast`

Options None

Command mode
- ROUTER-BGP
- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to enter IPv4 address-family configuration mode to configure BGP neighbors and peer-groups. This command applies to all IPv4 peers belonging to the template or neighbors only. The `no` version of this command deletes the subsequent address-family configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# address-family ipv4 unicast
sonic(conf-router-bgp-af)#
```

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)#
```

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)#
```

Releases 3.0 or later

address-family ipv6

Enters IPv6 unicast address-family configuration mode.

Command `address-family ipv6 unicast`

Options None

Command modes
- ROUTER-BGP
- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to enter IPv6 unicast address-family configuration mode to configure BGP neighbors and peer-groups. This command applies to all IPv6 peers belonging to the template or neighbors only. The `no` version of this command deletes the subsequent address-family configuration.

Examples

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# address-family ipv6 unicast
sonic(conf-router-bgp-af)#
```

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# address-family ipv6 unicast
sonic(conf-router-bgp-neighbor-af)#
```

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family ipv6 unicast
sonic(conf-router-bgp-pg-af)#
```

Releases 3.0 or later

address-family l2vpn

Enters L2VPN EVPN address-family configuration mode.

Command `address-family l2vpn evpn`

Options None

Command modes
- ROUTER-BGP
- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to enter L2VPN EVPN address-family configuration mode to configure BGP neighbors and peer-groups. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)#
```

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# address-family l2vpn evpn
sonic(conf-router-bgp-neighbor-af)#
```

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family l2vpn evpn
sonic(conf-router-bgp-pg-af)#
```

Releases 3.0 or later

advertise ipv4 unicast

Enables tenant VRFs to announce IPv4 prefixes as EVPN type-5 routes.

Command `advertise ipv4 unicast`

Options None

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to enable tenant VRFs to announce IPv4 prefixes as EVPN type-5 routes. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100 vrf Vrf1
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# advertise ipv4 unicast
```

Releases 3.0 or later

advertise ipv6 unicast

Enables tenant VRFs to announce IPv6 prefixes as EVPN type-5 routes.

Command `advertise ipv6 unicast`

Options None

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to enable tenant VRFs to announce IPv6 prefixes as EVPN type-5 routes. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100 vrf Vrf1
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# advertise ipv6 unicast
```

Releases 3.0 or later

advertise-all-vni

Enables BGP control plane for all locally-configured VNIs.

Command `advertise-all-vni`

Options None

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to enable BGP control plane for all locally-configured VNIs. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# advertise-all-vni
```

Releases 3.0 or later

advertise-default-gw

Enables gateway advertisement.

Command `advertise-default-gw`

Options None

Command mode
- ADDRESS-FAMILY-VNI
- BGP-ADDRESS-FAMILY

Usage Use this command to enable gateway advertisements for a specific VNI, or for gateways VTEPs to advertise their IP/MAC addresses. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# vni 100
sonic(conf-router-bgp-af-vni)# advertise-default-gw
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# advertise-default-gw
```

Releases 3.0 or later

advertisement-interval

Sets the minimum time interval between sending BGP routing updates to a neighbor, or neighbors in a peer-group.

Command `advertisement-interval` *`seconds`*

Options *`seconds`* — Time value (1 to 600 seconds; default eBGP 30, iBGP 5)

Command modes
- BGP-NEIGHBOR

- BGP-PEER-GROUP

Usage Use this command to configure the time in seconds between sending BGP route updates to neighbors, or neighbors in a peer-group. The `no` version of this command resets the advertisement interval value to the default.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# advertisement-interval 10
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# advertisement-interval 10
```

Releases 3.0 or later

aggregate-address

Configures an aggregate address and enables aggregation of routes that falls in the aggregate address subnet.

Command `aggregate-address` *`prefix`* `{[as-set] [summary-only] {[route-map]` *`rtemap_name`*`}}`

Options

- *`prefix`* — IP address prefix in A.B.C.D/mask format
- `as-set` — (Optional) Advertises the aggregate routes contained in the summary aggregate-prefix entry
- `summary-only` — (Optional) Suppresses the advertisement of specific routes in the prefix range to neighbors
- `route-map` *`rtemap_name`* — (Optional) Route-map name

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to configure an aggregate address entry in the BGP routing table. Aggregate entries reduce the size of the routing table. An aggregate prefix combines contiguous networks into one summarized set of IP addresses. This command enables you to turn on aggregation of BGP routes. The `summary-only` option filters out all the aggregates routes and only the aggregate address will be advertised by BGP. The `as-set` option makes sure that AS path of individual aggregated routes are also included in the resulting aggregate route. The `route-map` option provides a finer control over the route's attributes. The `no` version of this command disables the aggregate-address configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family ipv4 unicast
sonic(conf-router-bgp-af)# aggregate-address 17.35.0.0/16
```

Releases 3.0 or later

aging-interval

Configures an aging-interval for configured drop-monitor flows.

Command `aging-interval` *`seconds`*

Options *`seconds`* — Interval wait time

Draft comment: Need interval range and default values

Command mode DROP-MONITOR

Usage This command is used to configure an aging-interval for drop-monitor flows configured in the system. The aging interval determines how long the system waits before it decides that drops have ceased on a flow. The `no` version of this command removes the configuration.

Example

```
sonic(config)# drop-monitor
sonic(conf-drop-monitor)# aging-interval 6
```

Releases 3.0 or later

allowas-in

Enables the BGP speaker to accept a route with the local AS number in updates received from a peer for the specified number of times.

Command `allowas-in {[number] | [origin]}`

Options *number* — (Optional) Number of occurrences for a local AS number (1 to 10)

Command modes
- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to configure the number of times the local AS number can appear in the BGP AS_PATH path attribute before the switch rejects the route. This command enables the BGP speaker to accept a route with the local AS number in updates received from a peer for the specified number of times. Accepting own AS in an AS path usually results in an AS loop. You can add the AS number to influence the BGP route selection process. This command enables you to control when a route with as-path containing own AS number should be accepted or not. The command also provides flexibility in terms of maximum number of occurrences of AS number in an AS path. The `no` version of this command moves the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# allowas-in 5
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family l2vpn evpn
sonic(conf-router-bgp-neighbor-af)# allowas-in 5
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# remote-as 300
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# allowas-in
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# remote-as 300
sonic(conf-router-bgp-pg)# address-family l2vpn evpn
sonic(conf-router-bgp-pg-af)# allowas-in
```

Releases 3.0 or later

always-compare-med

Instructs BGP to always compare MED attributes in the paths that are received from different neighbors.

Command `always-compare-med`

Options None

Command mode ROUTER-BGP

Usage Use this command to always compare the MED on routes, even when they are received from different neighbors. Setting this option makes the order of preference of routes more defined, and should eliminate MED induced oscillations. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# always-compare-med
```

Releases 3.0 or later

as-override

Instructs BGP to override AS numbers in outbound updates if as-path equals remote-as.

Command `as-override`

Options None

Command mode
- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to override the outbound route updates to an AS path that includes the remote AS configured with the BGP neighbor or peer-group `remote-as` command. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# as-override
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# as-override
```

Releases 3.0 or later

attribute-unchanged

Instructs BGP to propagate route attributes unchanged to this neighbor, or neighbors in a peer-group.

Command `attribute-unchanged [as-path] [med] [next-hop]`

Options
- `as-path` — (Optional) Use the AS path attribute to propagate unchanged
- `med` — (Optional) Use the MED attribute to propagate unchanged
- `next-hop` — (Optional) Use the next-hop attribute to propagate unchanged

Command modes
- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to propagate BGP route attributes unchanged to this neighbor, or neighbors in a peer-group. You can control which attributes are propagated unchanged. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# attribute-unchanged as-path next-hop
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family l2vpn evpn
sonic(conf-router-bgp-neighbor-af)# attribute-unchanged as-path next-hop
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
```

```
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# attribute-unchanged as-path next-hop
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family l2vpn evpn
sonic(conf-router-bgp-pg-af)# attribute-unchanged as-path next-hop
```

Releases 3.0 or later

autoneg

Configures auto-negotiation on the Mangement interface.

Command `autoneg {true | false}`

Options

- `true` — Enables auto-negotiation on an interface
- `false` — Disables auto-negotiation on an interface (default)

Command mode INTERFACE-MGMT

Usage Use this command to set the Management interface to auto-negotiate speed with a connected device. Both sides of the link must have auto-negotiation enabled or disabled for the link to come up. The `no` version of this command resets the interface speed auto-negotiation to the default (false).

Example

```
sonic(conf-if-Mgmt)# autoneg true
```

Releases 3.0 or later

autort

Enables automatic derivation of route-distinguisher and route-targets.

Command `autort rfc8365-compatible`

Options None

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to enable automatic derivation of route-distinguisher and route-targets. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# autort rfc8365-compatible
```

Releases 3.0 or later

3

B commands

Topics:

bestpath as-path confed

Instructs BGP to consider confederation path length in as-path length comparison during best-path selection process.

Command	`bestpath as-path confed`
Options	None
Command mode	ROUTER-BGP
Usage	Use this command to consider confederation set and sequence path length for best-path selection process. The `no` version of this command removes the configuration.
Example	`sonic(config)# router bgp 65300` `sonic(conf-router-bgp)# bestpath as-path confed`
Releases	3.0 or later

bestpath as-path ignore

Instructs BGP to ignore the as-path comparison during best-path calculations.

Command	`bestpath as-path ignore`
Options	None
Command mode	ROUTER-BGP
Usage	Use this command to ignore as-path comparison during best-path selection process. The `no` version of this command disables configuration.
Example	`sonic(config)# router bgp 65300` `sonic(conf-router-bgp)# bestpath as-path ignore`
Releases	3.0 or later

bestpath as-path multipath-relax

Specifies that BGP decision process should consider paths of equal AS_PATH length candidates for multi-path computation.

Command `bestpath as-path multipath-relax [as-set]`

Options `as-set` — (Optional) Generates AS set-path information

Command mode ROUTER-BGP

Usage Use this command to ignore as-path check for paths for the same prefix, making all the paths equal irrespective of their as-path. This command relaxes as-path comparison for multipath during the best-path selection process. The `no` version of this command disables configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# bestpath as-path multipath-relax
```

Releases 3.0 or later

bestpath compare-routerid

Influences best-path selection algorithm by comparing router-IDs for identical eBGP routes.

Command `bestpath compare-routerid`

Options None

Command mode ROUTER-BGP

Usage Use this command to ensure that when comparing routes where both are equal on most metrics, including local-pref, AS_PATH length, IGP cost, and MED that the tie is broken based on the router-ID. If this option is enabled, the already-selected check where already selected eBGP routes are preferred is skipped. If a route has an ORIGINATOR_ID attribute because it has been reflected, that ORIGINATOR_ID is used. The router-ID of the peer the route was received from will otherwise be used. The advantage is that the route-selection (at this point) is more deterministic. The disadvantage is that a few or even one lowest-ID router may attract all traffic to otherwise-equal paths because of this check. It may increase the possibility of MED or IGP oscillation, unless other measures were taken to avoid these. The exact behavior will be sensitive to the iBGP and reflection topology. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# bestpath compare-routerid
```

Releases 3.0 or later

bestpath med

Changes the best-path MED attributes during MED comparison for path selection.

Command `bestpath med {{missing-as-worst [confed]} | {confed [missing-as-worst]}}`

Options

- `confed` — Compare MED among BGP confederation paths
- `missing-as-worst` — Treat missing MED as the least preferred method

Command mode ROUTER-BGP

Usage Use this command to consider MED for confederation paths for best-path selection process. If MED is missing, it should be considered as worst MED. The `no` version of this command resets the MED comparison influence.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# bestpath med missing-as-worst confed
```

Releases 3.0 or later

bfd

Enables bidirectional forwarding detection (BFD) liveliness check for BGP neighbors, and neighbors in a peer-group.

Command `bfd [check-control-plane-failure]`

Options `check-control-plane-failure` — (Optional) Links the data plane status to the BGP control plane

Command modes
- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to enable BFD to detect forwarding-path failures in BGP routes. This command reduces BGP convergence time if there is a link failure. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# bfd
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# bfd
```

Releases 3.0 or later

bgp as-path-list

Creates a BGP AS path list.

Command `bgp as-path-list` *`as-path-list-name`* `{regex {`*`regex_id`* `{[any] | [all]}}}`

Options
- *`as-path-list-name`* — AS path list name
- *`regex_id`* — REGEX ID
- `any` — Allows matching of any entry in an AS path list
- `all` — Allows matching of all entries in an AS path list

Command mode CONFIGURATION

Usage Use this command to create filters for the AS paths in route advertisements using regular expressions as match criteria. You can apply an AS list filter to the inbound and outbound address families of routes that are advertised to, and received from BGP neighbors and BGP peer-groups using a filter-list. You can enter a regular expression of AS paths to provide flexible and powerful match support. This command also provides any and all options to allow matching all or any entry in an AS path-list. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# bgp as-path-list asp_private regex ^65000.*6510565109$
```

```
sonic(config)# bgp as-path-list asp_private regex 65107.*65200
```

Releases 3.0 or later

bgp community-list

Creates a standard BGP community-list.

Command `bgp community-list {{standard {`*`community_list_name`* `{{`*`aann`* `[local-AS] [no-advertise] [no-export] [no-peer] {[any] | [all]}} | {local-AS [`*`aa:nn`*`] [no-advertise] [no-export] [no-peer] {[any] | [all]}} | {no-advertise [`*`aa:nn`*`] [local-AS] [no-export] [no-peer] {[any] | [all]}} | {no-export [`*`aann`*`] [local-AS] [no-advertise] [no-peer] {[any] | [all]}} | {no-peer [`*`aa:nn`*`] [local-AS] [no-advertise] [no-export] {[any] | [all]}}}}}} | {expanded {`*`community_expanded_list_name`* `{`*`line`* `{[any] | [all]}}}}}}`

Options

- *community_list_name* — Name of the community list used to identify one or more groups of communities
- *aa:nn* — Community number in *aa:nn* format, where *aa* is the number that identifies the autonomous system and *nn* is a number that identifies the community within the autonomous system
- `local-AS` — BGP does not advertise this route to external peers
- `no-advertise` — BGP does not advertise this route to any internal or external peers
- `no-export` — BGP does not advertise this route outside a BGP confederation boundary
- `no-peer` — BGP does not advertise this route outside a BGP peer group
- `any` — BGP does not advertise any routes that do not match the filter
- `all` — BGP does not advertise all routes that do not match the filter
- *community_expanded_list_name* — Name of the expanded community list
- *line* — Expanded community-list line which matches any or all routes

Command mode CONFIGURATION

Usage Use this command to create a standard BGP community list. By default, the communities attribute is not sent in BGP route updates. You must configure the switch to provide the communities attribute in BGP routes. You can configure a list of BGP community values to use in permit/deny statements in route maps. You can then apply the route maps to a BGP neighbor or BGP peer-group address-family. This command provides options to create expanded or standard community lists and accepts community in AA:NN, IP:NN and well-known communities format. This command also provides any and all constructs to enable you to design community filters with clause match any or all. For expanded community, you can specify a regular expression of communities. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# bgp community-list standard CommList_RT 100:200
```

```
sonic(config)# bgp community-list standard CommList_RT no-export
```

```
sonic(config)# bgp community-list standard CommList_RT no-peer
```

```
sonic(config)# bgp community-list standard CommList_RT 65100:3456
```

Releases 3.0 or later

bgp extcommunity-list

Creates BGP extended-community list entries.

Command

```
bgp extcommunity-list {{standard {extcommunity_list_name {{rt {aa | ipaddrnn}
{[any] | [all]}} | {soo {aa | ipaddrnn} {[any] | [all]}}}}} | {expanded
{extcommunity_list_name {line {[any] | [all]}}}}}
```

Options

- `standard` *extcommunity_list_name* — Name of the extended-community list used to identify one or more groups of communities
- `rt` *aa* — Target route to match against in AA:NN format
- `any` — BGP does not advertise any routes that do not match the filter
- `all` — BGP does not advertise all routes that do not match the filter
- `soo` *aa* — Route origin to match against in AA:NN format
- *ipaddrnn* — IP address to match against in :NN format
- `expanded` *extcommunity_list_name* — Name of expanded community list
- *line* — Expanded community-list line which matches any or all routes

Command mode CONFIGURATION

Usage Use this command to create BGP extended-community list entries. This command provides options to create expanded or standard extended community list entries. For standard extended community, you can create `rt` or `soo` type of communities to accept communities in AA:NN or IP:NN formats. For expanded extended community, this command accepts a regular expression of communities, which is very flexible and powerful for matching

communities in routes. This command also provides options for matching all or any extended communities. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# bgp extcommunity-list standard ExtComm_AllowInt rt
19.32.56.167:65011 all
```

```
sonic(config)# bgp extcommunity-list standard ExtComm_AllowInt rt
31.67.182.214:3001 all
```

```
sonic(config)# bgp extcommunity-list standard ExtComm_AllowInt soo
4001:65010 all
```

```
sonic(config)# bgp extcommunity-list standard ExtComm_AllowInt soo
98.13.175.21:65101 all
```

Releases 3.0 or later

binding

Creates a binding between an ACL and NAT pool.

Command `binding *binding_name pool_name* [*acl_name*] [*natType*] [twice-nat-id] *twice_nat_id_value*`

Options

- *binding_name* — Binding name
- *pool_name* — Pool name
- *acl_name* — (Optional) ACL table name
- *natType* — (Optional) NAT type; snat or dnat (default)
- *twice_nat_id_value* — (Optional) Twice NAT ID value

Command mode NAT

Usage Use this command to create a binding between an ACL and NAT pool. You can use access-control list (ACL) to determine the IP addresses in a global NAT address pool. By default, if you specify an ACL, traffic for all IP hosts are allowed. A permit statement allows an IP address, where a deny statement denies the address. NAT types are `snat` which translates a source IP address to a global IP address in the pool, and `dnat` which translates a destination IP address to a global IP address in the pool. `twice-nat-id-value` performs address translation on both source and destination IP addresses using the address pool for static entries which have the same ID value. The `no` version of this command removes the ACL pool binding.

Example

```
sonic(config)# nat
sonic(conf-nat)# binding Bind2 Pool2 12_ACL_IPV4 snat twice-nat-id 25
```

Releases 3.0 or later

4

C commands

Topics:

call

Jumps to another route-map after match_set.

Command	`call match-call`
Options	*match-call* — Route-map name
Command mode	ROUTE-MAP
Usage	The `no` version of this command removes the configuration.
Example	`sonic(conf-route-map)# call match-call rmap1`
Releases	3.0 or later

capability dynamic

Allows BGP to advertise dynamically to a neighbor, or neighbors in a peer-group.

Command	`capability dynamic`
Options	None
Command modes	• BGP-NEIGHBOR • BGP-PEER-GROUP
Usage	Use this command enable dynamic BGP peering to exchange route information with remote neighbors or neighbors in a peer-group. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# capability dynamic
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# capability dynamic
```

Releases 3.0 or later

capability extended-nexthop

Enables BGP to negotiate the extended next-hop capability with its peer, or peers in a peer-group.

Command `capability extended-nexthop`

Options None

Command mode

- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to allow BGP to negotiate the extended next-hop capability with its peer, or peers in a peer-group. This command is automatically enabled for IPv6 link-layer addressing. If you are peering over a IPv6 global address, this command allows BGP to install IPv4 routes with IPv6 next-hops if you do not have IPv4 configured on interfaces. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# capability extended-nexthop
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# capability extended-nexthop
```

Releases 3.0 or later

capability orf prefix-list

Enables BGP to advertise outbound route filtering (ORF) to a neighbor, or neighbors in a peer-group.

Command `capability orf prefix-list {[send] | [receive] | [both]}`

Options None

Command modes

- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to advertise outbound route filtering capability to neighbor, or neighbors in a peer-group. This capability can be enabled in inbound and outbound direction separately. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# capability orf prefix-list send
```

```
sonic# configure terminal
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
```

```
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# capability orf prefix-list send
```

Releases 3.0 or later

channel-group

Assigns and configures a physical interface to a PortChannel group.

Command `channel-group lag-id`

Options *lag-id* — PortChannel number (1 to 128)

Command mode INTERFACE

Usage Use this command to create a PortChannel group. When you delete the last physical interface from a PortChannel, the PortChannel remains. Configure these attributes on an individual member port. The member ports in a PortChannel must have the same setting for link speed capability and duplex capability. The `no` version of this command removes the interface from the PortChannel.

Example

```
sonic(config)# interface PortChannel 1 mode active min-links 2 fallback
sonic(conf-if-po-1)# channel-group 3
```

Releases 3.0 or later

clear bgp l2vpn evpn

Clears BGP information for EVPN address-family on neighbors.

Command

```
clear bgp l2vpn evpn {{[as-num-dot] {[in] | [out] | {[soft] [in] [out]}}} |
{[*] {[in] | [out] | {[soft] [in] [out]}}} | {[external] {[in] | [out] |
{[soft] [in] [out]}}} | {[interface] {ifname {[in] | [out] | {[soft] [in]
[out]}}}} | {[peer-group] {peer-group name {[in] | [out] | {[soft] [in]
[out]}}}} | {[neighbor-ipv6] {[in] | [out] | {[soft] [in] [out]}}} |
{[neighbor-ipv4] {[in] | [out] | {[soft] [in] [out]}}} | [in] | [out] | {[soft]
[in] [out]}} clear bgp l2vpn evpn {{[as-num-dot] {[in] | [out] | {[soft] [in]
[out]}}} | {[*] {[in] | [out] | {[soft] [in] [out]}}} | {[external] {[in] |
[out] | {[soft] [in] [out]}}} | {[interface] {ifname {[in] | [out] | {[soft]
[in] [out]}}}} | {[peer-group] {peer-group name {[in] | [out] | {[soft] [in]
[out]}}}} | {[neighbor-ipv6] {[in] | [out] | {[soft] [in] [out]}}} |
{[neighbor-ipv4] {[in] | [out] | {[soft] [in] [out]}}} | [in] | [out] | {[soft]
[in] [out]}}
```

Options

- *as-num-dot* — (Optional) AS number (1 to 4294967295)
- *ifname* — (Optional) Interface name
- *peer-group name* — (Optional) Peer-group name
- *neighbor-ipv6* — (Optional) IPv6 neighbor address in A::B format
- *neighbor-ipv4* — (Optional) IPv4 neighbor address in A.B.C.D format

Command mode EXEC

Usage Use this command to delete L2 VPN BGP information. Here is a partial list of available commands.

- `clear bgp l2vpn evpn *` — Clears all BGP neighbors with address-family l2vpn evpn activated
- `clear bgp l2vpn evpn {peer_ip} *` — Clear peers with address of peer_ip and address-family l2vpn evpn activated
- `clear bgp l2vpn evpn soft {in | out}` — `in` option sends route-refresh request unless using 'soft-reconfiguration inbound, and `out` option re-sends all outbound updates

Example

```
sonic# clear bgp l2vpn evpn *
```

Releases 3.0 or later

clear counters

Clears all counters or for a specific interface.

Command `clear counters interface {all | {Ethernet [ifId]} | {PortChannel [ifId]}}`

Options
- `all` — Clears all counters
- *`ifId`* — (Optional) Clears a specific Ethernet or PortChannel interface

Command mode EXEC

Usage Use this command to clear all interface counters, or for a specific Ethernet or PortChannel interface.

Example

```
sonic# clear counters interface Ethernet 0
Clear counters on Ethernet0 [confirm y/N]: y
```

Releases 3.0 or later

clear ip arp

Clears all IPv4 ARP entries.

Command `clear ip arp [ip-addr] [force]`

Options
- `arp` *`ip-addr`* — (Optional) IPv4 address of the ARP entry to clear in A.B.C.D format
- `force` — (Optional) Delete statically configured ARP entries

Command mode EXEC

Usage Use this command to delete dynamically learned IPv4 entries from the ARP table. Use `show ip arp` to verify the IPv4 entries have been deleted.

Example

```
sonic# clear ip arp 192.168.1.4 force

sonic# show ip arp
------------------------------------------------------------------------
Address         Hardware address    Interface        Egress Interface
------------------------------------------------------------------------
192.168.2.4     00:01:02:03:ab:cd   PortChannel200   -
10.11.48.254    00:01:e8:8b:44:71   eth0             -
10.14.8.102     00:01:e8:8b:44:71   eth0             -
```

Releases 3.0 or later

clear ip arp interface

Clears ARP interface entries.

Command `clear ip arp interface` *`if-type if-id`* `[force]`

Options
- `interface` *`if-type`* — IPv4 interface type; Ethernet, Vlan, PortChannel, or Management
- `if-type` *`if-id`* — IPv4 interface ID
- `force` — Delete statically configured ARP entries

Command mode EXEC

Usage Use this command to delete dynamically learned IPv4 interface entries from the ARP table.

Examples

```
sonic# clear ip arp interface Vlan 100 force

sonic# show ip arp
------------------------------------------------------------------------
Address         Hardware address    Interface        Egress Interface
```

```
--------------------------------------------------------------------------
192.168.1.4     00:01:02:03:44:55   Ethernet8        -
192.168.2.4     00:01:02:03:ab:cd   PortChannel200   -
10.11.48.254    00:01:e8:8b:44:71   eth0             -
10.14.8.102     00:01:e8:8b:44:71   eth0             -
```

```
sonic# clear ip arp interface Management 0

sonic# show ip arp
--------------------------------------------------------------------------
Address         Hardware address    Interface        Egress Interface
--------------------------------------------------------------------------
192.168.1.4     00:01:02:03:44:55   Ethernet8        -
192.168.2.4     00:01:02:03:ab:cd   PortChannel200   -
10.14.8.102     00:01:e8:8b:44:71   eth0
```

Releases 3.0 or later

clear ip bgp

Resets BGP IPv4 or IPv6 neighbor sessions.

Command

```
clear ip bgp {{[as-num-dot] {[in] | [out] | {[soft] [in] [out]}}} | {[*] {[in]
| [out] | {[soft] [in] [out]}}} | {[external] {[in] | [out] | {[soft] [in]
[out]}}} | {[interface] {ifname {[in] | [out] | {[soft] [in] [out]}}}} |
{[peer-group] {peer-group name {[in] | [out] | {[soft] [in] [out]}}}} |
{[neighbor-ipv4] {[in] | [out] | {[soft] [in] [out]}}} | {[prefix] {prefix-ip4
{[in] | [out] | {[soft] [in] [out]}}}} | {[neighbor-ipv6] {[in] | [out] |
{[soft] [in] [out]}}} | {[prefix] {prefix-ip6 {[in] | [out] | {[soft] [in]
[out]}}}} | {ipv4 {{[as-num-dot] {[in] | [out] | {[soft] [in] [out]}}} | {[*]
{[in] | [out] | {[soft] [in] [out]}}} | {[external] {[in] | [out] | {[soft]
[in] [out]}}} | {[interface] {ifname {[in] | [out] | {[soft] [in] [out]}}}} |
{[peer-group] {peer-group name {[in] | [out] | {[soft] [in] [out]}}}} |
{[neighbor-ipv4] {[in] | [out] | {[soft] [in] [out]}}} | {[prefix] {prefix-ip4
{[in] | [out] | {[soft] [in] [out]}}}} | [in] | [out] | {[soft] [in] [out]}}} |
{ipv6 {{[as-num-dot] {[in] | [out] | {[soft] [in] [out]}}} | {[*] {[in] | [out]
| {[soft] [in] [out]}}} | {[external] {[in] | [out] | {[soft] [in] [out]}}} |
{[interface] {ifname {[in] | [out] | {[soft] [in] [out]}}}} | {[peer-group]
{peer-group name {[in] | [out] | {[soft] [in] [out]}}}} | {[neighbor-ipv6]
{[in] | [out] | {[soft] [in] [out]}}} | {[prefix] {prefix-ip6 {[in] | [out] |
{[soft] [in] [out]}}}} | [in] | [out] | {[soft] [in] [out]}}} | {[vrf] {vrf-
name {{[as-num-dot] {[in] | [out] | {[soft] [in] [out]}}} | {[*] {[in] | [out]
| {[soft] [in] [out]}}} | {[external] {[in] | [out] | {[soft] [in] [out]}}} |
{[interface] {ifname {[in] | [out] | {[soft] [in] [out]}}}} | {[peer-group]
{peer-group name {[in] | [out] | {[soft] [in] [out]}}}} | {[neighbor-ipv4]
{[in] | [out] | { [soft] [in] [out]}}} | {[prefix] {prefix-ip4 {[in] | [out] |
{[soft] [in] [out]}}}} | {[neighbor-ipv6] {[in] | [out] | {[soft] [in] [out]}}}
| {[prefix] {prefix-ip6 {[in] | [out] | {[soft] [in] [out]}}}} | {ipv4 {{[as-
num-dot] {[in] | [out] | {[soft] [in] [out]}}} | {[*] {[in] | [out] | {[soft]
[in] [out]}}} | {[external] {[in] | [out] | {[soft] [in] [out]}}} |
{[interface] {ifname {[in] | [out] | {[soft] [in] [out]}}}} | {[peer-group]
{peer-group name {[in] | [out] | {[soft] [in] [out]}}}} | {[neighbor-ipv4]
{[in] | [out] | {[soft] [in] [out]}}} | {[prefix] {prefix-ip4 {[in] | [out] |
{[soft] [in] [out]}}}} | [in] | [out] | {[soft] [in] [out]}}} | {ipv6 {{[as-
num-dot] {[in] | [out] | {[soft] [in] [out]}}} | {[*] {[in] | [out] | {[soft]
[in] [out]}}} | {[external] {[in] | [out] | {[soft] [in] [out]}}} |
{[interface] {ifname {[in] | [out] | {[soft] [in] [out]}}}} | {[peer-group]
{peer-group name {[in] | [out] | {[soft] [in] [out]}}}} | {[neighbor-ipv6]
{[in] | [out] | {[soft] [in] [out]}}} | {[prefix] {prefix-ip6 {[in] | [out] |
{[soft] [in] [out]}}}} | [in] | [out] | {[soft] [in] [out]}}}}}}}
```

Options

- *as-num-dot* — (Optional) AS number (1 to 4294967295)
- soft — (Optional) Configures and activates policies without resetting the BGP TCP session

- `*` — (Optional) Clears all BGP sessions
- *`ifname`* — (Optional) Interface name
- *`peer-group name`* — (Optional) Peer-group name
- *`neighbor-ipv4`* — (Optional) Neighbor IPv4 address in A.B.C.D format
- *`prefix-ip4`* — (Optional) IPv4 prefix in A.B.C.D/mask format
- *`neighbor-ipv6`* — (Optional) Neighbor IPv6 address in A::B format
- *`prefix-ip6`* — (Optional) IPv6 prefix in A::B/mask format
- *`vrf-name`* — (Optional) VRF instance name (up to 15 characters)

Command mode EXEC

Usage Use this command to clear or reset BGP IPv4 or IPv6 neighbor sessions. Here is a partial list of possible command values.

- `clear ip bgp *`— Clears all BGP neighbors
- `clear ip bgp {ipv4 | ipv6} unicast *` — Clears all BGP neighbors with this address-family and sub-address-family activated
- `clear ip bgp {ipv4 | ipv6} peer_ip *` — Clear peers with address of peer_ip and this address-family activated
- `clear ip bgp {ipv6 | ipv4} soft {in | out}` — `in` option sends route-refresh request unless using soft-reconfiguration inbound, and `out` option resends all outbound updates

Example

```
sonic# clear ip bgp 14.14.14.1
```

Releases 3.0 or later

clear ipv6 neighbors

Clears entries in the IPv6 neighbor discovery cache, or neighbors of a specific interface.

Command `clear ipv6 neighbors [ip-addr] [force]`

Options

- *`ip-addr`* — (Optional) IPv6 address of the neighbor in A::B format
- `force` — (Optional) Deletes IPv6 neighbor entries

Command mode EXEC

Usage Use this command to delete IPv6 entries or neighbors of a specific interface.

Examples

```
sonic# clear ipv6 neighbors

sonic# show ipv6 neighbors
------------------------------------------------------------------------------
Address                   Hardware address    Interface   Egress Interface
------------------------------------------------------------------------------
20::2                     aa:bb:cc:dd:ee:ff   Ethernet8
-
fe80::e6f0:4ff:fe79:34c7  e4:f0:04:79:34:c7   eth0        -
```

```
sonic# clear ipv6 neighbors 20::2

sonic# show ipv6 neighbors 20::2
------------------------------------------------------------------------------
Address                   Hardware address    Interface   Egress Interface
------------------------------------------------------------------------------
20::2                     aa:bb:cc:dd:ee:ff   Ethernet8   -
```

Releases 3.0 or later

clear ipv6 neighbors interface

Clears IPv6 neighbors of a specific interface.

Command `clear ipv6 neighbors interface if-type if-id [force]`

Options
- *if-type* — Interface type; Ethernet, Vlan, PortChannel, or Management
- *if-id* — Interface ID
- `force` — (Optional) Deletes strategically configured interface entries

Command mode EXEC

Usage Use this command to delete or remove IPv6 neighbors of Ethernet, VLAN, PortChannel, or a Management interface. To specify entries to delete, enter an interface type and interface ID. Use `show ipv6 neighbors` to verify the entries have been deleted. The `no` version of this command removes the configuration.

Example

```
sonic# clear ipv6 neighbors 20::2 force

sonic# show ipv6 neighbors
------------------------------------------------------------------------------
Address                   Hardware address    Interface   Egress Interface
------------------------------------------------------------------------------
fe80::e6f0:4ff:fe79:34c7  e4:f0:04:79:34:c7   eth0        -
```

Releases 3.0 or later

clear mac address-table dynamic

Clears L2 dynamic address entries from the MAC address-table.

Command `clear mac address-table dynamic {all | {address mac-addr} | {Vlan vlan-id}}`

Options
- `all` — Deletes all MAC address-table entries
- `address mac-addr` — Deletes a configured MAC address from the address-table in nn:nn:nn:nn:nn:nn format
- `Vlan vlan-id` — Deletes all entries based on the VLAN number from the address-table (1 to 4093)

Command mode EXEC

Usage Use this command to clear all or specific entries in the MAC address table. Use the `all` option to remove all dynamic entries from the address-table. The `no` version of this command removes the configuration.

Examples

```
sonic# clear mac address-table dynamic all
```

```
sonic# clear mac address-table dynamic Vlan 20
```

Releases 3.0 or later

clear nat

Clears network address translations and statistics.

Command `clear nat {translations | statistics}`

Options None

Command mode EXEC

Usage Use this command to clear the entries in the NAT translation table.

Example

```
sonic# clear nat translations
```

Releases 3.0 or later

client-to-client reflection

Enables route reflection between clients in a cluster.

Command `client-to-client reflection`

Options None

Command mode ROUTER-BGP

Usage Use this command to configure the route-reflector to enable the sharing of route information between members of a peer-group that is configured as a BGP route-reflector client. Route information received from one peer-group member is sent to all other members. You must fully mesh all clients before you disable route-reflection. The `no` version of this command disables route-reflection in a cluster.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# client-to-client reflection
```

Releases 3.0 or later

cluster-id

Assigns a cluster ID to a BGP cluster with multiple route-reflectors.

Command `cluster-id` *`intval-ip`*

Options *`intval-ip`* — IP address in A.B.C.D format (default), or route-reflector cluster ID as a 32-bit number (1 to 4294967295)

Command mode ROUTER-BGP

Usage Use this command to configure a cluster ID (an IP address or a 32-bit number) on a BGP router. A cluster is a collection of route reflectors and their clients, and is used by route reflectors to avoid looping. If a cluster contains only one route-reflector, the cluster ID is the route-reflector's router ID. For redundancy, a BGP cluster may contain two or more route reflectors. Without a cluster ID, the route reflector cannot recognize route updates from the other route reflectors within the cluster. The default format to display the cluster ID is A.B.C.D format. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# cluster-id 23.79.154.17
```

Releases 3.0 or later

coalesce-time

Configures the coalesce timer interval.

Command `coalesce-time` *`coaltime`*

Options *`coaltime`* — Coalesce time in milliseconds (1 to 4294967295)

Command mode ROUTER-BGP

Usage Use this command to configure the coalesce timer interval. The coalesce timer is used in the sub-AS group to which the router belongs. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# coalesce-time 2000
```

Releases 3.0 or later

collector

Configures the external collector IP address and port.

Command `collector name type ip-type-val ip ip-addr-val port port-val`

Options

- *name* — Collector name (up to 63 characters)
- *ip-type-val* — Collector IP type; ipv4 or ipv6
- *ip-addr-val* — IPv4 or IPv6 address in A.B.C.D or A::B format
- *port-val* — Port number

Command mode CONFIGURATION-TAM

Usage Use this command to configure the external inband flow analyzer (IFA) collector IP address and port. Reports are forwarded to the configured collector. The `no` version of this command removes the configuration.

Example

```
sonic(config)# tam
sonic(conf-tam)# collector c1 type ipv4 ip 3.3.3.3 port 7777
```

Releases 3.0 or later

confederation

Configures an identifier for a BGP confederation.

Command `confederation {{identifier id-as} | {peers peer-as}}`

Options

- `identifier id-as` — AS number (0 to 65535 for 2 bytes, 1 to 4294967295 for 4 bytes, or 0.1 to 65535.65535 for dotted format)
- `peers peer-as` — AS number for peers in the BGP confederation (1 to 4294967295)

Command mode ROUTER-BGP

Usage Use this command to configure your system to accept 4-byte formats before entering a 4-byte AS number. All routers in the confederation must be 4-byte or 2-byte identified routers. You cannot have a mix of 2-byte and 4-byte identified routers. The autonomous system number that you configure in this command is visible to the eBGP neighbors. Each autonomous system is fully meshed and contains a few connections to other autonomous systems. The next-hop (MED) and local preference information is preserved throughout the confederation. The system accepts confederation eBGP peers without a LOCAL_PREF attribute. SONiC sends AS_CONFED_SET and accepts AS_CONFED_SET and AS_CONF_SEQ. The `no` version of this command deletes the confederation configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# confederation identifier 65000
sonic(conf-router-bgp)# confederation peers 65100
sonic(conf-router-bgp)# confederation peers 65200
sonic(conf-router-bgp)# confederation peers 65300
```

Releases 3.0 or later

configure terminal

Enters configuration mode.

Command `configure terminal`

Options None

Command mode EXEC

Usage Use this command to enter configuration mode. You can also use the shortcut `config t`. The `no` version of this command removes the configuration.

Example

```
sonic# config t
sonic(config)#
```

Releases 3.0 or later

copy

Copies the running configuration to the startup configuration.

Command `copy {{copy_config_url {running-configuration [overwrite]}} | {running-configuration filepath} | {startup-configuration {running-configuration [overwrite]}}}`

Options

- *copy_config_url* — URL to copy configuration to or from
- *filepath* — File path to copy files to or from

Command mode EXEC

Usage Use this command to save the running configuration to the startup configuration, transfer coredump files to a remote location, backup the startup configuration, retrieve a previously backed-up configuration, replace the startup configuration file, or transfer support bundles.

Example

```
sonic# copy https://10.206.28.174:/startup.xml https://10.206.28.174://
backup.xml
```

Releases 3.0 or later

5

D commands

Topics:

dampening

Enables BGP route-flap dampening and configures the dampening parameters.

Command `dampening [halflife] {[reusethr] {suppressthr maxsuppress}}`

Options

- *halflife* — (Optional) Half-life time, in minutes, after which the penalty decreases; after the router assigns a penalty of 1024 to a route, the penalty decreases by half after the half-life period expires (1 to 45; default 15)
- *reusethr* — (Optional) Reuse threshold value which compares to the flapping route's penalty value; if the penalty value is less than the reuse value, the flapping route advertises again and is not suppressed (1 to 20000; default 750)
- *suppressthr* — Suppress threshold value which compares to the flapping route's penalty value; if the penalty value is greater than the suppress value, the flapping route is no longer advertised (1 to 20000; default 2000)
- *maxsuppress* — Maximum number of minutes a route is suppressed (1 to 255; default 60)

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to reduce the instability of the BGP process. After you set up the dampening parameters, clear information about route dampening and return the suppressed routes to the Active state. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family ipv4 unicast
sonic(conf-router-bgp-af)# dampening 10 1200 2000 40
```

Releases 3.0 or later

default ipv4-unicast

Enables IPv4 unicast address-family for a BGP peer.

Command `default ipv4-unicast`

Options None

Command mode ROUTER-BGP

Usage Use this command to activate IPv4 unicast address-family on BGP neighbors by default. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# default ipv4-unicast
```

Releases 3.0 or later

default local-preference

Configures the default value for the local preference attribute.

Command `default local-preference` *`lprftime`*

Options *`lprftime`* — Local preference time

Command mode ROUTER-BGP

Usage Use this command to set the default value of the local preference parameter. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# default local-preference 200
```

Releases 3.0 or later

default show-hostname

Configures BGP to displays the hostname in specific display commands.

Command `default show-hostname`

Options None

Command mode ROUTER-BGP

Usage Use this command to instruct BGP to display hostname in certain display command outputs. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# default show-hostname
```

Releases 3.0 or later

default shutdown

Configures BGP to make newly created neighbors in shutdown state.

Command `default shutdown`

Options None

Command mode ROUTER-BGP

Usage Use this command to keep newly created BGP neighbors in admin shutdown state. By default, newly created BGP neighbors are in admin enabled state. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# default shutdown
```

Releases 3.0 or later

default subgroup-pkt-queue-max

Configures the maximum packet queue length for update groups.

Command `default subgroup-pkt-queue-max` *maxval*

Options *maxval* — Maximum packet queue length

Command mode ROUTER-BGP

Usage Use this command to set a default maximum packet queue length for update groups. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# default subgroup-pkt-queue-max 50
```

Releases 3.0 or later

default-originate

Configures the default route to a BGP peer or neighbor.

Command `default-originate [route-map]` *rtemap*

Options *rtemap* — Route-map name (up to 140 characters)

Command modeS

- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to configure the default route to this neighbor, or neighbors in a peer-group. You can optionally use route-map to specify criteria to originate a default. The `no` version of this command removes the default route.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# default-originate
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# default-originate
```

Releases 3.0 or later

default-originate ipv4

Enables border-leaf to originate IPv4 default type-5 EVPN routes.

Command `default-originate ipv4`

Options None

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to enable border leaf to originate IPv4 default type-5 EVPN routes. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100 vrf Vrf1
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# default-originate ipv4
```

Releases 3.0 or later

default-originate ipv6

Enables border-leaf to originate IPv6 default type-5 EVPN routes.

Command `default-originate ipv6`

Options None

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to enable border-leaf to originate IPv6 default type-5 EVPN routes. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100 vrf Vrf1
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# default-originate ipv6
```

Releases 3.0 or later

description

Configures a description for an interface, link state group, BGP neighbor, and BGP peer-group.

Command `description` *`string`*

Options *`string`* — Descriptive string

Command modes

- INTERFACE
- LINK-STATE-GROUP
- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to configure a descriptive string for an interface, link state group, BGP neighbor, and BGP peer-group. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# remote-as 65100
sonic(conf-router-bgp-neighbor)# description to_nyc_dc1
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# description My_PG_East_Cost_Nbrs
```

Releases 3.0 or later

destination

Configures an ERSPAN/SPAN mirror-session for port mirroring.

Command `destination phy-if-id [source] src-phy-if-id [direction] sess-direction [dst-ip] dst_ip [src-ip] src_ip [dscp] ip_dscp [gre] ip_gre [ttl] ip_ttl [queue] queue_val`

Options

- *phy-if-id* — Interface ID
- `source` *src-phy-if-id* — (Optional) Source interface ID
- `direction` *sess-direction* — (Optional) Port mirror session direction; rx, tx, both
- `dst-ip` *dst_ip* — (Optional) Destination IPv4 address in A.B.C.D format
- `src-ip` *src_ip* — (Optional) Source IPv4 address in A.B.C.D format
- `dscp` *ip_dscp* — (Optional) DSCP IP value
- `gre` *ip_gre* — (Optional) Greater or equal to the IP value
- `ttl` *ip_ttl* — (Optional) TTL IP value
- `queue` *queue_val* — (Optional) Queue value

Command mode CONFIGURATION-MIRROR

Usage Use this command to configure an ERSPAN/SPAN mirror-session for port mirroring. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# mirror-session Mirror1
sonic(conf-mirror-Mirror1)# destination Ethernet0 source Ethernet4 direction
rx
Success
sonic(conf-mirror-Mirror1)# exit
```

```
sonic(config)# mirror-session Mirror2
sonic(conf-mirror-Mirror2)# destination erspan dst-ip 10.1.1.1 src-ip
11.1.1.1 dscp 10
ttl 10 gre 0x88ee queue 10 source Ethernet4 direction rx
Success
sonic(conf-mirror-Mirror2)#
```

Releases 3.0 or later

detect-multiplier

Configures a detection multiplier for bidirectional forwarding detection (BFD) peers for timeout.

Command `detect-multiplier multiplier`

Options *multiplier* — Multiplier value; default 3

Command mode BFD-PEER-GROUP

Usage Use this command to configure detection multiplier for BFD peers for timeout. The `no` version of this command removes the configuration.

Example

```
sonic(config)# bfd
sonic(conf-bfd)# peer 192.168.0.5 interface Ethernet0
sonic(conf-bfd-peer)# detect-multiplier 5
```

Releases 3.0 or later

deterministic-med

Carries out route-selection that produces deterministic results locally.

Command `deterministic-med`

Options None

Command mode ROUTER-BGP

Usage Use this command to carry out route-selection in a way that produces deterministic results locally, even in the face of MED and the lack of a well-defined order of preference it can induce on routes. Without this option, the preferred route with MED may be determined largely by the order that routes were received in. Setting this option will have a performance cost that may be noticeable when there are many routes for each destination. BGP is implemented in a way that scales poorly as the number of routes per destination increases. By default deterministic-med is disabled. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# deterministic-med
```

Releases 3.0 or later

device-id

Configures the device ID.

Command `device-id` *`device-id`*

Options *`device-id`* — Switch device ID

Command mode CONFIGURATION

Usage Use this command to configure the switch device ID. The last 23 bits of the MAC address are used as the default device ID. The `no` version of this command removes the configuration.

Example

```
sonic(config)# tam
sonic(conf-tam)# device-id 5000
```

Releases 3.0 or later

disable-connected-check

Disables the restriction that eBGP peers must be directly connected.

Command `disable-connected-check`

Options None

Command modes

- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to allow peerings between directly connected eBGP peers using loopback addresses. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# disable-connected-check
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# disable-connected-check
```

Releases 3.0 or later

disable-ebgp-connected-route-check

Disables eBGP connected route check.

Command `disable-ebgp-connected-route-check`

Options None

Command mode ROUTER-BGP

Usage Use this command to disable checking if next-hop is connected on eBGP sessions. When BGP peering is between the loopback interfaces, enable this option. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# disable-ebgp-connected-route-check
```

Releases 3.0 or later

distance bgp

Sets the administrative distance for BGP routes.

Command `distance bgp external {internal local}`

Options

- *external* — Number to assign to routes learned from a neighbor external (eBGP) to the AS (1 to 255; default 20)
- *internal* — Number to assign to routes learned from a router within (iBGP) the AS (1 to 255; default 200)
- *local* — Number to assign to routes learned from networks (1 to 255; default 200)

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to configure the administrative distance for eBGP route, iBGP route, and local BGP route. The command allows finer control to change the distance values for external routes, internal routes, and local routes separately. Administrative distance indicates the reliability of the route; the lower the administrative distance, the more reliable the route is. Routes that are assigned an administrative distance of 255 are not installed in the routing table. Routes from confederations are treated as iBGP routes. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family ipv4 unicast
sonic(conf-router-bgp-af)# distance bgp 100 50 10
```

Releases 3.0 or later

dont-capability-negotiate

Disables capability negotiation for a BGP neighbor, or neighbors in a peer-group.

Command `dont-capability-negotiate`

Options None

Command modes

- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to disable capability negotiation for a BGP neighbor, or neighbors in a peer-group. This command suppresses the sending capability negotiation as OPEN message optional parameter to the peer. This command only affects the peer is configured other than IPv4 unicast configuration. When a remote peer does not have capability negotiation feature, remote peer will not send any capabilities at all — BGP configures the peer with configured capabilities. You may prefer locally configured capabilities more than the negotiated capabilities even though remote peer sends capabilities. If the peer is configured by override-capability, BGP ignores received capabilities then override negotiated capabilities with configured values. This feature fundamentally disables the ability to use widely deployed BGP features such as BGP unnumbered, hostname support, AS4, addpath, route

refresh, ORF, dynamic capabilities, and graceful restart. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# dont-capability-negotiate
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# dont-capability-negotiate
```

Releases 3.0 or later

downstream all-mclag

Configures downstream ports.

Command `downstream all-mclag`

Options None

Command mode LINK-STATE-TRACK

Usage Use this command to add an interface or range of interfaces as a downstream interface to the link-state track. The `no` version of this command removes the interface.

Example

```
sonic(conf-uplink-state-group-1)# downstream all-mclag
```

Releases 3.0 or later

drop-monitor

Enters drop-monitor configuration mode.

Command `drop-monitor`

Options None

Command mode TAM

Usage Use this command to configure drop-monitor. The `no` version of this command removes the configuration.

Example

```
sonic(config)# tam
sonic(conf-tam)# drop-monitor
```

Releases 3.0 or later

dup-addr-detection

Configures the threshold for address moves.

Command `dup-addr-detection [max-moves] {`*`nummoves`* `{time` *`timevalue`*`}}`

Options

- *`nummoves`* — Number of moves (2 to 1000; default 5)
- *`timevalue`* — Time in seconds (2 to 1800; default 180)

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to configure threshold address moves including maximum moves allowed and maximum time interval. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# dup-addr-detection max-moves 10 time 1200
```

Releases 3.0 or later

dup-addr-detection freeze

Configures duplicate address detection.

Command `dup-addr-detection freeze {permanent | time}`

Options *time* — Amount of time to freeze in seconds (30 to 3600; default 180)

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to configure the action to be taken on duplicate address detection. You can configure freezing the address permanently or for a specified duration. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# dup-addr-detection freeze permanent
```

Releases 3.0 or later

6

E, F, and G commands

Topics:

ebgp-multihop

Allows eBGP neighbors on indirectly connected networks.

Command `ebgp-multihop [hop-count]`

Options *hop-count* — (Optional) Maximum number of hops allowed to communicate with a peer in a remote network (1 to 255; default 1 for eBGP, 255 for iBGP)

Command modes
- BGP-ADDRESS-FAMILY
- BGP-PEER-GROUP

Usage Use this command to configure a peer-group with eBGP neighbors as members that are multiple hops away. You can optionally set the maximum hops that BGP neighbors in peer-group can be apart. This command avoids installation of default multi-hop peer routes to prevent loops and creates neighbor relationships between peers. Networks indirectly connected are not valid for best-path selection. The `no` version of this command removes the multi-hop session.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# dup-addr-detection freeze permanent
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# ebgp-multihop 10
```

Releases 3.0 or later

echo-interval

Configures the desired echo packet transmit interval for a bidirectional forwarding detection (BFD) peer.

Command `echo-interval echo_interval`

Options *echo_interval* — Echo packet transmit interval in ms (default 50)

Command mode BFD-PEER

Usage Use this command to configure the desired echo packet transmit interval for a BFD peer. The `no` version of this command removes the configuration.

Example

```
sonic(config)# bfd
sonic(conf-bfd)# peer 192.168.0.5 interface Ethernet0
sonic(conf-bfd-peer)# echo-interval 200
```

Releases 3.0 or later

echo-mode

Enables echo-mode for a bidirectional forwarding detection (BFD) peer.

Command `echo-mode`

Options None

Command mode BFD-PEER

Usage Use this command to enable echo mode for BFD single-hop peer. Echo mode is not supported for multi-hop peers. The `no` version of this command removes the configuration.

Example

```
sonic(config)# bfd
sonic(conf-bfd)# peer 192.168.0.5 interface Ethernet0
sonic(conf-bfd-peer)# echo-mode
```

Releases 3.0 or later

enable

Enables network address translation (NAT).

Command `enable`

Options

Command mode NAT

Usage Use this command to enable NAT for configuration. The `no` version of this command removes the configuration.

Example

```
sonic(config)# nat
sonic(if-nat)# enable
```

Releases 3.0 or later

enforce-first-as

Enforces the first AS in the AS path of the route received from an eBGP peer to be the same as the configured remote AS

Command `enforce-first-as`

Options None

Command modes

- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to enforce that first AS in as-path of route received from an eBGP peer must be the peer's local AS number. If routes are rejected, the session is reset. In the event of a failure, the existing BGP sessions flap. For updates received from eBGP peers, BGP ensures that the first AS of the first AS segment is always the AS of the peer, otherwise the update drops and the counter increments. The `no` version disables this command.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# enforce-first-as
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# enforce-first-as
```

Releases 3.0 or later

enforce-multihop

Configures eBGP neighbors or neighbors in a peer-group to perform multi-hop.

Command `enforce-multihop`

Options None

Command modes

- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to configure eBGP neighbors or neighbors in a peer-group to perform multi-hop. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# enforce-multihop
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# enforce-multihop
```

Releases 3.0 or later

evpn

Enters EVPN configuration mode.

Command `evpn` *`evpn_name`*

Options *`evpn_name`* — EVPN name (up to 63 characters)

Command mode INTERFACE-EVPN

Usage Use this command to enter EVPN configuration mode.

Example

```
sonic(config)# evpn evpn1
sonic(conf-if-evpn)#
```

Releases 3.0 or later

fast-external-failover

Causes BGP to take down eBGP peers immediately when a link flaps.

Command `fast-external-failover`

Options None

Command mode ROUTER-BGP

Usage Use this command to control how sensitive eBGP neighborship is to the underlying link failure. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# fast-external-failover
```

Releases 3.0 or later

feature

Enables or disables the drop-monitor, IFA, or IFA TS feature.

Command `feature {enable | disable}`

Options

- `enable` — Enables the specific feature
- `disable` — Disables the specific feature

Command mode

- DROP-MONITOR
- TAM-INT-IFA
- INT-IFA-TS

Usage Use this command to enable drop-monitor, INT IFA, or the INT IFA TS feature.

Example

```
sonic(config-drop-monitor)# feature enable
```

```
sonic(config-tam-int-ifa)# feature enable
```

```
sonic(config-int-ifa-ts)# feature enable
```

Releases 3.0 or later

filter-list

Configures a filter list for a BGP neighbor or peer-group.

Command `filter-list` *`fname`* `{in | out}`

Options *`fname`* — Filter-list name

Command modes

- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to define policy (route filtering) for a BGP neighbor or peer-group in outbound or/and inbound direction. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# filter-list fl_allow_remote in
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# filter-list fl_allow_remote in
```

Releases 3.0 or later

flow

Configures a drop-monitor flow.

Command

```
flow flow-name acl-table acl-table-name acl-rule acl-rule-name [collector collector-name] [sample sampling-name] flowgroup-id flowgroup_id
```

Options

- `flow flow-name` — Flow name (up to 63 characters)
- `acl-table acl-table-name` — ACL table name (up to 63 characters)
- `acl-rule acl-rule-name` — ACL rule name (up to 63 characters)
- `collector collector-name` — (Optional) Collector name (up to 63 characters)
- `sample sampling-name` — (Optional) Sampling name (up to 63 characters)
- `flowgroup-id flowgroup_id` — Flow-group ID

Command modes

- DROP-MONITOR
- TAM-INT-IFA
- TAM-INT-IFA-TS

Usage

Use this command to associate a flow configuration specified by an ACL table, ACL table rule, collector, and sample configuration. The `no` version of this command removes the configuration.

Examples

```
sonic(conf-drop-monitor)# flow f4 acl-table t4 acl-rule r4 collector c1
sample s1
flowgroup-id 4
```

```
sonic(conf-tam-int-ifa)# flow iflow1 acl-table iacl1 acl-rule irule1
sampling-rate 12
collector-name icol
```

```
sonic(conf-tam-int-ifa-ts)# flow flow32 acl-table tacl1 acl-rule trule1
```

Releases

3.0 or later

graceful-restart enable

Enables graceful restart for an instance of BGP.

Command

```
graceful-restart enable
```

Options

None

Command mode

ROUTER-BGP

Usage

Use this command to enable BGP graceful restart globally in an instance of BGP. Changing the graceful restart parameter will take effect only on the fly, and will not take effect immediately. It will require all the BGP neighbors to be reset to take effect. This is because graceful restart capability must be negotiated with neighbors to make this feature functional. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# graceful-restart enable
```

Releases

3.0 or later

graceful-restart preserve-fw-state

Configures BGP to preserve forwarding state during graceful restart for an instance of BGP.

Command

```
graceful-restart preserve-fw-state
```

Options

None

Command mode	ROUTER-BGP
Usage	Use this command to enable BGP to preserve forwarding state of BGP during graceful restart. The `no` version of this command removes the configuration.
Example	`sonic(config)# router bgp 65300` `sonic(conf-router-bgp)# graceful-restart preserve-fw-state`
Releases	3.0 or later

graceful-restart restart-time

Configures the restart timer interval for BGP.

Command	`graceful-restart restart-time` *`restart-time`*
Options	*`restart-time`* — Restart time in seconds (default 120)
Command mode	ROUTER-BGP
Usage	Use this command to configure the BGP restart timer interval in seconds. This is optional parameter and determines how long peer routers will wait to delete stale routes before a BGP open message is received. The default value is 120 seconds. The `no` version of this command removes the configuration.
Example	`sonic(config)# router bgp 65300` `sonic(conf-router-bgp)# graceful-restart restart-time 180`
Releases	3.0 or later

graceful-restart stalepath-time

Configures the stale path timer interval for BGP.

Command	`graceful-restart stalepath-time` *`stalepath-time`*
Options	*`stalepath-time`* — Stale path time in seconds (default 360)
Command mode	ROUTER-BGP
Usage	Use this command to set the maximum time to hold on to the stale paths of a gracefully restarted peer. All stale paths are deleted after the expiration of this timer. This is an optional parameter, and the default is 360 seconds. The `no` version of this command removes the configuration.
Example	`sonic(config)# router bgp 65300` `sonic(conf-router-bgp)# graceful-restart stalepath-time 300`
Releases	3.0 or later

graceful-shutdown

Enables the graceful shutdown feature.

Command	`graceful shutdown`
Options	None
Command mode	ROUTER-BGP
Usage	Use this command to gracefully remove a BGP router from service. This command will instruct BGP to enter into graceful shutdown mode by resending routes with GSHUT community to all neighbors. This will enable all neighbors to route traffic around it so that the router can be taken out of service without impact data forwarding. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# graceful-shutdown
```

Releases 3.0 or later

7

I commands

Topics:

image install

Installs a SONiC image.

Command `image install img-path`

Options *img-path* — Location of the image

- `http[s]://hostip:/filepath` — Enter the path to install from the remote HTTP or HTTPS server
- `image://filename` — Enter the path to use to install the image from a local file system

Command mode EXEC

Usage Use this command to install a SONiC image in the standby partition. If the active partition contains any modified text files or installed custom packages, they will be not available in the standby partition. Backup the modified files and re-install the packages after downloading the image.

Example

```
sonic# image install https://10.206.28.174:/DellEMC-OpenFabric-SONiC-
OS-3.0.0-Enterprise-Standard.bin
```

Releases 3.0 or later

image remove

Removes all or a specific image file.

Command `image remove {all | image}`

Options
- `all` — Removes all image files
- *image* — Removes a specific image file

Command mode EXEC

Usage Use this command to delete an unused SONiC image. You cannot remove the current running image.

Example

```
sonic# image remove SONiC-OS-HEAD.140-20200105.093102
Remove image SONiC-OS-HEAD.138-20200103.154042? [y/N]:y
```

Releases 3.0 or later

image set-default

Sets the default boot image.

Command `image set-default img-name`

Options *img-name* — Image name

Command mode EXEC

Usage Use this command to set the default boot image.

Example

```
sonic# image set default SONiC-OS-HEAD.140-20200103.154042
```

Releases 3.0 or later

interface

Enters interface configuration mode.

Command `interface {phy-if-name | vlan-if-name}`

Options
- *phy-if-name* — Physical interface name
- *vlan-if-name* — VLAN interface name

Command mode CONFIGURATION

Usage Use this command to select a physical interface (Ethernet, PortChannel, Management, or Loopback) or VLAN interface to configure.

Example

```
sonic(config)# interface Ethernet 2
sonic(conf-if-Eth2)#
```

Releases 3.0 or later

interface Loopback

Enters Loopback interface configuration mode.

Command `interface Loopback lo-id`

Options *lo-id* — Loopback interface ID number (0 to 16383)

Command mode CONFIGURATION

Usage Use this command to configure a Loopback interface. The `no` version of this command deletes the Loopback interface.

Example

```
sonic(config)# interface loopback 100
sonic(conf-if-lo-100)#
```

Releases 3.0 or later

interface Management

Enters Management interface configuration mode.

Command `interface Management mgmt-if-id`

Options *mgmt-if-id* — Management interface ID

Command mode CONFIGURATION

Usage Use this command to configure the Management interface. You cannot delete a Management interface port.

Example

```
sonic(config)# interface Management 0
sonic(conf-if-ma-0)#
```

Releases 3.0 or later

interface PortChannel

Enters PortChannel interface configuration mode.

Command `interface PortChannel lag-id [mode] PoMode [min-links] min-links-value [fallback]`

Options

- *lag-id* — PortChannel ID number (1 to 128)
- *PoMode* — (Optional) PortChannel mode
- *min-links-value* — (Optional) Minimum links

Command mode CONFIGURATION

Usage Use this command to create a PortChannel interface. The `no` version of this command deletes the PortChannel interface.

Example

```
sonic(config)# interface PortChannel 10
sonic(conf-if-po-10)#
```

Releases 3.0 or later

interface Vxlan

Enters VxLAN interface configuration mode.

Command `interface Vxlan vxlan-if-name`

Options *vxlan-if-name* — VxLAN interface name (up to 63 characters)

Command mode CONFIGURATION

Usage Use this command to configure a VxLAN interface. A virtual extensible LAN (VxLAN) extends Layer 2 (L2) server connectivity over an underlying Layer 3 (L3) transport network in a virtualized data center. A virtualized data center consists of virtual machines (VMs) in a multi-tenant environment.

Example

```
sonic(config)# interface vxlan vtep1
sonic(conf-if-vtep1)#
```

Releases	3.0 or later

int-ifa

Enters inband flow analyzer (IFA) configuration mode.

Command	`int-ifa`
Options	None
Command mode	TAM
Usage	Use this command to enable IFA data collection.
Example	`sonic(conf-tam)# int-ifa` `sonic(conf-tam-int-ifa)#`
Releases	3.0 or later

int-ifa-ts

Enters inband flow analyzer (IFA) tail timestamping configuration mode.

Command	`int-ifa-ts`
Options	None
Command mode	TAM
Usage	Use this command to enable IFA TS data collection.
Example	`sonic(conf-tam)# int-ifa-ts` `sonic(conf-tam-int-ifa-ts)#`
Releases	3.0 or later

ip access-group

Specifies access control for packets.

Command	`ip access-group` *`access-list-name`* `{in | out}`
Options	*`access-list-name`* — IPv4 access-list name (up to 63 characters)
Command mode	INTERFACE
Usage	Use this command to create an ingress (in) or egress (out) access-list on an interface. The `no` version of this command removes the configuration.
Example	`sonic(conf-if-eth1/1/28)# ip access-group abcd in`
Releases	3.0 or later

ip access-list

Creates an IP access-list to filter based on an IP address.

Command	`ip access-list` *`access-list-name`*
Options	*`access-list-name`* — Name of an IPv4 access-list (up to 63 characters)
Command mode	CONFIGURATION

Usage Use this command to assign an access-list to match the route-map. The `no` version of this command removes the configuration.

Example

```
sonic(config)# ip access-list ac15
```

Releases 3.0 or later

ip address

Configures an IPv4 address to an interface.

Command `ip address addr [[gwaddr] gw_addr]`

Options

- *addr* — IPv4 address in A.B.C.D/mask format
- `gwaddr` *gw_addr* — (Optional) Gateway address

Command mode INTERFACE

Usage Use this command to configure an IPv4 address to an Ethernet, Management, VLAN, PortChannel, or Loopback interface. The `no` version of this command removes the IP address set for the interface.

Example

```
sonic(config)# interface Ethernet 1/1/1
sonic(conf-if-eth1/1/1)# ip address 10.1.1.0/24
```

Releases 3.0 or later

ip anycast-address

Configures an IPv4 static anycast gateway address for an interface.

Command `ip anycast-address anycast-addr`

Options *anycast-addr* — IPv4 anycast address in A.B.C.D/mask format

Command mode INTERFACE-VLAN

Usage Use this command to configure an IPv4 static anycast gateway address for a VLAN interface. The `no` version of this command removes the configuration.

Example

```
sonic(conf-if-Vlan10)# ip anycast-address 10.1.1.100/24
```

Releases 3.0 or later

ip anycast-mac-address

Configures an IPv4 MAC address for all static anycast gateway addresses.

Command `ip anycast-mac-address anycast-mac`

Options *anycast-mac* — Anycast MAC address in nn:nn:nn:nn:nn:nn format

Command mode INTERFACE-VLAN

Usage Use this command to configure an IPv4 MAC address for all static anycast gateway addresses. The `no` version of this command removes the configuration.

Example

```
sonic(conf-if-Vlan10)# ip anycast-mac-address 00:00:00:00:00:01
```

Releases 3.0 or later

ip igmp snooping

Configures or unconfigures IGMP snooping parameters on a VLAN.

Command

```
ip igmp snooping {[querier] | [fast-leave] | {[query-interval] query-interval-val} | {[last-member-query-interval] last-mem-query-interval-val} | {[query-max-response-time] query-max-response-val} | {[version] igmps-version-val} | {[mrouter] {interface mrouter-if-name}} | {[static-group] {group-addr {interface grp-if-name}}}}
```

Options

- `querier` — (Optional) Enables IGMP querier processing for the specified VLAN interface
- `fast-leave` — (Optional) Enables fast-leave snooping for the specified VLAN interface
- `query-interval` *query-interval-val* — (Optional) Query interval time in seconds (default 125)
- `last-member-query-interval` *last-mem-query-interval-val* — (Optional) Last memory query value in ms (default 1000)
- `query-max-response-time` *query-max-response-val* — (Optional) Query maximum response time in seconds (default 10)
- `version` *igmps-version-val* — (Optional) IGMP version; 1 or 2 (default 2)
- `mrouter` *mrouter-if-name* — (Optional) Interface name
- `static-group` *group-addr* — (Optional) IPv4 address in A.B.C.D format
- `interface` *grp-if-name* — (Optional) Group interface name

Command mode

INTERFACE-VLAN

Usage

Use this command to configure or unconfigure IGMP snooping on a VLAN. The IGMP querier periodically sends a general query to discover which multicast groups are active. A group must have at least one host to be active. By default, the periodic query messages are sent every 60 seconds. When the IGMP querier receives a leave message, it sends a group-specific query message to ensure if any other host in the network is interested in the multicast flow. By default, the group-specific query messages are sent every 1000 milliseconds. The maximum response time is the amount of time that the querier waits for a response to a query before taking action. When a host receives a query, it does not respond immediately, but rather starts a delay timer. The delay time is set to a random value between 0 and the maximum response time. The host sends a response when the timer expires; in IGMP version 2, if another host responds before the timer expires, the timer nullifies, and no response is sent. The querier advertises the maximum response time in the query. Lowering this value decreases leave latency but increases response burstiness because all host membership reports are sent before the maximum response time expires. Inversely, increasing this value decreases burstiness, but increases leave latency. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# ip igmp snooping
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# ip igmp snooping querier
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# ip igmp snooping fast-leave
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# no ip igmp snooping fast-leave
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# ip igmp snooping query-interval 20
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# no ip igmp snooping query-interval
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# ip igmp snooping last-member-query-interval 2000
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# no ip igmp snooping last-member-query-interval
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# ip igmp snooping query-max-response-time 12
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# no ip igmp snooping query-max-response-time
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# ip igmp snooping version 3
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# no ip igmp snooping version
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# ip igmp snooping mrouter interface Ethernet4
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# no ip igmp snooping mrouter interface Ethernet4
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# ip igmp snooping static-group 225.0.0.1 interface
PortChannel2
```

```
sonic(config)# interface Vlan 200
sonic(conf-if-Vlan200)# no ip igmp snooping static-group 225.0.0.1 interface
PortChannel2
```

Releases 3.0 or later

ip prefix-list

Creates a prefix-list to permit or deny route filtering from a specified network address.

Command

```
ip prefix-list prefix-name {{permit {ipv4-prefix {[ge] ge-min-prefix-length}
{[le] le-max-prefix-length}}} | {deny {ipv4-prefix {[ge] ge-min-prefix-length}
{[le] le-max-prefix-length}}}}
```

Options

- *prefix-name* — New IPv4 prefix-list name
- *ipv4-prefix* — IPv4 prefix-list to permit or deny in A.B.C.D/mask format
- ge *ge-min-prefix-length* — (Optional) Network address is greater than or equal to the range specified
- le *le-max-prefix-length* — (Optional) Network address is less than or equal to the range specified

Command mode CONFIGURATION

Usage Use this command to configure IP filtering in BGP route advertisements, and create an IPv4 prefix-list with permit or deny statements for matching network prefixes. The no version of this command removes the specified prefix-list.

Example

```
sonic(config)# ip prefix-list prflst656 permit 156.1.1.0/24
```

Releases 3.0 or later

ip unnumbered

Configures an IPv4 unnumbered interface from an Ethernet or PortChannel donor interface.

Command

```
ip unnumbered donor-interface
```

Options *donor-interface* — IPv4 interface name

Command mode INTERFACE

Usage Use this command to configure an IPv4 unnumbered interface at the interface level. The no version of this command removes the IPv4 unnumbered interface.

Example

```
sonic(config)# interface Ethernet 0
sonic(conf-if-Ethernet0)# ip unnumbered Loopback1
```

```
sonic(config)# interface PortChannel 1
sonic(conf-if-po1)# ip unnumbered Loopback1
```

Releases 3.0 or later

ip vrf

Creates a non-default VRF instance.

Command

```
ip vrf vrf-name
```

Options *vrf-name* — Name of the non-default VRF that you want to create (up to 15 characters)

Command mode CONFIGURATION

Usage Use this command to create a VRF instance to leak routes in one VRF instance to another using route targets. The no version of this command removes the non-default instance.

Example

```
sonic(config)# ip vrf vrf1
sonic(conf-vrf-vrf1)#
```

Releases 3.0 or later

ip vrf forwarding

Configures the interface forwarding table.

Command `ip vrf forwarding vrf-name`

Options *vrf-name* — VRF name instance (up to 15 characters)

Command mode INTERFACE

Usage Use this command to assign the Ethernet, PortChannel, VLAN, or Loopback interface to the source VRF instance. The `no` version of this command removes the configuration.

Example

```
sonic(config)# ip vrf forwarding vrf1
```

Releases 3.0 or later

ip vrf management

Enters Management interface VRF configuration mode.

Command `ip vrf management`

Options None

Command mode CONFIGURATION

Usage Use this command to assign the Management interface back to the default VRF instance. The `no` version of this command removes the configuration.

Example

```
sonic(config)# ip vrf management
sonic(conf-vrf-mgmt)#
```

Releases 3.0 or later

ipv6

Enters IPv6 configuration mode.

Command `ipv6`

Options None

Command mode CONFIGURATION

Usage Use this command to configure an IPv6 prefix-list. The `no` version of this command removes the configuration.

Example

```
sonic(config)# interface Ethernet 1/1/8
sonic(conf-if-eth1/1/8)# ipv6
```

Releases 3.0 or later

ipv6 address

Configures a global unicast IPv6 address on an interface.

Command `ipv6 address addr [[gwaddr] gw_addr]`

Options

- *addr* — IPv6 address in A::B/mask format
- `gwaddr` *gw_addr* — (Optional) Gateway address

Command mode INTERFACE

Usage Use this command to configure an IPv6 address for a physical, Loopback, PortChannel, VLAN, or Management interface. The `no` version of this command removes the IPv6 address on the interface.

Example

```
sonic(config)# interface Ethernet 1/1/8
sonic(conf-if-Eth1/1/8)# ipv6 address 2111:dddd:0eee::22/64
```

Releases 3.0 or later

ipv6 anycast-address

Configures an IPv6 static anycast gateway address for an interface.

Command `ipv6 anycast-address` *`anycast-addr`*

Options *`anycast-addr`* — IPv6 address in A::B/mask format

Command mode INTERFACE

Usage Use this command to configure an IPv6 static anycast gateway address for a VLAN interface. The `no` version of this command removes the configuration.

Example

```
sonic(config)# interface Vlan 10
sonic(conf-if-Vlan10)# ipv6 anycast-address
```

Releases 3.0 or later

ipv6 prefix-list

Builds an IPv6 prefix-list.

Command `ipv6 prefix-list` *`prefix-name`* {{permit {*`ipv6-prefix`* {[ge] *`ge-min-prefix-length`*} {[le] *`le-max-prefix-length`*}}} | {deny {*`ipv6-prefix`* {[ge] *`ge- min-prefix-length`*} {[le] *`le-max-prefix-length`*}}}}

Options

- *`prefix-name`* — New prefix-list name
- *`ipv6-prefix`* — IPv6 prefix-list to permit or deny
- *`ge-min-prefix-length`* — (Optional) Indicates the prefix-list is greater than or equal to the range specified
- *`le-max-prefix-length`* — (Optional) Indicates the prefix-list is less than or equal to the range specified

Command mode CONFIGURATION

Usage Use this command to create an IPv6 prefix-list to permit or deny route filtering from a specified prefix-list. The `no` version of this command removes the configuration.

Example

```
sonic(config)# ipv6 prefix-list TEST deny AB10::1/128 ge 10 le 30
```

Releases 3.0 or later

8

K, L, and M commands

Topics:

kdump enable

Enables or disables kernel core dump operation.

Command	`kdump enable`
Options	None
Command mode	CONFIGURATION
Usage	Use this command to enable or disable kdump configuration in the startup configuration file. These commands typically require a reboot to complete. The `no` version of this command disables the kdump operation.

Examples

```
sonic(config)# kdump enable
KDUMP configuration has been updated in the startup configuration
Kdump configuration changes will be applied after the system reboots
```

```
sonic(config)# no kdump
KDUMP configuration has been updated in the startup configuration
ALERT! A system reboot is highly recommended.
Kdump configuration changes will be applied after the system reboots
```

Releases 3.0 or later

kdump memory

Sets or resets the amount of memory reserved for kernel core dump files.

Command `kdump memory` *`kdump_memory`*

Options *`kdump_memory`* — Amount of memory reserved for kdump

Command mode CONFIGURATION

Usage Use this command to set or reset the default amount of memory reserved for kernel core dump files. These commands typically require a reboot to complete. The `no` version of this command resets the default value.

Examples

```
sonic(config)# kdump memory 512M
KDUMP configuration has been updated in the startup configuration
kdump updated memory will be only operational after the system reboots
```

```
sonic(config)# no kdump memory
```

Releases 3.0 or later

kdump num_dumps

Sets or resests the maximum number of kernel core files stored locally.

Command `kdump num_dumps` *`kdump_num_dumps`*

Options *`kdump_num_dumps`* — Maximum number of kdump files to store locally

Command mode CONFIGURATION

Usage Use this command to set or reset to default the maximum number of kdump files which can be stored locally. These commands typically require a reboot to complete. The `no` version of this command resets the default value.

Examples

```
sonic(config)# kdump num_dump 5
```

```
sonic(config)# no kdump num_dump
```

Releases 3.0 or later

keepalive-interval

Configures MCLAG session keepalive intervals.

Command `keepalive-interval` *`KA`*

Options *`KA`* — Keepalive time interval in seconds

Command mode CONFIGURATION

Usage Use this command to configure the time interval between keepalive messages sent to the neighbor routers. The `no` version of this command removes the configuration.

Example

```
sonic(config)# keepalive-interval 20
```

Releases 3.0 or later

link state track

Creates a link state tracking group.

Command `link state track grp-name [upstream]`

Options *grp-name* — Group name (up to 63 characters)

Command mode INTERFACE

Usage Use this command to create a link state tracking group for Ethernet, PortChannel, or VLAN interfaces. The `no` version of this command removes the link state tracking group.

Example

```
sonic(config)# interface Ethernet 4
sonic(conf-if-Ethernet4)# link state track track1
```

Releases 3.0 or later

listen

Enables peer listening and sets the prefix range for dynamic peers.

Command `listen {{range {addr {peer-group pgname}}} | {limit lmt-val}}`

Options

- *addr* — BGP neighbor IPv4 or IPv6 address in A.B.C.D/mask or A::B/mask format
- *pgname* — Peer-group name
- *lmt-val* — Maximum dynamic peer count (1 to 4294967295)

Command mode ROUTER-BGP

Usage Use this command to enable a passive peer session for listening, and create dynamic neighbors. BGP will accept connections from any peers in the specified prefix. Configuration from the specified peer-group is used to configure these peers. The `no` version of this command disables a passive peering session.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# listen range 192.168.0.0/16 peer-group PG_Ext limit
10
```

Releases 3.0 or later

local-as

Configures a local AS number for a BGP neighbor or neighbors in a peer-group.

Command `local-as asnum {[no-prepend] [replace-as]}`

Options

- *asnum* — Local AS number (1 to 4294967295)
- `no-prepend` — (Optional) Local AS values are not prepended to the AS_PATH attribute
- `replace-as` — (Optional) Globally-configured AS values are are not prepended to the AS_PATH attribute

Command modes

- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to configure a local AS number for a BGP neighbor or neighbors in a peer-group, and control how the local AS number is prepended to the AS_PATH of incoming and outgoing routes. Without modifiers, the specified local-as is prepended to the received AS_PATH when receiving routing updates from the peer, and prepended to the outgoing AS_PATH (after the process local AS) when transmitting local routes to the peer. If the `no-prepend` option is specified, the supplied local-as is not prepended to the received AS_PATH. If the `replace-as` option is specified, only the supplied local-as is prepended to the AS_PATH when transmitting local-route updates to this peer. The `no` version of this command deletes the local-as configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# local-as 65200 no-prepend
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# local-as 65200 non-prepend
```

Releases 3.0 or later

log-neighbor-changes

Enables logging for changes in neighbor status.

Command `log-neighbor-changes`

Options None

Command mode ROUTER-BGP

Usage Use this command to enable logging of neighbor up/down events along with reason code for down event. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# log-neighbor-changes
```

Releases 3.0 or later

map

Configures VNI-VLAN and VNI-VRF mappings.

Command `map vni {`*`vnid`* `{{vlan {`*`vid`* `{[count] [`*`numvid`*`]}}} | {vrf `*`vrf-name`*`}}}`

Options

- *vnid* — VNI value (1 to 16777215)
- *vid* — VLAN ID (1 to 4094)
- *numvid* — (Optional) Number of mappings
- *vrf-name* — VRF instance name (up to 63 characters)

Command mode INTERFACE-VXLAN-VTEP

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# interface Vxlan vtep1
sonic(conf-if-Vxlan-vtep1)# map vni 100 vlan 100 count 2
sonic(conf-if-Vxlan-vtep1)# map vni 100 vrf vrf1
```

Releases 3.0 or later

match as-path

Configures a routing policy to match criteria to an AS path.

Command `match as-path as-path-name`

Options *as-path-name* — Name of the established AS-PATH ACL (up to 140 characters)

Command mode ROUTE-MAP

Usage Use this command to configure a filter to match routes that have a specific AS path in their BGP path. The `no` version of this command deletes a match AS path filter.

Example

```
sonic(conf-route-map)# match as-path pathtest1
```

Releases 3.0 or later

match community

Configures a routing policy to match criteria to a BGP community.

Command `match community community-name`

Options *community-name* — Name of the configured community list

Command mode ROUTE-MAP

Usage Use this command to configure a filter to match routes that have a specific COMMUNITY attribute in their BGP path. The `no` version of this command deletes the community match filter.

Example

```
sonic(conf-route-map)# match community commlist1
```

Releases 3.0 or later

match ext-community

Configures a routing policy to match criteria to a BGP extended community.

Command `match ext-community community-name`

Options *community-name* — Name of the configured ext-community list

Command mode ROUTE-MAP

Usage Use this command to configure a filter to match routes that have a specific EXTCOMMUNITY attribute in their BGP path. The `no` version of this command deletes the ext-community match filter.

Example

```
sonic(conf-route-map)# match ext-community extcommlist1
```

Releases 3.0 or later

match interface

Configures a routing policy to match criteria to an interface.

Command `match interface {{Ethernet phy-if-name} | {PortChannel lag-id} | {Vlan vlan-id}}`

Options

- `Ethernet phy-if-name` — Interface name as the next-hop interface
- `PortChannel lag-id` — PortChannel number as the next-hop interface (1 to 128)
- `Vlan vlan-id` — VLAN number as the next-hop interface (1 to 4093)

Command mode ROUTE-MAP

Usage Use this command to configure a filter to match routes whose next-hop is the configured interface. The `no` version of this command deletes the match.

Example

```
sonic(conf-route-map)# match interface Ethernet 1/1/1
sonic(conf-if-eth1/1/1/)#
```

Releases 3.0 or later

match ip address prefix-list

Configures routing policy match criteria to an IPv4 prefix-list.

Command `match ip address prefix-list` *`prefix-list-name`*

Options *`prefix-list-name`* — Name of the configured IPv4 prefix-list to match against

Command mode ROUTE-MAP

Usage Use this command to configure a filter to match routes based on a specified IPv4 prefix-list name. The `no` version of this command removes the configuration.

Example

```
sonic(conf-route-map)# match ip address prefix-list test100
```

Releases 3.0 or later

match ip next-hop prefix-list

Configures a routing policy to match criteria to a next-hop prefix-list.

Command `match ip next-hop prefix-list` *`match-hop`*

Options *`match-hop`* — Name of the configured IPv4 prefix-list to match against

Command mode ROUTE-MAP

Usage Use this command to configure a filter to match based on the next-hop IPv4 addresses specified in IP prefix lists. The `no` version of this command removes the configuration.

Example

```
sonic(conf-route-map)# match ip next-hop prefix-list test100
```

Releases 3.0 or later

match ipv6 address prefix-list

Configures a routing policy to match criteria to an IPv6 prefix-list.

Command `match ipv6 address prefix-list` *`prefix-list-name`*

Options *`prefix-list-name`* — Prefix-list name to match against

Command mode ROUTE-MAP

Usage Use this command to configure a filter to match routes based on a specified IPv6 prefix-list name. The `no` version of this command removes the configuration.

Example

```
sonic(conf-route-map)# match ipv6 address prefix-list test100
```

Releases 3.0 or later

match local-preference

Configures a routing policy to match criteria to local-preference.

Command `match local-preference match-loc`

Options *match-loc* — Local-preference to match against

Command mode ROUTE-MAP

Usage Use this command to configure a routing policy to match criteria to a local-preference value. The `no` version of this command removes the configuration.

Example

```
sonic(conf-route-map)# match local-preference 10000
```

Releases 3.0 or later

match metric

Configures a filter to match on a specific value.

Command `match metric match-met`

Options `metric` *match-met* — Value to match the route metric against (0 to 4294967295)

Command mode ROUTE-MAP

Usage Use this command to configure a routing policy to match criteria to a metric. The `no` version of this command removes the configuration.

Example

```
sonic(conf-route-map)# match metric 429132
```

Releases 3.0 or later

match origin

Configures a filter to match routes based on the origin attribute of BGP.

Command `match origin {egp | igp | incomplete}`

Options

- `egp`— Match only remote EGP routes
- `igp`— Match only local IGP routes
- `incomplete`— Match on unknown routes learned through some other means

Command mode ROUTE-MAP

Usage Use this command to configure a filter to match routes based on the origin attribute of BGP. The `no` version of this command removes the configuration.

Example

```
sonic(config)# route-map bgp
sonic(conf-route-map)# match origin egp
```

Releases 3.0 or later

match peer

Configures a routing policy to match criteria to a peer IP.

Command `match peer {`*match-peer* `| {Ethernet` *phy-if-name*`} | {PortChannel` *lag-id*`} | {Vlan` *vlan-id*`}}`

Options

- `peer match-peer` — Peer IPv4 or IPv6 address in A.B.C.D or A::B format
- `Ethernet phy-if-name` — Physical interface name
- `PortChannel lag-id` — LAG ID
- `Vlan vlan-id` — VLAN ID

Command mode ROUTE-MAP

Usage Use this command to configure a routing policy to match criteria to an IPv4 or IPv6 peer. The `no` version of this command removes the configuration.

Example

```
sonic(conf-route-map)# match peer 10.1.1.100 Vlan4
```

Releases 3.0 or later

match source-protocol

Configures the source protocol to match.

Command `match source-protocol {bgp | ospf | ospf3 | static | connected}`

Options None

Command mode ROUTE-MAP

Usage Use this command to match BGP, OSPF, OSPFv3, static, or connected source protocols. The `no` version of this command removes the configuration.

Example

```
sonic(conf-router-map)# match source-protocol bgp
```

Releases 3.0 or later

match tag

Creates a filter to redistribute only routes that match a specific tag value.

Command `match tag match-tag`

Options `match-tag` — Tag value to match with the tag number (1 to 4294967295)

Command mode ROUTER-BGP

Usage Use this command to redistribute routes based on specified tag values. The `no` version of this command deletes the match.

Example

```
sonic(conf-router-bgp)# match tag 656442
```

Releases 3.0 or later

maximum-paths

Configures the maximum number of equal-cost multipaths (ECMP) for load sharing.

Command `maximum-paths paths`

Options `paths` — Maximum ECMP routes

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to configure BGP to control the maximum number of equal cost multipath routes to eBGP destinations. This command is per address-family. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family ipv4 unicast
sonic(conf-router-bgp-af)# maximum-paths 32
```

Releases 3.0 or later

maximum-paths ibgp

Configures the maximum number of equal-cost multipath for internal BGP (iBGP) routes.

Command `maximum-paths ibgp` *`ipaths`* `[equal-cluster-length]`

Options

- *`ipaths`* — ECMP maximum routes
- `equal-cluster-length` — (Optional) Equal cluster lengths

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to configure BGP to control the maximum number of equal cost multipath routes to iBGP destinations. This command is per address-family. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family ipv4 unicast
sonic(conf-router-bgp-af)# maximum-paths ibgp 32
```

Releases 3.0 or later

maximum-prefix

Configures the maximum number of prefixes to accept from this BGP neighbor, peer, or neighbors in a peer-group.

Command `maximum-prefix` *`max-prefix-val`* `{[`*`threshold-val`*`] | {[warning-only] | {[restart] interval}}}`

Options

- *`max-prefix-val`* — Maximum prefix value (1 to 4294967295)
- *`threshold-val`* — (Optional) Threshold value (1 to 100; default 75)
- `warning-only` — (Optional) Sends a warning log messages when the maximum limit is exceeded
- `restart` *`interval`* — (Optional) Restart value

Command modes

- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to set the upper limit on the number of BGP prefixes to accept from this neighbor. If you configure this command and the neighbor receives more prefixes than the configuration allows, the neighbor goes down. The neighbor remains down until you use `clear ip bgp` for the neighbor, or the peer-group to which the neighbor belongs. This command has optional parameters for warning when a threshold is reached and restarting BGP neighborship when the maximum prefix limit has exceeded. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# maximum-prefix 2000 80 warning-only
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Int
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# maximum-prefix 2000 80 warning-only
```

Releases 3.0 or later

max-med

Configures BGP to advertise routes with maximum MED value under a given condition.

Command `max-med {{on-startup stime} | administrative} [maxmedval]`

Options
- `on-startup stime` — Startup time
- `maxmedval` — (Optional) Maximum MED value

Command mode ROUTER-BGP

Usage Use this command to instruct BGP to advertise routes with the maximum MED value. Set the condition under which routes with max MED value will be sent. Options include during the startup for a prespecified number of seconds, and he other is permanently (administrative). You can also specify the value for max MED. The `no` version of this command removes the configuration.

Example
```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# max-med on-startup 300 2000
```

Releases 3.0 or later

mclag

Configures an MCLAG interface.

Command `mclag domain_id`

Options `domain_id` — MCLAG interface ID

Command mode INTERFACE-PORT-CHANNEL

Usage Use this command to configure an MCLAG interface. The `no` version of this command removes the configuration.

Example
```
sonic(conf-if-po10)# mclag 100
```

Releases 3.0 or later

mclag domain

Configures the MCLAG domain ID.

Command `mclag domain mclag-domain-id`

Options `domain mclag-domain-id` — MCLAG domain ID number

Command mode CONFIGURATION

Usage Use this command to configure the MCLAG domain ID. The `no` version of this command removes the configuration.

Example
```
sonic(config)# mclag domain mlag20
```

Releases 3.0 or later

mclag-seperate-ip

Configures separate IPs on a VLAN interface for L3 protocol support over MCLAG.

Command `mclag-seperate-ip`

Options None

Command mode INTERFACE

Usage	Use this command to configure separate IPs on a VLAN interface for L3 protocol support over MCLAG. The `no` version of this command removes the configuration.
Example	`sonic(conf-if-Vlan10)# mclag-seperate-ip`
Releases	3.0 or later

mirror-session

Configures a mirror-session.

Command	`mirror-session` *`session-name`*
Options	*`session-name`* — Mirror-session name
Command mode	CONFIGURATION
Usage	Use this command to configure a mirror-session. The `no` version of this command removes the configuration.
Example	`sonic(config)# mirror-session sess10`
Releases	3.0 or later

mtu

Configures maximum transmission unit (MTU) on an interface.

Command	`mtu` *`mtu`*
Options	*`mtu`* — Interface; physical, VLAN, PortChannel, or Management
Command mode	INTERFACE
Usage	Use this command to configure MTU on a physical, VLAN, PortChannel, or Management interface. The `no` version of this command removes the MTU on an interface.
Example	`sonic(config)# interface Ethernet 4` `sonic(conf-if-Ethernet4)# mtu 20000`
Releases	3.0 or later

9

N, O, and P commands

Topics:

nat

Configures network address translation (NAT).

Command	`nat`
Options	None
Command mode	CONFIGURATION
Usage	Use this command to configure NAT. The `no` version of this command removes the configuration.
Example	`sonic(config)# nat` `sonic(conf-nat)#`
Releases	3.0 or later

nat-zone

Configures a NAT zone on a physical, Loopback, PortChannel, or VLAN interface.

Command `nat-zone zone`

Options *zone* — Zone number (0 to 3)

Command mode INTERFACE

Usage Use this command to configure a NAT zone on Layer 3 (L3) interfaces so that NAT address translation is performed on packets when a packet transverses a zone on a configured interface. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# interface Ethernet 4
sonic(conf-if-Vlan5)# ip address 20.20.20.20/24
sonic(conf-if-Ethernet4)# nat-zone 1
```

```
sonic(config)# interface Loopback 1
sonic(conf-if-lo1)# ip address 10.10.10.10/32
sonic(conf-if-lo1)# nat-zone 2
```

```
sonic(config)# interface PortChannel 2
sonic(conf-if-po2)# ip address 25.25.25.25/24
sonic(conf-if-po2)# nat-zone 1
```

```
sonic(config)# interface Vlan 5
sonic(conf-if-Vlan5)# ip address 23.23.23.23/24
sonic(conf-if-Vlan5)# nat-zone 1
```

Releases 3.0 or later

neigh_suppress

Enables ARP and ND suppression on a VLAN interface.

Command `neigh_suppress`

Options None

Command mode INTERFACE

Usage Use this command to enable ARP and ND suppression on a VLAN interface. The `no` version of this command removes the configuration.

Example

```
sonic(conf-int-vlan10)# neigh_suppress
```

Releases 3.0 or later

neighbor

Creates a remote IP or unnumbered peer, and enters into neighbor configuration mode.

Command `neighbor {ip | {interface {Ethernet | PortChannel | Vlan}}}`

Options

- *ip* — IPv4 or IPv6 address of the neighbor in A.B.C.D or A::B format
- `interface` — Ethernet, PortChanel, or Vlan Interface that connects to an unnumbered neighbor

Command mode BGP-NEIGHBOR

Usage Use this command to create an IPv4 or IPv6 BGP neighbor. Enter the neighbor's IPv4 or IPv6 address directly, or you can optional enter an interface name for an unnumbered BGP neighbor. The `no` version of this command disables the BGP neighbor configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)#
```

Releases 3.0 or later

network

Configures a network as local to this AS and adds it to the BGP routing table.

Command `network prefix {[backdoor] {[route-map] route-map-name}}`

Options

- *prefix* — IPv4 or IPv6 address and prefix number to the network in A.B.C.D/mask or A::B/mask format
- `backdoor` — (Optional) Backdoor route-map
- `route-map` *route-map-name* — (Optional) Name of the established route-map

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to enable routing of an IPv4 or IPv6 network to announce using BGP. This command can be used to statically inject routes into BGP. Use `route-map` to modify or set the various attributes of the route. The `no` version of this command removes the network.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family ipv4 unicast
sonic(conf-router-bgp-af)# network 10.10.0.0/16
```

Releases 3.0 or later

network import-check

Configure BGP to check if a BGP network route exists in the local route table before advertising the network.

Command `network import-check`

Options None

Command mode ROUTER-BGP

Usage Use this command to configure BGP to check if a BGP network route exists in the local route table before advertising the network. By default, BGP networks are advertised to neighbors irrespective of if the same route exists in local route table or not. This behavior may lead to data traffic blackholing. Use this command to place a restriction on BGP networks to get advertised only if a corresponding route from an internal gateway protocol (IGP) exists in local route table. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# network import-check
```

Releases 3.0 or later

next-hop-self

Disables the next-hop calculation for a BGP neighbor, or neighbors in a peer-group.

Command `next-hop-self [force]`

Options `force` — (Optional) Forces the next-hop attribute

Command modes

- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to disable BGP next-hop attribute computation and override the next-hop by the sender's address. This command influences next-hop processing of eBGP routes to iBGP peers. The `no` version of this command disables the next-hop calculation.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# next-hop-self
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# next-hop-self
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family l2vpn evpn
sonic(conf-router-bgp-neighbor-af)# next-hop-self
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family l2vpn evpn
sonic(conf-router-bgp-pg-af)# next-hop-self
```

Releases 3.0 or later

override-capability

Configures BGP to override the result of capability negotiation with local configuration, and ignore the remote peer's capability value.

Command `override-capability`

Options None

Command modes

- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to ignore the negotiated capability parameters with a BGP neighbor or neighbors in a peer-group, and instead use the locally configured parameters. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# override-capability
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# override-capability
```

Releases 3.0 or later

passive

Configures a BGP neighbor as passive.

Command `passive`

Options None

Command modes

- BGP-NEIGHBOR

- BGP-PEER-GROUP

Usage Use this command to configure a BGP neighbor or neighbor in a peer-group as passive. BGP neighbors will not initiate a session, and will listen to any incoming BGP session. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# passive
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# passive
```

Releases 3.0 or later

password

Configures a password for message digest 5 (MD5) authentication on the TCP connection between two neighbors.

Command `password String [encrypted]`

Options

- *String* — MD5 password (16-byte)
- `encrypted` — (Optional) Indicates if the password should be encrypted

Command modes

- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to configure an MD5 password to be used with the TCP socket connection to the remote peer. This command is for security purposes. When a password is configured for a BGP neighbor or peer-group, the sender will include a 16-byte MD5 digest in the TCP header of BGP message. The receiver validates the digest before accepting the BGP message. The `no` version of this command disables authentication.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# password jackandjillwentupthehill
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# password ilovebeansbecausetheyaremean
```

Releases 3.0 or later

peer

Configures an IPv4 or IPv6 single-hop or multi-hop bidirectional forwarding detection (BFD) peer.

Command `peer {peer_ipv4 | peer_ipv6} [vrf] vrfname [multihop] [local-address] {local_ipv4 | local_ipv6} [interface] interfacename`

Options

- *peer_ipv4* — Peer IPv4 address in A.B.C.D format
- *peer_ipv6* — Peer IPv6 address in A::B format
- `vrf` *vrfname* — VRF instance name
- *local_ipv4* — Local IPv4 address in A.B.C.D format
- *local_ipv6* — Local IPv6 address in A::B format
- `interface` *interfacename* — Interface name

Command mode CONFIGURATION-BFD

Usage Use this command to configure an IPv4 or IPv6 single-hop and multi-hop bidirectional forwarding detection (BFD) peer. A single-hop BFD peer interface must be configured, and for a multi-hop BFD peer a local address must be configured. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# bfd
sonic(conf-bfd)# peer 192.168.0.5 interface Ethernet0
```

```
sonic(config)# bfd
sonic(conf-bfd)# peer 192.168.0.5 interface Ethernet0 vrf Vrf1
```

```
sonic(config)# bfd
sonic(conf-bfd)# peer 192.168.0.2 multihop local-address 192.168.0.3
```

```
sonic(config)# bfd
sonic(conf-bfd)# peer 192.168.0.2 multihop local-address 192.168.0.3 vrf Vrf1
```

Releases 3.0 or later

peer-group

Creates a BGP peer-group, and assigns a BGP neighbor to a peer-group.

Command `peer-group *template-str*`

Options *`template-str`* — Peer group name

Command modes

- BGP-PEER-GROUP
- BGP-NEIGHBOR

Usage Use this command to create a BGP peer-group, and assign a BGP neighbor to a peer-group to inherit parameters from the peer-group. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)#
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# peer-group PG_Ext
```

Releases 3.0 or later

peer-ip

Configures a MCLAG peer IPv4 address.

Command `peer-ip *PIP*`

Options *`PIP`* — Peer IPv4 address in A.B.C.D format

Command mode MCLAG-DOMAIN

Usage Use this command to configure a MCLAG peer IPv4 address. The `no` version of this command removes the configuration.

Example

```
sonic(conf-mclag)# peer-ip 10.10.1.100
```

Releases 3.0 or later

peer-link

Configures a MCLAG peer on an Ethernet or PortChannel interface.

Command `peer-link {{Ethernet PLK} | {PortChannel PLK}}`

Options
- `Ethernet PLK` — Ethernet peer link
- `PortChannel PLK` — PortChannel peer link

Command mode MCLAG-DOMAIN

Usage Use this command to configure a MCLAG peer on an Ethernet or PortChannel interface. The `no` version of this command removes the configuration.

Example

```
sonic(conf-mclag)# peer-link PortChannel po1
```

Releases 3.0 or later

pool

Creates a network address translation (NAT) pool.

Command `pool pool-name global-ip-range [global-port-range]`

Options
- *pool-name* — NAT pool name
- *global-ip-range* — Global IP range
- *global-port-range* — (Optional) Global port range

Command mode CONFIGURATION-NAT

Usage This command create a NAT pool. The `no` version of this command removes the configuration.

Example

```
sonic(conf-nat)# pool pool1
```

Releases 3.0 or later

port

Configures a TCP port for a BGP neighbor, or neighbors in a peer-group.

Command `port tcpport`

Options *tcpport* — TCP port number

Command modes
- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to set a specific TCP port for a BGP neighbor or neighbors in a BGP peer group. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# port 61356
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# port 65001
```

Releases 3.0 or later

prefix-list

Configures a prefix-list for a BGP neighbor or peer-group.

Command `prefix-list pname {in | out}`

Options *pname* — Prefix-list name to filter inbound and/or outbound

Command modes
- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to define a policy (route filtering) for a BGP neighbor or BGP peer-group in an outbound or/and inbound direction. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# prefix-list pl_allow_remote in
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# prefix-list pl_allow_remote in
```

Releases 3.0 or later

ptp announce-timeout

Configures the PTP announce receipt timeout.

Command `ptp announce-timeout ptp_announce_timeout`

Options *ptp_announce_timeout* — PTP amount timeout; default 3

Command mode CONFIGURATION

Usage Use this command to configure the PTP announce receipt timeout value.

Example

```
sonic(config)# ptp announce-timeout 2
Success
```

Releases 3.0 or later

ptp domain

Configures the PTP domain.

Command `ptp domain ptp_domain`

Options *ptp_domain* — PTP domain number

Command mode CONFIGURATION

Usage Use this command to configure the PTP domain.

Example

```
sonic(config)# ptp domain 1
Success
```

Releases 3.0 or later

ptp domain-profile

Configures the PTP domain profile.

Command `ptp domain-profile ptp_domain_profile`

Options *ptp_domain_profile* — PTP domain profile; default, g8271.1, or g8275.2

Command mode CONFIGURATION

Usage Use this command to configure the PTP domain profile type.

Example

```
sonic(config)# ptp domain-profile default
Success
```

Releases 3.0 or later

ptp ipv6-scope

Configures the PTP IPv6 multicast address range.

Command `ptp ipv6-scope ptp_ipv6_scope`

Options *ptp_ipv6_scope* — PTP IPv6 address in hexadecimal range (0x0 to 0xf)

Command mode CONFIGURATION

Usage Use this command to configure the PTP IPv6 multicast address range.

Example

```
sonic(config)# ptp ipv6-scope 0xe
Success
```

Releases 3.0 or later

ptp log-announce-interval

Configures the PTP log announce interval.

Command `ptp log-announce-interval ptp_announce_interval`

Options *ptp_announce_interval* — PTP log announce interval specified as a power of two seconds; default 1 (2 seconds)

Command mode CONFIGURATION

Usage Use this command to configure the PTP log announce interval value. The interval value should be the same in the whole domain.

Example

```
sonic(config)# ptp log-announce-interval 0
Success
```

Releases 3.0 or later

ptp log-min-delay-req-interval

Configures the PTP log minimum delay request interval.

Command `ptp log-min-delay-req-interval ptp_delay_request_interval`

Options *ptp_delay_request_interval* — PTP delay request interval specified as a power of two seconds; default 0 (1 second)

Command mode CONFIGURATION

Usage Use this command to configure the PTP log minimum delay request interval.

Example

```
sonic(config)# ptp log-min-delay-req-interval 0
Success
```

Releases 3.0 or later

ptp log-sync-interval

Configures the PTP log sync interval value.

Command `ptp log-sync-interval ptp_sync_interval`

Options *ptp_sync_interval* — PTP sync interval in power of two seconds; default is 0 (1 second)

Command mode CONFIGURATION

Usage Use this command to configure the PTP log sync interval value.

Example

```
sonic(config)# ptp log-sync-interval 0
Success
```

Releases 3.0 or later

ptp mode

Configures the PTP clock mode type.

Command `ptp mode mode_type`

Options *mode_type* — PTP clock mode type; boundary-clock, peer-to-peer-transparent-clock, end-to-end-transparent-clock, or disable

Command mode CONFIGURATION

Usage Use this command to configure the PTP clock mode type.

Example

```
sonic(config)# ptp mode boundary-clock
Success
```

Releases 3.0 or later

ptp network-transport

Configures the PTP network transport type.

Command `ptp network-transport ptp_network_transport_type ptp_master_slave`

Options

- *ptp_network_transport_type* — PTP network transport type; l2, ipv4, or ipv6
- *ptp_master_slave* — PTP master slave type; unicast or multicast

Command mode CONFIGURATION

Usage Use this command to configure the PTP network transport type, and PTP master slave type.

Example

```
sonic(config)# ptp network-transport ipv4 unicast
Success
```

Releases 3.0 or later

ptp port add

Adds a PTP port.

Command `ptp port add {Ethernet ptp_port_number}`

Options `Ethernet ptp_port_number` — Ethernet port number

Command mode CONFIGURATION

Usage Use this command to add an Ethernet PTP port.

Example

```
sonic(config)# ptp port add Ethernet 64
Success
```

Releases 3.0 or later

ptp port del

Deletes a configured PTP port.

Command `ptp port del {Ethernet ptp_port_number}`

Options `Ethernet ptp_port_number` — Ethernet port number

Command mode CONFIGURATION

Usage Use this command to delete an Ethernet configured PTP port.

Example

```
sonic(config)# ptp port del Ethernet 64
Success
```

Releases 3.0 or later

ptp port master-table

Adds or deletes a master IP/MAC from the master table for the designated slave port.

Command `ptp port master-table {Ethernet ptp_port_number} {{add {l3_ip | mac}} | {del {l3_ip | mac}}}`

Options

- `Ethernet ptp_port_number` — PTP port number
- `l3_ip` — Layer 3 IPv4 or IPv6 address in A.B.C.D or A::B format
- `mac` — MAC address in nn:nn:nn:nn:nn:nn format

Command mode CONFIGURATION

Usage Use this command to add or delete a master IP or MAC address from the master table for the designated slave port.

Example

```
sonic(config)# ptp port master-table Ethernet 64 add 10.1.1.1
Success
```

```
sonic(config)# ptp port master-table Ethernet 64 del 10.1.1.1
Success
```

Releases 3.0 or later

ptp priority1

Configures a PTP priority1 value.

Command `ptp priority1 ptp_priority1`

Options `priority1 ptp_priority1` — Priority1 value

Command mode CONFIGURATION

Usage Use this command to configure a PTP priority1 value.

Example

```
sonic(config)# ptp priority1 128
Success
```

Releases 3.0 or later

ptp priority2

Configures a PTP priority2 value.

Command `ptp priority2 ptp_priority2`

Options `priority2 ptp_priority2` — Priority2 value

Command mode CONFIGURATION

Usage Use this command to configure a PTP priority2 value.

Example

```
sonic(config)# ptp priority2 128
Success
```

Releases 3.0 or later

ptp two-step

Enables or disables PTP two-step mode.

Command `ptp two-step ptp_two_step`

Options `two-step ptp_two_step` — Enable or disable

Command mode CONFIGURATION

Usage Use this command to enable or disable PTP two-step mode.

Example

```
sonic(config)# ptp two-step enable
Success
```

Releases 3.0 or later

10

R commands

Topics:

radius-server auth-type

Configures the global RADIUS server authentication type.

Command

```
radius-server auth-type [chap | pap | mschapv2]
```

Options

- `chap` — Enables `chap` for the authentication type (default)
- `pap` — Enables `pap` for the authentication type
- `mspapv2` — Enables `mschap` for the authentication type

Command mode CONFIGURATION

Usage Use this command to configure a default RADIUS server authentication type that is used for remote user access. The authentication type is used to encrypt or decrypt data that is sent and received between the switch and the RADIUS server. If you have not configured a server-specific authtype, this global value is used for that RADIUS server. The `no` version of this command resets the configuration to the default.

Example

```
sonic(config)# radius-server auth-type pap
```

Releases 3.0 or later

radius-server host

Configures a RADIUS server and the key used to authenticate the switch on the server.

Command

```
radius-server host host_ip [auth-port] port_no [timeout] seconds [retransmit]
attempts [key] secret_key [auth-type] authentication_type [priority]
port_priority [vrf] vrf_instance
```

Options

- `host host_ip` — IPv4 or IPv6 host in A.B.C.D or A::B format
- `auth-port port_no` — (Optional) Port number
- `timeout seconds` — (Optional) Timeout in seconds
- `retransmit attemps` — (Optional) Retransmit attempts
- `key secret_key` — (Optional) Shared secret key (up to 65 characters)
- `auth-type authentication_type` — (Optional) Authentication type; pap, chap, or mschapv2
- `priority port_priority` — (Optional) Priority (1 to 64)
- `vrf vrf_instance` — VRF instance (up to 32 characters; mgmt or prefixed by Vrf_)

Command mode CONFIGURATION

Usage Use this command to configure a RADIUS server host. The authentication key must match the key configured on the RADIUS server, and you cannot enter spaces in the key. You can configure global settings for the timeout and retransmit attempts allowed on RADIUS servers. The `no` version of this command removes the RADIUS server configuration.

Example

```
sonic(config)# radius-server host 100.1.1.200
```

Releases 3.0 or later

radius-server key

Configures the global authentication key for the RADIUS server.

Command `radius-server key secret-key`

Options `secret-key` — Authentication key (up to 65 characters)

Command mode CONFIGURATION

Usage Use this command to modify the global value for the RADIUS server authentication key. If you have not configured a server-specific authentication key, this global value is used for that RADIUS server. The authentication key can include all printable ASCII characters with a few exceptions (#, SPACE, and COMMA), and up to 65 characters. The `no` version of this command removes the configuration.

Example

```
sonic(config)# radius-server key abcabcabcabcabcabcabcabcabcabcabc
abcabcabcabcabcabcabcabcabcabc
```

Releases 3.0 or later

radius-server retransmit

Configures the number of authentication attempts allowed on the RADIUS server.

Command `radius-server retransmit retries`

Options `retries` — Number of retry attempts (0 to 100; default 3)

Command mode CONFIGURATION

Usage Use this command to globally configure the number of retransmit attempts allowed for authentication requests on RADIUS servers. If you have not configured a server-specific retransmit, this global value is used for that server. The `no` version of this command removes the configuration.

Example

```
sonic(config)# radius-server retransmit 10
```

Releases 3.0 or later

radius-server source-ip

Configures the global source IP address for the RADIUS server.

Command `radius-server source-ip ip_address`

Options `source-ip ip_address` — IPv4 or IPv6 address in A.B.C.D or A::B format

Command mode CONFIGURATION

Usage Use this command to configure the global source IP address for the RADIUS server. The `no` version of this command removes the configuration.

Example

```
sonic(config)# radius-server source-ip 100.1.1.10
```

Releases 3.0 or later

radius-server timeout

Configures the timeout used to resend RADIUS authentication requests.

Command `radius-server timeout seconds`

Options *seconds* — Retransmission time (1 to 60 seconds; default 5)

Command mode CONFIGURATION

Usage Use this command to globally configure the timeout value used on RADIUS servers. If you have not configured a server-specific timeout, this global value is used for that server. The `no` version of this command removes the configuration.

Example

```
sonic(config)# radius-server timeout 10
```

Releases 3.0 or later

rd

Specifies the route-distinguisher to attach to routes exported from the current VRF into EVPN.

Command `rd rdvalue`

Options *rdvalue* — Route-distinguisher

Command modes

- BGP-ADDRESS-FAMILY
- BGP-ADDRESS-FAMILY-VNI

Usage Use this command to specify the route-distinguisher to attach to routes exported from the current VRF into EVPN. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100 vrf Vrf1
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# rd 11:11
```

```
sonic(config)# router bgp 100 vrf Vrf1
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# vni 100
sonic(conf-router-bgp-af-vni)# rd 11:11
```

Releases 3.0 or later

read-quanta

Configures the maximum number of BGP packets to read from peer socket in one cycle.

Command `read-quanta rdval`

Options *rdval* — Maximum number of packets to read

Command mode ROUTER-BGP

Usage Use this command to configure the maximum number of BGP packets to read from the peer socket in one cycle of I/O. BGP packets are read off the wire, one loop at a time. This setting controls how many iterations the loop runs for. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# read-quanta 6
```

Releases 3.0 or later

receive-interval

Configures the packet receive interval from a bidirectional forwarding detection (BFD) peer.

Command `receive-interval receive_interval`

Options *receive_interval* — Receive interval in ms (default 300)

Command mode BFD-PEER

Usage Use this command to set the desired packet received interval from a BFD peer. The `no` version of this command removes the configuration.

Example

```
sonic(config)# bfd
sonic(conf-bfd)# peer 192.168.0.5 interface Ethernet0
sonic(conf-bfd-peer)# receive-interval 200
```

Releases 3.0 or later

redistribute

Redistributes connected, static, and OSPF routes in BGP.

Command `redistribute {connected | static | ospf} [route-map] route-map-name [metric] metvalue`

Options

- `connected` — Redistributes routes from physically connected interfaces
- `static` — Redistributes manually configured routes
- `ospf` — Redistributes OSPF internal routes
- `route-map route-map-name` — (Optional) Route-map name
- `metric metvalue` — (Optional) Default metric value for redistributed routes

Command mode BGP-ADDRESS-FAMILY

Usage Use this command to redistribute information from another protocol to BGP. You can provide a route-map while enabling redistribution of routes to control routes that go into BGP. You can also use a metric option to set the default metric for the redistributed routes. The `no` version of this command disables the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# address-family ipv4 unicast
sonic(conf-router-bgp-af)# redistribute connected
```

Releases 3.0 or later

remote-as

Adds a remote AS to the specified BGP neighbor or peer-group.

Command `remote-as {internal | external | as-num-dot}`

Options
- `internal` — Tag neighbor or peer-group as internal (iBGP)
- `external` — Tag neighbor or peer-group as external (eBGP)
- *`as-num-dot`* — Remote AS number (1 to 65535 for 2 byte; 1 to 4294967295 for 4 byte)

Command modes
- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to add a remote AS for a BGP neighbor or BGP peer-group. The `no` version of this command deletes the remote AS.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 30.30.30.3
sonic(conf-router-bgp-neighbor)# remote-as 65100
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# remote-as 65200
```

Releases 3.0 or later

remove-private-AS

Removes private AS numbers from receiving outgoing updates.

Command `remove-private-AS [all] [replace-AS]`

Options
- `all` — (Optional) Replace all private ASNs
- `replace-AS` — (Optional) Replace private ASN with local ASN

Command modes
- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command at the boundary of your BGP network to remove the internal/private ASNs from outbound route updates. You can optionally replace private ASN by local ASN. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# remove-private-AS all
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# remove-private-AS all
```

Releases 3.0 or later

route-map

Applies an established route-map to either incoming or outgoing routes of a BGP neighbor or peer-group.

Command `route-map` *`route-name-str`* `{in | out}`

Options

- *route-name-str* — Name of the configured route-map
- `in` — Attaches the route-map as the inbound policy
- `out` — Attaches the route-map as the outbound policy

Command modes

- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage

Use this command to configure policy for a BGP neighbor or peer-group. The policy can be applied in an inbound or outbound direction. The policy dictates if a subset of routes need to be filtered out or/and if attributes of some routes needs to be modified. The `no` version of this command deletes the route-map.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# route-map rmap_filter_intra_routes in
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# route-map RM_Blk_192 in
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family l2vpn evpn
sonic(conf-router-bgp-neighbor-af)# route-map rmap_filter_intra_routes in
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Ext
sonic(conf-router-bgp-pg)# address-family l2vpn evpn
sonic(conf-router-bgp-pg-af)# route-map RM_Blk_192 in
```

Releases

3.0 or later

route-map delay-timer

Sets the route-map delay interval.

Command

`route-map delay-timer` *delaytm*

Options

`delay-timer` *delaytm* — Delay timer value in seconds (0 to 600; no default)

Command mode

ROUTER-BGP

Usage

Use this command to set the interval in seconds to wait before processing a route-map change. You can apply a route map to filter the exchange of incoming and outgoing BGP IPv4 or IPv6 routes. Configure the time interval (in seconds) to wait before processing received filtered routes in the BGP routing table. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# route-map delay-timer 60
```

Releases

3.0 or later

route-reflector allow-outbound-policy

Sets the outbound policy for route-reflector neighbors.

Command

`route-reflector allow-outbound-policy`

Options

None

Command mode ROUTER-BGP

Usage Use this command to set the outbound policy for route-reflector neighbors. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# route-reflector allow-outbound-policy
```

Releases 3.0 or later

route-reflector-client

Configures a BGP neighbor or neighbors in a peer-group as a member of a route-reflector cluster.

Command `route-reflector-client`

Options None

Command modes

- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to configure a BGP neighbor or neighbors in a peer-group as a route-reflector cluster. This command will implicitly make the local router a route-reflector server. The `no` version of this command deletes all clients of a route-reflector — the router no longer functions as a route-reflector.

Examples

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# route-reflector-client
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Int
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# route-reflector-client
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family l2vpn evpn
sonic(conf-router-bgp-neighbor-af)# route-reflector-client
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Int
sonic(conf-router-bgp-pg)# address-family l2vpn evpn
sonic(conf-router-bgp-pg-af)# route-reflector-client
```

Releases 3.0 or later

route-server-client

Configures BGP neighbors or neighbors in a peer-group as route server client.

Command `route-server-client`

Options None

Command mode NEIGHBOR-ADDRESS-FAMILY

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
```

```
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family ipv4 unicast
sonic(conf-router-bgp-neighbor-af)# route-server-client
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Int
sonic(conf-router-bgp-pg)# address-family ipv4 unicast
sonic(conf-router-bgp-pg-af)# route-server-client
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# neighbor 20.20.20.2
sonic(conf-router-bgp-neighbor)# remote-as 300
sonic(conf-router-bgp-neighbor)# address-family l2vpn evpn
sonic(conf-router-bgp-neighbor-af)# route-server-client
```

```
sonic(config)# router bgp 100
sonic(conf-router-bgp)# peer-group PG_Int
sonic(conf-router-bgp-pg)# address-family l2vpn evpn
sonic(conf-router-bgp-pg-af)# route-server-client
```

Releases 3.0 or later

route-target

Configures the route-target or community to attach while exporting routes from the current VRF for a specified VNI.

Command `route-target rttype rt`

Options

- *rttype* — Advertise options; both, import, or export
- `rt` — Route target to match

Command modes

- BGP-ADDRESS-FAMILY
- BGP-ADDRESS-FAMILY-VNI

Usage Use this command to specify the route-target to be matched when importing routes into the current VRF, or for a specific address-family or VNI. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100 vrf Vrf1
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# route-target import 11:11
sonic(conf-router-bgp-af)# route-target export 22:22
sonic(conf-router-bgp-af)# route-target both 33:33
```

```
sonic(config)# router bgp 100 vrf Vrf1
sonic(conf-router-bgp)# address-family l2vpn evpn
sonic(conf-router-bgp-af)# vni 100
sonic(conf-router-bgp-af-vni)# route-target import 11:11
sonic(conf-router-bgp-af-vni)# route-target export 22:22
sonic(conf-router-bgp-af-vni)# route-target both 33:33
```

Releases 3.0 or later

router bgp

Enables BGP and assigns an AS number to the local BGP speaker.

Command `router bgp {as-num-dot {[vrf] vrf-name}}`

Options

- *as-num-dot* — AS number range (1 to 65535 in 2 byte; 1 to 4294967295 in 4 byte)
- `vrf` *vrf-name* — (Optional) VRF instance name (up to 15 characters)

Command mode CONFIGURATION

Usage Use this command to create a BGP routing instance in a VRF. If the vrf key is not supplied, the default-vrf is used. Only one instance of BGP can be created per VRF. The `no` version of this command resets the configuration.

Example

```
sonic(config)# router bgp 65300
```

Releases 3.0 or later

router-id

Assigned a user-provided router ID to a BGP router.

Command `router-id ip-addr`

Options *ip-addr* — IPv4 address in A.B.C.D format

Command mode ROUTER-BGP

Usage Use this command to assign the router ID for an instance of BGP protocol. Router ID configuration is optional. BGP automatically selects one interface IP address as the router ID if not configured explicitly. You can change the router ID of a BGP router to reset peer-sessions. The `no` version of this command resets the value to the first configured IP address.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# router-id 163.134.6.97
```

Releases 3.0 or later

11

S to show ptp commands

Topics:

sample

Creates a sample session that co-relates a sampling rate with a session name.

Command `sample` *`sample-name`* `rate` *`rate-name`*

Options
- `sample` *`sample-name`* — Sample name (up to 63 characters)
- `rate` *`rate-name`* — Rate name

Command mode CONFIGURATION

Usage Use this command to create a sample session that co-relates a sampling rate with a session name. The sample session is identified by the name and can be used by multiple features to indicate sampling configuration. One packet in every rate packets will be sampled. The `no` version of this command removes the configuration.

Example

```
sonic(config)# sample s1 rate 5000
```

Releases 3.0 or later

send-community

Sends a community attribute to a BGP neighbor or peer-group.

Command `send-community {standard | extended | both | large | all | none}`

Options

- `standard` — Standard community attribute
- `extended` — Extended community attribute
- `both` — Both standard and extended community attributes
- `large` — Large community attributes
- `all` — All community attributes
- `none` — No attributes

Command mode ADDRESS-FAMILY

Usage Use this command to enable sending of community attribute to a BGP neighbor or peer-group. A community attribute indicates that all routes with the same attribute belong in the same community grouping. The command option provides the flexibility to enable sending of standard, extended, and large communities. The `no` version of this command disables sending a community attribute to a BGP neighbor or peer-group.

Examples

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 20.20.20.2
sonic(config-router-bgp-neighbor)# remote-as 300
sonic(config-router-bgp-neighbor)# address-family ipv4 unicast
sonic(config-router-bgp-neighbor-af)# send-community
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Int
sonic(config-router-bgp-pg)# address-family ipv4 unicast
sonic(config-router-bgp-pg-af)# send-community
```

Releases 3.0 or later

seq

Assigns a sequence number to deny or permit IPv4 addresses while creating the filter.

Command `seq seq-no {{deny {{protocol {src-prefix | src-any} {dest-prefix | dest-any} {[dscp] dscp-val}} | {icmp {src-prefix | src-any} {dest-prefix | dest-any} {[dscp] dscp-val}} | {tcp {src-prefix | src-any} {{[src-eq] src-port-number}} {dest-prefix | dest-any} {{[dst-eq] dst-port-number}} {[dscp] dscp-val} [fin] [syn] [rst] [psh] [ack] [urg]} | {udp {src-prefix | src-any} {{[src-eq] src-port-number}} {dest-prefix | dest-any} {{[dst-eq] dst-port-number}} {[dscp] dscp-val}}}} | {permit {{proto-num {src-prefix | src-any} {dest-prefix | dest-any} {[dscp] dscp-val}} | {icmp {src-prefix | src-any} {dest-prefix | dest-any} {[dscp] dscp-val}} | {tcp {src-prefix | src-any} {{[src-eq] src-port-number}} {dest-prefix | dest-any} {{[dst-eq] dst-port-number}} {[dscp] dscp-val} [fin] [syn] [rst] [psh] [ack] [urg]} | {udp {src-prefix | src-any} {{[src-eq] src-port-number}} {dest-prefix | dest-any} {{[dst-eq] dst-port-number}} {[dscp] dscp-val}}}}}`

Options

- `seq seq-no` — Sequence number to identify the ACL for editing and sequencing number
- `protocol` — Protocol number
- `src-prefix` — Source prefix in A.B.C.D/mask format
- `dest-prefix` — Destination prefix in A.B.C.D/mask format
- `dscp-val` — Permit a packet based on the DSCP values
- `src-port-number` — Source port number
- `dst-port-number` — Destination port number
- `proto-num` — Protocol number

Command mode IPV4-ACL

Usage	Use this command to assign a sequence number to deny or permit IPv4 addresses while creating the filter. The `no` version of this command removes the filter.
Example	`sonic(conf-ipv4-acl)# seq 10 deny tcp any any`
Releases	3.0 or later

session-timeout

Sets the MCLAG session timeout value.

Command	`session-timeout` *`ST`*
Options	`session-timeout` *`ST`* — Session timeout value in seconds
Command mode	MCLAG-DOMAIN
Usage	Use this command to set the MCLAG session timeout value. The `no` version of this command removes the configuration.
Example	`sonic(conf-mclag)# session-timeout 5`
Releases	3.0 or later

set as-path

Sets the BGP AS path attribute.

Command	`set as-path {prepend {`*`as-number_list`*`}}`
Options	`prepend` *`as-number_list`* — AS number list
Command mode	ROUTE-MAP
Usage	Use the command to set the BGP AS path attribute. The `no` version of this command removes the configuration.
Example	`sonic(conf-route-map)# set as-path prepend ASlist1`
Releases	3.0 or later

set community

Sets the community attribute in BGP updates.

Command	`set community {`*`comm-num`* `| local-AS | no-advertise | no-export | no-peer}`
Options	`community` *`comm-num`* — Community name in AA::NN format
Command mode	ROUTE-MAP
Usage	Use this command to set community attribute in BGP updates. The `no` version of this command deletes the BGP community attribute assignment.
Example	`sonic(conf-route-map)# set community local-AS`
Releases	3.0 or later

set extcommunity

Sets the extended community attributes in a route-map for BGP updates.

Command `set extcommunity {{rt value} | {soo value}}`

Options
- `rt value` — Route target value in ASN:NN_OR_IP-ADDRESS:NN format
- `soo value` — Route origin or site-of-origin value

Command mode ROUTE-MAP

Usage Use this command to set the extended community attributes in a route-map for BGP updates. The `no` version of this command removes the configuration.

Example
```
sonic(conf-route-map)# set extcommunity rt 10.10.10.2:325
```

Releases 3.0 or later

set ip next-hop

Sets an IPv4 address as the next-hop.

Command `set ip next-hop ip-addr`

Options `next-hop ip-addr` — IP address in A.B.C.D format

Command mode ROUTE-MAP

Usage Use this command to set an IPv4 address as the next-hop. The `no` version of this command removes the configuration.

Example
```
sonic(conf-route-map)# set ip next-hop 10.10.10.2
```

Releases 3.0 or later

set local-preference

Sets the preference value for the AS path.

Command `set local-preference pvalue`

Options `local-preference pvalue` — LOCAL_PREF attribute value

Command mode ROUTE-MAP

Usage Use this command to set the LOCAL_PREF attribute for routes meeting the route-map criteria. The `no` version of this command removes the configuration.

Example
```
sonic(conf-route-map)# set local-preference 200
```

Releases 3.0 or later

set metric

Sets a metric value for a routing protocol.

Command `set metric met`

Options `metric met` — Metric value

Command mode ROUTE-MAP

Usage Use the command to set a metric value for a routing protocol. The `no` version of this command removes the configuration.

Example

```
sonic(conf-route-map)# set metric 10
```

Releases 3.0 or later

set origin

Sets the origin of advertised routes

Command `set origin {egp | igp | incomplete}`

Options

- `egp` — Adds an existing community
- `igp` — Sends inside the local-AS
- `incomplete` — Not advertised to peers

Command mode ROUTE-MAP

Usage Use this command to set the origin of advertised routes. The `no` version of this command deletes the set clause from a route-map.

Example

```
sonic(conf-route-map)# set origin egp
```

Releases 3.0 or later

sflow agent-id

Configures an sFlow agent interface.

Command `sflow agent-id` *`interface`*

Options `agent-id` *`interface`* — Interface name

Command mode CONFIGURATION

Usage Use this command to configure an sFlow agent interface. The interface name provides the IPv4 or IPv6 address for the collector to uniquely identify the source of packets it receives. The `no` version of this command removes the configuration.

Example

```
sonic(config)# sflow agent-id Ethernet0
```

Releases 3.0 or later

sflow collector

Adds an sFlow collector.

Command `sflow collector` *`name ip`* [*`port`*]

Options

- *`name`* — Collector name (up to 16 characters)
- *`ip`* — Collector IPv4 or IPv6 address in A.B.C.D or A:B:C:D:E:F:G:H format
- *`port`* — (Optional) UDP port of the collector (0 to 65535, default 6343)

Command mode CONFIGURATION

Usage Use this command to configure an sFlow collector IP address where sFlow datagrams are forwarded. You must enter a valid and reachable IPv4 or IPv6 address. If you configure two collectors, traffic samples are sent to both (up to two sFlow collectors is allowed). The `no` version of this command deletes the configured sFlow collector.

Examples

```
sonic(config)# sflow collector col1 1.1.1.1
```

```
sonic(config)# sflow collector col2 1.1.1.2 port 4451
```

Releases 3.0 or later

sflow enable

Enables sFlow configuration.

Command `sflow enable`

Options None

Command mode CONFIGURATION

Usage Use this command to enable sFlow configuration. The `no` version of this command disables sFlow configuration.

Example

```
sonic(config)# sflow enable
```

Releases 3.0 or later

sflow polling-interval

Sets the sFlow polling interval.

Command `sflow polling-interval` *`interval`*

Options *`interval`* — Polling interval size (5 to 300, default 20, 0 to disable)

Command mode CONFIGURATION

Usage Use this command to configure the sFlow polling interval. The polling interval for an interface is the number of seconds between successive samples of counters sent to the collector. You can configure the duration for polled interface statistics. The `no` version of this command resets the sFlow polling interval.

Example

```
sonic(config)# sflow polling-interval 44
```

Releases 3.0 or later

sflow sampling-rate

Sets the sFlow sampling rate.

Command `sflow sampling-rate` *`rate`*

Options `sampling-rate` *`rate`* — sFlow sampling rates:

- 1G link — 1 packet in 1000
- 10G link — 1 packet in 10,000
- 40G link — 1 packet in 40,000
- 50G link — 1 packet in 50,000
- 100G link — 1 packet in 100,000

Command mode INTERFACE

Usage Use this command to configure the sFlow sampling rate. The `no` version of this command removes the sFlow sampling rate.

Example

```
sonic(conf-if-eth1/1/9)# sflow sampling-rate 4400
```

Releases 3.0 or later

show aaa

Displays authentication, authorization, and accounting (AAA) configuration information.

Command `show aaa`

Options None

Command mode EXEC

Usage Use this command to view AAA information including if failthrough is enabled, and the configured login method (local or TACACS+).

Example

```
sonic# show aaa
-----------------------------
AAA Authentication Information
-----------------------------
failthrough  : False
login-method : local, tacacs+
```

Releases 3.0 or later

show bfd peer

Displays BFD peer information with specified filters.

Command `show bfd peer {peer_ipv4 | peer_ipv6} [vrf] vrfname [multihop] [local-address] {local_ipv4 | local_ipv6} [interface] interfacename`

Options

- `peer peer_ipv4` — Peer IPv4 address in A.B.C.D format
- `peer peer_ipv6` — Peer IPv6 address in A::B format
- `vrf vrfname` — (Optional) VRF instance name
- `local-address local_ipv4` — (Optional) Local IPv4 address in A.B.C.D format
- `local-address local_ipv6` — (Optional) Local IPv6 address in A::B format
- `interface interfacename` — Interface name

Command mode EXEC

Usage Use this command to view single hop and multihop BFD peer information.

Example

```
sonic# show bfd peer 192.168.2.1 interface Ethernet0
```

```
sonic# show bfd peer 192.168.2.1 mulithop local-address 192.168.2.2
```

Releases 3.0 or later

show bfd peer counters

Displays BFD peer counter information.

Command `show bfd peer counters {peer_ipv4 | peer_ipv6} [vrf] vrfname [multihop] [local-address] {local_ipv4 | local_ipv6} [interface] interfacename`

Options

- `counters peer_ipv4` — Peer IPv4 address in A.B.C.D format
- `counters peer_ipv6` — Peer IPv6 address in A::B format
- `vrf vrfname` — (Optional) VRF instance name
- `local-address local_ipv4` — (Optional) Local IPv4 address in A.B.C.D format
- `local-address local_ipv6` — (Optional) Local IPv6 address in A::B format

- `interface interfacename` — (Optional) Interface name

Command mode EXEC

Usage Use this command to view information for single hop and multihop BFD peer counter information.

Examples

```
sonic# show bfd peer counters 192.168.2.1 interface Ethernet0
```

```
sonic# show bfd peer counters 192.168.2.1 multihop local-address 192.168.2.2
```

Releases 3.0 or later

show bfd peers

Displays all BFD peer information.

Command `show bfd peers [vrf] vrfname [brief]`

Options

- `vrf vrfname` — (Optional) VRF instance name
- `brief` — Display brief information on BFD peers

Command mode EXEC

Usage Use this command to view BFD peers information for all, a specific VRF name, or brief.

Example

```
sonic# show bfd peers
```

Releases 3.0 or later

show bfd peers counters

Displays counters for all BFD peers.

Command `show bfd peers counters`

Options None

Command mode EXEC

Usage Use this command to view counters for all BFD peers.

Example

```
sonic# show bfd peers counters
```

Releases 3.0 or later

show bgp as-path-access-list

Displays BGP AS path lists configured on the device.

Command `show bgp as-path-access-list [list-name]`

Options `as-path-access-list list-name` — (Optional) Access-list name

Command mode EXEC

Usage Use this command to view the AS path access lists configured on this device. If an access-list name is not specified, all AS Path access lists display. A BGP AS path access-list is used in route-maps and with BGP neighbors to design routing policies.

Example

```
sonic# show bgp as-path-access-list
AS path list asp_private:
   members: ^65000.*6510565109$,65107.*65200
```

```
AS path list asp_public:
   members: ^107.*2301.*709$,97.*201
```

Releases 3.0 or later

show bgp community-list

Displays BGP community-list configuration information.

Command `show bgp community-list [list-name]`

Options `community-list list-name` — (Optional) Community-list name

Command mode EXEC

Usage Use this command to view the community lists configured on this device. If a community-list name is not specified, all community lists display. Community-lists are used in route-maps to design BGP routing policies.

Example

```
sonic# show bgp community-list
Standard community list com1:  match: ANY
     local-AS
Expanded community list com2:   match: ANY
     Extended1
```

Releases 3.0 or later

show bgp ext-community-list

Displays BGP extended community-list configuration information.

Command `show bgp ext-community-list [list-name]`

Options `ext-community list list-name` — (Optional) Extended community-list name

Command mode EXEC

Usage Use this command to view the extended community lists configured on this device. If extended community-list name is not specified, all extended community lists display.

Example

```
sonic# show bgp ext-community-list
Standard extended community list ExtComm_AllowInt:  match: ALL

rt:19.32.56.167:65011,rt:31.67.182.214:3001,soo:01:65010,soo:.13.175.21:65101
Standard extended community list ExtComm_BlockExt:  match: ANY
     rt:4020:65104
     soo:9.54.32.165:65200
```

Releases 3.0 or later

show bgp l2vpn evpn route

Displays BGP EVPN route information in a tabular format.

Command `show bgp l2vpn evpn route {[rd] {rdvalue {{[mac] {macvalue {ip ipvalue}}} | {[type] {ead | es | macip | multicast | prefix}}}}}`

Options

- `rd rdvalue` — (Optional) RD value in A.B.C.D:NN or ASN:NN format
- `mac macvalue` — MAC address value in nn:nn:nn:nn:nn:nn format
- `ip ipvalue` — IP address value in A.B.C.D or A::B format
- `ead` — Ethernet auto-discovery EVPN route type
- `es` — Ethernet segment EVPN route type
- `macip` — MAC + IP EVPN route type

- `multicast` — Multicast EVPN route type
- `prefix` — Prefix EVPN route type

Command mode EXEC

Usage Use this command to view all BGP EVPN route information.

Example

```
sonic# show bgp l2vpn evpn route
BGP table version is 2, local router ID is 10.59.142.127
Status codes: s suppressed, d damped, h history, * valid, > best, i -
internal
Origin codes: i - IGP, e - EGP, ? - incomplete
EVPN type-1 prefix: [1]:[ESI]:[EthTag]
EVPN type-2 prefix: [2]:[EthTag]:[MAClen]:[MAC]:[IPlen]:[IP]
EVPN type-3 prefix: [3]:[EthTag]:[IPlen]:[OrigIP]
EVPN type-4 prefix: [4]:[ESI]:[IPlen]:[OrigIP]
EVPN type-5 prefix: [5]:[EthTag]:[IPlen]:[IP]
Network Next Hop Metric LocPrf Weight Path
Extended Community
Route Distinguisher: 11:11
*> [5]:[0]:[0]:[0.0.0.0]
0.0.0.0 32768 i
ET:8
*> [5]:[0]:[0]:[::] 0.0.0.0 32768 i
ET:8
Route Distinguisher: 22:22
*> [2]:[0]:[48]:[52:54:00:76:be:f7]:[32]:[2.1.1.1]
1.1.1.1 32768 i
ET:8 RT:100:268435556 Default Gateway
*> [2]:[0]:[48]:[52:54:00:cb:f0:e3]
1.1.1.1 32768 i
ET:8 RT:100:268435556
*> [2]:[0]:[48]:[52:54:00:cb:f0:e3]:[32]:[2.1.1.2]
1.1.1.1 32768 i
ET:8 RT:100:268435556
*> [3]:[0]:[32]:[1.1.1.1]
1.1.1.1 32768 i
ET:8 RT:100:268435556
Route Distinguisher: 3.1.1.1:5096
*> [5]:[0]:[24]:[3.1.1.0]
1.1.1.1 0 32768 ?
ET:8 RT:100:200 Rmac:52:54:00:76:be:f7
Route Distinguisher: 4.1.1.2:5096
*> [5]:[0]:[24]:[4.1.1.0]
2.2.2.2 0 0 200 ?
RT:200:200 ET:8 Rmac:52:54:00:cb:f0:e3
Route Distinguisher: 10.59.143.68:100
*> [2]:[0]:[48]:[52:54:00:cb:f0:e3]:[32]:[2.1.1.2]
2.2.2.2 0 200 i
RT:200:100 ET:8 Default Gateway
*> [3]:[0]:[32]:[2.2.2.2]
2.2.2.2 0 200 i
RT:200:100 ET:8
Displayed 10 prefixes (10 paths)
```

Releases 3.0 or later

show bgp l2vpn evpn route detail

Displays BGP EVPN routes in detail.

Command `show bgp l2vpn evpn route detail`

Options None

Command mode EXEC

Usage Use this command to view all BGP EVPN detailed information on routes.

Example

```
sonic# show bgp l2vpn evpn route detail
EVPN type-2 prefix: [2]:[EthTag]:[MAClen]:[MAC]
EVPN type-3 prefix: [3]:[EthTag]:[IPlen]:[OrigIP]
EVPN type-5 prefix: [5]:[EthTag]:[IPlen]:[IP]
BGP routing table entry for 11:11:[5]:[0]:[0]:[0.0.0.0]
Paths: (1 available, best #1)
Zebra Add: 6d21h36m
Advertised to non peer-group peers:
10.1.1.2
Route [5]:[0]:[0]:[0.0.0.0] VNI 0
Local
0.0.0.0 from 0.0.0.0 (10.59.142.127)
Origin IGP, weight 32768, valid, sourced, local, best (First path received)
Extended Community: ET:8
Last update: Wed Feb 12 17:06:15 2020
BGP routing table entry for 11:11:[5]:[0]:[0]:[::]
Paths: (1 available, best #1)
Zebra Add: 6d21h36m
Advertised to non peer-group peers:
10.1.1.2
Route [5]:[0]:[0]:[::] VNI 0
Local
0.0.0.0 from 0.0.0.0 (10.59.142.127)
Origin IGP, weight 32768, valid, sourced, local, best (First path received)
Extended Community: ET:8
Last update: Wed Feb 12 17:06:15 2020
Displayed 2 prefixes (2 paths) with this RD (of requested type)
```

Releases 3.0 or later

show bgp l2vpn evpn route detail type

Displays BGP EVPN routes of a specified type in detail.

Command `show bgp l2vpn evpn route detail type {ead | es | macip | multicast | prefix}`

Options

- `ead` — Ethernet auto-discovery EVPN route type
- `es` — Ethernet segment EVPN route type
- `macip` — MAC + IP EVPN route type
- `multicast` — Multicast EVPN route type
- `prefix` — Prefix EVPN route type

Command mode EXEC

Usage Use this command to view BGP EVPN routes based on the type.

Example

```
sonic#  show bgp l2vpn evpn route detail type multicast
```

Releases 3.0 or later

show bgp l2vpn evpn route type

Displays BGP EVPN routes of a specified type.

Command `show bgp l2vpn evpn route type {ead | es | macip | multicast | prefix}`

Options

- `ead` — Ethernet auto-discovery EVPN route type
- `es` — Ethernet segment EVPN route type
- `macip` — MAC + IP EVPN route type
- `multicast` — Multicast EVPN route type
- `prefix` — Prefix EVPN route type

Command mode EXEC

Usage Use this command to view BGP EVPN routes of a specified type.

Example

```
sonic# show bgp l2vpn evpn route type multicast
```

Releases 3.0 or later

show bgp l2vpn evpn summary

Displays BGP summarized information for BGP L2VPN EVPN address-family.

Command `show bgp l2vpn evpn summary`

Options None

Command mode EXEC

Usage Use this command to view BGP summarized information for BGP L2VPN EVPN address-family including neighbors with EVPN address-family activated.

Example

```
sonic# show bgp l2vpn evpn summary
BGP router identifier 10.59.142.127, local AS number 100 vrf-id 0
BGP table version 0
Neighbor V AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down State/PfxRcd
10.1.1.2 4 200 11338 11337 0 0 0 6d21h29m 3
Total number of neighbors 1
Total number of neighbors established 1
```

Releases 3.0 or later

show bgp l2vpn evpn vni

Displays VNI information.

Command `show bgp l2vpn evpn vni` *vninum*

Options `vni` *vninum* — VNI number

Command mode EXEC

Usage Use this command to view VNI information including type, tenant VRF, RD, originating IP address, import route targets, and so on.

Example

```
sonic# show bgp l2vpn evpn vni 100
VNI: 100 (known to the kernel)
Type: L2
Tenant-Vrf: default
RD: 22:22
Originator IP: 1.1.1.1
Mcast group: 0.0.0.0
Advertise-gw-macip : Yes
Advertise-svi-macip : No
Import Route Target:
22:22
22:23
Export Route Target:
100:268435556
```

Releases 3.0 or later

show image list

Displays image list information.

Command `show image list`

Options None

Command mode EXEC

Usage Use this command to view current, next, and available software image information.

Example

```
sonic# show image list
Current: SONiC-OS-HEAD.138-dirty-20200103.154042
Next: SONiC-OS-HEAD.140-dirty-20200105.093102
Available:
SONiC-OS-HEAD.138-dirty-20200103.154042
SONiC-OS-HEAD.140-dirty-20200105.093102
```

Releases 3.0 or later

show interface

Displays all configured interface information.

Command `show interface {counters | {Ethernet [phy-if-id]} | {PortChannel [po-id]} | {Management [mgmt-if-id]} | {Vlan vlan-id} | {Loopback lo-id} | status}`

Options

- `counters` — Displays counter information for all interfaces
- `Ethernet phy-if-id` — Physical interface ID
- `PortChannel po-id` — PortChannel ID (1 to 128)
- `Management mgmt-if-id` — Management interface ID
- `Vlan vlan-id` — VLAN ID (1 to 4093)
- `Loopback lo-id` — Loopback interface ID (0 to 16383)
- `status` — Interface status

Command mode EXEC

Usage Use this command to view interface configuration information on counters, interfaces, and status. Use `do show interface` to view interface information from other command modes.

Example

```
sonic# show interface counters
-------------------------------------------------------------
Interface    State  RX_OK  RX_ERR  RX_DRP  TX_OK  TX_ERR  TX_DRP
-------------------------------------------------------------
Ethernet0    D      0       0       0       0      0       0
Ethernet4    U      1064    0       0       438    0       0
Ethernet8    D      0       0       0       0      0       0
Ethernet12   D      0       0       0       0      0       0
Ethernet16   D      0       0       0       0      0       0
Ethernet20   D      0       0       0       0      0       0
Ethernet24   D      0       0       0       0      0       0
Ethernet28   D      0       0       0       0      0       0
Ethernet32   U      431     0       0       438    0       0
Ethernet36   D      0       0       0       0      0       0
Ethernet40   D      0       0       0       0      0       0
eth0         U      23233   0       0       33220  0       0
```

```
sonic# show interface PortChannel 100
PortChannel100 is down, line protocol is down, mode LACP
Minimum number of links to bring PortChannel up is 1
Fallback: Enabled
MTU 9100
LACP mode ACTIVE interval SLOW priority 65535 address 90:b1:1c:f4:9c:9f
Members in this channel: Ethernet84(Selected), Ethernet80
LACP Actor port 85  address 90:b1:1c:f4:9c:9f key 100
LACP Partner port 0  address 00:00:00:00:00:00 key 0
Last clearing of "show interface" counters: 1970-01-01 00:00:00
Input statistics:
          2972 packets, 712962 octets
2972 Multicasts, 0 Broadcasts, 0 Unicasts
```

```
0 error, 2972 discarded
Output statistics:
          2969 packets, 712560 octets
2969 Multicasts, 0 Broadcasts, 0 Unicasts
0 error, 0 discarded
```

```
sonic# show interface Management 0
eth0 is up, line protocol is up
Hardware is Mgmt
IPV4 address is 44.2.3.4/24
Mode of IPV4 address assignment: MANUAL
IPV6 address is a::e/64
Mode of IPV6 address assignment: MANUAL
IP MTU 1500 bytes
LineSpeed 1000MB, Auto-negotiation on
Input statistics:
        0 packets, 0 octets
        0 Multicasts, 0 Broadcasts, 0 Unicasts
        0 error, 0 discarded
Output statistics:
        0 packets, 0 octets
        0 Multicasts, 0 Broadcasts, 0 Unicasts
        0 error, 0 discarded
```

```
sonic# show interface Vlan 2
Vlan2 is up
Mode of IPV4 address assignment: not-set
Mode of IPV6 address assignment: not-set
IP MTU 9100 bytes
```

```
sonic# show interface status
------------------------------------------------------------------
Name            Description     Admin    Oper     Speed    MTU
------------------------------------------------------------------
Ethernet0       -               up       down     40GB     9100
Ethernet4       -               up       up       40GB     9100
Ethernet8       -               up       down     40GB     9100
Ethernet12      Ethernet12      up       down     40GB     9100
Ethernet16      -               up       down     40GB     9100
Ethernet20      -               up       down     40GB     9100
Ethernet24      -               up       down     40GB     9100
eth0            Management0     up       up       1000MB   1500
```

Releases 3.0 or later

show ip access-group

Displays all IPv4 access-group information.

Command `show ip access-group`

Options None

Command mode EXEC

Usage Use this command to view ingress and egress IPv4 access-group configuration information.

Example

```
sonic# show ip access-group
Ingress IP access-list ACL1 on Ethernet0
Egress IP access-list ACL3 on Ethernet4
```

Releases 3.0 or later

show ip access-lists

Displays IPv4 access-lists information.

Command `show ip access-lists [access-list-name]`

Options `access-lists access-list-name` — (Optional) Access-list name (up to 63 characters)

Command mode EXEC

Usage Use this command to view information on all IPv4 access-lists, or for a specific access-list name.

Example

```
sonic# show ip access-lists
ip access-list ACL1
     1 permit icmp 11.1.1.1/32 21.1.1.1/32 dscp 1 (0 matches)
     2 deny tcp 11.1.1.2/32 eq 102 21.1.1.2/32 eq 202 fin dscp 2(5 matches)
     3 permit pim 11.1.1.3/32 21.1.1.3/32 dscp 3 (10 matches)
     4 deny tcp 11.1.1.4/32 21.1.1.4/32 dscp 4 (12 matches)
     5 permit tcp 11.1.1.5/32 21.1.1.5/32 dscp 5 (21 matches)
ip access-list ACL2
     1 permit tcp 12.1.1.1/32 eq 101 22.1.1.1/32 eq 201 dscp 1 (4 matches)
     2 permit tcp 12.1.1.2/32 eq 102 22.1.1.2/32 eq 202 dscp 2 (6 matches)
ip access-list ACL3
     1 permit tcp 0.0.0.0/0 0.0.0.0/0 eq 24 (0 matches)
```

Releases 3.0 or later

show ip arp

Displays all ARP entries.

Command `show ip arp [ip-addr]`

Options `arp ip-addr` — (Optional) IP address in A.B.C.D format

Command mode EXEC

Usage Use this command to view all ARP entry configuration information, or for a specific IP address.

Examples

```
sonic# show ip arp
------------------------------------------------------------------------
Address         Hardware address    Interface        Egress Interface
------------------------------------------------------------------------
192.168.1.4     00:01:02:03:44:55   Ethernet8        -
192.168.2.4     00:01:02:03:ab:cd   PortChannel200   -
192.168.3.6     00:01:02:03:04:05   Vlan100          Ethernet4
10.11.48.254    00:01:e8:8b:44:71   eth0             -
10.14.8.102     00:01:e8:8b:44:71   eth0             -
0.0.0.0         00:00:00:00:00:00   lo
```

```
sonic# show ip arp 20.0.0.2
Address        Hardware address    Interface         Egress Interface
------------------------------------------------------------------------
20.0.0.2       90:b1:1c:f4:9d:ba   Vlan20            Ethernet0
```

Releases 3.0 or later

show ip arp interface

Displays ARP entries for an interface.

Command `show ip arp interface {{Ethernet {phy-if-name [summary]}} | {Loopback {lo-id [summary]}} | {Management {mgmt-if-id [summary]}} | {PortChannel {lag-id [summary]}} | {Vlan {vlan-id [summary]}} | {Vxlan {vxlan-if-name [summary]}}}`

Options

- `Ethernet phy-if-name` — Physical interface name
- `Loopback lo-id` — Loopback interface ID
- `Mangement mgmt-if-id` — Management interface ID
- `PortChannel lag-id` — PortChannel ID
- `Vlan vlan-id` — VLAN ID
- `Vxlan vxlan-if-name` — VxLAN name (up to 63 characters)

Command mode EXEC

Usage Use this command to view a summary of ARP interface entries, or entries for a specific interface.

Example

```
sonic# show ip arp interface Vlan 20
Address         Hardware address    Interface         Egress Interface
------------------------------------------------------------------------
20.0.0.2        90:b1:1c:f4:9d:ba   Vlan20            Ethernet0
20.0.0.5        00:11:22:33:44:55   Vlan20            Ethernet0
```

Releases 3.0 or later

show ip arp mac-address

Displays ARP entries for a specific MAC address.

Command `show ip arp mac-address mac-addr`

Options `mac-address mac-addr` — MAC address in nn:nn:nn:nn:nn:nn format

Command mode EXEC

Usage Use this command to view ARP entries for a specific MAC address.

Example

```
sonic# show ip arp mac-address 90:b1:1c:f4:9d:ba
Address         Hardware address    Interface         Egress Interface
------------------------------------------------------------------------
20.0.0.2        90:b1:1c:f4:9d:ba   Vlan20            Ethernet0
```

Releases 3.0 or later

show ip arp summary

Displays a summary of ARP entries.

Command `show ip arp summary`

Options None

Command mode EXEC

Usage Use this command to view a summary of ARP entries.

Example

```
 sonic# show ip arp summary
Total Entries
-------------
     2
```

Releases 3.0 or later

show ip bgp

Displays all IP BGP routing information.

Command

```
show ip bgp {{[ipv4] {{[neighbors] {{[neighbor-ip] {[routes] | [received-
routes] | [advertised-routes]}} | {[interface] {Ethernet | PortChannel | Vlan}
{[routes] | [received-routes] | [advertised-routes]}}}} | [prefix] |
[summary]}} | {[ipv6] {{[neighbors] {{[neighbor-ip] {[routes] | [received-
routes] | [advertised-routes]}} | {[interface] {Ethernet | PortChannel | Vlan}
{[routes] | [received-routes] | [advertised-routes]}}}} | [prefix] |
[summary]}} | {[neighbors] {{[neighbor-ip] {[routes] | [received-routes] |
[advertised-routes]}} | {[interface] {Ethernet | PortChannel | Vlan} {[routes]
| [received-routes] | [advertised-routes]}}}} | {[peer-group] [peer-group-
name]} | [prefix] | [summary] | {[vrf] {vrf-name {{[ipv4] {{[neighbors]
{{[neighbor-ip] {[routes] | [received-routes] | [advertised-routes]}} |
{[interface] {Ethernet | PortChannel | Vlan} {[routes] | [received-routes] |
[advertised-routes]}}}} | [prefix] | [summary]}} | {[ipv6] {{[neighbors]
{{[neighbor-ip] {[routes] | [received-routes] | [advertised-routes]}} |
{[interface] {Ethernet | PortChannel | Vlan} {[ routes] | [received-routes] |
[advertised-routes]}}}} | [prefix] | [summary]}} | {[neighbors] {{[neighbor-ip]
{[routes] | [received-routes] | [advertised-routes]}} | {[interface] {Ethernet
| PortChannel | Vlan} {[routes] | [received-routes] | [advertised-routes]}}}} |
{[peer-group] [peer-group-name]} | [prefix] | [summary]}}}}
```

Options

- `neighbors` *neighbor-ip* — Neighbor IP address in A.B.C.D or A::B format
- `interface` *prefix* — Interface prefix in A.B.C.D/mask format
- `peer-group` *peer-group-name* — Peer-group name
- `vrf` *vrf-name* — VRF instance name (up to 15 characters)

Command mode

EXEC

Usage

Use this command to display BGP neighbors, routes, interfaces, and peer-group information. There are various options available to display BGP information. Use the `vrf` option to view information from a particular VRF instance of BGP. Select IPv4 or IPv6 to display information from either address-family.

- `show ip bgp summary` — Displays BGP global parameters and brief information about BGP neighbors
- `show ip bgp` — Displays BGP local RIB routes; use filtering options to zoom into a subset of routes
- `show ip bgp neighbor` — Displays one or all BGP neighbors information in detail
- `show ip bgp peer-group` — Displays one or all BGP peer-group information in detail

Examples

```
sonic# show ip bgp summary
BGP router identifier 2.2.2.2, local AS number 100
Neighbor     V   AS    MsgRcvd   MsgSent   InQ   OutQ   Up/Down    State/
PfxRcd
101.2.2.2    4   100   36        37        0     0      00:19:31   10
```

```
sonic# show ip bgp ipv4 summary
BGP router identifier 2.2.2.2, local AS number 100
Neighbor     V   AS    MsgRcvd   MsgSent   InQ   OutQ   Up/Down    State/
PfxRcd
101.2.2.2    4   100   36        37        0     0      00:19:31   10
```

```
sonic# show ip bgp ipv4
BGP routing table information for VRF default
Router identifier 2.2.2.2, local AS number 100
Route status codes: * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network          Next Hop    Metric   LocPref   Path
*    10.0.0.0/8       101.2.2.2   0        100       ?
*>                    0.0.0.0     0                  ?
*>   21.0.0.0/8       101.2.2.2   0        100       i
*>   22.0.0.0/8       101.2.2.2   0        100       i
*>   31.0.0.0/8       101.2.2.2   0        100       i
*>   32.0.0.0/8       101.2.2.2   0        100       i
```

```
*>   34.0.0.0/8      101.2.2.2   0        100       i
*    101.2.2.0/24    101.2.2.2   0        100       ?
*>                   0.0.0.0     0                  ?
```

```
sonic# show ip bgp ipv4 neighbors

BGP neighbor is 101.2.2.2, remote AS 100, local AS 100, internal link
  BGP version 4, remote router ID 1.1.1.1 , local router ID 2.2.2.2
  BGP state = Established, up for 00:20:23
  Last read 00:00:22, Last write 00:00:22
  Hold time is 180 seconds, keepalive interval is 60 seconds
  Minimum time between advertisement runs is 30 seconds
  Neighbor capabilities:
    4 Byte AS: advertised and received
    AddPath: advertised and received
    Route refresh: advertised and received
    Multiprotocol Extension: advertised and received
  Message statistics:
    InQ depth is 0
    OutQ depth is 0
                         Sent       Rcvd
    Opens:               2          2
    Notifications:       2          0
    Updates:             8          8
    Keepalive:           22         22
    Route Refresh:       0          0
    Capability:          0          0
    Total:               34         32

  For address family: IPv4 Unicast
    Address-family enabled
    Prefixes received 2
  For address family: IPv6 Unicast
    Address-family enabled
    Prefixes received 1
  Connections established 2, dropped 1
  Last reset 00:20:24, Last reset reason BGP Notification send
  Local host: 101.2.2.1, Local port: 55388
  Foreign host: 101.2.2.2, Foreign port: 179
  BGP Connect Retry Timer in Seconds 120

BGP neighbor is 1001:2222::2, remote AS 100, local AS 100, internal link
  BGP version 4, remote router ID 1.1.1.1 , local router ID 2.2.2.2
  BGP state = Established, up for 00:03:55
  Last read 00:00:54, Last write 00:00:54
  Hold time is 180 seconds, keepalive interval is 60 seconds
  Minimum time between advertisement runs is 30 seconds
  Neighbor capabilities:
    4 Byte AS: advertised and received
    AddPath: advertised and received
    Route refresh: advertised and received
    Multiprotocol Extension: advertised and received
  Message statistics:
    InQ depth is 0
    OutQ depth is 0
                         Sent       Rcvd
    Opens:               3          3
    Notifications:       2          2
    Updates:             11         11
    Keepalive:           7          7
    Route Refresh:       0          0
    Capability:          0          0
    Total:               23         23

  For address family: IPv4 Unicast
    Address-family enabled
    Prefixes received 2
  For address family: IPv6 Unicast
    Address-family enabled
    Prefixes received 1
  Connections established 3, dropped 2
```

```
  Last reset 00:03:56, Last reset reason Peer closed the session
  Local host: 1001:2222::1, Local port: 40018
  Foreign host: 1001:2222::2, Foreign port: 179
  BGP Connect Retry Timer in Seconds 120
```

```
sonic# show ip bgp ipv6
BGP routing table information for VRF default
Router identifier 2.2.2.2, local AS number 100
Route status codes: * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network          Next Hop            Metric   LocPref   Path
*>   2121::/64        fe80::92b1:1cff:fef4:ab9b0   100       i
*>   2122::/64        fe80::92b1:1cff:fef4:ab9b0   100       i
*>   2123:3322::/64   fe80::92b1:1cff:fef4:ab9b0   100       i
```

```
sonic# show ip bgp ipv6 neighbors

BGP neighbor is 101.2.2.2, remote AS 100, local AS 100, internal link
  BGP version 4, remote router ID 1.1.1.1 , local router ID 2.2.2.2
  BGP state = Established, up for 00:20:13
  Last read 00:00:12, Last write 00:00:12
  Hold time is 180 seconds, keepalive interval is 60 seconds
  Minimum time between advertisement runs is 30 seconds
  Neighbor capabilities:
    4 Byte AS: advertised and received
    AddPath: advertised and received
    Route refresh: advertised and received
    Multiprotocol Extension: advertised and received
  Message statistics:
    InQ depth is 0
    OutQ depth is 0
                         Sent        Rcvd
    Opens:               2           2
    Notifications:       2           0
    Updates:             8           8
    Keepalive:           22          22
    Route Refresh:       0           0
    Capability:          0           0
    Total:               34          32

  For address family: IPv4 Unicast
    Address-family enabled
    Prefixes received 2
  For address family: IPv6 Unicast
    Address-family enabled
    Prefixes received 1
  Connections established 2, dropped 1
  Last reset 00:20:14, Last reset reason BGP Notification send
  Local host: 101.2.2.1, Local port: 55388
  Foreign host: 101.2.2.2, Foreign port: 179
  BGP Connect Retry Timer in Seconds 120

BGP neighbor is 1001:2222::2, remote AS 100, local AS 100, internal link
  BGP version 4, remote router ID 1.1.1.1 , local router ID 2.2.2.2
  BGP state = Established, up for 00:03:45
  Last read 00:00:45, Last write 00:00:45
  Hold time is 180 seconds, keepalive interval is 60 seconds
  Minimum time between advertisement runs is 30 seconds
  Neighbor capabilities:
    4 Byte AS: advertised and received
    AddPath: advertised and received
    Route refresh: advertised and received
    Multiprotocol Extension: advertised and received
  Message statistics:
    InQ depth is 0
    OutQ depth is 0
                         Sent        Rcvd
    Opens:               3           3
   Notifications:       2           2
    Updates:             11          11
```

```
    Keepalive:           7           7
    Route Refresh:       0           0
    Capability:          0           0
    Total:               23          23

  For address family: IPv4 Unicast
    Address-family enabled
    Prefixes received 2
  For address family: IPv6 Unicast
    Address-family enabled
    Prefixes received 1
  Connections established 3, dropped 2
  Last reset 00:03:47, Last reset reason Peer closed the session
  Local host: 1001:2222::1, Local port: 40018
  Foreign host: 1001:2222::2, Foreign port: 179
  BGP Connect Retry Timer in Seconds 120
```

```
sonic# show ip bgp ipv6 summary
BGP router identifier 2.2.2.2, local AS number 100
Neighbor     V   AS    MsgRcvd   MsgSent   InQ   OutQ   Up/Down     State/
PfxRcd
101.2.2.2    4   100   37        38        0     0      00:20:14    10
```

```
sonic# show ip bgp peer-group

BGP peer-group pgrp0, remote AS 100
  Configured address-families: IPv4 Unicast;
  Peer-group members:
   1001:2222::2 Established
   101.2.2.2 Established
```

Releases 3.0 or later

show ip igmp snooping

Display IPv4 IGMP snooping membership details.

Command `show ip igmp snooping {{[vlan] vlan-id} {[groups] {[vlan] vlan-id}}}`

Options `vlan vlan-id` — (Optional) VLAN ID (1 to 4093)

Command mode EXEC

Usage Use this command to view IGMP snooping configuration across all VLANs, a specified VLAN, or display IGMP snooping groups across all VLANs or a specified VLAN.

Examples

```
sonic# show ip igmp snooping
Vlan ID: 100
Querier: Disabled
IGMP Operation mode: IGMPv1
Is Fast-Leave Enabled: Disabled
Query interval: 125
Last Member Query Interval: 1000
Max Response time: 10

Vlan ID: 200
Querier: Enabled
IGMP Operation mode: IGMPv2
Is Fast-Leave Enabled: Disabled
Query interval: 125
Last Member Query Interval: 1000
Max Response time: 10

Vlan ID: 300
Querier: Enabled
IGMP Operation mode: IGMPv3
Is Fast-Leave Enabled: Disabled
```

```
Query interval: 20
Last Member Query Interval: 1000
Max Response time: 10
```

```
sonic# show ip igmp snooping vlan 200
Vlan ID: 200
Querier: Enabled
IGMP Operation mode: IGMPv2
Is Fast-Leave Enabled: Disabled
Query interval: 125
Last Member Query Interval: 1000
Max Response time: 10
```

```
sonic# show ip igmp snooping groups
Vlan ID: 100
-----------
1 (*, 225.1.1.1)
  Outgoing Ports: Ethernet4,PortChannel3
2 (*, 225.1.1.2)
  Outgoing Ports: Ethernet8
Total number of entries: 2

Vlan ID : 300
-------------
1 (100.10.2.3, 226.0.0.1 )
  Outgoing Ports: Ethernet8,Portchannel2
Total number of entries: 1
```

```
sonic# show ip igmp snooping groups vlan 100
Vlan ID: 100
-----------
1 (*, 225.1.1.1)
  Outgoing Ports: Ethernet4, PortChannel3
2 (*, 225.1.1.2)
  Outgoing Ports: Ethernet8
Total number of entries: 2
```

Releases 3.0 or later

show ip interfaces

Displays IPv4 interface configuration information.

Command `show ip interfaces`

Options None

Command mode EXEC

Usage Use this command to view information on all IPv4 interfaces configured.

Example

```
sonic# show ip interfaces
```

Releases 3.0 or later

show ip prefix-list

Displays IPv4 prefix-list configuration information.

Command `show ip prefix-list [list-name]`

Options `prefix-list list-name` — (Optional) Prefix-list name

Command mode EXEC

Usage Use this command to view configured IPv4 prefix-list information.

Examples

```
sonic# show ip prefix-list
IP prefix list prflst656:
     permit 156.1.1.0/24
IP prefix list prflst657:
     permit 157.1.1.0/24
```

```
sonic# show ip prefix-list prflst657
IP prefix list prflst657:
     permit 157.1.1.0/24
```

Releases 3.0 or later

show ip route

Displays information about IPv4 BGP routing table entries.

Command `show ip route {{[vrf] {vrfname [prefix]}} | [prefix]}`

Options

- `vrf vrfname` — (Optional) Name of the VRF to view information that is exchanged between BGP neighbors corresponding to that VRF
- `prefix` — (Optional) Prefix in A.B.C.D/mask format

Command mode EXEC

Usage Use this command to display information about the IPv4 BGP routing table entries.

Example

```
sonic# show ip route
```

Releases 3.0 or later

show ip static-anycast-gateway

Displays IPv4 static Anycast gateway information.

Command `show ip static-anycast-gateway`

Options None

Command mode EXEC

Usage Use this command to view IPv4 static Anycast gateway configuration information.

Example

```
sonic# show ip static-anycast-gateway
```

Releases 3.0 or later

show ip vrf

Display all IPv4 VRF instance information, or for a specific VRF instance.

Command `show ip vrf [vrf-name]`

Options `vrf vrf-name` — (Optional) VRF name (up to 15 characters)

Command mode EXEC

Usage Use this command to view all IPv4 VRF instance information, or for a specific VRF instance.

Examples

```
sonic# show ip vrf
VRF-NAME            INTERFACES
```

```
------------------------------
mgmt                 eth0
Vrf_blue             Ethernet8
                     Loopback20
                     PortChannel20
                     Vlan20
Vrf_red              Ethernet4
                     Loopback10
                     PortChannel10
                     Vlan10
```

```
sonic# show ip vrf Vrf_blue
VRF-NAME             INTERFACES
-----------------------------
Vrf_blue             Ethernet8
                     Loopback20
                     PortChannel20
                     Vlan20
```

```
sonic# show ip vrf Vrf_red
VRF-NAME             INTERFACES
-----------------------------
Vrf_red              Ethernet4
                     Loopback10
                     PortChannel0
                     Vlan10
```

Releases 3.0 or later

show ip vrf management

Displays IPv4 management VRF configuration information.

Command `show ip vrf management`

Options None

Command mode EXEC

Usage Use this command to view IPv4 management VRF configuration information.

Example

```
sonic# show ip vrf management
VRF-Name                    Interfaces
-----------------------------------
management                  eth0
```

Releases 3.0 or later

show ipv6 interfaces

Displays IPv6 configuration information for all interfaces.

Command `show ipv6 interfaces`

Options None

Command mode EXEC

Usage Use this command to view all IPv6 interface configuration information.

Example

```
sonic# show ipv6 interfaces
```

Releases 3.0 or later

show ipv6 neighbors

Displays IPv6 discovery information.

Command `show ipv6 neighbors [ipv6-addr]`

Options `neighbors ipv6-addr` — (Optional) IPv6 address in A::B format

Command mode EXEC

Usage Use this command to view IPv6 discovery information. If you do not specify an IPv6 address, all IPv6 neighbor address information displays.

Example

```
sonic# show ipv6 neighbors
```

Releases 3.0 or later

show ipv6 neighbors interface

Displays IPv6 neighbors interface configuration information.

Command `show ipv6 neighbors interface if-type {if-id [summary]}`

Options

- `interface if-type` — Select Ethernet, Vlan, PortChannel, or Management
- `interface if-id` — Interface ID
- `summary` — (Optional) View a summary of IPv6 neighbors interface information

Command mode EXEC

Usage Use this command to view IPv6 neighbors entries for a specific interface.

Example

```
sonic# show ipv6 neighbors interface Management 0
------------------------------------------------------------------------------
Address                   Hardware address    Interface    Egress Interface
------------------------------------------------------------------------------
fe80::e6f0:4ff:fe79:34c7  e4:f0:04:79:34:c7   eth0         -
```

Releases 3.0 or later

show ipv6 neighbors mac-address

Display IPv6 neighbors MAC address configuration information.

Command `show ipv6 neighbors mac-address mac-addr`

Options `mac-address mac-addr` — MAC address in nn:nn:nn:nn:nn:nn format

Command mode EXEC

Usage Use this command to view IPv6 neighbors MAC address configuration information.

Example

```
sonic# show ipv6 neighbors mac-address e4:f0:04:79:34:c7
------------------------------------------------------------------------------
Address                    Hardware address    Interface   Egress Interface
------------------------------------------------------------------------------
fe80::e6f0:4ff:fe79:34c7   e4:f0:04:79:34:c7   eth0        -
```

Releases 3.0 or later

show ipv6 neighbors summary

Displays a summary of IPv6 neighbors configuration information.

Command `show ipv6 neighbors summary`

Options None

Command mode EXEC

Usage Use this command to view a summary of IPv6 neighbors configuration information.

Example

```
sonic# show ipv6 neighbors summary
```

Releases 3.0 or later

show ipv6 prefix-list

Displays IPv6 prefix-list configuration information.

Command `show ipv6 prefix-list [list-name]`

Options `prefix-list list-name` — (Optional) Prefix-list name

Command mode EXEC

Usage Use this command to view IPv6 prefix-list configuration information.

Examples

```
sonic# show ipv6 prefix-list
IPv6 prefix list prflst758:
     permit 1758:5523::/64 ge 67 le 68
IPv6 prefix list prflst759:
     permit 1759:5567::/64
```

```
sonic# show ipv6 prefix-list prflst758
IPv6 prefix list prflst758:
     permit 1758:5523::/64 ge 67 le 68
```

Releases 3.0 or later

show ipv6 route

Displays information about IPv6 BGP routing table entries.

Command `show ipv6 route {{[vrf] {vrfname [prefix]}} | [prefix]}`

Options

- `vrf vrfname` — (Optional) Name of the VRF to view information that is exchanged between BGP neighbors corresponding to that VRF
- `prefix` — (Optional) Route prefix in A::B/mask format

Command mode EXEC

Usage Use this command to view information about all IPv6 BGP routing table entries, or for a specific prefix.

Example

```
sonic# show ipv6 route vrf vrf1 10.1.1.1/100
```

Releases 3.0 or later

show ipv6 static-anycast-gateway

Displays IPv6 static Anycast gateway configuration information.

Command `show ipv6 static-anycast-gateway`

Options None

Command mode EXEC

Usage Use this command to view IPv6 static Anycast gateway information.

Example

```
sonic#  show ipv6 static-anycast-gateway
```

Releases 3.0 or later

show kdump files

Displays the local stored kernel core dump files.

Command `show kdump files`

Options None

Command mode EXEC

Usage Use this command to show the kdump kernel core dump files which are stored locally.

Example

```
sonic# show kdump files
Record Key           Filename
-----------------------------------------------------------
     1 202002101809 /var/crash/202002101809/dmesg.202002101809
                    /var/crash/202002101809/kdump.202002101809
```

Releases 3.0 or later

show kdump log

Displays kernel core dump log from a locally stored file.

Command `show kdump log` *`record`* `[`*`lines`*`]`

Options

- *`record`* — Record number
- *`lines`* — (Optional) Number of lines to retrieve from the kernel log file; default 20

Command mode EXEC

Usage Use this command to view the kernel core dump file log from a file stored locally. The mandatory parameter is the number of the kernel core dump files which are stored locally. The optional parameter is the number of lines displayed (20 is the default number of lines to view).

Example

```
sonic# show kdump log 1 5
File: /var/crash/202002101809/dmesg.202002101809
[326785.222049]  [<ffffffffa0c0484e>] ? entry_SYSCALL_64_after_swapgs
+0x58/0xc6
[326785.229926] Code: 41 5c 41 5d 41 5e 41 5f e9 6c 2f cf ff 66 2e 0f 1f 84
00 00 00 00 00 66 90 0f 1f 44 00 00 c7 05 29 28 a8 00 01 00 00 00 0f ae f8
<c6> 04 25 00 00 00 00 01 c3 0f 1f 44 00 00 0f 1f 44 00 00 53 8d
[326785.251451] RIP  [<ffffffffa0a2a562>] sysrq_handle_crash+0x12/0x20
[326785.258463]  RSP <ffffafd2c6523e78>
[326785.262453] CR2: 0000000000000000

In this example, we show the kernel log for the first kernel core dump file
stored locally. We display only the first 5 lines of the log.
```

Releases 3.0 or later

show kdump memory

Displays the amount of memory reserved for kernel core dump.

Command `show kdump memory`

Options None

Command mode EXEC

Usage Use this command to show the amount of memory reserved for the kernel core dump operation.

Example

```
sonic# show kdump memory
Memory Reserved: 0M-2G:256M,2G-4G:320M,4G-8G:384M,8G-:448M
```

Releases 3.0 or later

show kdump num_dumps

Displays the maximum number of kernel core dump files that can be stored locally.

Command `show kdump num_dumps`

Options None

Command mode EXEC

Usage Use this command to show the maximum number of kernel core dump files which can be stored locally.

Example

```
sonic# show kdump num_dumps
Maximum number of Kernel Core files Stored: 3
```

Releases 3.0 or later

show kdump status

Displays kernel core dump status information.

Command `show kdump status`

Options None

Command mode EXEC

Usage Use this command to view kernel core dump status information.

Example

```
sonic# show kdump status
Kdump Administrative Mode:  Enabled
Kdump Operational State:    Ready
Memory Reserved: 512M
Maximum number of Kernel Core files Stored: 3
Record Key           Filename
-------------------------------------------------------------
     1 202002101809 /var/crash/202002101809/dmesg.202002101809
                    /var/crash/202002101809/kdump.202002101809
```

Releases 3.0 or later

show link state tracking

Displays link state tracking configuration information.

Command `show link state tracking [grp-name]`

Options `tracking grp-name` — (Optional) Group name (up to 63 characters)

Command mode EXEC

Usage Use this command to view link state tracking configuration information.

Example

```
sonic# show link state tracking
```

Releases 3.0 or later

show lldp

Displays LLDP information.

Command `show lldp`

Options None

Command mode EXEC

Usage Use this command to view all LLDP configuration information.

Example

```
sonic# show lldp
```

Releases 3.0 or later

show lldp neighbor

Displays LLDP neighbor configuration information.

Command `show lldp neighbor [ifname]`

Options `neighbor ifname` — (Optional) Interface name

Command mode EXEC

Usage Use this command to view LLDP neighbor configuration information.

Example

```
sonic# show lldp neighbor
------------------------------------------------------------
LLDP Neighbors
------------------------------------------------------------
Interface:   Ethernet72,via: LLDP
  Chassis:
    ChassisID:    54:bf:64:b8:ce:c0
    SysName:      sonic
    SysDescr:     Debian GNU/Linux 9 (stretch) Linux 4.9.0-11-2-amd64 #1
                  SMP Debian 4.9.189-3+deb9u2 (2019-11-11) x86_64
    MgmtIP:
    Capability:   MAC_BRIDGE, ON
    Capability:   ROUTER, ON
  Port
    PortID:       tenGigE1/19/2
    PortDescr:    Ethernet73
------------------------------------------------------------
Interface:   Ethernet73,via: LLDP
  Chassis:
    ChassisID:    54:bf:64:b8:ce:c0
    SysName:      sonic
    SysDescr:     Debian GNU/Linux 9 (stretch) Linux 4.9.0-11-2-amd64 #1
```

```
                    SMP Debian 4.9.189-3+deb9u2 (2019-11-11) x86_64
    MgmtIP:
    Capability:   MAC_BRIDGE, ON
    Capability:   ROUTER, ON
  Port
    PortID:       tenGigE1/19/1
    PortDescr:    Ethernet72
------------------------------------------------------------
```

Releases 3.0 or later

show lldp table

Displays brief LLDP neighbor configuration information.

Command `show lldp table`

Options None

Command mode EXEC

Usage Use this command to view brief LLDP neighbor configuration information.

Example

```
sonic# show lldp table
-------------------------------------------------------------------------------
LocalPort   RemoteDevice        RemotePortID      Capability   RemotePortDescr
-------------------------------------------------------------------------------
Ethernet0   OS10                ethernet1/1/1:1   BOR          ethernet1/1/1:1
eth0        swtor-b2lab2-1409   ethernet1/1/9     BOR          ethernet1/1/9
```

Releases 3.0 or later

show mac address-table

Displays all MAC address-table configuration information.

Command `show mac address-table`

Options None

Command mode EXEC

Usage Use this command to view all MAC address-table configuration information.

Example

```
sonic# show mac address-table
-----------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-----------------------------------------------------------
10          00:00:00:00:00:01   STATIC       Ethernet0
11          00:00:00:00:00:01   STATIC       Ethernet0
100         00:00:00:00:00:10   DYNAMIC      Ethernet36
20          00:00:00:00:00:02   DYNAMIC      Ethernet4
30          00:00:00:00:00:03   STATIC       Ethernet8
40          00:00:00:00:00:04   DYNAMIC      Ethernet12
50          00:00:00:00:00:05   STATIC       Ethernet16
60          00:00:00:00:00:06   DYNAMIC      Ethernet20
70          00:00:00:00:00:07   STATIC       Ethernet24
80          00:00:00:00:00:08   DYNAMIC      Ethernet28
90          00:00:00:00:00:09   STATIC       Ethernet32
10          00:00:00:00:00:98   STATIC       Ethernet0
99          00:00:00:00:00:99   STATIC       PortChannel10
```

Releases 3.0 or later

show mac address-table address

Displays address-table configuration information for a specific MAC address.

Command `show mac address-table address mac-addr`

Options `address mac-addr` — MAC address in nn:nn:nn:nn:nn:nn format

Command mode EXEC

Usage Use this command to view address-table configuration information for a specific MAC address.

Example

```
sonic# show mac address-table address 00:00:00:00:00:01
-------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------
10          00:00:00:00:00:01   STATIC        Ethernet0
11          00:00:00:00:00:01   STATIC        Ethernet0
```

Releases 3.0 or later

show mac address-table count

Displays address-table count information.

Command `show mac address-table count`

Options None

Command mode EXEC

Usage Use this command to view MAC address-table count information.

Example

```
sonic# show mac address-table count
MAC Entries for all vlans :  13
Dynamic Address Count :  5
Static Address (User-defined) Count :  8
Total MAC Addresses in Use: 13
```

Releases 3.0 or later

show mac address-table dynamic

Displays address-table information for dynamic MAC addresses.

Command `show mac address-table dynamic {{[address] mac-addr} | {[Vlan] vlan-id} | {[interface] {{Ethernet phy-if-id} | {PortChannel port-channel-id}}} }`

Options

- `address mac-addr` — MAC address in nn:nn:nn:nn:nn:nn format
- `Vlan vlan-id` — VLAN ID
- `Ethernet phy-if-id` — Physical interface ID
- `PortChannel port-channel-id` — PortChannel ID

Command mode EXEC

Usage Use this command to view VLAN and interface configuration information for dynamic MAC addresses.

Examples

```
sonic# show mac address-table dynamic
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
100         00:00:00:00:00:010  DYNAMIC       Ethernet36
20          00:00:00:00:00:02   DYNAMIC       Ethernet4
40          00:00:00:00:00:04   DYNAMIC       Ethernet12
```

```
60          00:00:00:00:00:06   DYNAMIC       Ethernet20
80          00:00:00:00:00:08   DYNAMIC       Ethernet28
```

```
sonic# show mac address-table dynamic address 00:00:00:00:00:06
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
60          00:00:00:00:00:06   DYNAMIC       Ethernet20
```

```
sonic# show mac address-table dynamic Vlan 60
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
60          00:00:00:00:00:06   DYNAMIC       Ethernet20
```

```
sonic# show mac address-table dynamic interface Ethernet 12
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
40          00:00:00:00:00:04   DYNAMIC       Ethernet12
```

```
sonic# show mac address-table dynamic interface PortChannel 11
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
98          00:00:00:00:00:95   DYNAMIC       PortChannel11
```

Releases 3.0 or later

show mac address-table interface

Displays MAC address-table information for static Ethernet and PortChannel interfaces.

Command `show mac address-table interface {{Ethernet` *`phy-if-id`*`} | {PortChannel` *`port-channel-id`*`}}`

Options

- `Ethernet` *`phy-if-id`* — Physical interface ID
- `PortChannel` *`port-channel-id`* — PortChannel ID

Command mode EXEC

Usage Use this command to view MAC address-table information for static Ethernet and PortChannel interfaces.

Examples

```
sonic# show mac address-table interface Ethernet 0
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
10          00:00:00:00:00:01   STATIC        Ethernet0
11          00:00:00:00:00:01   STATIC        Ethernet0
10          00:00:00:00:00:98   STATIC        Ethernet0
```

```
sonic# show mac address-table interface PortChannel 10
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
99          00:00:00:00:00:99   STATIC        PortChannel10
```

Releases 3.0 or later

show mac address-table static

Displays address-table information for static MAC addresses.

Command

```
show mac address-table static {{[address] mac-addr} | {[Vlan] vlan-id} |
{[interface] {{Ethernet phy-if-id} | {PortChannel port-channel-id}}} }
```

Options

- `address mac-addr` — MAC address in nn:nn:nn:nn:nn:nn format
- `Vlan vlan-id` — VLAN ID
- `Ethernet phy-if-id` — Physical interface ID
- `PortChannel port-channel-id` — PortChannel ID

Command mode EXEC

Usage Use this command to view VLAN and interface configuration information for static MAC addresses.

Examples

```
sonic# show mac address-table static
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
10          00:00:00:00:00:01   STATIC        Ethernet0
11          00:00:00:00:00:01   STATIC        Ethernet0
30          00:00:00:00:00:03   STATIC        Ethernet8
50          00:00:00:00:00:05   STATIC        Ethernet16
70          00:00:00:00:00:07   STATIC        Ethernet24
90          00:00:00:00:00:09   STATIC        Ethernet32
10          00:00:00:00:00:98   STATIC        Ethernet0
99          00:00:00:00:00:99   STATIC        PortChannel10
```

```
sonic# show mac address-table static address 00:00:00:00:00:01
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
10          00:00:00:00:00:01   STATIC        Ethernet0
11          00:00:00:00:00:01   STATIC        Ethernet0
```

```
show mac address-table static Vlan 11
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
11          00:00:00:00:00:01   STATIC        Ethernet0
```

```
sonic# show mac address-table static interface Ethernet 8
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
30          00:00:00:00:00:03   STATIC        Ethernet8
```

```
sonic# show mac address-table static interface PortChannel 10
-------------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------------
99          00:00:00:00:00:99   STATIC        PortChannel10
```

Releases 3.0 or later

show mac address-table Vlan

Displays the MAC address-table for a specific VLAN.

Command

```
show mac address-table Vlan vlan-id
```

Options `Vlan vlan-id` — VLAN ID

Command mode EXEC

Usage Use this command to view the MAC address-table for a specific VLAN.

Example

```
sonic# show mac address-table Vlan 10
-------------------------------------------------------
VLAN         MAC-ADDRESS         TYPE         INTERFACE
-------------------------------------------------------
10          00:00:00:00:00:01   STATIC        Ethernet0
10          00:00:00:00:00:101  STATIC        Ethernet0
```

Releases 3.0 or later

show mclag brief

Displays MCLAG domain and interface information.

Command `show mclag brief`

Options None

Command mode EXEC

Usage Use this command to view MCLAG domain and interface information.

Example

```
sonic# show mclag brief
```

Releases 3.0 or later

show mclag interface

Displays MCLAG interface information.

Command `show mclag interface ifid domain_id`

Options

- `interface ifid` — MCLAG interface ID
- `interface domain-id` — MCLAG domain ID

Command mode EXEC

Usage Use this command to view MCLAG interface information.

Example

```
sonic# show mclag interface Eth0 100
```

Releases 3.0 or later

show mirror-session

Displays configured mirror-session information.

Command `show mirror-session [session-name]`

Options `mirror-session session-name` — (Optional) Session name

Command mode EXEC

Usage Use this command to view mirror-session configuration information.

Example

```
sonic# show mirror-session
ERSPAN Sessions
------------------------------------------------------------------------------------
Name    Status SRC-IP   DST-IP   GRE    DSCP TTL Queue Policer SRC-Port  Direction
------------------------------------------------------------------------------------
```

```
Mirror2 active 11.1.1.1 10.1.1.1 0x88ee 10   10  10            Ethernet4 rx

SPAN Sessions
------------------------------------------
Name    Status DST-Port  SRC-Port  Direction
------------------------------------------
Mirror1 active Ethernet0 Ethernet4 rx
```

Releases 3.0 or later

show NeighbourSuppressStatus

Displays ARP and ND suppression information.

Command `show NeighbourSuppressStatus [id]`

Options *id* — (Optional) Neighbor ID

Command mode EXEC

Usage Use this command to view ARP and ND suppression status information.

Example

```
sonic# show NeighbourSuppressStatus
```

Releases 3.0 or later

show platform

Displays platform information.

Command `show platform`

Options None

Command mode EXEC

Usage Use this command to view all platform information.

Example

```
sonic# show platform
```

Releases 3.0 or later

show platform environment

Displays platform environment information.

Command `show platform environment`

Options None

Command mode EXEC

Usage Use this command to view all platform environment information including fan trays, temperature sensors, PSUs, and core temperatures.

Example

```
sonic# show platform environment

Fan Trays:
   Fan Tray 1:
       Fan State:                    Normal
       Fan1 Speed:                   12300 RPM
       Fan2 Speed:                   14850 RPM
   Fan Tray 2:
       Fan State:                    Normal
```

```
        Fan1 Speed:                     12150 RPM
        Fan2 Speed:                     15000 RPM
    Fan Tray 3:
        Fan State:                      Normal
        Fan1 Speed:                     12150 RPM
        Fan2 Speed:                     14850 RPM
    ...

Onboard Temperature Sensors:
    Fan U52:                           25 degrees C
        Baseboard U3:                   25 degrees C
        Fan U17:                        21 degrees C
        Near CPU:                       51 degrees C
        PSU1 hotspot:                   30 degrees C
        PSU1 inlet:                     22 degrees C
        PSU2 hotspot:                   31 degrees C
        PSU2 inlet:                     23 degrees C
        SW U04:                         28 degrees C
        SW U14:                         22 degrees C
        SW U16:                         21 degrees C
        SW U4403:                       40 degrees C
        SW U52:                         21 degrees C
        SW interal:                     43 degrees C

PSUs:
    PSU1:
        FAN RPM:                        15800 RPM
        Hotspot Temperature:            30 degrees C
        Inlet Temperature:              22 degrees C
        Input Current:                  1.04 Amps
        Input Power:                    210 Watts
        Input Voltage:                  207.90 Volts
        Output Current:                 16.50 Amps
        Output Power:                   198 Watts
        Output Voltage:                 12 Volts
    ...

coretemp-isa-0000
    Adapter: ISA adapter
        Core 0:                         +52.0 C  (high = +82.0 C, crit = +104.0
C)
        Core 1:                         +52.0 C  (high = +82.0 C, crit = +104.0
C)
        Physical id 0:                  +52.0 C  (high = +82.0 C, crit = +104.0
C)
```

Releases 3.0 or later

show platform syseeprom

Displays platform system EEPROM information.

Command `show platform syseeprom`

Options None

Command mode EXEC

Usage Use this command to view platform system EEPROM values.

Example

```
sonic# show platform syseeprom
------------------------------------------------------------
Attribute            Value/State
------------------------------------------------------------
Hardware-version    :A00
Mfg-name            :Dell
Name                :System Eeprom
Oper-status         :ACTIVE
Empty               :False
```

```
Part-no              :08YWFG
Id                   :S6000-ON
Location             :Slot 1
Removable            :False
Serial-no            :CN08YWFG282983AR0146A00
```

Releases 3.0 or later

show PortChannel summary

Displays PortChannel summary information.

Command `show PortChannel summary`

Options None

Command mode EXEC

Usage Use this command to view a summary of PortChannel configuration information.

Example

```
sonic# show PortChannel summary
Flags(oper-status):  D - Down U - Up
-------------------------------------------------------------
Group   PortChannel           Type    Protocol   Member Ports
-------------------------------------------------------------
100     PortChannel100 (D)    Eth     LACP       Ethernet84(D)
100     PortChannel200 (D)    Eth     NONE       Ethernet44(U)
100     PortChannel300 (D)    Eth     LACP       Ethernet40(U)
```

Releases 3.0 or later

show ptp

Displays all PTP status and configuration information.

Command `show ptp`

Options None

Command mode EXEC

Usage Use this information to view PTP interface and state status information.

Example

```
sonic# show ptp
-------------------------------------------------------------
Interface            State
-------------------------------------------------------------
Ethernet56           master
Ethernet64           slave
```

Releases 3.0 or later

show ptp clock

Displays PTP clock configuration information.

Command `show ptp clock`

Options None

Command mode EXEC

Usage Use this command to view PTP clock configuration and status information.

Example

```
sonic# show ptp clock
Mode                    BC
Domain Profile          ieee1588
Network Transport       UDPv4 unicast
Domain Number           1
Clock Identity          3c2c99.fffe.2d7c35
Priority1               128
Priority2               128
Two Step                Enabled
Slave Only              False
Number Ports            2
Clock Quality:
Clock Class          248
Clock Accuracy       254
Ofst Scaled Log Var 65535
Mean Path Delay        0
Steps Removed          0
Offset from master     0
```

Releases 3.0 or later

show ptp parent

Displays PTP parent status information.

Command `show ptp parent`

Options None

Command mode EXEC

Usage Use this command to view PTP parent status information.

Example

```
sonic# show ptp parent
Parent Clock Identity           3c2c99.fffe.2d7c35
Port Number                     0
Grandmaster Clock Class         248
Grandmaster Off Scaled Log Var  65535
Grandmaster Clock Accuracy      254
Grandmaster Identity            3c2c99.fffe.2d7c35
Grandmaster Priority1           128
Grandmaster Priority2           128
Stats Valid                     False
Observed Off Scaled Log Var     65535
Observed Clock Phase Chg Rate   2147483647
```

Releases 3.0 or later

show ptp port

Displays PTP port status configuration information.

Command `show ptp port {Ethernet` *`ptp_port_number`*`}`

Options `Ethernet` *`ptp_port_number`* — Ethernet PTP port number

Command mode EXEC

Usage Use this command to view PTP port status configuration information.

Example

```
sonic# show ptp port Ethernet 64
Port Number                     64
Port State                     master
Log Min delay Req Intvl         0
Peer Mean Path Delay            0
Log Announce Interval           1
```

```
Log Sync Interval               0
Log Min PDelay Req Interval     0
Version Number                  2
Unicast Master Table:
                                10.1.1.1
```

Releases 3.0 or later

show ptp time-property

Displays PTP time-property information.

Command `show ptp time-property`

Options None

Command mode EXEC

Usage Use this command to view PTP time-property configuration information.

Example

```
sonic# show ptp time-property
Curr UTC Offset Vld  False
Curr UTC Offset      37
Leap59               False
Leap61               False
Time Traceable       False
Freq Traceable       False
PTP Timescale        True
```

Releases 3.0 or later

12

show radius to switchport commands

Topics:

show radius-server

Displays RADIUS server configuration information.

Command	`show radius-server`
Options	None
Command mode	EXEC
Usage	Use this command to view the global fields, and the list of all RADIUS servers and their corresponding configuration.
Example	**Draft comment:** Need command example

```
sonic# show radius-server
```

Releases 3.0 or later

show route-map

Displays route-map configuration information.

Command `show route-map [rt-map-name]`

Options `route-map rt-map-name` — (Optional) Route-map name (up to 63 characters)

Command mode EXEC

Usage Use this command to view route-map configuration information.

Example

```
sonic# show route-map
Route map map1:
   permit, sequence 10
    Match clauses:
    Set clauses:
      local preference 10
    Call clauses:
    Actions:
      Exit routemap
Route map map2:
   permit, sequence 2
    Match clauses:
      med 10
    Set clauses:
    Call clauses:
    Actions:
      Exit routemap
```

Releases 3.0 or later

show sample

Displays all configured samples with their sampling rates.

Command `show sample [sample-name]`

Options `sample sample-name` — (Optional) Sample name (up to 63 characters)

Command mode EXEC

Usage Use this command to view all configured samples with their sampling rates.

Examples

```
sonic# show sample
-----------------------------------------------------
SAMPLE NAME    SAMPLE RATE
-----------------------------------------------------
s1             200
s2             5000
```

```
sonic# show sample s1
-----------------------------------------------------
SAMPLE NAME    SAMPLE RATE
-----------------------------------------------------
s1             200
```

Releases 3.0 or later

show sflow

Displays global sFlow configuration information.

Command `show sflow`

Options None

Command mode EXEC

Usage Use this command to view global sFlow configuration information.

Example

```
sonic # show sflow
------------------------
Global sFlow Information
------------------------
admin state:            up
polling-interval:       44
agent-id:               default
configured collectors:  1
   collector_1    190.167.1.1        6343
```

Releases 3.0 or later

show sflow interface

Displays sFlow interface configuration information.

Command `show sflow interface`

Options None

Command mode EXEC

Usage Use this command to view sFlow interface configuration information.

Example

```
sonic# show sflow interface
------------------------------------------------------------
sFlow interface configurations
Interface            Admin State             Sampling Rate
Ethernet0            up                      4000
Ethernet1            up                      4000
Ethernet2            up                      4000
Ethernet3            up                      4000
Ethernet4            up                      4000
Ethernet5            up                      4000
Ethernet6            up                      4000
Ethernet7            up                      4000
Ethernet8            up                      4000
Ethernet9            up                      4000
Ethernet10           up                      4000
Ethernet11           up                      4000
Ethernet12           up                      4000
Ethernet13           up                      4000
Ethernet14           up                      4000
Ethernet15           up                      4000
Ethernet16           up                      4000
Ethernet17           up                      4000
Ethernet18           up                      4000
Ethernet19           up                      4000
Ethernet20           up                      4000
Ethernet21           up                      4000
Ethernet22           up                      4000
Ethernet23           up                      4000
Ethernet24           up                      4000
Ethernet25           up                      4000
Ethernet26           up                      4000
Ethernet27           up                      4000
```

```
Ethernet28           up                    4000
Ethernet29           up                    4000
```

Releases 3.0 or later

show snmp-server

Displays global SNMP server information.

Command `show snmp-server`

Options None

Command mode EXEC

Usage Use this command to view simple network management protocol (SNMP) server information including the physical location of the switch, the organization responsible for the network, SNMP engine identification, trap status and the agent addresses, if configured. SNMP engine identification is derived from the device MAC address on an initial boot.

Example

```
sonic# show snmp-server
Location   : Lab1, Rack-10
Contact    : Dell Support
EngineID   : 800001370352540 0f6817e
Traps      : enable

Agent Addresses:

               IP Address               UDP Port            Interface
------------------------------------ --------
---------------------------------
1.2.3.4                                 161
1.2.3.4                                 1024
1.2.3.5                                 1024     Ethernet10
```

Releases 3.0 or later

show snmp-server community

Displays the configured SNMP communities.

Command `show snmp-server community`

Options None

Command mode EXEC

Usage Use this command to view the SNMP communities configured on the switch and the community group association, if configured. Communities are used by SNMPv2 protocol to access the switch.

Example

```
sonic# show snmp-server community
         Community Name                  Group Name
------------------------------ ------------------------------
comm1                          group-lab
comm2                          None
```

Releases 3.0 or later

show snmp-server group

Displays the configured SNMP groups.

Command `show snmp-server group`

Options None

Command mode EXEC

Usage Use this command to view the SNMP groups configured on the switch. The model and security information indicate the SNMP protocol and security level used to access the system via the group. View names indicate the view that a group provides read, write or trap access to.

Example

```
sonic# show snmp-server group
   Group Name      Model: Security       Read View        Write View
Notify View
---------------- -------------------- ---------------- ----------------
----------------
group-floor1     v2c: no-auth-no-priv ro_view          wr_view          None
group-floor2     v3 : auth-priv       r_view           None             None
group-lab        v2c: no-auth-no-priv None             None             None
```

Releases 3.0 or later

show snmp-server host

Displays the configured SNMP hosts.

Command `show snmp-server host`

Options None

Command mode EXEC

Usage Use this command to view the SNMP hosts to which the trap or inform messages are sent by the SNMP agent. Timeout indicates the number of seconds before the traps/informs time out when sending to a host. Retries indicate the number of times the traps/informs are sent after timing out.

Example

```
sonic# show snmp-server host
             Target Address               Type    Community    Ver T-Out
Retries
---------------------------------------- ------ -------------- --- -----
-------
1.2.3.4                                  trap   comm1          v2c 15    3


             Target Address               Type    User Name
Security    T-Out Retries
---------------------------------------- ------ --------------
--------------- ----- -------
1001::1                                  inform user1          auth-
priv       200   10
```

Releases 3.0 or later

show snmp-server user

Displays the configured SNMPv3 users.

Command `show snmp-server user`

Options None

Command mode EXEC

Usage Use this command to view the SNMPv3 users configured on the switch including any authentication and/or encryption algorithm for the user. The group name indicates a group that defines the SNMPv3 access parameters.

Example

```
sonic# show snmp-server user
          User Name                        Group Name            Auth
```

```
Privacy
------------------------------ ------------------------------ -----
-------
user1                          group-lab                      md5
aes-128
user2                          group-floor2                   None  None
```

Releases 3.0 or later

show snmp-server view

Displays SNMP views configuration information.

Command `show snmp-server view`

Options None

Command mode EXEC

Usage Use this command to display SNMP views configured on the switch including the OID tree that the view includes or excludes.

Example

```
sonic# show snmp-server view
           View Name                        OID Tree                 Type
------------------------------ ------------------------------ --------
view1                          1.2.3.4.5.6.7.8.9.1            included
view2                          1.2.3.4.5.6.7.8.9.5.1          excluded
```

Releases 3.0 or later

show spanning-tree

Displays spanning-tree configuration information.

Command `show spanning-tree [vlan] {`*`vlan-id`* `{[interface]` *`name`*`}}`

Options

- `vlan` *`vlan-id`* — (Optional) VLAN ID
- `interface` *`name`* — (Optional) Interface name

Command mode EXEC

Usage Use this command to view spanning-tree configuration information.

Example

```
sonic# show spanning-tree

Spanning-tree Mode: PVST

VLAN 100 - STP instance 0
------------------------------------------------------------------------------
STP Bridge Parameters:

Bridge            Bridge    Bridge    Bridge    Hold  LastTopology  Topology
Identifier        MaxAge    Hello     FwdDly    Time  Change        Change
hex               sec       sec       sec       sec   sec           cnt
80643c2c99a704a0  20        2         15        1     515           4

RootBridge        RootPath  DesignatedBridge  Root        Max  Hel  Fwd
Identifier        Cost      Identifier        Port        Age  lo   Dly
hex                         hex                           sec  sec  sec
00646cb9c51613ca  1600      10643c2c992d8235  PortChannel1120   2    15

STP Port Parameters:
Port         Prio Path Port Uplink BPDU   State      Designated Designated
Designated
Num          rity Cost Fast Fast   Filter            Cost       Root             bridg
```

```
PortChannel1 128  800  N    N      N      FORWARDING 800        00646cb9c51613ca
10643c2c992d8235
```

Releases 3.0 or later

show spanning-tree bpdu-guard

Displays spanning-tree BPDU guard information for the ports.

Command `show spanning-tree bpdu-guard`

Options None

Command mode EXEC

Usage Use this command to view spanning-tree BPDU guard port information.

Example

```
sonic# show spanning-tree bpdu-guard
PortNum          Shutdown    Port shut
                 Configured  due to BPDU guard
Ethernet64        Y           N
PortChannel2      Y           N
```

Releases 3.0 or later

show spanning-tree counters

Displays spanning-tree counter information.

Command `show spanning-tree counters [vlan] vlan-id`

Options `vlan vlan-id` — (Optional) VLAN ID

Command mode EXEC

Usage Use this command to view spanning-tree counter information.

Example

```
sonic# show spanning-tree counters

VLAN 100 - STP instance 0
--------------------------------------------------------------------------
PortNum          BPDU Tx    BPDU Rx    TCN Tx     TCN Rx
--------------------------------------------------------------------------
Ethernet0        0          0          0          0
Ethernet48       53         7          1          0
Ethernet62       0          0          0          0
Ethernet64       0          0          0          0
PortChannel1     0          76         1          0
PortChannel2     0          0          0          0
PortChannel3     0          0          0          0
```

Releases 3.0 or later

show spanning-tree inconsistentports

Displays spanning-tree root guard inconsistent port information.

Command `show spanning-tree inconsistentports [vlan] vlan-id`

Options `vlan vlan-id` — (Optional) VLAN ID

Command mode EXEC

Usage Use this command to view spanning-tree root guard inconsistent port information.

Example

```
sonic# show spanning-tree inconsistentports

Root guard timeout: 30  secs

------------------------------------------------------------------------
PortNum          VLAN  Inconsistency State
------------------------------------------------------------------------
Ethernet48       100   Root Inconsistent (29 seconds left on timer)
```

Releases 3.0 or later

show system

Displays system information.

Command `show system`

Options None

Command mode EXEC

Usage Use this command to view system information.

Example

```
sonic# show system
-------------------------------------------------------------
Attribute            Value/State
-------------------------------------------------------------
Hostname            :sonic
Boot Time           :1563911113
Current Datetime    :2019-07-24 18:34:43+00:00
Domain Name         :None
```

Releases 3.0 or later

show system cpu

Displays system CPU information.

Command `show system cpu`

Options None

Command mode EXEC

Usage Use this command to view system CPU information.

Example

```
sonic# show system cpu
------------------------------------------------------------------------
CPU                 %KERNEL             %USER               %IDLE
------------------------------------------------------------------------
CPU-total            17                   17                   62
CPU-1                17                   17                   62
CPU-2                17                   16                   63
CPU-3                17                   16                   62
CPU-4                17                   17                   61
```

Releases 3.0 or later

show system memory

Displays system memory information.

Command `show system memory`

Options None

Command mode EXEC

Usage Use this command to view system memory information.

Example

```
sonic# show system memory
-------------------------------------------------------
Attribute            Value/State
-------------------------------------------------------
Used                :1304976
Total               :8162872
```

Releases 3.0 or later

show system processes

Displays all system processes information.

Command `show system processes`

Options None

Command mode EXEC

Usage Use this command to view all system processes information.

Example

```
sonic# show system processes
-------------------------------------------------------------
PID        %CPU    %MEMORY    MEM-USAGE(Bytes)    NAME
-------------------------------------------------------------
1          0        0        58761216    /sbin/init
10         0        0        0           [lru-add-drain]
100        0        0        0           [scsi_eh_0]
1000       0        0        409763840   docker
101        0        0        0           [scsi_tmf_0]
10179      0        0        12124160    /bin/bash
102        0        0        0           [scsi_eh_1]
10217      0        0        42135552    python
103        0        0        0           [scsi_tmf_1]
107        0        0        0           [bioset]
10862      0        0        256139264   /usr/sbin/rsyslogd
109        0        0        0           [kworker/3:1H]
11         0        0        0           [watchdog/0]
110        0        0        0           [kworker/2:1H]
11044      0        0        111427584   containerd-shim
11088      0        0        109920256   containerd-shim
111        0        0        0           [kworker/0:1H]
11119      0        0        109920256   containerd-shim
11140      0        0        59592704    /usr/bin/python
11177      0        0        111362048   containerd-shim
112        0        0        0           [kworker/1:1H]
11204      0        0        59600896    /usr/bin/python
11223      0        0        61095936    /usr/bin/python
11272      0        0        189263872   /usr/bin/orchagent
11308      0        0        58249216    /usr/bin/python
```

Releases 3.0 or later

show system processes pid

Displays system process information for a specific process ID.

Command `show system processes pid` *pid-no*

Options `pid` *pid-no* — Process ID

Command mode EXEC

Usage Use this command to view system process information for a specific process ID.

Example

```
sonic# show system processes pid 1
-----------------------------------------
Attribute            Value/State
-----------------------------------------
Memory Usage        :58761216
Uptime              :84024
Start Time          :1563911114000000000
Name                :/sbin/init
Args                :None
Pid                 :1
Memory Utilization  :0
Cpu Utilization     :0
Cpu Usage System    :494
Cpu Usage User      :443
```

Releases 3.0 or later

show tacacs-server global

Displays global TACACS+ server information.

Command `show tacacs-server global`

Options None

Command mode EXEC

Usage Use this command to view global TACACS+ server information.

Example

```
sonic# show tacacs-server global
 --------------------------
TACACS Global Configuration
--------------------------
key        : mykey
```

Releases 3.0 or later

show tacacs-server host

Displays TACACS+ server host information.

Command `show tacacs-server host [address]`

Options `host address` — (Optional) Host IP address in A.B.C.D format

Command mode EXEC

Usage Use this command to view TACACS+ server host information.

Examples

```
sonic# show tacacs-server host
-------------------------------------------------------------------------------
HOST                AUTH-TYPE       KEY        PORT       PRIORITY   TIMEOUT
-------------------------------------------------------------------------------
1.1.1.1             pap             mykey      11         11         10
```

```
sonic# show tacacs-server host 1.1.1.1
-------------------------------------------------------------------------------
HOST                AUTH-TYPE       KEY        PORT       PRIORITY   TIMEOUT
-------------------------------------------------------------------------------
1.1.1.1             pap             mykey      11         11         10
```

Releases 3.0 or later

show tam collector

Displays TAM collector information.

Command `show tam collector [name]`

Options `collector name` — (Optional) TAM collector name (up to 63 characters)

Command mode EXEC

Usage Use this command to view TAM collector information.

Examples

```
sonic#  show tam collector
------------------------------------------------
NAME            IP TYPE         IP ADDRESS      PORT
------------------------------------------------
cnew            ipv4            1.1.1.1         10
```

```
sonic# show tam collector cnew
------------------------------------------------
NAME            IP TYPE         IP ADDRESS      PORT
------------------------------------------------
cnew            ipv4            1.1.1.1         10
```

Releases 3.0 or later

show tam device

Displays the TAM device identifier.

Command `show tam device`

Options None

Command mode EXEC

Usage Use this command to view the configured TAM device ID.

Example

```
sonic# show tam device
----------------------
TAM Device Information
----------------------
device-id: 10
```

Releases 3.0 or later

show tam drop-monitor aging-interval

Displays system-level drop-monitor aging interval configuration information.

Command `show tam drop-monitor aging-interval`

Options None

Command mode EXEC

Usage Use this command to view the system-level drop-monitor aging interval configured.

Example

```
sonic# show tam drop-monitor aging-interval
Aging interval : 6 seconds
```

Releases 3.0 or later

show tam drop-monitor flow

Displays all configured drop-monitor flows and their corresponding flows.

Command `show tam drop-monitor flow [flow-name]`

Options `flow flow-name` — (Optional) Drop-monitor flow name (up to 63 characters)

Command mode EXEC

Usage Use this command to view information about all configured flows, or a specific flow identified by flow-name.

Examples

```
sonic# show tam drop-monitor flow
------------------------------------------------------------------------
FLOW     ACL TABLE     ACL RULE     COLLECTOR     SAMPLE     FLOWGROUP ID
------------------------------------------------------------------------
f2       t2            r2           c1            s1         5
f3       t3            r3           c1            s1         6
f4       t4            r4           c1            s1         4
```

```
sonic# show tam drop-monitor flow f2
------------------------------------------------------------------------
FLOW     ACL TABLE     ACL RULE     COLLECTOR     SAMPLE     FLOWGROUP ID
------------------------------------------------------------------------
f2       t2            r2           c1            s1         5
```

Releases 3.0 or later

show tam drop-monitor statistics

Displays per-flow statistics for drop-monitor.

Command `show tam drop-monitor statistics [flow-name]`

Options `statistics flow-name` — (Optional) Drop-monitor flow name (up to 63 characters)

Command mode EXEC

Usage Use this command to view match statistics for the given flow.

Example

```
sonic# show tam drop-monitor statistics
------------------------------------------------------------------------------
FLOW            ACL TABLE      ACL RULE       PACKET COUNT    BYTE COUNT
------------------------------------------------------------------------------
f2              t2             r2             0               0
f3              t3             r3             0               0
f4              t4             r4             0               0
```

Releases 3.0 or later

show tam drop-monitor supported

Displays drop-monitor functionality support.

Command `show tam drop-monitor supported`

Options None

Command mode EXEC

Usage Use this command to view the status of the drop-monitor feature.

Example

```
sonic# show tam drop-monitor supported
Feature Supported : True
```

Releases 3.0 or later

show tam int-ifa flow

Displays IFA flow configuration.

Command `show tam int-ifa flow [name]`

Options `flow name` — (Optional) IFA flow name (up to 63 characters)

Command mode EXEC

Usage Use this command to view IFA flow configuration.

Example

```
sonic# show tam int-ifa flow
------------------------------------------------------------
FLOW     ACL TABLE     ACL RULE     SAMPLING RATE  COLLECTOR
------------------------------------------------------------
fnew     acl1          rule1        10              cnew
```

```
sonic# show tam int-ifa flow fnew
------------------------------------------------------------
FLOW     ACL TABLE     ACL RULE     SAMPLING RATE  COLLECTOR
------------------------------------------------------------
fnew     acl1          rule1        10              cnew
```

Releases 3.0 or later

show tam int-ifa statistics

Displays the IFA flow statistics.

Command `show tam int-ifa statistics [name]`

Options `statistics name` — (Optional) IFA flow name (up to 63 characters)

Command mode EXEC

Usage Use this command to view packet count and byte count for matching IFA flow.

Examples

```
sonic# show tam int-ifa statistics
------------------------------------------------------------------------------
FLOW           ACL TABLE      ACL RULE       PACKET COUNT   BYTE COUNT
------------------------------------------------------------------------------
flow1          t1             r1             0              0
flow2          t2             r2             0              0
```

```
sonic# show tam int-ifa statistics flow1
------------------------------------------------------------------------------
FLOW           ACL TABLE      ACL RULE       PACKET COUNT   BYTE COUNT
------------------------------------------------------------------------------
flow1          t1             r1             0
```

Releases 3.0 or later

show tam int-ifa status

Displays the overall IFA status.

Command `show tam int-ifa status`

Options None

Command mode EXEC

Usage Use this command to view the overall IFA status.

Example

```
sonic# show tam int-ifa status
------------------------------------------------------
TAM/IFA Status
------------------------------------------------------
Device Identifier    : 7765
Number of collectors : 1
Number of flows      : 1
Feature Enabled      : True
```

Releases 3.0 or later

show tam int-ifa supported

Displays IFA feature status.

Command `show tam int-ifa supported`

Options None

Command mode EXEC

Usage Use this command to display IFA feature status.

Example

```
sonic# show tam int-ifa supported
---------------------------
TAM IFA Feature Information
---------------------------
IFA Feature Supported: True
```

Releases 3.0 or later

show tam int-ifa-ts flow

Displays IFA tail timestamping flow configuration information.

Command `show tam int-ifa-ts flow [name]`

Options `flow name` — (Optional) Flow name to display configuration information (up to 63 characters)

Command mode EXEC

Usage Use this command to view flow configuration.

Examples

```
sonic# show tam int-ifa-ts flow
----------------------------------------
FLOW           ACL TABLE      ACL RULE
----------------------------------------
flow1          t1             r1
flow2          t2             r2
```

```
sonic# show tam int-ifa-ts flow flow1
----------------------------------------
FLOW           ACL TABLE      ACL RULE
----------------------------------------
flow1          t1             r1
```

Releases 3.0 or later

show tam int-ifa-ts statistics

Displays IFA tail timestamping flow statistics.

Command `show tam int-ifa-ts statistics [name]`

Options `statistics name` — (Optional) Flow name to display statistics (up to 63 characters)

Command mode EXEC

Usage Use this command to view IFA tail timestamping flow statistics.

Examples

```
sonic# show tam int-ifa-ts statistics
------------------------------------------------------------
FLOW     ACL TABLE    ACL RULE    PACKET COUNT    BYTE COUNT
------------------------------------------------------------
flow1    t1           r1          0               0
flow2    t2           r2          0               0
```

```
sonic# show tam int-ifa-ts statistics flow1
------------------------------------------------------------
FLOW     ACL TABLE    ACL RULE    PACKET COUNT    BYTE COUNT
------------------------------------------------------------
flow1    t1           r1          0               0
```

Releases 3.0 or later

show tam int-ifa-ts status

Displays IFA tail timestamping overall status.

Command `show tam int-ifa-ts status`

Options None

Command mode EXEC

Usage Use this command to view the overall status of IFA tail timestamping.

Example

```
sonic# show tam int-ifa-ts status
------------------------------------------------------------
TAM INT IFA TS Status
------------------------------------------------------------
Device Identifier    : 2345
Number of flows      : 2
Feature Enabled      : True
```

Releases 3.0 or later

show tam int-ifa-ts supported

Displays IFA tail timestamping status.

Command `show tam int-ifa-ts supported`

Options None

Command mode EXEC

Usage After enabling IFA tail timestamping, use this command to view the feature status.

Examples

```
sonic(config)# tam
sonic(config-tam)# int-ifa-ts
```

```
sonic(config-int-ifa-ts)# feature enable
sonic(config-int-ifa-ts)# exit
```

```
sonic# show tam int-ifa-ts supported
Feature Supported : True
```

Releases 3.0 or later

show tech-support

Collects technical support information.

Command `show tech-support [since] date`

Options `since date` — (Optional) Date to collection technical support information since

Command mode EXEC

Usage Use this command to collect technical support information.

Example

```
sonic# show tech-support
```

Releases 3.0 or later

show udld global

Displays global-level UDLD information.

Command `show udld global`

Options None

Command mode EXEC

Usage After enabling UDLD at a global-level or modifying UDLD attributes, use this command to check global-level UDLD information.

Examples

```
sonic(config)# udld enable
sonic(config)# exit
```

```
sonic# show udld global
UDLD Global Information
  Admin State        : UDLD Enabled
  Mode               : Normal
  UDLD Message Time  : 1 seconds
  UDLD Multiplier    : 3
```

Releases 3.0 or later

show udld interface

Displays UDLD information and neighbors detail for a specific interface.

Command `show udld interface interface-name`

Options `interface interface-name` — Name of interface to display UDLD information

Command mode EXEC

Usage After enabling UDLD at an interface-level, use this command to view UDLD information and neighbors attached to this interface.

Example

```
sonic(config)# udld enable
sonic(config)# interface Ethernet 0
sonic(conf-if-Ethernet0)# udld enable
sonic(conf-if-Ethernet0)# exit
```

```
sonic# show udld interface Ethernet0
UDLD information for Ethernet0
  UDLD Admin State:                  Enabled
  Mode:                              Normal
  Status:                            Bidirectional
  Local device id:                   3c2c.992d.8201
  Local port id :                    Ethernet0
  Local device name:                 Sonic
  Message time:                      1
  Timeout interval:                  3
    Neighbor Entry 1
    ------------------------------------------------------
    Neighbor device id:         3c2c.992d.8235
    Neighbor port id:           Ethernet0
    Neighbor device name:       Sonic
    Neighbor message time:      1
    Neighbor timeout interval:  3
```

Releases 3.0 or later

show udld neighbors

Displays UDLD neighbors information

Command `show udld neighbors`

Options None

Command mode EXEC

Usage After enabling UDLD at global and interface levels, use this command to view all UDLD neighbors information.

Examples

```
sonic(config)# udld enable
sonic(config)# interface Ethernet 0
sonic(conf-if-Ethernet0)# udld enable
sonic(conf-if-Ethernet0)# exit
```

```
sonic# show udld neighbors
Port         Device Name    Device ID         Port ID      Neighbor State
-------------------------------------------------------------------------
Ethernet1    Sonic          3c2c.992d.8201    Ethernet0    Bidirectional
Ethernet3    Sonic          3c2c.992d.8201    Ethernet3    Bidirectional
```

Releases 3.0 or later

show udld statistics

Displays UDLD statistics for all interfaces.

Command `show udld statistics`

Options None

Command mode EXEC

Usage Use this command to get UDLD statistics for all interfaces.

Example

```
sonic# show udld statistics
UDLD Interface statistics for Ethernet0
Frames transmitted:         10
```

```
Frames received:            9
Frames with error:          0

UDLD Interface statistics for Ethernet1
Frames transmitted:         5
Frames received:            8
Frames with error:          0
```

Releases 3.0 or later

show udld statistics interface

Displays UDLD statistics for a specific interface.

Command `show udld statistics interface interface-name`

Options `interface interface-name` — Interface name to view UDLD statistics

Command mode EXEC

Usage Use this command to view UDLD statistics for a specific interface.

Example

```
sonic# show udld statistics interface Ethernet0
UDLD Interface statistics for Ethernet0
Frames transmitted:         10
Frames received:            9
Frames with error:          0
```

Releases 3.0 or later

show version

Displays software version information.

Command `show version`

Options None

Command mode EXEC

Usage Use this command to view software version information.

Example

```
sonic# show version

SONiC Software Version: SONiC-OS-3.0.0-Cloud-Premium
Product: Dell EMC OpenFabric Powered by SONiC
```

Releases 3.0 or later

show Vlan

Displays the current VLAN configuration.

Command `show Vlan [id]`

Options `Vlan id` — (Optional) VLAN ID (1 to 4093)

Command mode EXEC

Usage Use this command to view the current VLAN configuration, or for a specific VLAN interface.

Examples

```
sonic# show Vlan
Q: A - Access (Untagged), T - Tagged
    NUM        Status       Q Ports
```

```
    5         Active       T Ethernet24
    10        Inactive
    20        Inactive     A PortChannel20
```

```
sonic# show Vlan 5
Q: A - Access (Untagged), T - Tagged
    NUM    Status     Q Ports
    5      Active     T Ethernet24
                      T PortChannel10
                      A Ethernet20
```

Releases 3.0 or later

show vxlan interface

Displays VxLAN VTEP SIP information.

Command `show vxlan interface`

Options None

Command mode EXEC

Usage Use this command to view the name, SIP, associated NVO name, and the loopback interface configured with the VTEP SIP.

Example

```
sonic# show vxlan interface

VTEP Information:

    VTEP Name : VTEP1, SIP  : 4.4.4.4
    NVO Name  : nvo1,  VTEP : VTEP1
    Source interface  : Loopback33
```

Releases 3.0 or later

show vxlan remote mac

Displays all MACs learned from the specified remote IP, or all remote IPs for a specific/all VLANs.

Command `show vxlan remote mac [remote_ip_addr]`

Options `remote mac remote_ip_addr` — (Optional) Remote IP address in A.B.C.D format

Command mode EXEC

Usage Use this command to view all MACs learned from the specified remote IP, or all remote IPs for a specific/all VLANs..

Examples

```
sonic# show vxlan remote mac
+---------+-------------------+-------------+-------+--------+
| VLAN    | MAC               | RemoteVTEP  |   VNI | Type   |
+=========+===================+=============+=======+========+
| Vlan101 | 00:00:00:00:00:01 | 4.4.4.4     |  1001 | static |
+---------+-------------------+-------------+-------+--------+
| Vlan101 | 00:00:00:00:00:02 | 3.3.3.3     |  1001 | static |
+---------+-------------------+-------------+-------+--------+
| Vlan101 | 00:00:00:00:00:03 | 4.4.4.4     |  1001 | static |
+---------+-------------------+-------------+-------+--------+
| Vlan101 | 00:00:00:00:00:04 | 4.4.4.4     |  1001 | static |
+---------+-------------------+-------------+-------+--------+
| Vlan101 | 00:00:00:00:00:05 | 4.4.4.4     |  1001 | static |
+---------+-------------------+-------------+-------+--------+
| Vlan101 | 00:00:00:00:00:99 | 3.3.3.3     |  1001 | static |
```

```
+---------+-------------------+-------------+-------+--------+
Total count : 6
```

```
sonic# show vxlan remote mac 3.3.3.3
+---------+-------------------+-------------+-------+--------+
| VLAN    | MAC               | RemoteVTEP  |   VNI | Type   |
+=========+===================+=============+=======+========+
| Vlan101 | 00:00:00:00:00:02 | 3.3.3.3     |  1001 | static |
+---------+-------------------+-------------+-------+--------+
| Vlan101 | 00:00:00:00:00:99 | 3.3.3.3     |  1001 | static |
+---------+-------------------+-------------+-------+--------+
Total count : 2
```

Releases 3.0 or later

show vxlan remote vni

Displays all VLANs learned from the specified remote IP or all remote IPs.

Command `show vxlan remote vni [remote_ip_addr]`

Options `vni remote_ip_addr` — (Optional) Remote IP address in A.B.C.D format

Command mode EXEC

Usage Use this command to view all VLANs learned from the specified remote IP, or all remote IPs.

Examples

```
sonic# show vxlan remote vni
+---------+-------------+-------+
| VLAN    | RemoteVTEP  |   VNI |
+=========+=============+=======+
| Vlan101 | 3.3.3.3     |  1001 |
+---------+-------------+-------+
| Vlan101 | 4.4.4.4     |  1001 |
+---------+-------------+-------+
Total count : 2
```

```
sonic# show vxlan remote vni 3.3.3.3
+---------+-------------+-------+
| VLAN    | RemoteVTEP  |   VNI |
+=========+=============+=======+
| Vlan101 | 3.3.3.3     |  1001 |
+---------+-------------+-------+
Total count : 1
```

Releases 3.0 or later

show vxlan tunnel

Displays all discovered tunnels.

Command `show vxlan tunnel`

Options None

Command mode EXEC

Usage Use this command to view all discovered tunnels.

Example

```
sonic# show vxlan tunnel
+---------+---------+-------------------+-------------+
| SIP     | DIP     | Creation Source   | OperStatus  |
+=========+=========+===================+=============+
| 2.2.2.2 | 4.4.4.4 | EVPN              | oper_up     |
+---------+---------+-------------------+-------------+
```

```
| 2.2.2.2 | 3.3.3.3 | EVPN                | oper_up      |
+---------+---------+---------------------+--------------+
Total count : 2
```

Releases 3.0 or later

show vxlan vlanvnimap

Displays all VLAN VNI mappings.

Command `show vxlan vlanvnimap`

Options None

Command mode EXEC

Usage Use this command to view all VLAN VNI mappings.

Example

```
sonic# show vxlan vlanvnimap
+---------+-------+
| VLAN    |   VNI |
+=========+=======+
| Vlan100 |   100 |
+---------+-------+
| Vlan101 |   101 |
+---------+-------+
Total count : 2
```

Releases 3.0 or later

show vxlan vrfvnimap

Displays all VRF VNI mappings.

Command `show vxlan vrfvnimap`

Options None

Command mode EXEC

Usage Use this command to view all VRF VNI mappings.

Example

```
sonic# show vxlan vrfvnimap
+-------+-------+
| VRF   |   VNI |
+=======+=======+
| Vrf1  |   600 |
+-------+-------+
Total count : 1
```

Releases 3.0 or later

show ztp-status

Displays the current zero-touch provisioning (ZTP) status.

Command `show ztp-status`

Options None

Command mode EXEC

Usage Use this command to show the status of ZTP. These are the possible current states or result of a ZTP session:

- IN-PROGRESS — ZTP session is currently in progress. ZTP service is processing switch provisioning information
- FAILED — ZTP service has failed to process the switch provisioning information
- SUCCESS — ZTP service has successfully processed the switch provisioning information
- Not Started — ZTP service has not started processing the discovered switch provisioning information

These are the state and result of a configuration session:

- IN-PROGRESS — Corresponding configuration session is currently being processed
- SUCCESS — Corresponding configuration session was processed successfully
- FAILED — Corresponding configuration session failed to execute successfully
- Not Started — ZTP service has not started processing the corresponding configuration session
- DISABLED — Corresponding configuration session has been marked as disabled and will not be processed

Example

```
sonic# show ztp-status
=======================================
ZTP
=======================================
ZTP Admin Mode      : True
ZTP Service         : Inactive
ZTP Status          : Not Started
```

Releases 3.0 or later

shutdown

Administratively shuts down a BGP neighbor or peer.

Command `shutdown {[message] MSG}`

Options `message MSG` — (Optional)

Command modes

- INTERFACE
- BGP-PEER
- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to mark a physical interface or BGP peer or neighbor as unavailable for traffic. Disabling a VLAN or a PortChannel causes different behavior. When you disable a VLAN, the L3 functions within that VLAN are disabled, and L2 traffic continues to flow. Use this command on a PortChannel to disable all traffic on the PortChannel, and the individual interfaces. The `shutdown` and `description` commands are the only commands that you can configure on an interface that is a PortChannel member. This command also changes the BFD session state to DOWN, disables the interface, and administratively shuts down a BGP neighbor or peer-group sessions. The `no` version of this command enables an interface, BGP peer, neighbor, or peer-group.

Examples

```
sonic(config)# bfd
sonic(conf-bfd)# peer 192.168.0.5 interface Ethernet0
sonic(conf-bfd-peer)# shutdown
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 30.30.30.3
sonic(config-router-bgp-neighbor)# shutdown
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Ext
sonic(config-router-bgp-pg)# shutdown
```

Releases 3.0 or later

snmp-server agentaddress

Configures one or more SNMP agent addresses.

Command

```
snmp-server agentaddress host_ip_addr {{[port] udp_port} {[interface] interface_name}}
```

Options

- `agentaddress host_ip_addr` — Host IP address in A.B.C.D or A:B:C:D:E:F:G:H format
- `port udp_port` — (Optional) UDP port number; default 161
- `interface interface_name` — (Optional) Interface name (up to 32 characters)

Command mode CONFIGURATION

Usage Use this command to configure an SNMP agent address, UDP port number, and VRF interface. When the Management VRF is enabled, you must configure the SNMP agent to listen to the Management VRF. You can also set the UDP port number on which the SNMP server listens for requests, and the VRF interface used by the Management port to access SNMP. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# snmp-server agentaddress 1.2.3.4
```

```
sonic(config)# snmp-server agentaddress 1.2.3.4 port 1024
```

```
sonic(config)# snmp-server agentaddress 1.2.3.5 port 1024 interface
Ethernet10
```

Releases 3.0 or later

snmp-server community

Configures an SNMP user community.

Command

```
snmp-server community community_name {[group] group_name}
```

Options

- *community_name* — Community name string for to act as a password for SNMP server access (up to 32 characters)
- *group_name* — (Optional) Group name string for SNMP server access (up to 32 characters)

Command mode CONFIGURATION

Usage Use this command to configure an SNMP community and group. Configure one or more SNMP communities and optionally associate them with a group. The `no` version of this command removes the configured community text string.

Example

```
sonic(config)# snmp-server community comm1
```

```
sonic(config)# snmp-server community comm1 group group-lab
```

Releases 3.0 or later

snmp-server contact

Configures contact information for troubleshooting the local SNMP switch.

Command

```
snmp-server contact contact_name
```

Options *contact_name* — Contact name (up to 32 characters; default support)

Command mode CONFIGURATION

Usage Use this command to configure the SNMP server contact information. The `no` version of this command resets the SNMP server contact to support.

Example

```
sonic(config)# snmp-server contact administrator
```

Releases 3.0 or later

snmp-server enable trap

Enables SNMP traps on a switch.

Command `snmp-server enable trap`

Options None

Command mode CONFIGURATION

Usage Use this command to enable SNMP traps. The command is disabled by default. The `no` version of this command disables SNMP traps on the switch.

Example

```
sonic(config)# snmp-server enable trap
```

Releases 3.0 or later

snmp-server engine

Configures the SNMP engine ID.

Command `snmp-server engine` *`engineID`*

Options *`engineID`* — Engine ID to identify the local SNMP agent on the switch as an octet colon-separated number (5 to 32 octets)

Command mode CONFIGURATION

Usage Use this command to configure the SNMP engine ID used for localizing configuration. The engine ID generates the localized keys for the authentication and privilege passwords. These passwords authenticate SNMP users and encrypt SNMP messages. If you reconfigure the engine ID, the localized keys also change, and the existing values are no longer valid. You must reconfigure SNMP users with new localized password keys. The default engine ID is derived from the device MAC address of the Management interface. The `no` version of this command resets the default engine ID values.

Example

```
sonic(config)# snmp-server engine 80:00:02:b8:04:61:62:63
```

Releases 3.0 or later

snmp-server group

Configures the views allowed for SNMP group users.

Command `snmp-server group` *`group_name`* `{{any | v2c | {v3 {noauth | auth | priv}}} {[read]` *`view_name`*`} {[write]` *`view_name`*`} {[notify]` *`view_name`*`}}`

Options

- `group` *`group_name`* — Name of the group (up to 32 characters)
- `any` — Use any authentication method on the group
- `v2c` — Use no user authentication or privacy protection on the group
- `v3` — Use optional user authentication and encryption for SNMP messages on the group
- `auth` — Authenticate group users in SNMP messages
- `noauth` — Do not authenticate group users or encrypt SNMP messages
- `priv` — Authenticate group users and encrypt or decrypt SNMP messages
- `read` *`view_name`* — Name of a read-only view (up to 32 characters)
- `write` *`view_name`* — Name of a write-only view (up to 32 characters)

- `notify view_name` — Name of a notification view (up to 32 characters)

Command mode CONFIGURATION

Usage Use this command to set up the access privileges for a group of SNMP users. Configure the security level for receiving SNMP messages. Specify read-view, write-only, and/or notification access to the SNMP agent. To configure an SNMPv3 user's authentication and privacy settings, use `snmp-server user`. The `no` version of this command deletes an SNMP group.

Example

```
sonic(config)# snmp-server group group1 v2c
```

```
sonic(config)# snmp-server group group1 v2c notify no_view
```

```
sonic(config)# snmp-server group group-floor2 v3 priv
```

```
sonic(config)# snmp-server group group-floor2 v3 priv read r_view write
w_view notify n_view
```

Releases 3.0 or later

snmp-server host

Configures a host to receive SNMP notifications.

Command

```
snmp-server host host-ipaddr {{community {community_name {{[traps] v2c} |
{[informs] {[timeout] seconds} {[retries] attempts}}}}} | {user {username
{{[traps] {noauth | auth | priv}} | {[informs] {noauth | auth | priv}
{[timeout] seconds} {[retries] attempts}}}}}}
```

Options

- `host host-ipaddr` — IPv4 or IPv6 of the SNMP host in A.B.C.D or A:B:C:D:E:F:G:H format
- `community community_name` — Community string name (up to 32 characters)
- `traps` — Send trap messages to the SNMP host
- `v2c` — Send inform messages to the SNMP host
- `informs` — Send inform messages to the SNMP host
- `timeout seconds` — Timeout value in seconds
- `retries retries` — Retry value in seconds
- `user username` — Username (up to 32 characters)
- `noauth` — Send SNMPv3 traps without user authentication and privacy encryption
- `auth` — Include a user authentication key for SNMPv3 messages sent to the host
- `priv` — Configure encryption for SNMPv3 messages sent to the host

Command mode CONFIGURATION

Usage Use this command to configure an SNMP agent to send SNMP notifications, traps, and inform SNMP managers configured as host receivers. You can configure multiple host receivers. An SNMP host does not acknowledge the trap messages and notifications received from an SNMP agent. Set the timeout and number of retries for the inform messages sent to an SNMP host. Timeout indicates the number of seconds before the informs time out when sending to a host. Retries indicate the number of times the informs are sent after timing out. The `no` version of this command disables the local agent from sending SNMP traps, informs, or notifications to a host receiver.

Examples

```
sonic(config)# snmp-server host 1.2.3.4 community comm1 traps v2c
```

```
sonic(config)# snmp-server host 1.2.3.5 user user1 informs noauth timeout
200 retries 10
```

```
sonic(config)# snmp-server host 2001::1 community comm2 informs timeout 150
retries 5
```

```
sonic(config)# snmp-server host 3001::1 user u1 traps priv
```

Releases 3.0 or later

snmp-server location

Configures the location of the SNMP server.

Command `snmp-server location location_name`

Options *location_name* — Location name in alphanumeric string (up to 55 characters)

Command mode CONFIGURATION

Usage Use this command to configure the SNMP server location information. The `no` version of this command removes the SNMP location.

Example

```
sonic(config)# snmp-server location "Lab1, Rack-10"
```

Releases 3.0 or later

snmp-server user

Configures user access to the SNMP agent on the switch using group membership.

Command

```
snmp-server user username {{[group] group-name} {{[encrypted] {auth {{md5
{auth-password {authpassword {[priv] {{des {priv-password privpassword}} |
{aes-128 {priv-password privpassword}}}}}}} | {sha {auth-password {authpassword
{[priv] {{des {priv-password privpassword}} | {aes-128 {priv- password
privpassword}}}}}}}}} | {[auth] {noauth | {md5 {auth-password {authpassword
{[priv] {{des {priv-password privpassword}} | {aes-128 {priv-password
privpassword}}}}}}} | {sha {auth-password {authpassword {[priv] {{des { priv-
password privpassword}} | {aes-128 {priv-password privpassword}}}}}}}}}}
```

Options

- `user` *username* — SNMP username (up to 32 characters)
- `group` *group-name* — (Optional) SNMP group-name (up to 32 characters)
- `auth-password` *authpassword* — (Optional) Authentication password (16-byte hex string)
- `priv-password` *privpassword* — (Optional) Privacy password (16-byte hex string)

Command mode CONFIGURATION

Usage Use this command to assign each user to a group and configure SNMPv3-specific authentication and encryption settings. Authentication passwords can be encrypted. If password encryption is desired, it must be specified prior to setting the authentication type. The `no` version of this command removes the configuration.

Example

```
sonic(config)# snmp-server user user1
```

```
sonic(config)# snmp-server user user1 group group-lab auth md5 auth-password
pwd priv aes-128 priv-password pwd
```

```
sonic(config)# snmp-server user user2 group group-floor2 encrypted auth sha
authpassword
abcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcd
abcdabcd priv des priv-password
abcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcd
abcdabcdabcd
```

Releases 3.0 or later

snmp-server view

Configures one or more SNMP views and set the OID tree to include or exclude from the view.

Command `snmp-server view` *view-name* {*oid-tree* {included | excluded}}

Options

- *view-name* — View name (up to 32 characters)
- *oid-tree* — OID tree name (up to 255 characters)
- `included` — Included in the SNMP views
- `excluded` — Excluded from the SNMP views

Command mode CONFIGURATION

Usage Use this command to configure a SNMP view. SNMP views are used by the groups for the GET/SET requests and to send traps. The `no` version of this command removes the configuration.

Example

```
sonic(config)# snmp-server view view2 1.2.3.4.5.6.7.8.9.2 excluded
```

Releases 3.0 or later

soft-reconfiguration

Enables soft-reconfiguration for a BGP neighbor.

Command `soft-reconfiguration inbound`

Options None

Command mode

- NEIGHBOR-ADDRESS-FAMILY
- PEER-GROUP-ADDRESS-FAMILY

Usage Use this command to store routes received (RIB-In) from a BGP neighbor. This command is not supported on a peer-group level. To enable soft-reconfiguration for peers in a peer-group, you must enable this command at a per-peer level. With soft-reconfiguration inbound, all updates that are received from this neighbor are stored unmodified, regardless of the inbound policy. When inbound soft-reconfiguration is performed later, the stored information generates a new set of inbound updates. These stored routes could be used to refresh the Loc-RIB in future as needed. If inbound policy changes, these stored routes will be used to generate LocRIB after applying the modified inbound policy. The `no` version of this command disables soft-reconfiguration inbound for a BGP neighbor.

Example

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 20.20.20.2
sonic(config-router-bgp-neighbor)# remote-as 300
```

```
sonic(config-router-bgp-neighbor)# address-family ipv4 unicast
sonic(config-router-bgp-neighbor-af)# soft-reconfiguration
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Int
sonic(config-router-bgp-pg)# address-family ipv4 unicast
sonic(config-router-bgp-pg-af)# soft-reconfiguration
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 20.20.20.2
sonic(config-router-bgp-neighbor)# remote-as 300
sonic(config-router-bgp-neighbor)# address-family l2vpn evpn
sonic(config-router-bgp-neighbor-af)# soft-reconfiguration
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Int
sonic(config-router-bgp-pg)# address-family l2vpn evpn
sonic(config-router-bgp-pg-af)# soft-reconfiguration
```

Releases 3.0 or later

solo

Command `solo`

Options None

Command mode
- NEIGHBOR
- PEER-GROUP

Usage This command is used to indicate that routes advertised by the peer should not be reflected back to the peer. This command is only meaningful when there is a single peer defined in the peer-group. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 30.30.30.3
sonic(config-router-bgp-neighbor)# solo
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Ext
sonic(config-router-bgp-pg)# solo
```

Releases 3.0 or later

source-ip

Sets the MCLAG source IPv4 address.

Command `source-ip` *`source-ip-address`*

Options *`source-ip-address`* — Source IP address in A.B.C.D format

Command mode INTERFACE

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# interface Vxlan vtep1
sonic(conf-if-Vxlan-vtep1)# source-ip 1.1.1.1
```

Releases 3.0 or later

source-vtep

Configures the VxLAN interface to source the IP address.

Command `source-vtep vxlan_name`

Options *vxlan_name* — Source VTEP interface name (up to 63 characters)

Command mode INTERFACE-EVPN

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# evpn evpn1
sonic(config-if-evpn)# source-vtep vtep1
```

Releases 3.0 or later

spanning-tree

Configures interface-specific spanning-tree parameters.

Command `spanning-tree {{bpduguard [port-shutdown]} | {cost value} | enable | {link-type {point-to-point | shared}} | portfast | {port-priority value} | {port {type edge}} | uplinkfast}`

Options

- `bpduguard port-shutdown` — Enables the BPDU filter on an interface
- `cost value` — Sets the spanning-tree cost on an interface
- `enable` — Enables spanning-tree on an interface
- `link-type` — Sets the link-type on an interface
- `portfast` — Enables spanning-tree portfast on an interface
- `port-priority value` — Sets the spanning-tree port-priority on an interface
- `uplinkfast` — Enables spanning-tree uplink fast on an interface
- `link-type` — Sets the spanning-tree link-type on an interface
- `port-type` — Sets the spanning-tree edge-port type on an interface

Command mode INTERFACE

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# interface Ethernet 0
sonic(conf-if-Ethernet0)# spanning-tree bpduguard port-shutdown
sonic(conf-if-Ethernet0)# spanning-tree cost 1000
sonic(conf-if-Ethernet0)# spanning-tree enable
sonic(conf-if-Ethernet0)# spanning-tree portfast
sonic(conf-if-Ethernet0)# spanning-tree port-priority 16
sonic(conf-if-Ethernet0)# spanning-tree uplinkfast
sonic(conf-if-Ethernet0)# spanning-tree link-type point-to-point
sonic(conf-if-Ethernet0)# spanning-tree port type edge
```

Releases 3.0 or later

spanning-tree bpdufilter

Enables or disables BPDU filtering on an interface.

Command `spanning-tree bpdufilter {enable | disable}`

Options

- `enable` — Enables the BPDU filter on an interface
- `disable` — Disables the BPDU filter on an interface

Command mode INTERFACE

Usage Use this command to enable BPDU filtering on an interface.

Examples

```
sonic(config)# interface Ethernet 0
sonic(conf-if-Ethernet0)# spanning-tree bpdufilter enable
```

```
sonic(config)# interface PortChannel 1
sonic(conf-if-po1)# spanning-tree bpdufilter enable
```

Releases 3.0 or later

spanning-tree bpduguard

Enables or disables the BPDU guard on an interface.

Command `spanning-tree bpduguard [port-shutdown]`

Options `port-shutdown` — (Optional) Enables the BPDU filter on an interface

Command mode INTERFACE

Usage Use this command to prevent a port from receiving BPDUs. If the port receives a BPDU, it is placed in the Error-Disabled state.

Example

```
sonic(config)# interface PortChannel 1
sonic(conf-if-po1)# spanning-tree bpduguard port-shutdown
```

Releases 3.0 or later

spanning-tree cost

Configures the spanning-tree cost on an interface.

Command `spanning-tree cost` *value*

Options `cost` *value* — Port-level cost value (1 to 200000000)

Command mode INTERFACE-PORT-CHANNEL

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# interface PortChannel 1
sonic(conf-if-po1)# spanning-tree cost 1000
```

Releases 3.0 or later

spanning-tree edge-port bpdufilter

Enables the spanning-tree edge-port BPDU filter default.

Command `spanning-tree edge-port bpdufilter default`

Options None

Command mode CONFIGURATION

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# spanning-tree edge-port bpdufilter default
```

Releases 3.0 or later

spanning-tree enable

Enables spanning-tree on an interface.

Command `spanning-tree enable`

Options None

Command mode INTERFACE-PORT-CHANNEL

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# interface PortChannel 1
sonic(conf-if-po1)# spanning-tree enable
```

Releases 3.0 or later

spanning-tree forward-time

Configures the spanning-tree forward delay time in seconds.

Command `spanning-tree forward-time` *`seconds`*

Options `forward time` *`seconds`* — Sets the forward delay time in seconds (4 to 30; default 15)

Command mode CONFIGURATION

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# spanning-tree forward-time 20
```

Releases 3.0 or later

spanning-tree guard root timeout

Configures the spanning-tree root guard timeout on an interface.

Command `spanning-tree guard root {timeout` *`value`*`}`

Options `timeout` *`value`* — Timeout value in seconds (5 to 600; default 30)

Command modes

- CONFIGURATION
- INTERFACE

Usage This command allows configuring a root guard timeout value. Once superior BPDUs stop coming on the port, the device waits for a period until root guard timeout before moving the port to forwarding state. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# spanning-tree guard root timeout 10
```

```
sonic(config)# interface Ethernet 0
sonic(conf-if-Ethernet0)# spanning-tree guard root
```

```
sonic(config)# interface PortChannel 1
sonic(conf-if-po1)# spanning-tree guard root
```

Releases 3.0 or later

spanning-tree hello-time

Configures the spanning-tree hello time value.

Command `spanning-tree hello-time seconds`

Options *seconds* — Time value in seconds (1 to 10; default 2)

Command mode CONFIGURATION

Usage This command allow configuring the hello interval in seconds for transmission of BPDUs. The `no` version of this command removes the configuration.

Example

```
sonic(config)# spanning-tree hello-time 3
```

Releases 3.0 or later

spanning-tree link-type

Sets spanning-tree link-type on a PortChannel interface.

Command `spanning-tree link-type {auto | point-to-point | shared}`

Options

- `auto` — Sets the link-type based on the duplex settings of the interface (default)
- `point-to-point` — Specifies that the interface is a point-to-point or full-duplex link
- `shared` — Specifies that the interface is a half-duplex link

Command mode INTERFACE

Usage This command allows setting the link-type of spanning-tree on an interface. The `no` version of this command removes the configuration.

Example

```
sonic(config)# interface PortChannel 1
sonic(conf-if-po1)# spanning-tree link-type point-to-point
```

Releases 3.0 or later

spanning-tree max-age

Configures the spanning-tree max-age timeout value.

Command `spanning-tree max-age value`

Options *value* — Sets the maximum time to listen for the root bridge in seconds (6 to 40; default 20)

Command mode CONFIGURATION

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# spanning-tree max-age 22
```

Releases 3.0 or later

spanning-tree mode

Configures global spanning-tree mode for the device.

Command `spanning-tree mode {[pvst] | [rapid-pvst]}`

Options

- `pvst` — (Optional) Sets spanning-tree mode to PVST
- `rapid-pvst` — (Optional) Sets spanning-tree mode to RPVST

Command mode	CONFIGURATION
Usage	This command allows configuring the spanning-tree mode for the device. The `no` version of this command removes the configuration.
Example	`sonic(config)# spanning-tree mode pvst`
Releases	3.0 or later

spanning-tree port type edge

Sets the port type as the EdgePort.

Command	`spanning-tree port type edge`
Options	None
Command mode	INTERFACE
Usage	Use this command to set the port type as the EdgePort. When you configure an EdgePort on a device running spanning-tree protocol, the port immediately transitions to the Forwarding state. Only configured ports connected to end hosts act as EdgePorts. The `no` version of this command resets the port-type configuration.
Example	`sonic(config)# interface PortChannel 1` `sonic(conf-if-po1)# spanning-tree port type edge`
Releases	3.0 or later

spanning-tree portfast

Enables spanning-tree portfast on an interface.

Command	`spanning-tree portfast`
Options	None
Command mode	INTERFACE
Usage	Use this command to allow enabling portfast on an interface. Portfast allows edge ports to move to a Forwarding state quickly when the connected device is not participating in spanning-tree. The `no` version of this command disables portfast.
Example	`sonic(config)# interface PortChannel 1` `sonic(conf-if-po1)# spanning-tree portfast`
Releases	3.0 or later

spanning-tree port-priority

Configures spanning-tree port priority on an interface.

Command	`spanning-tree port-priority` *`value`*
Options	*`value`* — Port priority value (0 to 240; default 128)
Command mode	INTERFACE
Usage	The `no` version of this command removes the configuration.
Example	`sonic(config)# interface PortChannel 1` `sonic(conf-if-po1)# spanning-tree port-priority 16`
Releases	3.0 or later

spanning-tree priority

Configures the global-level spanning-tree bridge priority value.

Command `spanning-tree priority value`

Options *value* — Bridge priority value in increments of 4096 (0 to 61440)

Command mode CONFIGURATION

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# spanning-tree priority 4096
```

Releases 3.0 or later

spanning-tree uplinkfast

Configures spanning-tree uplink fast on an interface.

Command `spanning-tree uplinkfast`

Options None

Command mode INTERFACE

Usage Uplink fast enhances STP performance for switches with redundant uplinks. Using the default value for the standard STP forward delay, convergence following a transition from an active link to a redundant link can take 30 seconds (15 seconds for listening and an additional 15 seconds for learning). When uplink fast is configured on the redundant uplinks, it reduces the convergence time to just one second by moving to forwarding state (bypassing listening and learning modes) in just once second when the active link goes down. The `no` version of this command removes the configuration.

Example

```
sonic(config)# interface PortChannel 1
sonic(conf-if-po1)# spanning-tree uplinkfast
```

Releases 3.0 or later

spanning-tree vlan

Configures interface and spanning-tree parameters on a per VLAN-basis.

Command `spanning-tree vlan vlan-range {{[forward-time] seconds} | {[hello-time] seconds} | {[max-age] seconds} | {[cost] value} | {[priority] value}}`

Options

- *vlan-range* — VLAN ID
- `forward-time` *seconds* — Forward-time interval in seconds
- `hello-time` *seconds* — Hello-time interval in seconds
- `max-age` *seconds* — Max-age time interval in seconds
- `cost` *value* — Cost value
- `priority` *value* — Priority value

Command mode

- SPANNING-TREE
- INTERFACE
- PORT-CHANNEL

Usage This command is similar to the global-level commands but allows configuring spanning-tree parameters on per VLAN basis. The `no` version of this command removes the configuration.

Example

```
sonic(config)# spanning-tree vlan 100
sonic(config)# spanning-tree vlan 100 forward-time 11
sonic(config)# spanning-tree vlan 100 hello-time 3
```

```
sonic(config)# spanning-tree vlan 100 max-age 22
sonic(config)# spanning-tree vlan 100 priority 4096
```

```
sonic(config)# interface Ethernet 0
sonic(conf-if-Ethernet0)# spanning-tree vlan 100 cost 1000
sonic(conf-if-Ethernet0)# spanning-tree vlan 100 port-priority 16
```

```
sonic(config)# interface PortChannel 1
sonic(conf-if-po1)# spanning-tree vlan 100 cost 1000
sonic(conf-if-po1)# spanning-tree vlan 100 port-priority 16
```

Releases 3.0 or later

speed

Configures the transmission speed of the Management interface.

Command `speed {10 | 100 | 1000 | auto}`

Options Set the Management port speed:

- `10` — 10M
- `100` — 100M
- `1000` — 1000M
- `auto` — Auto-negotiate speed with a connected device (default)

Command mode INTERFACE

Usage Use the command to configure the Management interface transmission speed. This command is supported only on the Management interface. This command is not supported on Ethernet interfaces. The `no` version of this command resets the resets the port to the default value (`auto`).

Example

```
sonic(conf-if-ma-1/1/1)# speed auto
```

Releases 3.0 or later

static

Adds a static NAT entry based on all ports, or global or local IP addresses.

Command `static {all | {basic` *`global-ip local-ip`* `[`*`natType`*`] {[twice-nat-id]` *`twice-nat-id-value`*`}} | {`*`natPortType global-ip global-port local-ip local-port`* `[`*`natType`*`] {[twice-nat-id]` *`twice-nat-id-value`*`}}}`

Options

- *`global-ip`* — Global IP in A.B.C.D format
- *`local-ip`* — Local IP in A.B.C.D format
- *`natType`* — NAT authentication type; snat or dnat
- *`twice-nat-id-value`* — NAT ID
- *`natPortType`* — Port type; tcp or udp
- *`global-port`* — Global port ID
- *`local-port`* — Local port ID

Command mode NAT

Usage The `no` version of this command removes the configuration.

Example

```
sonic(conf-nat)# static all
```

Releases 3.0 or later

strict-capability-match

Configures a BGP neighbor or peer-group to strictly compare remote capabilities and local capabilities

Command `strict-capability-match`

Options None

Command modes
- BGP-NEIGHBOR
- PEER-GROUP

Usage Use this command for a BGP neighbor to enforce exact matching of sent and received capabilities. If remote and local capabilities are different, this command sends an unsupported capability error then resets the connection. The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 30.30.30.3
sonic(config-router-bgp-neighbor)# strict-capability-match
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Ext
sonic(config-router-bgp-pg)# strict-capability-match
```

Releases 3.0 or later

switchport access Vlan

Assigns access VLAN membership to a port in L2 Access or Trunk mode.

Command `switchport access Vlan` *`vlan-id`*

Options *`vlan-id`* — VLAN ID number (1 to 4093)

Command mode INTERFACE

Usage Use this command to enable L2 switching for untagged traffic and assign a port interface to default VLAN1. Use this command to change the assignment of the access VLAN that carries untagged traffic. You must create the VLAN before you can assign an access interface. The `no` version of this command resets access VLAN membership to VLAN1.

Example

```
sonic(config)# interface ethernet 1/1/2
sonic(conf-if-eth1/1/2)# switchport access Vlan 100
```

Releases 3.0 or later

switchport trunk allowed Vlan

Configures the tagged VLAN traffic that an L2 trunk interface can carry.

Command `switchport trunk allowed {Vlan` *`vlan_id_list`*`}`

Options *`vlan_id_list`* — VLAN numbers of the tagged traffic that the L2 trunk port can carry; comma-separated and hyphenated VLAN number ranges are supported

Command mode INTERFACE

Usage Use this command to configure the tagged VLAN traffic that an L2 trunk interface can carry. An L2 trunk port has no tagged VLAN membership and does not transmit tagged traffic. The `no` version of this command removes the configuration.

Example

```
sonic(config)# switchport trun allowed vlan 55
```

Releases 3.0 or later

13

T, U, V, W, and Z commands

Topics:

table-map

Applies a BGP table to RIB manager route download filter.

Command	`table-map` *`route-map`*
Options	*`route-map`* — Route-map name
Command mode	ADDRESS-FAMILY
Usage	This command enables route-map on route updates from BGP to RIB manager. All applicable match operations are allowed, including match on prefix, next-hop, communities, and so on. Set operations for this attach-point are limited to metric and next-hop only. Any operation does not affect BGPs internal RIB. The `no` version of this command removes the configuration.
Example	

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# address-family ipv4 unicast
sonic(config-router-bgp-af)# table-map rmap_block_private
```

Releases 3.0 or later

tacacs-server auth-type

Configures the global TACACS+ server authentication type.

Command
```
tacacs-server auth-type [pap | chap | mschap]
```

Options
- `pap` — Enables `pap` for the authentication type (default)
- `chap` — Enables `chap` for the authentication type
- `mspap` — Enables `mschap` for the authentication type

Command mode CONFIGURATION

Usage Use this command to configure a default TACACS+ authentication type that is used for remote user access. The authentication type is used to encrypt or decrypt data that is sent and received between the switch and the TACACS+ server. If you have not configured a server-specific authtype, this global value is used for that server. The `no` version of this command resets the configuration to the default.

Example
```
sonic(config)# tacacs-server auth-type chap
```

Releases 3.0 or later

tacacs-server host

Configures a TACACS+ server and the key used to authenticate the switch on the server.

Command
```
tacacs-server host ip_address [port] port_number [timeout] seconds [key]
authentication_key [type] authentication_type [priority] port_priority
```

Options
- `host` *`ip_address`* — IPv4 or IPv6 host address in A.B.C.D or A::B format
- `port` *`port_number`* — Port number
- `timeout` *`seconds`* — Port timeout in seconds
- `key` *`authentication_key`* — Authentication key (up to 65 characters)
- `type` *`authentication_type`* — Authentication type; select pap, chap, or mschap
- `priority` *`port_priority`* — Port priority (1 to 64)

Command mode CONFIGURATION

Usage Use this command to configure the TACACS+ server. The authentication key must match the key configured on the TACACS+ server, and you cannot enter spaces in the key. You can configure the global timeout allowed for authentication requests on TACACS+ servers using `radius-server timeout`. The `no` version of this command removes a TACACS+server configuration.

Example
```
sonic(config)# tacacs-server host 10.0.0.100 key secret1 type chap
```

Releases 3.0 or later

tacacs-server key

Configures the global shared secret authentication key for the TACACS+ server.

Command
```
tacacs-server key secret-key
```

Options *`secret-key`* — Authentication key (up to 65 characters)

Command mode CONFIGURATION

Usage Use this command to modify the global value for the TACACS+ server authentication key. If you have not configured a server-specific authentication key, this global value is used for that TACACS+ server. The

authentication key can include all printable ASCII characters with a few exceptions (#, SPACE, and COMMA), and up to 65 characters. The `no` version of this command removes the configuration.

Example

```
sonic(config)# tacacs-server key secret1
```

Releases 3.0 or later

tacacs-server source-ip

Configures the global IPv4 or IPv6 TACACS+ server address.

Command `tacacs-server source-ip` *ip-address*

Options *ip-address* — Source IP address for the TACACS+ server in A.B.C.D or A::B format

Command mode CONFIGURATION

Usage Use this command for TACACS+ server authentication. The `no` version of this command removes the configuration.

Example

```
sonic(config)# tacacs-server source-ip 10.1.1.1
```

Releases 3.0 or later

tacacs-server timeout

Configures the global timeout used for authentication attempts on TACACS+ servers.

Command `tacacs-server timeout` *seconds*

Options *seconds* — Timeout period to wait for an authentication response from a TACACS+ server (1 to 60 seconds; default 5)

Command mode CONFIGURATION

Usage Use this command to modify the global value for the TACACS+ server timeout. If you have not configured a server-specific timeout, this global value is used for that TACACS+ server. The `no` version of this command resets the TACACS+ server timeout to the default.

Example

```
sonic(config)# tacacs-server timeout 60
```

Releases 3.0 or later

tam

Enters TAM device configuration mode.

Command `tam`

Options None

Command mode CONFIGURATION

Usage None

Example

```
sonic(config)# tam
```

Releases 3.0 or later

tcp-timeout

Configures the TCP NAT entry aging timeout in seconds.

Command `tcp-timeout tcp-timeout-value`

Options *tcp-timeout-value* — NAT entry aging timeout in seconds (integer)

Command mode NAT

Usage The `no` version of this command removes the configuration.

Example

```
sonic(conf-nat)# tcp-timeout 5
```

Releases 3.0 or later

terminal length

Configures the number of lines to display on the terminal.

Command `terminal length length-value`

Options *length-value* — Number of lines to display (0 to 512; default 24)

Command mode EXEC

Usage Enter zero (0) for the terminal to display without pausing.

Example

```
sonic# terminal length 0
```

Releases 3.0 or later

timeout

Configures the aging timeout in seconds for link state tracking and NAT.

Command `timeout timeout-value`

Options *timeout-value* — Aging timeout in seconds; 1 to 999

Command mode CONFIGURATION

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# timeout 100
```

Releases 3.0 or later

timers

Adjusts BGP keepalive and holdtime timers.

Command `timers {keepalive-intvl hold-time} {{connect connect-time}`

Options

- *keepalive-intvl* — Keepalive time interval, in seconds, between keepalive messages sent to the neighbor routers (1 to 65535; default 60)
- *hold-time* — Hold time interval, in seconds, betweem the last keepalive message and declaring a router dead (3 to 65535; default 180)
- *connect-time* — (Optional) Connect time interval in seconds

Command modes

- ROUTER-BGP

- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to configure keepalive, hold timer, and connect neighbor and peer-group interval values for an instance of BGP. The configured timer value becomes effective after a BGP hard restart. The timer values negotiate from peers. The `no` version of this command resets the value to the default.

Examples

```
sonic(config)# router bgp 65300
sonic(config-router-bgp)# timers 10 30
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 30.30.30.3
sonic(config-router-bgp-neighbor)# timers 3 9
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Ext
sonic(config-router-bgp-pg)# timers 3 9
```

Releases 3.0 or later

transmit-interval

Configures the packet transmit interval for a BFD peer.

Command `transmit-interval` *`transmit_interval`*

Options *`transmit_interval`* — Transmit interval value; default 300 ms (integer)

Command mode BFD-PEER

Usage Use this command to configure the packet transmit interval for IPv4, IPv6, multi-hop, and VRF interfaces.

Example

```
sonic(config)# bfd
sonic(conf-bfd)# peer 192.168.0.5 interface Ethernet0
sonic(conf-bfd-peer)# transmit-interval 200
```

Releases 3.0 or later

ttl-security hops

Configures neighbors that are a specified number of hops away to be allowed to become neighbors using GTSM.

Command `ttl-security hops` *`nhops`*

Options *`nhops`* — Specified number of hops to be allowed to become a neighbor (integer)

Command modes
- BGP-NEIGHBOR
- PEER-GROUP

Usage This command enforces generalized TTL security mechanism (GTSM), as specified in RFC 5082, and is mutually exclusive with ebgp-multihop. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 30.30.30.3
sonic(config-router-bgp-neighbor)# ttl-security hops 6
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Ext
sonic(config-router-bgp-pg)# ttl-security hops 8
```

Releases 3.0 or later

udld aggressive

Configures UDLD mode to aggressive on an interface level or globally.

Command `udld agressive`

Options None

Command mode
- CONFIGURATION
- INTERFACE

Usage Use this command to change UDLD mode to aggressive on an interface level or globally. Default UDLD mode is normal. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# udld enable
sonic(config)# udld aggressive
```

```
sonic(config)# interface Ethernet 0
sonic(conf-if-Ethernet0)# udld enable
sonic(conf-if-Ethernet0)# udld aggressive
```

Releases 3.0 or later

udld enable

Enables UDLD at an interface or global level.

Command `udld enable`

Options None

Command modes
- CONFIGURATION
- INTERFACE

Usage Use this command to enable UDLD at an interface level or globally. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# udld enable
```

```
sonic(config)# interface Ethernet 0
sonic(conf-if-Ethernet0)# udld enable
```

Releases 3.0 or later

udld message-time

Configures the UDLD message time interval at which periodic hellos are exchanged.

Command `udld message-time` *`msg-time`*

Options *`msg-time`* — Time interval period; default 1 second (integer)

Command mode CONFIGURATION

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# udld enable
sonic(config)# udld message-time 3
```

Releases 3.0 or later

udld multiplier

Configures the UDLD multiplier value.

Command `udld multiplier multiplier`

Options *multiplier* — UDLD multiplier value; default 3 (integer)

Command mode CONFIGURATION

Usage Use this command to set UDLD multiplier value. This multiplier value is used to determine the timeout interval (message-time times the multiplier value) after which UDLD declares the link as unidirectonal. Default multiplier value is 3. The `no` version of this command removes the configuration.

Example

```
sonic(config)# udld enable
sonic(config)# udld multiplier 8
```

Releases 3.0 or later

udp-timeout

Configures UDP NAT entry aging timeout in seconds.

Command `udp-timeout udp-timeout-value`

Options *udp-timeout-value* — UDP timeout value (integer)

Command mode NAT

Usage The `no` version of this command removes the configuration.

Example

```
sonic(conf-nat)# udp-timeout 20
```

Releases 3.0 or later

unsuppress-map

Configures a route policy using a route-map to unsuppress suppressed routes.

Command `unsuppress-map map`

Options *map* — Route-map (string)

Command mode

- ADDRESS-FAMILY
- PEER-GROUP

Usage The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 20.20.20.2
sonic(config-router-bgp-neighbor)# remote-as 300
sonic(config-router-bgp-neighbor)# address-family ipv4 unicast
sonic(config-router-bgp-neighbor-af)# unsuppress-map rm_unsup_ext_rt
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Int
sonic(config-router-bgp-pg)# address-family ipv4 unicast
sonic(config-router-bgp-pg-af)# unsuppress-map rm_unsup_ext_rt
```

Releases 3.0 or later

update-delay

Sets the update delay which controls how long to wait before running best-path selection after graceful restart.

Command `update-delay time [maxmedval]`

Options
- *time* — Time value (integer)
- *maxmedval* — Maximum delay value; default 0 (integer)

Command mode ROUTER-BGP

Usage This command is used to enable read-only mode on BGP process restart or when BGP process is cleared using `clear ip bgp *`. Read-only mode begins as soon as the first peer reaches the established status, and a timer for max-delay seconds is started. During this mode, BGP does not run any best-path or generate any updates to its peers. This mode continues until all the configured peers (except the shutdown peers) have sent explicit EOR (End-Of-RIB) or an implicit-EOR. The first keep-alive after BGP has reached established is considered an implicit-EOR. If the establish-wait optional value is given, BGP waits for peers to reach established from the beginning of the update-delay until the establish-wait period is over (the minimum set of established peers for which EOR is expected would be peers established during the establish-wait window), not necessarily all the configured neighbors and max-delay period is over. On hitting any of the above two conditions, BGP resumes the decision process and generates updates to its peers. Default max-delay is 0 (feature is off by default). The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(config-router-bgp)# update-delay 120
```

Releases 3.0 or later

update-source ip interface

Specifies the IPv4 or IPv6 source address to use for the BGP session or peer-group.

Command `update-source {ip | {interface {Ethernet | PortChannel | Vlan | Loopback}}}`

Options
- *ip* — IPv4 or IPv6 address in A.B.C.D/A::B format
- `interface` — Interface type

Command modes
- BGP-NEIGHBOR
- BGP-PEER-GROUP

Usage Use this command to configure the source interface for a BGP neighbor session or peer-group. Source address may be specified as either an IPv4/IPv6 address directly or as an interface name. The interface name could be router port, PortChannel, Loopback, or Vlan interface with IPv4/IPv6 address configured. The `no` version of this command removes the configuration.

Examples

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 30.30.30.3
sonic(config-router-bgp-neighbor)# update-source 12.56.36.74
```

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# peer-group PG_Ext
sonic(config-router-bgp-pg)# update-source Ethernet16
```

Releases 3.0 or later

username

Adds a new user.

Command `username user-name password {passwd {role rl}}`

Options

- *user-name* — Name of this user (string)
- *passwd* — Password for this user (string)
- *rl* — Role for this user (string)

Command mode CONFIGURATION

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# username asmith password admin5 role admin
```

Releases 3.0 or later

vni

Enables configuration per-VNI EVPN parameter.

Command `vni vninum`

Options *vninum* — VNI number (integer)

Command mode ADDRESS-FAMILY

Usage The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# address-family l2vpn evpn
sonic(config-router-bgp-af)# vni 100
sonic(config-router-bgp-af-vni)#
```

Releases 3.0 or later

weight

Assigns a default weight for routes from neighbor interfaces.

Command `weight val`

Options *val* — Weight value for routes (1 to 4294967295; default 0)

Command mode ADDRESS-FAMILY

Usage Use this command to assign a default weight to BGP routes received from this neighbor. The path with the highest weight value is preferred in best-path selection. The `no` version of this command resets the value to the default.

Example

```
sonic(config)# router bgp 100
sonic(config-router-bgp)# neighbor 20.20.20.2
sonic(config-router-bgp-neighbor)# remote-as 300
sonic(config-router-bgp-neighbor)# address-family ipv4 unicast
sonic(config-router-bgp-neighbor-af)# weight 4096
```

Releases 3.0 or later

write

Saves the current running configuration to the startup configuration file.

Command `write {memory}`

Options `memory` — Write the current running configuration to the startup configuration.

Command mode EXEC

Usage None

Example

```
sonic# write memory
```

Releases 3.0 or later

write erase

Erases the existing switch configuration files except the Management interface configuration, or cancel the configuration erase operation.

Command `write erase`

Options None

Command mode CONFIGURATION

Usage Use this command to delete the startup configuration JSON file and all application configuration files in the */etc/sonic* directory. The Management interface configuration in the startup configuration file is retained so that you can access the switch using the same management address after the switch reboot. For this command to take effect, you must reboot the switch after issuing this command. If you do not want to not proceed with the configuration removal operation, the `write erase cancel` command can be used to undo the previously issued `write erase` command. The `no` version of this command cancels the configuration erase operation.

Examples

```
sonic(config)# write erase
Existing switch configuration files except management interface
configuration will be
removed, continue? [y/N]:
```

```
sonic(config)# no write erase
Switch configuration erase operation will be cancelled, continue? [y/N]:
```

Releases 3.0 or later

write erase boot

Erases the configuration files including the Management interface configuration.

Command `write erase boot`

Options None

Command mode CONFIGURATION

Usage Use this command to delete the startup configuration JSON file, and all application configuration files in the */etc/sonic* directory. The Management interface configuration in the startup configuration JSON file is also removed. The SONiC switch boots with a factory default configuration file.

Example

```
sonic(config)# write erase boot
Existing switch configuration files will be removed, continue? [y/N]:
```

Releases 3.0 or later

write erase install

Restores all SONiC switch content to default values, and removes all changes made by the user.

Command `write erase install`

Options None

Command mode CONFIGURATION

Usage All user installed packages and file changes are removed. It also deletes the startup configuration JSON file and the files in the */etc/sonic* directory. The SONiC switch is reverted to the same state as a newly installed image.

After the SONiC switch is rebooted, if the ZTP is enabled, the switch starts to discover and download the switch configuration.

Example

```
sonic(config)# write erase install
All SONiC switch content will be restored to default values, continue? [y/N]:
```

Releases 3.0 or later

write-quanta

Configures the maximum number of BGP packets to write to, peer socket, in one I/O cycle.

Command `write-quanta wrval`

Options *wrval* — Write value (integer)

Command mode ROUTER-BGP

Usage The BGP message transmission I/O is vectored which means that multiple packets are written to the peer socket at the same time each I/O cycle to minimize system call overhead. This value controls how many are written at a time. Under certain load conditions, reducing this value could make peer traffic less bursty. It is recommended to leave this settings as the default (64). The `no` version of this command removes the configuration.

Example

```
sonic(config)# router bgp 65300
sonic(conf-router-bgp)# write-quanta 50
```

Releases 3.0 or later

ztp enable

Administratively enables or disables zero-touch provisioning (ZTP).

Command `ztp enable`

Options None

Command mode CONFIGURATION

Usage The `no` version of this command disables ZTP.

Examples

```
sonic(config)# ztp enable
```

```
sonic(config)# no ztp enable
```

Releases 3.0 or later