
Networking Baremetal Documentation

Release 6.0.1.dev2

OpenStack Foundation

Sep 13, 2022

CONTENTS

| | |
|--------------------------------------|-----------|
| 1 networking-baremetal plugin | 1 |
| 2 Installation Guide | 3 |
| 2.1 Installation | 3 |
| 3 Configuration Guide | 7 |
| 3.1 Configuration Options | 7 |
| 4 Contributor Guide | 37 |
| 4.1 Contributing | 37 |
| 5 Release Notes | 67 |
| 6 Indices and tables | 69 |
| Python Module Index | 71 |
| Index | 73 |

NETWORKING-BAREMETAL PLUGIN

This projects goal is to provide deep integration between the Networking service and the Bare Metal service and advanced networking features like notifications of port status changes and routed networks support in clouds with Bare Metal service.

- Free software: Apache license
- Documentation: <http://docs.openstack.org/networking-baremetal/latest>
- Source: <http://opendev.org/openstack/networking-baremetal>
- Bugs: <https://storyboard.openstack.org/#!/project/955>
- Release notes: <https://docs.openstack.org/releasenotes/networking-baremetal/>

INSTALLATION GUIDE

2.1 Installation

This section describes how to install and configure the `networking-baremetal` plugin and `ironic-neutron-agent`.

The `ironic-neutron-agent` is a neutron agent that populates the host to physical network mapping for baremetal nodes in neutron. Neutron uses this to calculate the segment to host mapping information.

2.1.1 Install the networking-baremetal plugin and agent

At the command line:

```
$ pip install networking-baremetal
```

Or, if you have neutron installed in a virtualenv, install the `networking-baremetal` plugin to the same virtualenv:

```
$ . <path-to-neutron-venv>/bin/activate  
$ pip install networking-baremetal
```

Or, use the package from your distribution. For RHEL7/CentOS7:

```
$ yum install python2-networking-baremetal python2-ironic-neutron-agent
```

2.1.2 Enable baremetal mechanism driver in the Networking service

To enable mechanism drivers in the ML2 plug-in, edit the `/etc/neutron/plugins/ml2/ml2_conf.ini` configuration file. For example, this enables the `openvswitch` and `baremetal` mechanism drivers:

```
[ml2]  
mechanism_drivers = openvswitch,baremetal
```

2.1.3 Add devices (switches) to manage

The baremetal mechanism ML2 plug-in provides a device driver plug-in interface. If a device driver for the switch model exist the baremetal ML2 plug-in can be configured to manage switch configuration, adding tenant VLANs and setting switch port VLAN configuration etc.

To add a device to manage, edit the `/etc/neutron/plugins/ml2/ml2_conf.ini` configuration file. The example below enables devices: `device_a.example.net` and `device_b.example.net`. Both devices in the example is using the `netconf-openconfig` device driver. For each device a separate section in configuration defines the device and driver specific configuration.

```
[networking_baremetal]
enabled_devices = device_a.example.net,device_b.example.net

[device_a.example.net]
driver = netconf-openconfig
switch_info = device_a
switch_id = 00:53:00:0a:0a:0a
host = device_a.example.net
username = user
key_filename = /etc/neutron/ssh_keys/device_a_sshkey
hostkey_verify = false

[device_b.example.net]
driver = netconf-openconfig
switch_info = device_b
switch_id = 00:53:00:0b:0b:0b
host = device_a.example.net
username = user
key_filename = /etc/neutron/ssh_keys/device_a_sshkey
hostkey_verify = false
```

2.1.4 Configure ironic-neutron-agent

To configure the baremetal neutron agent, edit the neutron configuration `/etc/neutron/plugins/ml2/ironic_neutron_agent.ini` file. Add an `[ironic]` section. For example:

```
[ironic]
project_domain_name = Default
project_name = service
user_domain_name = Default
password = password
username = ironic
auth_url = http://identity-server.example.com/identity
auth_type = password
os_region = RegionOne
```


2.1.5 Start ironic-neutron-agent service

To start the agent either run it from the command line like in the example below or add it to the init system.

```
$ ironic-neutron-agent \  
  --config-dir /etc/neutron \  
  --config-file /etc/neutron/plugins/ml2/ironic_neutron_agent.ini \  
  --log-file /var/log/neutron/ironic_neutron_agent.log
```

You can create a systemd service file `/etc/systemd/system/ironic-neutron-agent.service` for `ironic-neutron-agent` for systemd based distributions. For example:

```
[Unit]  
Description=OpenStack Ironic Neutron Agent  
After=syslog.target network.target  
  
[Service]  
Type=simple  
User=neutron  
PermissionsStartOnly=true  
TimeoutStartSec=0  
Restart=on-failure  
ExecStart=/usr/bin/ironic-neutron-agent --config-dir /etc/neutron --config-  
↪file /etc/neutron/plugins/ml2/ironic_neutron_agent.ini --log-file /var/log/  
↪neutron/ironic-neutron-agent.log  
PrivateTmp=true  
KillMode=process  
  
[Install]  
WantedBy=multi-user.target
```

Note: systemd service file may be already available if you are installing from package released by linux distributions.

Enable and start the `ironic-neutron-agent` service:

```
$ sudo systemctl enable ironic-neutron-agent.service  
$ sudo systemctl start ironic-neutron-agent.service
```


CONFIGURATION GUIDE

3.1 Configuration Options

3.1.1 Configuration Reference

The following pages describe configuration options that can be used to adjust the `ironic-neutron-agent` service to your particular situation.

`ironic-neutron-agent` - Configuration Options

The following is an overview of all available configuration options in `networking-baremetal`. For a sample configuration file, refer to *Sample Configuration File*.

DEFAULT

`debug`

Type boolean

Default False

Mutable This option can be changed without restarting.

If set to true, the logging level will be set to DEBUG instead of the default INFO level.

`log_config_append`

Type string

Default <None>

Mutable This option can be changed without restarting.

The name of a logging configuration file. This file is appended to any existing logging configuration files. For details about logging configuration files, see the Python logging module documentation. Note that when logging configuration files are used then all logging configuration is set in the configuration file and other logging configuration options are ignored (for example, `log-date-format`).

Table 1: Deprecated Variations

| Group | Name |
|---------|-------------------------|
| DEFAULT | <code>log-config</code> |
| DEFAULT | <code>log_config</code> |

log_date_format**Type** string**Default** %Y-%m-%d %H:%M:%S

Defines the format string for `%(asctime)s` in log records. Default: the value above. This option is ignored if `log_config_append` is set.

log_file**Type** string**Default** <None>

(Optional) Name of log file to send logging output to. If no default is set, logging will go to `stderr` as defined by `use_stderr`. This option is ignored if `log_config_append` is set.

Table 2: Deprecated Variations

| Group | Name |
|---------|---------|
| DEFAULT | logfile |

log_dir**Type** string**Default** <None>

(Optional) The base directory used for relative `log_file` paths. This option is ignored if `log_config_append` is set.

Table 3: Deprecated Variations

| Group | Name |
|---------|--------|
| DEFAULT | logdir |

watch_log_file**Type** boolean**Default** False

Uses logging handler designed to watch file system. When log file is moved or removed this handler will open a new log file with specified path instantaneously. It makes sense only if `log_file` option is specified and Linux platform is used. This option is ignored if `log_config_append` is set.

use_syslog**Type** boolean**Default** False

Use syslog for logging. Existing syslog format is DEPRECATED and will be changed later to honor RFC5424. This option is ignored if `log_config_append` is set.

use_journal**Type** boolean**Default** False

Enable journald for logging. If running in a systemd environment you may wish to enable journal support. Doing so will use the journal native protocol which includes structured metadata in addition to log messages. This option is ignored if `log_config_append` is set.

syslog_log_facility

Type string

Default LOG_USER

Syslog facility to receive log lines. This option is ignored if `log_config_append` is set.

use_json

Type boolean

Default False

Use JSON formatting for logging. This option is ignored if `log_config_append` is set.

use_stderr

Type boolean

Default False

Log output to standard error. This option is ignored if `log_config_append` is set.

use_eventlog

Type boolean

Default False

Log output to Windows Event Log.

log_rotate_interval

Type integer

Default 1

The amount of time before the log files are rotated. This option is ignored unless `log_rotation_type` is set to `interval`.

log_rotate_interval_type

Type string

Default days

Valid Values Seconds, Minutes, Hours, Days, Weekday, Midnight

Rotation interval type. The time of the last file change (or the time when the service was started) is used when scheduling the next rotation.

max_logfile_count

Type integer

Default 30

Maximum number of rotated log files.

max_logfile_size_mb

Type integer

Default 200

Log file maximum size in MB. This option is ignored if `log_rotation_type` is not set to `size`.

log_rotation_type

Type string

Default none

Valid Values interval, size, none

Log rotation type.

Possible values

interval Rotate logs at predefined time intervals.

size Rotate logs once they reach a predefined size.

none Do not rotate log files.

logging_context_format_string

Type string

Default `%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s
[%s] %(request_id)s %(user_identity)s
%(instance)s%(message)s`

Format string to use for log messages with context. Used by `oslo_log.formatters.ContextFormatter`

logging_default_format_string

Type string

Default `%(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s
[-] %(instance)s%(message)s`

Format string to use for log messages when context is undefined. Used by `oslo_log.formatters.ContextFormatter`

logging_debug_format_suffix

Type string

Default `%(funcName)s %(pathname)s:%(lineno)d`

Additional data to append to log message when logging level for the message is `DEBUG`. Used by `oslo_log.formatters.ContextFormatter`

logging_exception_prefix

Type string

Default `%(asctime)s.%(msecs)03d %(process)d ERROR %(name)s
%(instance)s`

Prefix each line of exception output with this format. Used by oslo_log.formatters.ContextFormatter

logging_user_identity_format

Type string

Default %(user)s %(project)s %(domain)s %(system_scope)s
%(user_domain)s %(project_domain)s

Defines the format string for %(user_identity)s that is used in logging_context_format_string. Used by oslo_log.formatters.ContextFormatter

default_log_levels

Type list

Default ['amqp=WARN', 'amqpplib=WARN', 'boto=WARN', 'qpid=WARN', 'sqlalchemy=WARN', 'suds=INFO', 'oslo.messaging=INFO', 'oslo_messaging=INFO', 'iso8601=WARN', 'requests.packages.urllib3.connectionpool=WARN', 'urllib3.connectionpool=WARN', 'websocket=WARN', 'requests.packages.urllib3.util.retry=WARN', 'urllib3.util.retry=WARN', 'keystonemiddleware=WARN', 'routes.middleware=WARN', 'stevedore=WARN', 'taskflow=WARN', 'keystoneauth=WARN', 'oslo.cache=INFO', 'oslo_policy=INFO', 'dogpile.core.dogpile=INFO']

List of package logging levels in logger=LEVEL pairs. This option is ignored if log_config_append is set.

publish_errors

Type boolean

Default False

Enables or disables publication of error events.

instance_format

Type string

Default "[instance: %(uuid)s] "

The format for an instance that is passed with the log message.

instance_uuid_format

Type string

Default "[instance: %(uuid)s] "

The format for an instance UUID that is passed with the log message.

rate_limit_interval

Type integer

Default 0

Interval, number of seconds, of log rate limiting.

rate_limit_burst

Type integer

Default 0

Maximum number of logged messages per `rate_limit_interval`.

rate_limit_except_level

Type string

Default CRITICAL

Log level name used by rate limiting: CRITICAL, ERROR, INFO, WARNING, DEBUG or empty string. Logs with level greater or equal to `rate_limit_except_level` are not filtered. An empty string means that all levels are filtered.

fatal_deprecations

Type boolean

Default False

Enables or disables fatal status of deprecations.

agent

report_interval

Type floating point

Default 30

Seconds between nodes reporting state to server; should be less than `agent_down_time`, best if it is half or less than `agent_down_time`.

log_agent_heartbeats

Type boolean

Default False

Log agent heartbeats

ironic

auth_strategy

Type string

Default keystone

Valid Values keystone, noauth

Method to use for authentication: noauth or keystone.

Warning: This option is deprecated for removal. Its value may be silently ignored in the future.

Reason This option is no longer used, please use the `[ironic]/auth_type` option instead.

service_type

Type string

Default <None>

The default service_type for endpoint URL discovery.

service_name

Type string

Default <None>

The default service_name for endpoint URL discovery.

valid_interfaces

Type list

Default <None>

List of interfaces, in order of preference, for endpoint URL.

region_name

Type string

Default <None>

The default region_name for endpoint URL discovery.

Table 4: Deprecated Variations

| Group | Name |
|--------|-----------|
| ironic | os_region |

endpoint_override

Type string

Default <None>

Always use this endpoint URL for requests for this client. NOTE: The unversioned endpoint should be specified here; to request a particular API version, use the *version*, *min-version*, and/or *max-version* options.

Table 5: Deprecated Variations

| Group | Name |
|--------|------------|
| ironic | ironic_url |

version**Type** string**Default** <None>

Minimum Major API version within a given Major API version for endpoint URL discovery. Mutually exclusive with min_version and max_version

min_version**Type** string**Default** <None>

The minimum major version of a given API, intended to be used as the lower bound of a range with max_version. Mutually exclusive with version. If min_version is given with no max_version it is as if max version is latest.

max_version**Type** string**Default** <None>

The maximum major version of a given API, intended to be used as the upper bound of a range with min_version. Mutually exclusive with version.

connect_retries**Type** integer**Default** <None>

The maximum number of retries that should be attempted for connection errors.

connect_retry_delay**Type** floating point**Default** <None>

Delay (in seconds) between two retries for connection errors. If not set, exponential retry starting with 0.5 seconds up to a maximum of 60 seconds is used.

status_code_retries**Type** integer**Default** <None>

The maximum number of retries that should be attempted for retrieable HTTP status codes.

Table 6: Deprecated Variations

| Group | Name |
|--------|-------------|
| ironic | max_retries |

status_code_retry_delay**Type** floating point**Default** <None>

Delay (in seconds) between two retries for retrievable status codes. If not set, exponential retry starting with 0.5 seconds up to a maximum of 60 seconds is used.

Table 7: Deprecated Variations

| Group | Name |
|--------|----------------|
| ironic | retry_interval |

interface**Type** string**Default** <None>

The default interface for endpoint URL discovery.

Warning: This option is deprecated for removal. Its value may be silently ignored in the future.

Reason Using valid-interfaces is preferable because it is capable of accepting a list of possible interfaces.

cafile**Type** string**Default** <None>

PEM encoded Certificate Authority to use when verifying HTTPs connections.

certfile**Type** string**Default** <None>

PEM encoded client certificate cert file

keyfile**Type** string**Default** <None>

PEM encoded client certificate key file

insecure**Type** boolean**Default** False

Verify HTTPS connections.

timeout**Type** integer**Default** <None>

Timeout value for http requests

collect_timing

Type boolean

Default False

Collect per-API call timing information.

split_loggers

Type boolean

Default False

Log requests to multiple loggers.

auth_url

Type unknown type

Default <None>

Authentication URL

system_scope

Type unknown type

Default <None>

Scope for system operations

domain_id

Type unknown type

Default <None>

Domain ID to scope to

domain_name

Type unknown type

Default <None>

Domain name to scope to

project_id

Type unknown type

Default <None>

Project ID to scope to

project_name

Type unknown type

Default <None>

Project name to scope to

project_domain_id**Type** unknown type**Default** <None>

Domain ID containing project

project_domain_name**Type** unknown type**Default** <None>

Domain name containing project

trust_id**Type** unknown type**Default** <None>

ID of the trust to use as a trustee use

user_id**Type** unknown type**Default** <None>

User ID

username**Type** unknown type**Default** <None>

Username

Table 8: Deprecated Variations

| Group | Name |
|--------|-----------|
| ironic | user-name |
| ironic | user_name |

user_domain_id**Type** unknown type**Default** <None>

Users domain id

user_domain_name**Type** unknown type**Default** <None>

Users domain name

password

Type unknown type

Default <None>

Users password

Sample Configuration File

The following is a sample ironic-neutron-agent configuration for adaptation and use. For a detailed overview of all available configuration options, refer to *ironic-neutron-agent - Configuration Options*.

The sample configuration can also be viewed in `file` form.

Important: The sample configuration file is auto-generated from networking-baremetal when this documentation is built. You must ensure your version of networking-baremetal matches the version of this documentation.

```
[DEFAULT]

#
# From oslo.log
#

# If set to true, the logging level will be set to DEBUG instead of the
↪default
# INFO level. (boolean value)
# Note: This option can be changed without restarting.
#debug = false

# The name of a logging configuration file. This file is appended to any
# existing logging configuration files. For details about logging
↪configuration
# files, see the Python logging module documentation. Note that when logging
# configuration files are used then all logging configuration is set in the
# configuration file and other logging configuration options are ignored (for
# example, log-date-format). (string value)
# Note: This option can be changed without restarting.
# Deprecated group/name - [DEFAULT]/log_config
#log_config_append = <None>

# Defines the format string for %(asctime)s in log records. Default:
# %(default)s . This option is ignored if log_config_append is set. (string
# value)
#log_date_format = %Y-%m-%d %H:%M:%S

# (Optional) Name of log file to send logging output to. If no default is set,
# logging will go to stderr as defined by use_stderr. This option is ignored
↪if
```

(continues on next page)

(continued from previous page)

```
# log_config_append is set. (string value)
# Deprecated group/name - [DEFAULT]/logfile
#log_file = <None>

# (Optional) The base directory used for relative log_file paths. This option
# is ignored if log_config_append is set. (string value)
# Deprecated group/name - [DEFAULT]/logdir
#log_dir = <None>

# Uses logging handler designed to watch file system. When log file is moved,
↳or
# removed this handler will open a new log file with specified path
# instantaneously. It makes sense only if log_file option is specified and
# Linux platform is used. This option is ignored if log_config_append is set.
# (boolean value)
#watch_log_file = false

# Use syslog for logging. Existing syslog format is DEPRECATED and will be
# changed later to honor RFC5424. This option is ignored if log_config_append
# is set. (boolean value)
#use_syslog = false

# Enable journald for logging. If running in a systemd environment you may,
↳wish
# to enable journal support. Doing so will use the journal native protocol
# which includes structured metadata in addition to log messages. This option,
↳is
# ignored if log_config_append is set. (boolean value)
#use_journal = false

# Syslog facility to receive log lines. This option is ignored if
# log_config_append is set. (string value)
#syslog_log_facility = LOG_USER

# Use JSON formatting for logging. This option is ignored if log_config_append
# is set. (boolean value)
#use_json = false

# Log output to standard error. This option is ignored if log_config_append is
# set. (boolean value)
#use_stderr = false

# Log output to Windows Event Log. (boolean value)
#use_eventlog = false

# The amount of time before the log files are rotated. This option is ignored
# unless log_rotation_type is set to "interval". (integer value)
#log_rotate_interval = 1
```

(continues on next page)

(continued from previous page)

```

# Rotation interval type. The time of the last file change (or the time when
# the service was started) is used when scheduling the next rotation. (string
# value)
# Possible values:
# Seconds - <No description provided>
# Minutes - <No description provided>
# Hours - <No description provided>
# Days - <No description provided>
# Weekday - <No description provided>
# Midnight - <No description provided>
#log_rotate_interval_type = days

# Maximum number of rotated log files. (integer value)
#max_logfile_count = 30

# Log file maximum size in MB. This option is ignored if "log_rotation_type"
↳is
# not set to "size". (integer value)
#max_logfile_size_mb = 200

# Log rotation type. (string value)
# Possible values:
# interval - Rotate logs at predefined time intervals.
# size - Rotate logs once they reach a predefined size.
# none - Do not rotate log files.
#log_rotation_type = none

# Format string to use for log messages with context. Used by
# oslo_log.formatters.ContextFormatter (string value)
#logging_context_format_string = %(asctime)s.%(msecs)03d %(process)d
↳%(levelname)s %(name)s [%s(global_request_id)s %(request_id)s %(user_
↳identity)s] %(instance)s%(message)s

# Format string to use for log messages when context is undefined. Used by
# oslo_log.formatters.ContextFormatter (string value)
#logging_default_format_string = %(asctime)s.%(msecs)03d %(process)d
↳%(levelname)s %(name)s [-] %(instance)s%(message)s

# Additional data to append to log message when logging level for the message
# is DEBUG. Used by oslo_log.formatters.ContextFormatter (string value)
#logging_debug_format_suffix = %(funcName)s %(pathname)s:%(lineno)d

# Prefix each line of exception output with this format. Used by
# oslo_log.formatters.ContextFormatter (string value)
#logging_exception_prefix = %(asctime)s.%(msecs)03d %(process)d ERROR
↳%(name)s %(instance)s

# Defines the format string for %(user_identity)s that is used in
# logging_context_format_string. Used by oslo_log.formatters.ContextFormatter

```

(continues on next page)

(continued from previous page)

```

# (string value)
#logging_user_identity_format = %(user)s %(project)s %(domain)s %(system_
↳scope)s %(user_domain)s %(project_domain)s

# List of package logging levels in logger=LEVEL pairs. This option is ignored
# if log_config_append is set. (list value)
#default_log_levels = amqp=WARN,amqplib=WARN,boto=WARN,qpidd=WARN,
↳sqlalchemy=WARN,suds=INFO,oslo.messaging=INFO,oslo_messaging=INFO,
↳iso8601=WARN,requests.packages.urllib3.connectionpool=WARN,urllib3.
↳connectionpool=WARN,websocket=WARN,requests.packages.urllib3.util.
↳retry=WARN,urllib3.util.retry=WARN,keystonemiddleware=WARN,routes.
↳middleware=WARN,stevedore=WARN,taskflow=WARN,keystoneauth=WARN,oslo.
↳cache=INFO,oslo_policy=INFO,dogpile.core.dogpile=INFO

# Enables or disables publication of error events. (boolean value)
#publish_errors = false

# The format for an instance that is passed with the log message. (string
# value)
#instance_format = "[instance: %(uuid)s] "

# The format for an instance UUID that is passed with the log message. (string
# value)
#instance_uuid_format = "[instance: %(uuid)s] "

# Interval, number of seconds, of log rate limiting. (integer value)
#rate_limit_interval = 0

# Maximum number of logged messages per rate_limit_interval. (integer value)
#rate_limit_burst = 0

# Log level name used by rate limiting: CRITICAL, ERROR, INFO, WARNING, DEBUG
# or empty string. Logs with level greater or equal to rate_limit_except_level
# are not filtered. An empty string means that all levels are filtered.↳
↳(string
# value)
#rate_limit_except_level = CRITICAL

# Enables or disables fatal status of deprecations. (boolean value)
#fatal_deprecations = false

[agent]

#
# From ironic-neutron-agent
#

# Seconds between nodes reporting state to server; should be less than

```

(continues on next page)

(continued from previous page)

```
# agent_down_time, best if it is half or less than agent_down_time. (floating
# point value)
#report_interval = 30

# Log agent heartbeats (boolean value)
#log_agent_heartbeats = false

[ironic]

#
# From ironic-client
#

# DEPRECATED: Method to use for authentication: noauth or keystone. (string
# value)
# Possible values:
# keystone - <No description provided>
# noauth - <No description provided>
# This option is deprecated for removal.
# Its value may be silently ignored in the future.
# Reason: This option is no longer used, please use the [ironic]/auth_type
# option instead.
#auth_strategy = keystone

# The default service_type for endpoint URL discovery. (string value)
#service_type = <None>

# The default service_name for endpoint URL discovery. (string value)
#service_name = <None>

# List of interfaces, in order of preference, for endpoint URL. (list value)
#valid_interfaces = <None>

# The default region_name for endpoint URL discovery. (string value)
# Deprecated group/name - [ironic]/os_region
#region_name = <None>

# Always use this endpoint URL for requests for this client. NOTE: The
# unversioned endpoint should be specified here; to request a particular API
# version, use the `version`, `min-version`, and/or `max-version` options.
# (string value)
# Deprecated group/name - [ironic]/ironic_url
#endpoint_override = <None>

# Minimum Major API version within a given Major API version for endpoint URL
# discovery. Mutually exclusive with min_version and max_version (string
# value)
#version = <None>
```

(continues on next page)

(continued from previous page)

```
# The minimum major version of a given API, intended to be used as the lower
# bound of a range with max_version. Mutually exclusive with version. If
# min_version is given with no max_version it is as if max version is "latest
↳".
# (string value)
#min_version = <None>

# The maximum major version of a given API, intended to be used as the upper
# bound of a range with min_version. Mutually exclusive with version. (string
# value)
#max_version = <None>

# The maximum number of retries that should be attempted for connection_
↳errors.
# (integer value)
#connect_retries = <None>

# Delay (in seconds) between two retries for connection errors. If not set,
# exponential retry starting with 0.5 seconds up to a maximum of 60 seconds is
# used. (floating point value)
#connect_retry_delay = <None>

# The maximum number of retries that should be attempted for retrieable HTTP
# status codes. (integer value)
# Deprecated group/name - [ironic]/max_retries
#status_code_retries = <None>

# Delay (in seconds) between two retries for retrieable status codes. If not
# set, exponential retry starting with 0.5 seconds up to a maximum of 60
# seconds is used. (floating point value)
# Deprecated group/name - [ironic]/retry_interval
#status_code_retry_delay = <None>

# DEPRECATED: The default interface for endpoint URL discovery. (string value)
# This option is deprecated for removal.
# Its value may be silently ignored in the future.
# Reason: Using valid-interfaces is preferrable because it is capable of
# accepting a list of possible interfaces.
#interface = <None>

# PEM encoded Certificate Authority to use when verifying HTTPs connections.
# (string value)
#cafile = <None>

# PEM encoded client certificate cert file (string value)
#certfile = <None>

# PEM encoded client certificate key file (string value)
```

(continues on next page)

(continued from previous page)

```
#keyfile = <None>

# Verify HTTPS connections. (boolean value)
#insecure = false

# Timeout value for http requests (integer value)
#timeout = <None>

# Collect per-API call timing information. (boolean value)
#collect_timing = false

# Log requests to multiple loggers. (boolean value)
#split_loggers = false

# Authentication URL (string value)
#auth_url = <None>

# Scope for system operations (string value)
#system_scope = <None>

# Domain ID to scope to (string value)
#domain_id = <None>

# Domain name to scope to (string value)
#domain_name = <None>

# Project ID to scope to (string value)
#project_id = <None>

# Project name to scope to (string value)
#project_name = <None>

# Domain ID containing project (string value)
#project_domain_id = <None>

# Domain name containing project (string value)
#project_domain_name = <None>

# ID of the trust to use as a trustee use (string value)
#trust_id = <None>

# User ID (string value)
#user_id = <None>

# Username (string value)
# Deprecated group/name - [ironic]/user_name
#username = <None>

# User's domain id (string value)
```

(continues on next page)

(continued from previous page)

```
#user_domain_id = <None>

# User's domain name (string value)
#user_domain_name = <None>

# User's password (string value)
#password = <None>
```

3.1.2 Configuration Reference

The following pages describe configuration options that can be used to adjust the neutron ML2 configuration and the baremetal ML2 plug-in and device drivers to your particular situation.

To enable mechanism drivers in the ML2 plug-in, edit the `/etc/neutron/plugins/ml2/ml2_conf.ini` configuration file. For example, this enables the `openvswitch` and `baremetal` mechanism drivers:

```
[ml2]
mechanism_drivers = openvswitch,baremetal
```

To add a device to manage, edit the `/etc/neutron/plugins/ml2/ml2_conf.ini` configuration file. The example below enables devices: `device_a.example.net` and `device_b.example.net`. For each device a separate section in the same configuration file defines the device and driver specific configuration. Please refer to *Device drivers* for details.

```
[networking_baremetal]
enabled_device = device_a.example.net,device_b.example.net
```

Device drivers

The baremetal mechanism ML2 plug-in provides a device driver plug-in interface, this interface can be used to add device (switch) configuration capabilities. The interface uses `stevedore` for dynamic loading.

Individual drivers may have independent configuration requirements depending on the implementation. *Driver specific options* are documented separately.

Common configuration options for all device drivers

This page describes configuration options that is common to all networking- baremetal device drivers. Individual drivers may have independent configuration requirements depending on the implementation, refer to the device driver specific documentation.

Configuration options

common-example

driver

Type string

Default <None>

The driver to use when configuring the device

switch_id

Type string

Default <None>

The switch ID, MAC address of the device.

switch_info

Type string

Default <None>

Optional string field to be used to store any vendor-specific information.

physical_networks

Type list

Default []

A list of physical networks mapped to this device.

manage_vlans

Type boolean

Default True

Set this to False for the device if VLANs should not be create and deleted on the device.

networking_baremetal

enabled_devices

Type list

Default ['common-example', 'netconf-openconfig-example']

This option has a sample default set, which means that its actual default value may vary from the one documented above.

Enabled devices for which the plugin should manage configuration. Driver specific configuration for each device must be added in separate sections.

Sample Configuration File

The following is a sample configuration section that would be added to `/etc/neutron/plugins/ml2/ml2_conf.ini`.

The sample configuration can also be viewed in `file` form.

Important: The sample configuration file is auto-generated from `networking-baremetal` when this documentation is built. You must ensure your version of `networking-baremetal` matches the version of this documentation.

```
[DEFAULT]

[common-example]

#
# From common-device-driver-opts
#

# The driver to use when configuring the device (string value)
#driver = <None>

# The switch ID, MAC address of the device. (string value)
#switch_id = <None>

# Optional string field to be used to store any vendor-specific information.
# (string value)
#switch_info = <None>

# A list of physical networks mapped to this device. (list value)
#physical_networks =

# Set this to False for the device if VLANs should not be create and deleted
↪on
# the device. (boolean value)
#manage_vlans = true

[networking_baremetal]

#
# From common-device-driver-opts
#

# Enabled devices for which the plugin should manageconfiguration. Driver
# specific configuration for each device must be added in separate sections.
# (list value)
#
```

(continues on next page)

(continued from previous page)

```
# This option has a sample default set, which means that
# its actual default value may vary from the one documented
# below.
#enabled_devices = common-example,netconf-openconfig-example
```

Available device drivers

Device driver - netconf-openconfig

The `netconf-openconfig` device driver uses the Network Configuration Protocol (NETCONF) and open source vendor-neutral OpenConfig YANG models.

This driver has been tested with the following switch vendor/operating systems:

- Cisco NXOS
- Arista vEOS

Example configuration for Cisco NXOS device:

```
[networking_baremetal]
enabled_devices = nexus.example.net

[nexus.example.net]
driver = netconf-openconfig
device_params = name:nexus
switch_info = nexus
switch_id = 00:53:00:0a:0a:0a
host = nexus.example.net
username = user
key_filename = /etc/neutron/ssh_keys/nexus_sshkey
```

Example configuration for Arista EOS device:

```
[networking_baremetal]
enabled_devices = arista.example.net

[arista.example.net]
driver = netconf-openconfig
device_params = name:default
switch_info = arista
switch_id = 00:53:00:0b:0b:0b
host = arista.example.net
username = user
key_filename = /etc/neutron/ssh_keys/arista_sshkey
```


Configuration options

netconf-openconfig-example

driver

Type string

Default <None>

The driver to use when configuring the device

switch_id

Type string

Default <None>

The switch ID, MAC address of the device.

switch_info

Type string

Default <None>

Optional string field to be used to store any vendor-specific information.

physical_networks

Type list

Default []

A list of physical networks mapped to this device.

manage_vlans

Type boolean

Default True

Set this to False for the device if VLANs should not be create and deleted on the device.

network_instance

Type string

Default default

Advanced Option Intended for advanced users and not used by the majority of users, and might have a significant effect on stability and/or performance.

The L2, L3, or L2+L3 forwarding instance to use when defining VLANs on the device.

port_id_re_sub

Type dict

Default {'pattern': 'Ethernet', 'repl': 'eth'}

This option has a sample default set, which means that its actual default value may vary from the one documented above.

Regular expression pattern and replacement string. Some devices do not use the port description from LLDP in Netconf configuration. If the regular expression pattern and replacement string is set the port_id will be modified before passing configuration to the device.

disabled_properties

Type list

Default []

A list of properties that should not be used, currently only port_mtu is valid

manage_lacp_aggregates

Type boolean

Default True

When set to true the driver will manage LACP aggregates if link_group_information is defined in the binding:profile. When this is false the driver expect the link aggregation to be pre-configured on the device, and only perform vlan plugging.

link_aggregate_prefix

Type string

Default Port-Channel

The device specific prefix used for link-aggregation ports. Common values: po, port-channel or Port-Channel.

link_aggregate_range

Type string

Default 1000..2000

Range of link aggregation interface IDs that the driver can use when managing link aggregates.

host

Type string

Default device.example.com

This option has a sample default set, which means that its actual default value may vary from the one documented above.

The hostname or IP address to use for connecting to the netconf device.

username

Type string

Default netconf

This option has a sample default set, which means that its actual default value may vary from the one documented above.

The username to use for SSH authentication.

port

Type integer

Default 830

The port to use for connection to the netconf device.

password

Type string

Default secret

This option has a sample default set, which means that its actual default value may vary from the one documented above.

The password used if using password authentication, or the passphrase to use for unlocking keys that require it. (To disable attempting key authentication altogether, set options *allow_agent* and *look_for_keys* to *False*.)

key_filename

Type string

Default ~/.ssh/id_rsa

Private key filename

hostkey_verify

Type boolean

Default True

Enables hostkey verification from ~/.ssh/known_hosts

device_params

Type dict

Default {'name': 'default'}

ncclient device handler parameters, see ncclient documentation for supported device handlers.

allow_agent

Type boolean

Default True

Enables querying SSH agent (if found) for keys.

look_for_keys

Type boolean

Default True

Enables looking in the usual locations for ssh keys (e.g. ~/.ssh/id_*)

networking_baremetal

enabled_devices

Type list

Default ['common-example', 'netconf-openconfig-example']

This option has a sample default set, which means that its actual default value may vary from the one documented above.

Enabled devices for which the plugin should manage configuration. Driver specific configuration for each device must be added in separate sections.

Sample Configuration File

The following is a sample configuration section that would be added to `/etc/neutron/plugins/ml2/ml2_conf.ini`.

The sample configuration can also be viewed in `file` form.

Important: The sample configuration file is auto-generated from `networking-baremetal` when this documentation is built. You must ensure your version of `networking-baremetal` matches the version of this documentation.

```
[DEFAULT]

[netconf-openconfig-example]

#
# From netconf-openconfig-driver-opts
#

# The driver to use when configuring the device (string value)
#driver = <None>

# The switch ID, MAC address of the device. (string value)
#switch_id = <None>

# Optional string field to be used to store any vendor-specific information.
# (string value)
#switch_info = <None>

# A list of physical networks mapped to this device. (list value)
#physical_networks =

# Set this to False for the device if VLANs should not be create and deleted.
↪on
# the device. (boolean value)
```

(continues on next page)

(continued from previous page)

```
#manage_vlans = true

# Regular expression pattern and replacement string. Some devices do not use
# the port description from LLDP in Netconf configuration. If the regular
# expression pattern and replacement string is set the port_id will be
↳modified
# before passing configuration to the device. (dict value)
#
# This option has a sample default set, which means that
# its actual default value may vary from the one documented
# below.
#port_id_re_sub = pattern:Ethernet,repl:eth

# A list of properties that should not be used, currently only "port_mtu" is
# valid (list value)
#disabled_properties =

# When set to true the driver will manage LACP aggregates if
# link_group_information is defined in the binding:profile. When this is false
# the driver expect the link aggregation to be pre-configured on the device,
# and only perform vlan plugging. (boolean value)
#manage_lacp_aggregates = true

# The device specific prefix used for link-aggregation ports. Common values:
# "po", "port-channel" or "Port-Channel". (string value)
#link_aggregate_prefix = Port-Channel

# Range of link aggregation interface IDs that the driver can use when
↳managing
# link aggregates. (string value)
#link_aggregate_range = 1000..2000

# The hostname or IP address to use for connecting to the netconf device.
# (string value)
#
# This option has a sample default set, which means that
# its actual default value may vary from the one documented
# below.
#host = device.example.com

# The username to use for SSH authentication. (string value)
#
# This option has a sample default set, which means that
# its actual default value may vary from the one documented
# below.
#username = netconf

# The port to use for connection to the netconf device. (integer value)
#port = 830
```

(continues on next page)

(continued from previous page)

```
# The password used if using password authentication, or the passphrase to use
# for unlocking keys that require it. (To disable attempting key_
↪authentication
# altogether, set options *allow_agent* and *look_for_keys* to `False`. (string
# value)
#
# This option has a sample default set, which means that
# its actual default value may vary from the one documented
# below.
#password = secret

# Private key filename (string value)
#key_filename = ~/.ssh/id_rsa

# Enables hostkey verification from ~/.ssh/known_hosts (boolean value)
#hostkey_verify = true

# ncclient device handler parameters, see ncclient documentation for supported
# device handlers. (dict value)
#device_params = name:default

# Enables querying SSH agent (if found) for keys. (boolean value)
#allow_agent = true

# Enables looking in the usual locations for ssh keys (e.g.
# :file:`~/.ssh/id_*`) (boolean value)
#look_for_keys = true

# The L2, L3, or L2+L3 forwarding instance to use when defining VLANs on the
# device. (string value)
# Advanced Option: intended for advanced users and not used
# by the majority of users, and might have a significant
# effect on stability and/or performance.
#network_instance = default

[networking_baremetal]

#
# From netconf-openconfig-driver-opts
#
# Enabled devices for which the plugin should manageconfiguration. Driver
# specific configuration for each device must be added in separate sections.
# (list value)
#
# This option has a sample default set, which means that
# its actual default value may vary from the one documented
```

(continues on next page)

(continued from previous page)

```
# below.  
#enabled_devices = common-example,netconf-openconfig-example
```


CONTRIBUTOR GUIDE

4.1 Contributing

This document provides some necessary points for developers to consider when writing and reviewing networking-baremetal code.

4.1.1 Getting Started

If you're completely new to OpenStack and want to contribute to the networking-baremetal project, please start by familiarizing yourself with the [Infra Teams Developer Guide](#). This will help you get your accounts set up in Launchpad and Gerrit, familiarize you with the workflow for the OpenStack continuous integration and testing systems, and help you with your first commit.

LaunchPad Project

Most of the tools used for OpenStack require a launchpad.net ID for authentication.

See also:

- <https://launchpad.net>
- <https://launchpad.net/ironic>

Related Projects

Networking Baremetal is tightly integrated with the ironic and neutron projects. Ironic and its related projects are developed by the same community.

See also:

- <https://launchpad.net/ironic>
- <https://launchpad.net/neutron>

Project Hosting Details

Bug tracker <https://bugs.launchpad.net/networking-baremetal>

Mailing list (prefix Subject line with [ironic][networking-baremetal]) <http://lists.openstack.org/cgi-bin/mailman/listinfo/openstack-discuss>

Code Hosting <https://opendev.org/openstack/networking-baremetal>

Code Review <https://review.opendev.org/#/q/status:open+project:openstack/networking-baremetal,n,z>

4.1.2 Developer quick-starts

These are quick walk throughs to get you started developing code for networking-baremetal. These assume you are already familiar with submitting code reviews to an OpenStack project.

Deploying networking-baremetal with DevStack

DevStack may be configured to deploy networking-baremetal Networking service plugin. It is highly recommended to deploy on an expendable virtual machine and not on your personal work station. Deploying networking-baremetal with DevStack requires a machine running Ubuntu 14.04 (or later) or Fedora 20 (or later).

See also:

<http://docs.openstack.org/devstack/latest>

Create `devstack/local.conf` with minimal settings required to enable networking-baremetal with ironic. Here is an example of `local.conf`:

```
cd devstack
cat >local.conf <<END
[[local|localrc]]
# Credentials
ADMIN_PASSWORD=password
DATABASE_PASSWORD=password
RABBIT_PASSWORD=password
SERVICE_PASSWORD=password
SERVICE_TOKEN=password
SWIFT_HASH=password
SWIFT_TEMPURL_KEY=password

# Enable networking-baremetal plugin
enable_plugin networking-baremetal https://opendev.org/openstack/networking-
↪baremetal.git
enable_service ir-neutronagt

# Enable ironic plugin
enable_plugin ironic https://opendev.org/openstack/ironic
enable_service networking_baremetal

# Enable neutron which is required by ironic and disable nova-network.
```

(continues on next page)

(continued from previous page)

```
disable_service n-net
disable_service n-novnc
enable_service q-svc
enable_service q-agt
enable_service q-dhcp
enable_service q-l3
enable_service q-meta
enable_service neutron

# Enable swift for agent_* drivers
enable_service s-proxy
enable_service s-object
enable_service s-container
enable_service s-account

# Disable horizon
disable_service horizon

# Disable heat
disable_service heat h-api h-api-cfn h-api-cw h-eng

# Disable cinder
disable_service cinder c-sch c-api c-vol

# Swift temp URL's are required for agent_* drivers.
SWIFT_ENABLE_TEMPURLS=True

# Create 3 virtual machines to pose as ironic's baremetal nodes.
IRONIC_VM_COUNT=3
IRONIC_VM_SSH_PORT=22
IRONIC_BAREMETAL_BASIC_OPS=True
DEFAULT_INSTANCE_TYPE=baremetal

# Enable additional hardware types, if needed.
#IRONIC_ENABLED_HARDWARE_TYPES=ipmi,fake-hardware
# Don't forget that many hardware types require enabling of additional
# interfaces, most often power and management:
#IRONIC_ENABLED_MANAGEMENT_INTERFACES=ipmitool,fake
#IRONIC_ENABLED_POWER_INTERFACES=ipmitool,fake
# The 'ipmi' hardware type's default deploy interface is 'iscsi'.
# This would change the default to 'direct':
#IRONIC_DEFAULT_DEPLOY_INTERFACE=direct

# Change this to alter the default driver for nodes created by devstack.
# This driver should be in the enabled list above.
IRONIC_DEPLOY_DRIVER=ipmi

# The parameters below represent the minimum possible values to create
# functional nodes.
```

(continues on next page)

(continued from previous page)

```
IRONIC_VM_SPECS_RAM=1280
IRONIC_VM_SPECS_DISK=10

# Size of the ephemeral partition in GB. Use 0 for no ephemeral partition.
IRONIC_VM_EPHEMERAL_DISK=0

# To build your own IPA ramdisk from source, set this to True
IRONIC_BUILD_DEPLOY_RAMDISK=False

VIRT_DRIVER=ironic

# By default, DevStack creates a 10.0.0.0/24 network for instances.
# If this overlaps with the hosts network, you may adjust with the
# following.
NETWORK_GATEWAY=10.1.0.1
FIXED_RANGE=10.1.0.0/24
FIXED_NETWORK_SIZE=256

# Log all output to files
LOGFILE=$HOME/devstack.log
LOGDIR=$HOME/logs
IRONIC_VM_LOG_DIR=$HOME/ironic-bm-logs

END
```

Deploying networking-baremetal and multi-tenant networking with DevStack

DevStack may be configured to deploy networking-baremetal Networking service plugin together with networking-generic-switch for multi-tenant networking. It is highly recommended to deploy on an expendable virtual machine and not on your personal work station. Deploying networking-baremetal with DevStack requires a machine running Ubuntu 14.04 (or later) or Fedora 20 (or later).

See also:

<http://docs.openstack.org/devstack/latest>

Create `devstack/local.conf` with minimal settings required to enable networking-baremetal with ironic and networking-generic-switch for multi-tenant networking. Here is an example of `local.conf`:

```
[[local|localrc]]

# Credentials
ADMIN_PASSWORD=password
DATABASE_PASSWORD=password
RABBIT_PASSWORD=password
SERVICE_PASSWORD=password
SERVICE_TOKEN=password
SWIFT_HASH=password
SWIFT_TEMPURL_KEY=password
```

(continues on next page)

(continued from previous page)

```
# Install networking-generic-switch Neutron ML2 driver that interacts with OVS
enable_plugin networking-generic-switch https://opendev.org/openstack/
↳networking-generic-switch

# Enable networking-baremetal plugin
enable_plugin networking-baremetal https://opendev.org/openstack/networking-
↳baremetal.git
enable_service networking_baremetal
enable_service ir-neutronagt

# Add link local info when registering Ironic node
IRONIC_USE_LINK_LOCAL=True

IRONIC_ENABLED_NETWORK_INTERFACES=flat,neutron
IRONIC_NETWORK_INTERFACE=neutron

#Networking configuration
OVS_PHYSICAL_BRIDGE=brbm
PHYSICAL_NETWORK=mynetwork
IRONIC_PROVISION_NETWORK_NAME=ironic-provision
IRONIC_PROVISION_PROVIDER_NETWORK_TYPE=vlan
IRONIC_PROVISION_SUBNET_PREFIX=10.0.5.0/24
IRONIC_PROVISION_SUBNET_GATEWAY=10.0.5.1
Q_PLUGIN=m12
ENABLE_TENANT_VLANS=True
Q_ML2_TENANT_NETWORK_TYPE=vlan
TENANT_VLAN_RANGE=100:150
Q_USE_PROVIDERNET_FOR_PUBLIC=False

# Enable segments service_plugin for routed networks
Q_SERVICE_PLUGIN_CLASSES=neutron.services.l3_router.l3_router_plugin.
↳L3RouterPlugin,segments
IRONIC_USE_NEUTRON_SEGMENTS=True

# Configure ironic from ironic devstack plugin.
enable_plugin ironic https://opendev.org/openstack/ironic

# Enable Ironic API and Ironic Conductor
enable_service ironic
enable_service ir-api
enable_service ir-cond

# Enable Neutron which is required by Ironic and disable nova-network.
disable_service n-net
disable_service n-novnc
enable_service q-svc
enable_service q-agt
enable_service q-dhcp
enable_service q-l3
```

(continues on next page)

(continued from previous page)

```
enable_service q-meta
enable_service neutron

# Enable Swift for agent_* drivers
enable_service s-proxy
enable_service s-object
enable_service s-container
enable_service s-account

# Disable Horizon
disable_service horizon

# Disable Heat
disable_service heat h-api h-api-cfn h-api-cw h-eng

# Disable Cinder
disable_service cinder c-sch c-api c-vol

# Swift temp URL's are required for agent_* drivers.
SWIFT_ENABLE_TEMPURLS=True

# Create 3 virtual machines to pose as Ironic's baremetal nodes.
IRONIC_VM_COUNT=3
IRONIC_BAREMETAL_BASIC_OPS=True
DEFAULT_INSTANCE_TYPE=baremetal

# Enable additional hardware types, if needed.
#IRONIC_ENABLED_HARDWARE_TYPES=ipmi,fake-hardware
# Don't forget that many hardware types require enabling of additional
# interfaces, most often power and management:
#IRONIC_ENABLED_MANAGEMENT_INTERFACES=ipmitool,fake
#IRONIC_ENABLED_POWER_INTERFACES=ipmitool,fake
# The 'ipmi' hardware type's default deploy interface is 'iscsi'.
# This would change the default to 'direct':
#IRONIC_DEFAULT_DEPLOY_INTERFACE=direct

# Change this to alter the default driver for nodes created by devstack.
# This driver should be in the enabled list above.
IRONIC_DEPLOY_DRIVER=ipmi

# The parameters below represent the minimum possible values to create
# functional nodes.
IRONIC_VM_SPECS_RAM=1024
IRONIC_VM_SPECS_DISK=10

# Size of the ephemeral partition in GB. Use 0 for no ephemeral partition.
IRONIC_VM_EPHEMERAL_DISK=0

# To build your own IPA ramdisk from source, set this to True
```

(continues on next page)

(continued from previous page)

```

IRONIC_BUILD_DEPLOY_RAMDISK=False

VIRT_DRIVER=ironic

# By default, DevStack creates a 10.0.0.0/24 network for instances.
# If this overlaps with the hosts network, you may adjust with the
# following.
NETWORK_GATEWAY=10.1.0.1
FIXED_RANGE=10.1.0.0/24
FIXED_NETWORK_SIZE=256

# Log all output to files
LOGFILE=$HOME/devstack.log
LOGDIR=$HOME/logs
IRONIC_VM_LOG_DIR=$HOME/ironic-bm-logs

```

Virtual lab with virtual switch and netconf-openconfig Device Driver

Ansible playbooks that can be used to set up a lab for developing networking- baremetal network device integration is hosted on [GitHub](#).

4.1.3 Full networking-baremetal python API reference

- [modindex](#)

[networking_baremetal](#)

[networking_baremetal package](#)

[Subpackages](#)

[networking_baremetal.agent package](#)

[Submodules](#)

[networking_baremetal.agent.ironic_neutron_agent module](#)

```
class networking_baremetal.agent.ironic_neutron_agent.BaremetalNeutronAgent
```

```
    Bases: oslo_service.service.ServiceBase
```

```
    get_template_node_state(node_uuid)
```

```
    reset()
```

```
        Reset service.
```

```
        Called in case service running in daemon mode receives SIGHUP.
```

start()

Start service.

stop()

Stop service.

wait()

Wait for service to complete.

class networking_baremetal.agent.ironic_neutron_agent.
HashRingMemberManagerNotificationEndpoint

Bases: object

Class variables members and hashring is shared by all instances

filter_rule = <oslo_messaging.notify.filter.NotificationFilter object>

hashring = <tooz.hashring.HashRing object>

info(*ctxt, publisher_id, event_type, payload, metadata*)

members = []

networking_baremetal.agent.ironic_neutron_agent.**list_opts()**

networking_baremetal.agent.ironic_neutron_agent.**main()**

Module contents

networking_baremetal.drivers package

Submodules

networking_baremetal.drivers.base module

class networking_baremetal.drivers.base.**BaseDeviceClient**(*device*)

Bases: object

edit_config(*config*)

Edit configuration on the device

Parameters **config** The configuration to apply to the device

get(***kwargs*)

Get current configuration/state from device

get_client_args()

Get client connection arguments from configuration

class networking_baremetal.drivers.base.**BaseDeviceDriver**(*device*)

Bases: object

SUPPORTED_BOND_MODES = {}

create_network(*context*)

Create network on device

Parameters **context** NetworkContext instance describing the new network.

create_port(*context, segment, links*)

Create/Configure port on device

Parameters

- **context** PortContext instance describing the new state of the port, as well as the original state prior to the `update_port` call.
- **segment** segment dictionary describing segment to bind
- **links** Local link information filtered for the device.

delete_network(*context*)

Delete network on device

Parameters **context** NetworkContext instance describing the new network.

delete_port(*context, links, current=True*)

Delete/Un-configure port on device

Parameters

- **context** PortContext instance describing the new state of the port, as well as the original state prior to the `update_port` call.
- **links** Local link information filtered for the device.
- **current** Boolean, when true use `context.current`, when false use `context.original`

load_config()

Register driver specific configuration

All drivers should register driver specific options in the device specific config group. This method will be called during mechanism driver initialization.

update_network(*context*)

Update network on device

Parameters **context** NetworkContext instance describing the new network.

update_port(*context, links*)

Update port on device

Parameters

- **context** PortContext instance describing the new state of the port, as well as the original state prior to the `update_port` call.
- **links** Local link information filtered for the device.

validate()

Driver validation

This method will be called during mechanism driver initialization. Raising any exception other than `DriverValidationError` will cause service initialization failure.

Raises *DriverValidationError* On validation failure.

Module contents

networking_baremetal.openconfig package

Subpackages

networking_baremetal.openconfig.interfaces package

Submodules

networking_baremetal.openconfig.interfaces.aggregate module

class

networking_baremetal.openconfig.interfaces.aggregate.InterfacesAggregation

Bases: object

Options for logical interfaces representing aggregates

NAMESPACE = 'http://openconfig.net/yang/interfaces/aggregate'

PARENT = 'interface'

TAG = 'aggregation'

property config

property switched_vlan

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

class networking_baremetal.openconfig.interfaces.aggregate.InterfacesAggregationConfig(*Operation*

str
=
Net-
con-
fEd-
it-
Con-
fig-
Op-
er-
a-
tion.M

Bases: object

NAMESPACE = 'http://openconfig.net/yang/interfaces/aggregate'

PARENT = 'aggregation'
TAG = 'config'
property lag_type
property min_links
property operation
RFC 6241 - <edit-config> operation attribute
to_xml_element()
Create XML Element
Returns ElementTree Element with SubElements

networking_baremetal.openconfig.interfaces.ethernet module

class networking_baremetal.openconfig.interfaces.ethernet.InterfacesEthernet

Bases: object

Ethernet configuration and state

NAMESPACE = 'http://openconfig.net/yang/interfaces/ethernet'

PARENT = 'interface'

TAG = 'ethernet'

property config

Configuration parameters for interface

property switched_vlan

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

class networking_baremetal.openconfig.interfaces.ethernet.InterfacesEthernetConfig(*operation=*

Bases: object

OpenConfig interface ethernet configuration

NAMESPACE = 'http://openconfig.net/yang/interfaces'

PARENT = 'interface'

TAG = 'config'

property aggregate_id

Logical aggregate interface for interface

property operation

RFC 6241 - <edit-config> operation attribute

`to_xml_element()`

Create XML Element

Returns ElementTree Element with SubElements

`networking_baremetal.openconfig.interfaces.interfaces` module

class `networking_baremetal.openconfig.interfaces.interfaces.BaseInterface`(*name:*
str)

Bases: object

Base interface

NAMESPACE = 'http://openconfig.net/yang/interfaces'

PARENT = 'interfaces'

TAG = 'interface'

property `config`

Configuration parameters for interface

property `name`

The name of the interface.

`to_xml_element()`

Create XML Element

Returns ElementTree Element with SubElements

class `networking_baremetal.openconfig.interfaces.interfaces.InterfaceAggregate`(*name:*
str,
op-
er-
a-
tion:
str
=
Net-
con-
fEd-
it-
Con-
fig-
Op-
er-
a-
tion.MERGE)

Bases: `networking_baremetal.openconfig.interfaces.interfaces.BaseInterface`

property `aggregation`

Ethernet configuration and state

property operation

RFC 6241 - <edit-config> operation attribute

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

class networking_baremetal.openconfig.interfaces.interfaces.**InterfaceConfig**(*operation=NetconfE*

name:
Op-
tional[str]
 =
None,
de-
scrip-
tion:
Op-
tional[str]
 =
None,
en-
abled:
Op-
tional[bool]
 =
None,
mtu:
Op-
tional[int]
 =
None)

Bases: object

OpenConfig interface configuration

NAMESPACE = 'http://openconfig.net/yang/interfaces'

PARENT = 'interface'

TAG = 'config'

property description

A textual description of the interface

property enabled

The configured, desired state of the interface

property mtu

The max transmission unit size in octets

property name

The name of the interface.

property operation

RFC 6241 - <edit-config> operation attribute

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

class networking_baremetal.openconfig.interfaces.interfaces.**InterfaceEthernet** (*name:*
str)

Bases: *networking_baremetal.openconfig.interfaces.interfaces.BaseInterface*

property ethernet

Ethernet configuration and state

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

class networking_baremetal.openconfig.interfaces.interfaces.**Interfaces**

Bases: *collections.abc.Collection*

Group/List of interfaces

NAMESPACE = 'http://openconfig.net/yang/interfaces'

TAG = 'interfaces'

add(*name: str, interface_type: str = 'ethernet'*)

Add interface

Parameters

- **name** Interface name
- **interface_type** Interface type (ethernet, aggregate, base)

Type str

Type str

property interfaces

List of interfaces

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

networking_baremetal.openconfig.interfaces.types module

class networking_baremetal.openconfig.interfaces.types.**AggregationType**(*value*)

Bases: enum.Enum

An enumeration.

LACP = 'LACP'

STATIC = 'SATIC'

Module contents

networking_baremetal.openconfig.lacp package

Submodules

networking_baremetal.openconfig.lacp.lacp module

class networking_baremetal.openconfig.lacp.lacp.**LACP**

Bases: object

LACP Top level

LACP configuration and state variable containers

NAMESPACE = 'http://openconfig.net/yang/lacp'

TAG = 'lacp'

property interfaces

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

class networking_baremetal.openconfig.lacp.lacp.**LACPInterface**(*name: str, operation=NetconfEditConfigOperation.ME*)

Bases: object

Base LACP aggregate interface

NAMESPACE = 'http://openconfig.net/yang/lacp'

PARENT = 'interfaces'

TAG = 'interface'

property config

Configuration data for each LACP aggregate interface

property name

The name of the LACP aggregate interface.

property operation

RFC 6241 - <edit-config> operation attribute

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements**class** networking_baremetal.openconfig.lacp.lacp.LACPInterfaceConfig(*name: str,**operation=NetconfEditConfigOperation,*
interval=LACPPeriod.SLOW,
lacp_mode=LACPActivity.ACTIVE)

Bases: object

OpenConfig LACP aggregate interface configuration

NAMESPACE = 'http://openconfig.net/yang/lacp'**PARENT** = 'interface'**TAG** = 'config'**property interval**

The period between LACP messages

property lacp_mode

The LACP mode if the aggregate interface

property name

The name of the interface.

property operation

RFC 6241 - <edit-config> operation attribute

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements**class** networking_baremetal.openconfig.lacp.lacp.LACPInterfaces

Bases: collections.abc.Collection

Top-level grouping for LACP-enabled interfaces

NAMESPACE = 'http://openconfig.net/yang/lacp'**PARENT** = 'lacp'**TAG** = 'interfaces'**add**(*name: str*)

Add interface

Parameters **name** Interface name**Type** str

property interfaces

List of interfaces

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

networking_baremetal.openconfig.lacp.types module

class networking_baremetal.openconfig.lacp.types.LACPActivity(*value*)

Bases: enum.Enum

Describes the LACP membership type

Active or passive, of the interface in the aggregate. reference IEEE 802.1AX-2008

ACTIVE: Interface is an active member, i.e., will detect and maintain aggregates

PASSIVE: Interface is a passive member, i.e., it participates with an active partner

ACTIVE = 'ACTIVE'

PASSIVE = 'PASSIVE'

class networking_baremetal.openconfig.lacp.types.LACPPeriod(*value*)

Bases: enum.Enum

Defines the time between sending LACP messages

reference IEEE 802.3ad FAST: Send LACP packets every second SLOW: Send LACP packets every 30 seconds

FAST = 'FAST'

SLOW = 'SLOW'

Module contents**networking_baremetal.openconfig.network_instance package****Submodules****networking_baremetal.openconfig.network_instance.network_instance module**

class networking_baremetal.openconfig.network_instance.network_instance.NetworkInstance(*name*)

Bases: object

An OpenConfig description of a network_instance.

This may be a Layer 3 forwarding construct such as a virtual routing and forwarding (VRF) instance, or a Layer 2 instance such as a virtual switch instance (VSI). Mixed Layer 2 and Layer 3 instances are also supported.

NAMESPACE = 'http://openconfig.net/yang/network-instance'

TAG = 'network-instance'

property name

A unique name identifying the network instance

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

property vlans

Group/List of VLANs - keyed by id

class networking_baremetal.openconfig.network_instance.network_instance.
NetworkInstances

Bases: collections.abc.Collection

Top-level grouping containing a list of network instances.

NAMESPACE = 'http://openconfig.net/yang/network-instance'

TAG = 'network-instances'

add(*name: str*)

Add network instance

Parameters **name** A unique name identifying the network instance

Type str

Keyword arguments Network instance arguments

property network_instances

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

Module contents

networking_baremetal.openconfig.vlan package

Submodules

networking_baremetal.openconfig.vlan.types module

class networking_baremetal.openconfig.vlan.types.**VlanId**(*vlan_id: int*)

Bases: object

Type definition representing a single-tagged VLAN

property vlan_id

```
class networking_baremetal.openconfig.vlan.types.VlanInterfaceMode(value)
```

Bases: `enum.Enum`

VLAN interface mode (trunk or access)

```
ACCESS = 'ACCESS'
```

```
TRUNK = 'TRUNK'
```

```
class networking_baremetal.openconfig.vlan.types.VlanRange(vlan_range: str)
```

Bases: `object`

Type definition representing a range of single-tagged VLANs.

A range is specified as x..y where x and y are valid VLAN IDs (1 <= vlan-id <= 4094). The range is assumed to be inclusive, such that any VLAN-ID matching x <= VLAN-ID <= y falls within the range.

```
pattern =
```

```
re.compile('^(409[0-4]|40[0-8][0-9]|[1-3][0-9]{3}|[1-9][0-9]{1,2}|[1-9])\\  
\\.\\.(409[0-4]|40[0-8][0-9]|[1-3][0-9]{3}|[1-9][0-9]{1,2}|[1-9])$')
```

```
property vlan_range
```

```
class networking_baremetal.openconfig.vlan.types.VlanStatus(value)
```

Bases: `enum.Enum`

VLAN Admin state

ACTIVE: VLAN is active SUSPENDED: VLAN is inactive / suspended

```
ACTIVE = 'ACTIVE'
```

```
SUSPENDED = 'SUSPENDED'
```

networking_baremetal.openconfig.vlan.vlan module

```
class networking_baremetal.openconfig.vlan.vlan.TrunkVlans
```

Bases: `collections.abc.Collection`

```
add(value)
```

Add vlan or range of vlans (range: 100..200)

```
class networking_baremetal.openconfig.vlan.vlan.Vlan(vlan_id: int, operation=NetconfEditConfigOperation.MERGE)
```

Bases: `object`

Base vlan

```
NAMESPACE = 'http://openconfig.net/yang/vlan'
```

```
PARENT = 'vlans'
```

```
TAG = 'vlan'
```

property config

Configuration parameters for VLAN

property operation

RFC 6241 - <edit-config> operation attribute

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

property vlan_id

The id of the VLAN

```
class networking_baremetal.openconfig.vlan.vlan.VlanConfig(operation=NetconfEditConfigOperation.M
    vlan_id: Optional[int] =
    None, name:
    Optional[str] = None,
    status: Optional[str] =
    None)
```

Bases: object

OpenConfig VLAN configuration

NAMESPACE = 'http://openconfig.net/yang/vlan'

PARENT = 'vlan'

TAG = 'config'

property name

Interface VLAN name.

property operation

RFC 6241 - <edit-config> operation attribute

property status

Admin state of the VLAN

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

property vlan_id

The id of the VLAN

```
class networking_baremetal.openconfig.vlan.vlan.VlanSwitchedConfig(operation: str
    = NetconfEdit-
    ConfigOpera-
    tion.MERGE,
    inter-
    face_mode:
    Optional[str]
    = None,
    native_vlan:
    Optional[int]
    = None,
    access_vlan:
    Optional[int]
    = None)
```

Bases: object

Ethernet interface VLAN config

VLAN related configuration that is part of the physical Ethernet interface.

NAMESPACE = 'http://openconfig.net/yang/vlan'

PARENT = 'switched-vlan'

TAG = 'config'

property access_vlan

Access VLAN assigned to the interfaces

property interface_mode

Get the interface to access or trunk mode for VLANs

property native_vlan

Native VLAN

is valid for trunk mode interfaces

property operation

RFC 6241 - <edit-config> operation attribute

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

property trunk_vlans

Allowed VLANs may be specified for trunk mode interfaces

```
class networking_baremetal.openconfig.vlan.vlan.VlanSwitchedVlan
```

Bases: object

VLAN interface-specific data on Ethernet interfaces.

Enclosing container for VLAN interface-specific data on Ethernet interfaces. These are for standard L2, switched-style VLANs.

NAMESPACE = 'http://openconfig.net/yang/vlan'

PARENT = 'ethernet'

TAG = 'switched-vlan'

property config

Configuration parameters for VLANs

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

class networking_baremetal.openconfig.vlan.vlan.Vlans

Bases: collections.abc.Collection

Group/List of VLANs

NAMESPACE = 'http://openconfig.net/yang/vlan'

TAG = 'vlans'

add(vlan_id: int)

Add VLAN

Parameters vlan_id VLAN ID

Type int

Keyword arguments VLAN configuration

remove(vlan_id: int)

Remove VLAN

Parameters vlan_id VLAN ID

Type int

to_xml_element()

Create XML Element

Returns ElementTree Element with SubElements

property vlans

List of VLANs

Module contents

Module contents

networking_baremetal.plugins package

Subpackages

networking_baremetal.plugins.ml2 package

Submodules

networking_baremetal.plugins.ml2.baremetal_mech module

class `networking_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`

Bases: `neutron.plugins.ml2.drivers.mech_agent.SimpleAgentMechanismDriverBase`

property connectivity

Return the mechanism driver connectivity type

The possible values are l2, l3 and legacy (default).

Returns a string in (l2, l3, legacy)

create_network_postcommit(*context*)

Create a network.

Called after the transaction commits. Call can block, though will block the entire process so care should be taken to not drastically affect performance. Raising an exception will cause the deletion of the resource.

Parameters context NetworkContext instance describing the new network.

create_network_precommit(*context*)

Allocate resources for a new network.

Create a new network, allocating resources as necessary in the database. Called inside transaction context on session. Call cannot block. Raising an exception will result in a rollback of the current transaction.

Parameters context NetworkContext instance describing the new network.

create_port_postcommit(*context*)

Create a port.

Called after the transaction completes. Call can block, though will block the entire process so care should be taken to not drastically affect performance. Raising an exception will result in the deletion of the resource.

Parameters context PortContext instance describing the port.

create_port_precommit(*context*)

Allocate resources for a new port.

Create a new port, allocating resources as necessary in the database. Called inside transaction context on session. Call cannot block. Raising an exception will result in a rollback of the current transaction.

Parameters context PortContext instance describing the port.

create_subnet_postcommit(*context*)

Create a subnet.

Called after the transaction commits. Call can block, though will block the entire process so care should be taken to not drastically affect performance. Raising an exception will cause the deletion of the resource.

Parameters context SubnetContext instance describing the new subnet.

create_subnet_precommit(*context*)

Allocate resources for a new subnet.

Create a new subnet, allocating resources as necessary in the database. Called inside transaction context on session. Call cannot block. Raising an exception will result in a rollback of the current transaction.

Parameters context SubnetContext instance describing the new subnet.

delete_network_postcommit(*context*)

Delete a network.

Called after the transaction commits. Call can block, though will block the entire process so care should be taken to not drastically affect performance. Runtime errors are not expected, and will not prevent the resource from being deleted.

Parameters context NetworkContext instance describing the current state of the network, prior to the call to delete it.

delete_network_precommit(*context*)

Delete resources for a network.

Delete network resources previously allocated by this mechanism driver for a network. Called inside transaction context on session. Runtime errors are not expected, but raising an exception will result in rollback of the transaction.

Parameters context NetworkContext instance describing the current state of the network, prior to the call to delete it.

delete_port_postcommit(*context*)

Delete a port.

state of the port, prior to the call to delete it. Called after the transaction completes. Call can block, though will block the entire process so care should be taken to not drastically affect performance. Runtime errors are not expected, and will not prevent the resource from being deleted.

Parameters context PortContext instance describing the current state of the port, prior to the call to delete it.

delete_port_precommit(*context*)

Delete resources of a port.

Called inside transaction context on session. Runtime errors are not expected, but raising an exception will result in rollback of the transaction.

Parameters context PortContext instance describing the current state of the port, prior to the call to delete it.

delete_subnet_postcommit(*context*)

Delete a subnet.

Called after the transaction commits. Call can block, though will block the entire process so care should be taken to not drastically affect performance. Runtime errors are not expected, and will not prevent the resource from being deleted.

Parameters context SubnetContext instance describing the current state of the subnet, prior to the call to delete it.

delete_subnet_precommit(*context*)

Delete resources for a subnet.

Delete subnet resources previously allocated by this mechanism driver for a subnet. Called inside transaction context on session. Runtime errors are not expected, but raising an exception will result in rollback of the transaction.

Parameters context SubnetContext instance describing the current state of the subnet, prior to the call to delete it.

get_allowed_network_types(*agent*)

Return the agents or drivers allowed network types.

For example: return (flat,). You can also refer to the configuration the given agent exposes.

get_mappings(*agent*)

Return the agents bridge or interface mappings.

For example: agent[configurations].get(bridge_mappings, {}).

try_to_bind_segment_for_agent(*context, segment, agent*)

Try to bind with segment for agent.

Parameters

- **context** PortContext instance describing the port
- **segment** segment dictionary describing segment to bind
- **agent** agents_db entry describing agent to bind

Returns True iff segment has been bound for agent

Neutron segments api-ref: <https://docs.openstack.org/api-ref/network/v2/#segments>

Example segment dictionary: {**segmentation_id:** *segmentation_id*, **network_type:** *network_type*, **id:** *segment_uuid*}

Called outside any transaction during bind_port() so that derived MechanismDrivers can use agent_db data along with built-in knowledge of the corresponding agents capabilities to attempt to bind to the specified network segment for the agent.

If the segment can be bound for the agent, this function must call context.set_binding() with appropriate values and then return True. Otherwise, it must return False.

update_network_postcommit(*context*)

Update a network.

Called after the transaction commits. Call can block, though will block the entire process so care should be taken to not drastically affect performance. Raising an exception will cause the deletion of the resource. update_network_postcommit is called for all changes to the network state. It is up to the mechanism driver to ignore state or state changes that it does not know or care about.

Parameters context NetworkContext instance describing the new state of the network, as well as the original state prior to the update_network call.

update_network_precommit(*context*)

Update resources of a network.

Update values of a network, updating the associated resources in the database. Called inside transaction context on session. Raising an exception will result in rollback of the transaction. `update_network_precommit` is called for all changes to the network state. It is up to the mechanism driver to ignore state or state changes that it does not know or care about.

Parameters context NetworkContext instance describing the new state of the network, as well as the original state prior to the `update_network` call.

update_port_postcommit(*context*)

Update a port.

Called after the transaction completes. Call can block, though will block the entire process so care should be taken to not drastically affect performance. Raising an exception will result in the deletion of the resource. `update_port_postcommit` is called for all changes to the port state. It is up to the mechanism driver to ignore state or state changes that it does not know or care about.

Parameters context PortContext instance describing the new state of the port, as well as the original state prior to the `update_port` call.

update_port_precommit(*context*)

Update resources of a port.

Called inside transaction context on session to complete a port update as defined by this mechanism driver. Raising an exception will result in rollback of the transaction. `update_port_precommit` is called for all changes to the port state. It is up to the mechanism driver to ignore state or state changes that it does not know or care about.

Parameters context PortContext instance describing the new state of the port, as well as the original state prior to the `update_port` call.

update_subnet_postcommit(*context*)

Update a subnet.

Called after the transaction commits. Call can block, though will block the entire process so care should be taken to not drastically affect performance. Raising an exception will cause the deletion of the resource. `update_subnet_postcommit` is called for all changes to the subnet state. It is up to the mechanism driver to ignore state or state changes that it does not know or care about.

Parameters context SubnetContext instance describing the new state of the subnet, as well as the original state prior to the `update_subnet` call.

update_subnet_precommit(*context*)

Update resources of a subnet.

Update values of a subnet, updating the associated resources in the database. Called inside transaction context on session. Raising an exception will result in rollback of the transaction. `update_subnet_precommit` is called for all changes to the subnet state. It is up to the mechanism driver to ignore state or state changes that it does not know or care about.

Parameters context SubnetContext instance describing the new state of the subnet, as well as the original state prior to the `update_subnet` call.

Module contents

Module contents

Submodules

networking_baremetal.common module

networking_baremetal.common.config_to_xml(*config*)

networking_baremetal.common.driver_mgr(*device_id*)

networking_baremetal.common.txt_subelement(*parent*, *tag*, *text*, **args*, ***kwargs*)

networking_baremetal.config module

networking_baremetal.config.get_devices()

Get enabled network devices from configuration

This is called during driver initialization, during initialization additional driver specific configuration is loaded and the drivers validation method is called.

networking_baremetal.config.list_common_device_driver_opts()

networking_baremetal.config.list_opts()

networking_baremetal.constants module

class networking_baremetal.constants.NetconfEditConfigOperation(*value*)

Bases: enum.Enum

RFC 6241 - <edit-config> operation attribute

The operation attribute has one of the following values:

merge: The configuration data identified by the element containing this attribute is merged with the configuration at the corresponding level in the configuration datastore identified by the <target> parameter. This is the default behavior.

replace: The configuration data identified by the element containing this attribute replaces any related configuration in the configuration datastore identified by the <target> parameter. If no such configuration data exists in the configuration datastore, it is created. Unlike a <copy-config> operation, which replaces the entire target configuration, only the configuration actually present in the <config> parameter is affected.

create: The configuration data identified by the element containing this attribute is added to the configuration if and only if the configuration data does not already exist in the configuration datastore. If the configuration data exists, an <rpc-error> element is returned with an <error-tag> value of data-exists.

delete: The configuration data identified by the element containing this attribute is deleted from the configuration if and only if the configuration data currently exists in

the configuration datastore. If the configuration data does not exist, an <rpc-error> element is returned with an <error-tag> value of data-missing.

remove: The configuration data identified by the element containing this attribute is deleted from the configuration if the configuration data currently exists in the configuration datastore. If the configuration data does not exist, the remove operation is silently ignored by the server.

CREATE = 'create'

DELETE = 'delete'

MERGE = 'merge'

REMOVE = 'remove'

REPLACE = 'replace'

networking_baremetal.exceptions module

exception networking_baremetal.exceptions.DeviceConnectionError(**kwargs)

Bases: neutron_lib.exceptions.NeutronException

message = 'Driver failed connecting to device %(device)s: %(err)s'

exception networking_baremetal.exceptions.DriverEntrypointLoadError(**kwargs)

Bases: neutron_lib.exceptions.NeutronException

message = 'Failed to load entrypoint %(entry_point)s: %(err)s'

exception networking_baremetal.exceptions.DriverValidationError(**kwargs)

Bases: neutron_lib.exceptions.NeutronException

message = 'Failed driver validation for device %(device)s: %(err)s'

exception networking_baremetal.exceptions.PreConfiguredAggrergateNotFound(**kwargs)

Bases: neutron_lib.exceptions.NeutronException

message = 'Driver could not find the aggregate ID for the pre-configured link aggregate for links %(links)s on device %(device)s.'

networking_baremetal.ironic_client module

networking_baremetal.ironic_client.get_client()

Get an ironic client connection.

networking_baremetal.ironic_client.get_session(group)

networking_baremetal.ironic_client.list_opts()

Module contents

RELEASE NOTES

Release Notes

INDICES AND TABLES

- genindex
- search

PYTHON MODULE INDEX

a

`networking_baremetal.agent`, 44

`networking_baremetal.agent.ironic_neutron_agent`,
43

c

`networking_baremetal.common`, 63

`networking_baremetal.config`, 63

`networking_baremetal.constants`, 63

d

`networking_baremetal.drivers`, 46

`networking_baremetal.drivers.base`, 44

e

`networking_baremetal.exceptions`, 64

i

`networking_baremetal.ironic_client`, 64

n

`networking_baremetal`, 65

o

`networking_baremetal.openconfig`, 58

`networking_baremetal.openconfig.interfaces`,
51

`networking_baremetal.openconfig.interfaces.aggregate`,
46

`networking_baremetal.openconfig.interfaces.ethernet`,
47

`networking_baremetal.openconfig.interfaces.interfaces`,
48

`networking_baremetal.openconfig.interfaces.types`,
51

`networking_baremetal.openconfig.lacp`,
53

`networking_baremetal.openconfig.lacp.lacp`,
51

`networking_baremetal.openconfig.lacp.types`,
53

`networking_baremetal.openconfig.network_instance`,
54

`networking_baremetal.openconfig.network_instance.ne`
53

`networking_baremetal.openconfig.vlan`,
58

`networking_baremetal.openconfig.vlan.types`,
54

`networking_baremetal.openconfig.vlan.vlan`,
55

p

`networking_baremetal.plugins`, 63

`networking_baremetal.plugins.ml2`, 63

`networking_baremetal.plugins.ml2.baremetal_mech`,
59

INDEX

- A**
- ACCESS** (network-*ing_baremetal.drivers.base*), 44
 - ing_baremetal.openconfig.vlan.types.VlanInterfaceMode* (class in *network-ing_baremetal.drivers.base*), 44
 - ing_baremetal.openconfig.vlan.types.VlanInterfaceMode* (attribute), 55
 - access_vlan** (network-*ing_baremetal.openconfig.interfaces.interfaces*), 48
 - ing_baremetal.openconfig.vlan.vlan.VlanSwitchedConfig* (property), 57
 - ACTIVE** (network-*ing_baremetal.openconfig.lacp.types.LACPActivity*) (network-*ing_baremetal.openconfig.interfaces.aggregate.Interfaces*), 46
 - ing_baremetal.openconfig.lacp.types.LACPActivity* (attribute), 53
 - ACTIVE** (network-*ing_baremetal.openconfig.vlan.types.VlanStatus*) (network-*ing_baremetal.openconfig.interfaces.ethernet.InterfacesEthernet*), 47
 - ing_baremetal.openconfig.vlan.types.VlanStatus* (attribute), 55
 - add()** (*networking_baremetal.openconfig.interfaces.interfaces.Interfaces*) (network-*ing_baremetal.openconfig.interfaces.interfaces.BaseInterface*), 48
 - method), 50
 - add()** (*networking_baremetal.openconfig.lacp.lacp.LACPInterfaces*) (network-*ing_baremetal.openconfig.interfaces.interfaces.BaseInterface*), 48
 - method), 52
 - add()** (*networking_baremetal.openconfig.network_instance.network_instances*) (network-*ing_baremetal.openconfig.lacp.lacp.LACPInterface*), 51
 - method), 54
 - add()** (*networking_baremetal.openconfig.vlan.vlan.TrunkVlans*) (network-*ing_baremetal.openconfig.vlan.vlan.Vlan*), 55
 - method), 55
 - add()** (*networking_baremetal.openconfig.vlan.vlan.Vlans*) (network-*ing_baremetal.openconfig.vlan.vlan.Vlan*), 55
 - method), 58
 - aggregate_id** (network-*ing_baremetal.openconfig.interfaces.ethernet.InterfacesEthernetConfig*) (network-*ing_baremetal.openconfig.vlan.vlan.VlanSwitchedVlan*), 47
 - ing_baremetal.openconfig.interfaces.ethernet.InterfacesEthernetConfig* (property), 47
 - aggregation** (network-*ing_baremetal.openconfig.interfaces.interfaces.InterfaceAggregateConnectivity*) (network-*ing_baremetal.common*), 63
 - ing_baremetal.openconfig.interfaces.interfaces.InterfaceAggregateConnectivity* (property), 48
 - AggregationType** (class in *network-ing_baremetal.openconfig.interfaces.types*), 51
 - ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver* (property), 59
 - CREATE** (network-*ing_baremetal.constants.NetconfEditConfigOperation*) (network-*ing_baremetal.constants.NetconfEditConfigOperation*), 64
 - ing_baremetal.constants.NetconfEditConfigOperation* (attribute), 64
- B**
- BaremetalMechanismDriver** (class in *network-ing_baremetal.plugins.ml2.baremetal_mech*), 59
 - ing_baremetal.drivers.base.BaseDeviceDriver* (method), 44
 - BaremetalNeutronAgent** (class in *network-ing_baremetal.agent.ironic_neutron_agent*), 43
 - ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver* (method), 59
 - BaseDeviceClient** (class in *network-*ing_baremetal.drivers.base**) (network-*ing_baremetal.drivers.base*), 44
 - ing_baremetal.drivers.base.BaseDeviceDriver* (method), 44
 - create_network()** (network-*ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver*), 59
 - method), 59
 - create_network_postcommit()** (network-*ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver*), 59
 - method), 59
 - create_network_precommit()** (network-*ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver*), 59
 - method), 59

`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`), 63
`method`), 59 `DriverEntryLoadError`, 64
`create_port()` (`network- DriverValidationError`, 64
`ing_baremetal.drivers.base.BaseDeviceDriver`
`method`), 45 **E**
`create_port_postcommit()` (`network- edit_config()` (`network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 59 `method`), 44
`create_port_precommit()` (`network- enabled` (`network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 59 `openconfig.interfaces.interfaces.InterfaceC`
`property`), 49
`create_subnet_postcommit()` (`network- ethernet` (`network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 59 `openconfig.interfaces.interfaces.InterfaceE`
`property`), 50
`create_subnet_precommit()` (`network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 60 **F**
D `FAST` (`networking_baremetal.openconfig.lacp.types.LACPPeriod`
`attribute`), 53
`filter_rule` (`network-`
DELETE (`network- ing_baremetal.agent.ironic_neutron_agent.HashRingMem`
`ing_baremetal.constants.NetconfEditConfigOperation`
`attribute`), 64 `attribute`), 44
delete_network() (`network-`
`ing_baremetal.drivers.base.BaseDeviceDriver`
`method`), 45 **G**
`delete_network_postcommit()` (`network- get()` (`networking_baremetal.drivers.base.BaseDeviceClient`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 60 `method`), 44
`delete_network_precommit()` (`network- get_allowed_network_types()` (`network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 60 `ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMe`
`method`), 61
`delete_port()` (`network- get_client()` (`in module network-`
`ing_baremetal.drivers.base.BaseDeviceDriver`
`method`), 60 `ing_baremetal.agent.ironic_client`), 64
`delete_port_postcommit()` (`network- get_client_args()` (`network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 60 `ing_baremetal.drivers.base.BaseDeviceClient`
`method`), 44
`delete_port_precommit()` (`network- get_devices()` (`in module network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 61 `ing_baremetal.config`), 63
`delete_subnet_postcommit()` (`network- get_machines()` (`network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 60 `ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMe`
`method`), 61
`delete_subnet_precommit()` (`network- get_session()` (`in module network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 60 `ing_baremetal.agent.ironic_client`), 64
`delete_subnet_precommit()` (`network- get_template_node_state()` (`network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 60 `ing_baremetal.agent.ironic_neutron_agent.BaremetalNeu`
`method`), 43 `method`), 43
delete_subnet_precommit() (`network-`
`ing_baremetal.plugins.ml2.baremetal_mech.BaremetalMechanismDriver`
`method`), 61 **H**
description (`network- ing_baremetal.agent.ironic_neutron_agent.HashRingMem`
`ing_baremetal.openconfig.interfaces.interfaces.InterfaceConfig`
`property`), 49 `attribute`), 44
`DeviceConnectionError`, 64 `HashRingMemberManagerNotificationEndpoint`
`driver_mgr()` (`in module network- ing_baremetal.agent.ironic_neutron_agent`,
44

I
info() (*network-ing_baremetal.agent.ironic_neutron_agent.HashRingMemberManagerNotificationEndpoint* method), 44
interface_mode (*network-ing_baremetal.openconfig.vlan.vlan.VlanSwitchedConfig* property), 57
InterfaceAggregate (class in *network-ing_baremetal.openconfig.interfaces.interfaces*), 48
InterfaceConfig (class in *network-ing_baremetal.openconfig.interfaces.interfaces*), 49
InterfaceEthernet (class in *network-ing_baremetal.openconfig.interfaces.interfaces*), 50
Interfaces (class in *network-ing_baremetal.openconfig.interfaces.interfaces*), 50
interfaces (*network-ing_baremetal.openconfig.interfaces.aggregate.Interfaces* property), 50
interfaces (*network-ing_baremetal.openconfig.lacp.lacp.LACP* property), 51
interfaces (*network-ing_baremetal.openconfig.lacp.lacp.LACPInterfaces* property), 52
InterfacesAggregation (class in *network-ing_baremetal.openconfig.interfaces.aggregate*), 46
InterfacesAggregationConfig (class in *network-ing_baremetal.openconfig.interfaces.aggregate*), 46
InterfacesEthernet (class in *network-ing_baremetal.openconfig.interfaces.ethernet*), 47
InterfacesEthernetConfig (class in *network-ing_baremetal.openconfig.interfaces.ethernet*), 47
interval (*network-ing_baremetal.openconfig.lacp.lacp.LACPInterfaceConfig* property), 52

L
LACP (class in *network-ing_baremetal.openconfig.lacp.lacp*), 51
LACP (*networking_baremetal.openconfig.interfaces.types.AggregationType* attribute), 51
lacp_mode (*network-ing_baremetal.openconfig.lacp.lacp.LACPInterfaceConfig* attribute), 52
LACPActivity (class in *network-ing_baremetal.openconfig.lacp.types*), 53
LACPInterface (class in *network-ing_baremetal.openconfig.lacp.lacp*), 51
LACPInterfaceConfig (class in *network-ing_baremetal.openconfig.lacp.lacp*), 52
LACPInterfaces (class in *network-ing_baremetal.openconfig.lacp.lacp*), 52
LACPPeriod (class in *network-ing_baremetal.openconfig.lacp.types*), 53
lag_type (*network-ing_baremetal.openconfig.interfaces.aggregate.Interfaces* property), 47
list_common_device_driver_opts() (in module *networking_baremetal.config*), 63
list_opts() (in module *network-ing_baremetal.agent.ironic_neutron_agent*), 44
list_opts() (in module *network-ing_baremetal.config*), 63
list_opts() (in module *network-ing_baremetal.ironic_client*), 64
load_config() (*network-ing_baremetal.drivers.base.BaseDeviceDriver* method), 45

M
main() (in module *network-ing_baremetal.agent.ironic_neutron_agent*), 44
members (*network-ing_baremetal.agent.ironic_neutron_agent.HashRingMember* attribute), 44
MERGE (*networking_baremetal.constants.NetconfEditConfigOperation* attribute), 64
message (*network-ing_baremetal.exceptions.DeviceConnectionError* attribute), 64
message (*network-ing_baremetal.exceptions.DriverEntrypointLoadError* attribute), 64
message (*network-ing_baremetal.exceptions.DriverEntrypointLoadError* attribute), 64

ing_baremetal.exceptions.DriverValidationError (networking_baremetal.plugins.ml2.baremetal_mechanism, attribute), 64
 message (network-
ing_baremetal.exceptions.PreConfiguredAggregateNetwork (networking_baremetal.openconfig.interfaces.interfaces.InterfaceC), 49
 attribute), 64
 min_links (network-
ing_baremetal.openconfig.interfaces.aggregate.NetworkAggregateConfig (networking_baremetal.openconfig.interfaces.interfaces.BaseInterf), 47
 property), 47
 module
 networking_baremetal, 65
 networking_baremetal.agent, 44
 networking_baremetal.agent.ironic_neutron_agent, 43
 networking_baremetal.common, 63
 networking_baremetal.config, 63
 networking_baremetal.constants, 63
 networking_baremetal.drivers, 46
 networking_baremetal.drivers.base, 44
 networking_baremetal.exceptions, 64
 networking_baremetal.ironic_client, 64
 networking_baremetal.openconfig, 58
 networking_baremetal.openconfig.interfaces, 51
 networking_baremetal.openconfig.interfaces.aggregate, 46
 networking_baremetal.openconfig.interfaces.ethernet, 47
 networking_baremetal.openconfig.interfaces.interfaces, 48
 networking_baremetal.openconfig.interfaces.types, 51
 networking_baremetal.openconfig.lacp, 53
 networking_baremetal.openconfig.lacp.lacp, 51
 networking_baremetal.openconfig.lacp.types, 53
 networking_baremetal.openconfig.network_instance, 54
 networking_baremetal.openconfig.network_instance.network_instance, 53
 networking_baremetal.openconfig.vlan, 58
 networking_baremetal.openconfig.vlan.types, 54
 networking_baremetal.openconfig.vlan.vlan, 55
 networking_baremetal.plugins, 63
 networking_baremetal.plugins.ml2, 63

N

name (networking_baremetal.openconfig.interfaces.interfaces.BaseInterf), 48
 name (networking_baremetal.openconfig.interfaces.interfaces.InterfaceC), 49
 name (networking_baremetal.openconfig.lacp.lacp.LACPInterface), 51
 name (networking_baremetal.openconfig.lacp.lacp.LACPInterfaceConfig), 52
 name (networking_baremetal.openconfig.network_instance.network_instance), 54
 name (networking_baremetal.openconfig.vlan.vlan.VlanConfig), 56
 NAMESPACE (network-
ing_baremetal.openconfig.interfaces.aggregate.Interfaces attribute), 46
 NAMESPACE (network-
ing_baremetal.openconfig.interfaces.aggregate.Interfaces attribute), 46
 NAMESPACE (network-
ing_baremetal.openconfig.interfaces.ethernet.Interfaces attribute), 47
 NAMESPACE (network-
ing_baremetal.openconfig.interfaces.ethernet.Interfaces attribute), 47
 NAMESPACE (network-
ing_baremetal.openconfig.interfaces.interfaces.Interfaces attribute), 48
 NAMESPACE (network-
ing_baremetal.openconfig.interfaces.interfaces.InterfaceC attribute), 49
 NAMESPACE (network-
ing_baremetal.openconfig.interfaces.interfaces.Interfaces attribute), 50
 NAMESPACE (network-
ing_baremetal.openconfig.lacp.lacp.LACP attribute), 51
 NAMESPACE (network-
ing_baremetal.openconfig.lacp.lacp.LACPInterfaceConfig attribute), 52
 NAMESPACE (network-
ing_baremetal.openconfig.lacp.lacp.LACPInterfaces attribute), 52

| | | |
|---|--|--|
| NAMESPACE | (network- <i>ing_baremetal.openconfig.network_instance.network_instance</i> <i>attribute</i>), 53 | networking_baremetal.openconfig.interfaces module, 46 |
| NAMESPACE | (network- <i>ing_baremetal.openconfig.network_instance.network_instance</i> <i>attribute</i>), 54 | networking_baremetal.openconfig.interfaces.aggregate module, 47 |
| NAMESPACE | (network- <i>ing_baremetal.openconfig.vlan.vlan.Vlan</i> <i>attribute</i>), 55 | networking_baremetal.openconfig.interfaces.ethernet module, 48 |
| NAMESPACE | (network- <i>ing_baremetal.openconfig.vlan.vlan.Vlan</i> <i>attribute</i>), 56 | networking_baremetal.openconfig.interfaces.interface module, 48 |
| NAMESPACE | (network- <i>ing_baremetal.openconfig.vlan.vlan.Vlan</i> <i>attribute</i>), 56 | networking_baremetal.openconfig.interfaces.types module, 51 |
| NAMESPACE | (network- <i>ing_baremetal.openconfig.vlan.vlan.Vlans</i> <i>attribute</i>), 58 | networking_baremetal.openconfig.lacp module, 53 |
| NAMESPACE | (network- <i>ing_baremetal.openconfig.vlan.vlan.Vlans</i> <i>attribute</i>), 58 | networking_baremetal.openconfig.lacp.lacp module, 51 |
| NAMESPACE | (network- <i>ing_baremetal.openconfig.vlan.vlan.VlanSwitch</i> <i>attribute</i>), 57 | networking_baremetal.openconfig.lacp.types module, 53 |
| NAMESPACE | (network- <i>ing_baremetal.openconfig.vlan.vlan.VlanSwitch</i> <i>attribute</i>), 57 | networking_baremetal.openconfig.network_instance module, 54 |
| native_vlan | (network- <i>ing_baremetal.openconfig.vlan.vlan.VlanSwitch</i> <i>property</i>), 57 | networking_baremetal.openconfig.network_instance.network_instance module, 53 |
| NetconfEditConfigOperation (class in net- working_baremetal.constants), 63 | | networking_baremetal.openconfig.vlan module, 58 |
| network_instances | (network- <i>ing_baremetal.openconfig.network_instance.network_instance</i> <i>property</i>), 54 | networking_baremetal.openconfig.vlan.types module, 54 |
| networking_baremetal | | networking_baremetal.openconfig.vlan.vlan module, 55 |
| networking_baremetal.agent | | networking_baremetal.plugins module, 63 |
| networking_baremetal.agent.ironic_neutron_agent | | networking_baremetal.plugins.ml2 module, 63 |
| networking_baremetal.common | | networking_baremetal.plugins.ml2.baremetal_mech module, 59 |
| networking_baremetal.config | | NetworkInstance (class in network- <i>ing_baremetal.openconfig.network_instance.network_instance</i> 53 |
| networking_baremetal.constants | | NetworkInstances (class in network- <i>ing_baremetal.openconfig.network_instance.network_instance</i> 54 |
| networking_baremetal.drivers | | O |
| networking_baremetal.drivers.base | | operation (network- <i>ing_baremetal.openconfig.interfaces.aggregate.Interfaces</i> <i>property</i>), 47 |
| networking_baremetal.exceptions | | operation (network- <i>ing_baremetal.openconfig.interfaces.ethernet.InterfacesEt</i> <i>property</i>), 47 |
| networking_baremetal.ironic_client | | operation (network- <i>ing_baremetal.openconfig.interfaces.interfaces.InterfaceA</i> <i>property</i>), 48 |
| networking_baremetal.openconfig | | operation (network- <i>ing_baremetal.openconfig.interfaces.interfaces.InterfaceC</i> |

| | | | |
|-----------|---|----------------------------------|--|
| | <i>property</i>), 49 | PARENT | (<i>network- ing_baremetal.openconfig.vlan.vlan.VlanSwitchedConfig</i> |
| operation | (<i>network- ing_baremetal.openconfig.lacp.lacp.LACPInterface</i> | | attribute), 57 |
| | <i>property</i>), 51 | PARENT | (<i>network- ing_baremetal.openconfig.vlan.vlan.VlanSwitchedVlan</i> |
| operation | (<i>network- ing_baremetal.openconfig.lacp.lacp.LACPInterface</i> | | attribute), 57 |
| | <i>property</i>), 52 | PASSIVE | (<i>network- ing_baremetal.openconfig.lacp.types.LACPActivity</i> |
| operation | (<i>network- ing_baremetal.openconfig.vlan.vlan.Vlan</i> | | attribute), 53 |
| | <i>property</i>), 56 | pattern | (<i>network- ing_baremetal.openconfig.vlan.types.VlanRange</i> |
| operation | (<i>network- ing_baremetal.openconfig.vlan.vlan.VlanConfig</i> | | attribute), 55 |
| | <i>property</i>), 56 | PreConfiguredAggrergateNotFound, | 64 |
| operation | (<i>network- ing_baremetal.openconfig.vlan.vlan.VlanSwitchedConfig</i> | | attribute), 57 |
| | <i>property</i>), 57 | R | |
| | | REMOVE | (<i>network- ing_baremetal.constants.NetconfEditConfigOperation</i> |
| P | | | attribute), 64 |
| PARENT | (<i>network- ing_baremetal.openconfig.interfaces.aggregate.InterfacesAggregate</i> | remove() | (<i>network- ing_baremetal.openconfig.vlan.vlan.Vlans</i> |
| | attribute), 46 | | method), 58 |
| PARENT | (<i>network- ing_baremetal.openconfig.interfaces.aggregate.InterfacesAggregate</i> | REPLACE | (<i>network- ing_baremetal.constants.NetconfEditConfigOperation</i> |
| | attribute), 46 | | attribute), 64 |
| PARENT | (<i>network- ing_baremetal.openconfig.interfaces.ethernet.InterfacesEthernet</i> | reset() | (<i>network- ing_baremetal.agent.ironic_neutron_agent.BaremetalNeu</i> |
| | attribute), 47 | | method), 43 |
| PARENT | (<i>network- ing_baremetal.openconfig.interfaces.ethernet.InterfacesEthernetConfig</i> | S | |
| | attribute), 47 | SLOW | (<i>networking_baremetal.openconfig.lacp.types.LACPPeriod</i> |
| PARENT | (<i>network- ing_baremetal.openconfig.interfaces.interfaces.BaseInterface</i> | | attribute), 53 |
| | attribute), 48 | start() | (<i>network- ing_baremetal.agent.ironic_neutron_agent.BaremetalNeu</i> |
| PARENT | (<i>network- ing_baremetal.openconfig.interfaces.interfaces.InterfaceConfig</i> | | method), 43 |
| | attribute), 49 | STATIC | (<i>network- ing_baremetal.openconfig.interfaces.types.AggregationTyp</i> |
| PARENT | (<i>network- ing_baremetal.openconfig.lacp.lacp.LACPInterface</i> | | attribute), 51 |
| | attribute), 51 | status | (<i>network- ing_baremetal.openconfig.vlan.vlan.VlanConfig</i> |
| PARENT | (<i>network- ing_baremetal.openconfig.lacp.lacp.LACPInterface</i> | | property), 56 |
| | attribute), 52 | stop() | (<i>network- ing_baremetal.agent.ironic_neutron_agent.BaremetalNeu</i> |
| PARENT | (<i>network- ing_baremetal.openconfig.lacp.lacp.LACPInterfaces</i> | | method), 44 |
| | attribute), 52 | SUPPORTED_BOND_MODES | (<i>network- ing_baremetal.drivers.base.BaseDeviceDriver</i> |
| PARENT | (<i>network- ing_baremetal.openconfig.vlan.vlan.Vlan</i> | | attribute), 44 |
| | attribute), 55 | SUSPENDED | (<i>network- ing_baremetal.openconfig.vlan.types.VlanStatus</i> |
| PARENT | (<i>network- ing_baremetal.openconfig.vlan.vlan.VlanConfig</i> | | attribute), 55 |
| | attribute), 56 | switched_vlan | (<i>network- ing_baremetal.openconfig.interfaces.aggregate.Interfaces</i> |
| | | | property), 46 |

| | | | |
|------------------|--|--|---|
| switched_vlan | (network- ing_baremetal.openconfig.interfaces.ethernet.InterfacesE property), 47 | ing_baremetal.openconfig.interfaces.ethernet.InterfacesE to_xml_element() | ing_baremetal.openconfig.interfaces.ethernet.InterfacesE (network- ing_baremetal.openconfig.interfaces.interfaces.BaseInterf method), 48 |
| T | | | |
| TAG | (networking_baremetal.openconfig.interfaces.aggregate.Interface attribute), 46 | to_xml_element() | (network- ing_baremetal.openconfig.interfaces.interfaces.InterfaceA method), 49 |
| TAG | (networking_baremetal.openconfig.interfaces.aggregate.Interface attribute), 47 | to_xml_element() | (network- ing_baremetal.openconfig.interfaces.interfaces.InterfaceA method), 50 |
| TAG | (networking_baremetal.openconfig.interfaces.ethernet.Interface attribute), 47 | to_xml_element() | (network- ing_baremetal.openconfig.interfaces.interfaces.InterfaceC method), 50 |
| TAG | (networking_baremetal.openconfig.interfaces.ethernet.Interface attribute), 47 | to_xml_element() | (network- ing_baremetal.openconfig.interfaces.interfaces.InterfaceE method), 50 |
| TAG | (networking_baremetal.openconfig.interfaces.interfaces.BaseInter attribute), 48 | to_xml_element() | (network- ing_baremetal.openconfig.interfaces.interfaces.InterfaceE method), 50 |
| TAG | (networking_baremetal.openconfig.interfaces.interfaces.Interface attribute), 49 | to_xml_element() | (network- ing_baremetal.openconfig.interfaces.interfaces.InterfaceE method), 50 |
| TAG | (networking_baremetal.openconfig.interfaces.interfaces.Interface attribute), 50 | to_xml_element() | (network- ing_baremetal.openconfig.lacp.lacp.LACP method), 51 |
| TAG | (networking_baremetal.openconfig.lacp.lacp.LACP attribute), 51 | to_xml_element() | (network- ing_baremetal.openconfig.lacp.lacp.LACPInterface method), 52 |
| TAG | (networking_baremetal.openconfig.lacp.lacp.LACPInterface attribute), 51 | to_xml_element() | (network- ing_baremetal.openconfig.lacp.lacp.LACPInterface method), 52 |
| TAG | (networking_baremetal.openconfig.lacp.lacp.LACPInterface attribute), 52 | to_xml_element() | (network- ing_baremetal.openconfig.lacp.lacp.LACPInterfaceConfig method), 52 |
| TAG | (networking_baremetal.openconfig.lacp.lacp.LACPInterface attribute), 52 | to_xml_element() | (network- ing_baremetal.openconfig.lacp.lacp.LACPInterfaceConfig method), 52 |
| TAG | (networking_baremetal.openconfig.network_instance.network_inst attribute), 54 | to_xml_element() | (network- ing_baremetal.openconfig.lacp.lacp.LACPInterfaces method), 53 |
| TAG | (networking_baremetal.openconfig.network_instance.network_inst attribute), 54 | to_xml_element() | (network- ing_baremetal.openconfig.network_instance.network_inst method), 54 |
| TAG | (networking_baremetal.openconfig.vlan.vlan.Vlan attribute), 55 | to_xml_element() | (network- ing_baremetal.openconfig.network_instance.network_inst method), 54 |
| TAG | (networking_baremetal.openconfig.vlan.vlan.VlanConfig attribute), 56 | to_xml_element() | (network- ing_baremetal.openconfig.vlan.vlan.Vlan method), 56 |
| TAG | (networking_baremetal.openconfig.vlan.vlan.VlanSwitched attribute), 57 | to_xml_element() | (network- ing_baremetal.openconfig.vlan.vlan.Vlan method), 56 |
| TAG | (networking_baremetal.openconfig.vlan.vlan.VlanSwitched attribute), 58 | to_xml_element() | (network- ing_baremetal.openconfig.vlan.vlan.VlanConfig method), 56 |
| to_xml_element() | (network- ing_baremetal.openconfig.interfaces.aggregate.Interface method), 46 | to_xml_element() | (network- ing_baremetal.openconfig.vlan.vlan.Vlans method), 58 |
| to_xml_element() | (network- ing_baremetal.openconfig.interfaces.aggregate.Interface method), 47 | to_xml_element() | (network- ing_baremetal.openconfig.vlan.vlan.VlanSwitchedConfig method), 57 |
| to_xml_element() | (network- ing_baremetal.openconfig.interfaces.ethernet.Interface method), 47 | to_xml_element() | (network- ing_baremetal.openconfig.vlan.vlan.VlanSwitchedVlan method), 58 |
| to_xml_element() | (network- ing_baremetal.openconfig.vlan.vlan.VlanSwitchedVlan method), 47 | to_xml_element() | (network- ing_baremetal.openconfig.vlan.types.VlanInterface method), 58 |

- attribute), 55
- trunk_vlans (network-
ing_baremetal.openconfig.vlan.vlan.VlanSwitchedConfig
property), 57
- TrunkVlans (class in network-
ing_baremetal.openconfig.vlan.vlan),
55
- try_to_bind_segment_for_agent()
(network-
ing_baremetal.plugins.ml2.baremetal_mech.
method), 61
- txt_subelement() (in module network-
ing_baremetal.common), 63
- ## U
- update_network()
(network-
ing_baremetal.drivers.base.BaseDeviceDriver
method), 45
- update_network_postcommit()
(network-
ing_baremetal.plugins.ml2.baremetal_mech.
method), 61
- update_network_precommit()
(network-
ing_baremetal.plugins.ml2.baremetal_mech.
method), 61
- update_port()
(network-
ing_baremetal.drivers.base.BaseDeviceDriver
method), 45
- update_port_postcommit()
(network-
ing_baremetal.plugins.ml2.baremetal_mech.
method), 62
- update_port_precommit()
(network-
ing_baremetal.plugins.ml2.baremetal_mech.
method), 62
- update_subnet_postcommit()
(network-
ing_baremetal.plugins.ml2.baremetal_mech.
method), 62
- update_subnet_precommit()
(network-
ing_baremetal.plugins.ml2.baremetal_mech.
method), 62
- ## V
- validate()
(network-
ing_baremetal.drivers.base.BaseDeviceDriver
method), 45
- Vlan (class in network-
ing_baremetal.openconfig.vlan.vlan),
55
- vlan_id (network-
ing_baremetal.openconfig.vlan.types.VlanId
property), 54
- vlan_id (network-
ing_baremetal.openconfig.vlan.vlan.Vlan
- property), 56
- VlanSwitchedConfig (network-
ing_baremetal.openconfig.vlan.vlan.VlanConfig
property), 56
- vlan_range (network-
ing_baremetal.openconfig.vlan.types.VlanRange
property), 55
- VlanConfig (class in network-
ing_baremetal.openconfig.vlan.vlan),
54
- VlanId (class in network-
ing_baremetal.openconfig.vlan.types),
54
- VlanInterfaceMode (class in network-
ing_baremetal.openconfig.vlan.types),
54
- VlanRange (class in network-
ing_baremetal.openconfig.vlan.types),
55
- VlanStatus (class in network-
ing_baremetal.openconfig.vlan.types),
58
- VlanSwitchedConfig (class in network-
ing_baremetal.openconfig.vlan.vlan),
58
- VlanSwitchedVlan (class in network-
ing_baremetal.openconfig.vlan.vlan),
58
- ## W
- W (network-
ing_baremetal.agent.ironic_neutron_agent.
method), 44