Shaker Documentation

Release 1.4.2

OpenStack Foundation

Jan 26, 2021

Contents

The distributed data-plane testing tool built for OpenStack.

Shaker wraps around popular system network testing tools like iperf, iperf3 and netperf (with help of flent). Shaker is able to deploy OpenStack instances and networks in different topologies. Shaker scenario specifies the deployment and list of tests to execute. Additionally tests may be tuned dynamically in command-line.

CHAPTER 1

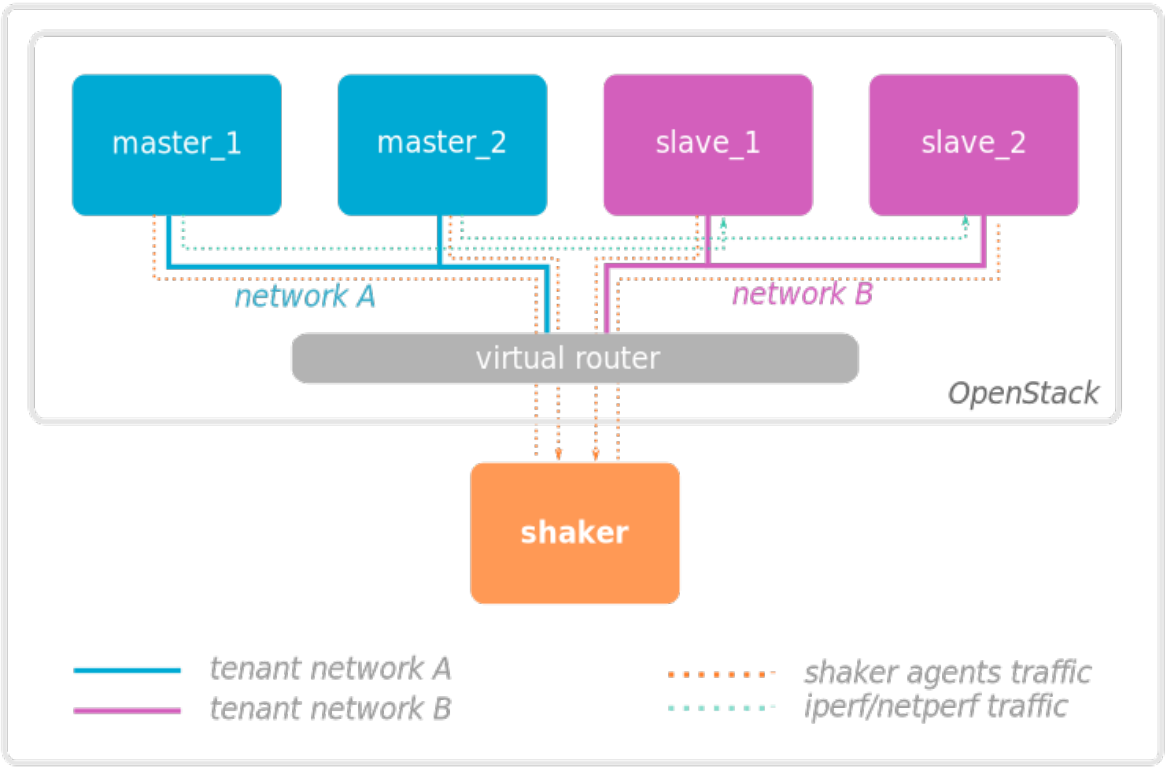
Installation

1.1 Installation in Python environment

Shaker is distributed as Python package and available through PyPi (https://pypi.org/project/pyshaker/).

```
$ pip install --user pyshaker
```

1.2 OpenStack Deployment



Requirements:

- Computer where Shaker is executed should be routable from OpenStack instances and should have open port to accept connections from agents running on instances

For full features support it is advised to run Shaker by admin user. However with some limitations it works for non-admin user - see *Running Shaker by non-admin user* for details.

1.3 Base image

1.3.1 Automatic build in OpenStack

The base image can be built using *shaker-image-builder* tool.

```
$ shaker-image-builder
```

There are 2 modes available:

- *heat* - using Heat template (requires Glance v1 for base image upload);
- *dib* - using diskimage-builder elements (requires qemu-utils and debootstrap to build Ubuntu-based image).

By default the mode is selected automatically preferring *heat* if Glance API v1 is available. Created image is uploaded into Glance and made available for further executions of Shaker. For full list of parameters refer to *shaker-image-builder*.

1.3.2 Manual build with disk-image-builder

Shaker image can also be built using diskimage-builder tool.

1. Install disk-image-builder. Refer to diskimage-builder installation
2. Clone Shaker repo: `git clone https://opendev.org/performa/shaker`
3. Add search path for diskimage-builder elements: `export ELEMENTS_PATH=shaker/shaker/resources/image_elements`
4. Build the image based on Ubuntu Xenial: `disk-image-create -o shaker-image.qcow2 ubuntu vm shaker`
5. Upload image into Glance: `openstack image create --public --file shaker-image.qcow2 --disk-format qcow2 shaker-image`
6. Create flavor: `openstack flavor create --ram 512 --disk 3 --vcpus 1 shaker-flavor`

1.4 Running Shaker by non-admin user

While the full feature set is available when Shaker is run by admin user, it works with some limitations for non-admin user too.

1.4.1 Image builder limitations

Image builder requires flavor name to be specified via command line parameter *–flavor-name*. Create flavor prior running Shaker, or choose one that satisfies instance template requirements. For Ubuntu-based image the requirement is 512 Mb RAM, 3 Gb disk and 1 CPU

1.4.2 Execution limitations

Non-admin user has no permissions to list compute nodes and to deploy instances to particular compute nodes.

When instances need to be deployed on low number of compute nodes it is possible to use server groups and specify anti-affinity policy within them. Note however that server group size is limited by *quota_server_group_members* parameter in *nova.conf*. The following is part of Heat template adds server groups.

Add to resources section:

```
server_group:
  type: OS::Nova::ServerGroup
  properties:
    name: {{ unique }}_server_group
    policies: [ 'anti-affinity' ]
```

Add attribute to server definition:

```
scheduler_hints:
  group: { get_resource: server_group }
```

The similar patch is needed to implement dense scenarios. The difference is in server group policy, it should be *'affinity'*.

Alternative approach is to specify number of compute nodes. Note that the number must always be specified. If Nova distributes instances evenly (or with normal random distribution) then the chances that instances are placed on unique nodes are quite high (well, there will be collisions due to https://en.wikipedia.org/wiki/Birthday_problem, so expect that number of unique pair will be lower than specified number of compute nodes).

1.5 Non-OpenStack Deployment (aka Spot mode)

To run scenarios against remote nodes (`shaker-spot` command) install shaker on the local host. Make sure all necessary tools are installed too. Refer to *Spot Scenarios* for more details.

1.6 Run Shaker against OpenStack deployed by Fuel-CCP on Kubernetes

Shaker can be run in Kubernetes environment and can execute scenarios against OpenStack deployed by Fuel-CCP tool.

Shaker app consists of *service*:

```
apiVersion: v1
kind: Service
metadata:
  name: shaker
spec:
  ports:
  - nodePort: 31999
    port: 31999
    protocol: TCP
    targetPort: 31999
  selector:
    app: shaker
  type: NodePort
```

and *pod*:

```
apiVersion: v1
kind: Pod
metadata:
  name: shaker
  labels:
    app: shaker
spec:
  containers:
  - args:
    - --debug
    - --nocleanup
    env:
```

(continues on next page)

(continued from previous page)

```
    - name: OS_USERNAME
      value: admin
    - name: OS_PASSWORD
      value: password
    - name: OS_PROJECT_NAME
      value: admin
    - name: OS_AUTH_URL
      value: http://keystone.ccp:5000/
    - name: SHAKER_SCENARIO
      value: openstack/perf_l2
    - name: SHAKER_SERVER_ENDPOINT
      value: 172.20.9.7:31999
    image: performa/shaker
    imagePullPolicy: Always
    name: shaker
    securityContext:
      privileged: false
    volumeMounts:
    - mountPath: /artifacts
      name: artifacts
  dnsPolicy: ClusterFirst
  restartPolicy: Never
  volumes:
  - name: artifacts
    hostPath:
      path: /tmp
```

You may need to change values for variables defined in config files:

- *SHAKER_SERVER_ENDPOINT* should point to external address of Kubernetes cluster, and OpenStack instances must have access to it
- *OS_*** parameters describe connection to Keystone endpoint
- *SHAKER_SCENARIO* needs to be altered to run the needed scenario
- Pod is configured to write logs into /tmp on the node that hosts the pod
- *port*, *nodePort* and *targetPort* must be equal and not to conflict with other exposed services

CHAPTER 2

Usage

2.1 Configuration

For OpenStack scenarios the connection is configured using standard `openrc` file (refer to Set environment variables using the OpenStack RC file on how to retrieve it). The config can be passed to Shaker rather by sourcing into system env `source openrc` or via set of CLI parameters `--os-project-name`, `--os-username`, `--os-password`, `--os-auth-url` and `--os-region-name`. Connection to SSL endpoints is configured by parameters `--os-cacert` and `--os-insecure` (to disable certificate verification). Configuration can also be specified in config file, refer to *Shaker config parameters*. Config file name can be passed by parameter `--config-file`.

Note: Shaker is better run under user with admin privileges. However, it's possible to run under ordinary user too - refer to *Running Shaker by non-admin user*

2.2 Common Parameters

The following parameters are applicable for both OpenStack mode (*shaker*) and spot mode (*shaker-spot*).

1. Run the scenario with defaults and generate interactive report into file *report.html*:

```
shaker --scenario <scenario> --report report.html
```

2. Run the scenario and store raw result:

```
shaker --scenario <scenario> --output output.json
```

3. Run the scenario and store SLA verification results in subunit stream file:

```
shaker --scenario <scenario> --subunit report.subunit
```

4. Generate report from the raw data:

```
shaker-report --input output.json --output report.html
```

2.3 Scenario Explained

Shaker scenario is file in YAML format. It describes how agents are deployed (at OpenStack instances or statically) and sequence of tests to execute. When agents are deployed at OpenStack instances a reference to Heat template is provided.

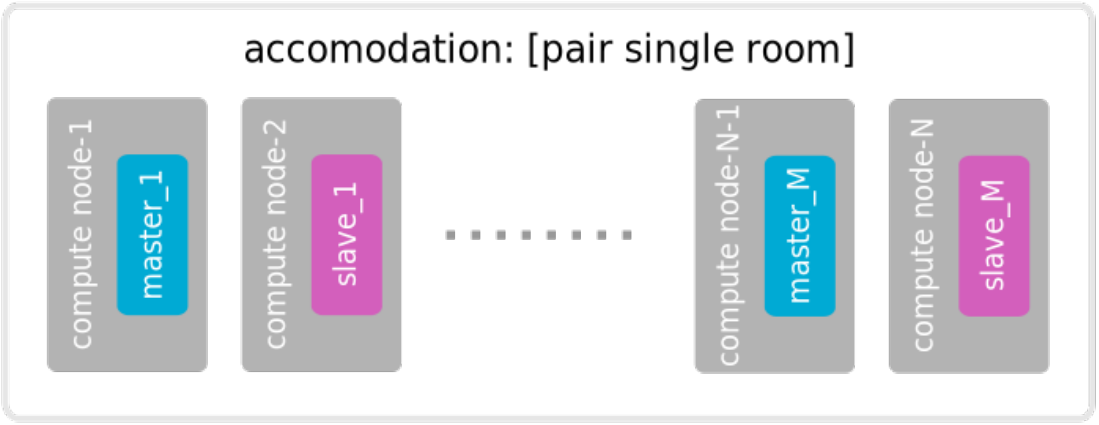
```
description:
  This scenario launches pairs of VMs in the same private network. Every VM is
  hosted on a separate compute node.

deployment:
  template: l2.hot
  accommodation: [pair, single_room]

execution:
  progression: quadratic
  tests:
  -
    title: Iperf TCP
    class: iperf_graph
    time: 60
```

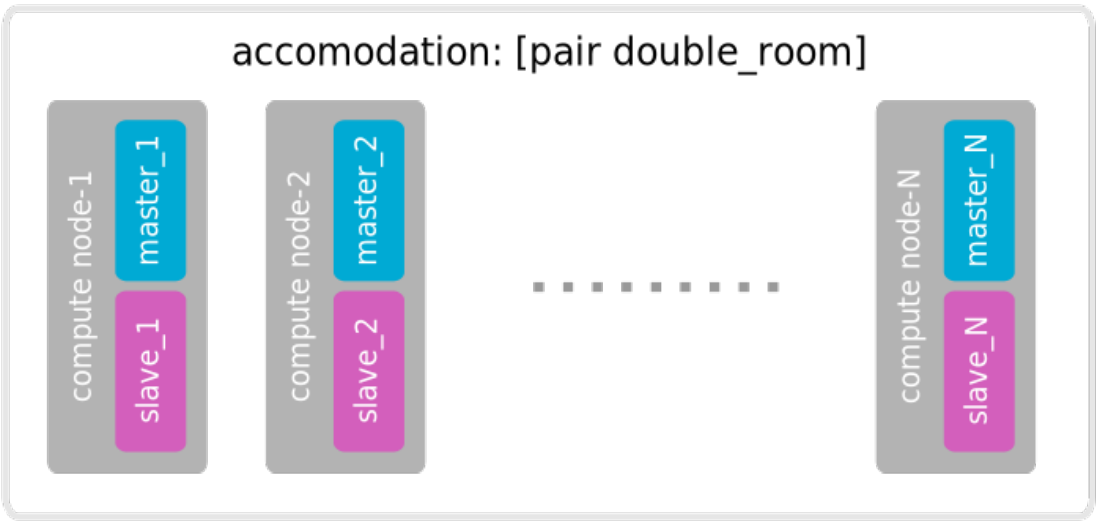
2.3.1 Deployment

By default Shaker spawns instances on every available compute node. The distribution of instances is configured by parameter `accommodation`. There are several instructions that allow control the scheduling precisely:

- `pair` - instances are grouped in pairs, meaning that one can be used as source of traffic and the other as a consumer (needed for networking tests)
- `single_room` - 1 instance per compute node
- `double_room` - 2 instances per compute node
- `density: N` - the multiplier for number of instances per compute node
- `compute_nodes: N` - how many compute nodes should be used (by default Shaker use all of them *see note below)
- `zones: [Z1, Z2]` - list of Nova availability zones to use
- `best_effort` - proceed even if the number of available compute nodes is less than what was requested

Examples:





As result of deployment the set of agents is produced. For networking testing this set contains agents in `primary` and `minion` roles. Primary agents are controlled by `shaker` tool and execute commands. Minions are used as back-ends and do not receive any commands directly.

*If a flavor is chosen, which has aggregate_instance_extra_specs metadata set to match a host aggregate, Shaker will only use matching computes for compute_nodes calculations. If no aggregate_instance_extra_specs is set on a flavor Shaker will use all computes by default.

> For example if we have 10 computes in a host aggregate with metadata special_hardware=true and use a flavor with aggregate_instance_extra_specs:special_hardware=true Shaker will only take into account the 10 matching computes, and by default try to use all of them

2.3.2 Execution

The execution part of scenario contains a list of tests that are executed one by one. By default Shaker runs the test simultaneously on all available agents. The level of concurrency can be controlled by option `progression`. There

are 3 values available:

- no value specified - all agents are involved;
- `linear` - the execution starts with 1 agent and increases by 1 until all agents are involved;
- `quadratic` - the execution starts with 1 agent (or 1 pair) and doubles until all agents are involved.

Tests are executed in order of definition. The exact action is defined by option `class`, additional attributes are provided by respective parameters. The following classes are available:

- `iperf3` - runs `iperf3` tool and shows chart and statistics
- `flent` - runs `flent` (http://flent.org) and shows chart and statistics
- `iperf` - runs `iperf` tool and shows plain output
- `netperf` - runs `netpers` tool and shows plain output
- `shell` - runs any shell command or process and shows plain output
- `iperf_graph` - runs `iperf` tool and shows chart and statistics (deprecated)

2.3.3 Test classes

Tools are configured via key-value attributes in test definition. For all networking tools Shaker offers unified parameters, that are translated automatically.

iperf3, iperf, iperf_graph:

- `time` - time in seconds to transmit for, defaults to *60*
- `udp` - use UDP instead of TCP, defaults to *TCP*
- `interval` - seconds between periodic bandwidth reports, defaults to *1 s*
- `bandwidth` - for UDP, bandwidth to send at in bits/sec, defaults to *1 Mbit/s*
- `threads` - number of parallel client threads to run
- `host` - the address of destination host to run the tool against, defaults to IP address of minion agent
- `datagram_size` - the size of UDP datagrams
- `mss` - set TCP maximum segment size

flent:

- `time` - time in seconds to transmit for, defaults to *60*
- `interval` - seconds between periodic bandwidth reports, defaults to *1*
- `method` - which flent scenario to use, see https://github.com/tohojo/flent/tree/master/flent/tests for the whole list, defaults to *tcp_download*
- `host` - the address of destination host to run the tool against, defaults to IP address of minion agent

netperf:

- `time` - time in seconds to transmit for, defaults to *60*
- `method` - one of built-in test names, see http://linux.die.net/man/1/netperf for the whole list, defaults to *TCP_STREAM*
- `host` - the address of destination host to run the tool against, defaults to IP address of minion agent

shell:

- `program` - run single program
- `script` - run bash script

2.3.4 SLA validation

Test case can contain SLA rules that are calculated upon test completion. Every rule has 2 parts: record selector and condition. The record selector allows to filter only subset of all records, e.g. of type *agent* to filter records produced by a single agent. The condition applies to particular statistics.

SLA examples:

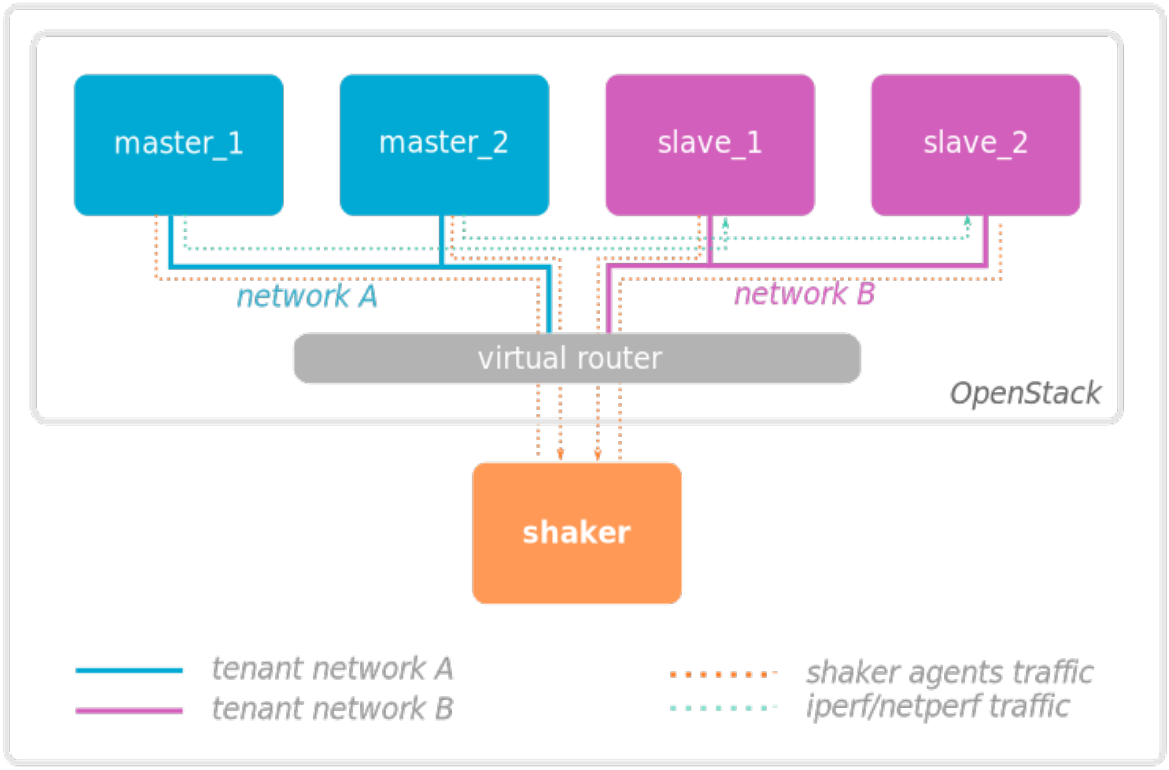
- `[type == 'agent'] >> (stats.bandwidth.min > 1000)` - require min bandwidth on every agent be at least 1000 Mbit
- `[type == 'agent'] >> (stderr == '')` - require stderr to be empty

Results of SLA validation can be obtained by generating output in subunit format. To do this a file name should be provided via *–subunit* parameter.

CHAPTER 3

Architecture

Shaker tool consists of server and agent modules. The server is executed by `shaker` command and is responsible for deployment of instances, execution of tests specified in scenario file, for results processing and report generation. The agent is light-weight and polls tasks from the server and replies with the results. Agents have connectivity to the server, but the server does not (so it is easy to keep agents behind NAT).



3.1 Under the Hood

Scenario execution involves the following steps:

1. User launches shaker with the following minimum set of parameters:

```
shaker --server-endpoint <host:port> --scenario <scenario> --report <report>
```

   where:

   - host:port - address of the machine where Shaker is installed and port is some arbitrary free port to bind the server to;
   - scenario - file name of the scenario (yaml file);
   - report - file name where report will be saved.

2. Shaker verifies connection to OpenStack. The parameters are taken from set of os-* params or from the env (`openrc`).
3. Based on `accommodation` parameter the list of agents is generated.
4. The topology is deployed with help of Heat. The list of agents is extended with IP addresses and instance names.
5. Shaker waits for all agents to join. Once all agents are alive it means that the quorum exists and everyone ready to execute the tests.
6. Shaker starts tests one by one in order they are listed in the scenario. Test definition is converted into the actual command that will be executed by agent. Shaker schedules the command to be started at the same time on all agents. For networking testing only agents in `primary` role are involved. Minion agents are used as back-end for corresponding commands (i.e. they run iperf in server mode).
7. Agents send their results to the server. Once all replies are received the test execution meant to be finished. If some agent didn't make it in dedicated time it is marked as lost.
8. Once all tests are executed Shaker can output the raw result in JSON format (if option `--output` is set).
9. Shaker clears the topology by calling Heat.
10. Shaker calculates statistics and aggregated charts. If there are any SLA statements they are also evaluated, the result can be stored in subunit format (if option `--subunit` is set).
11. Shaker generates report in HTML format into file specified by `--report` option.

CHAPTER 4

CLI Tools Reference

4.1 shaker

Executes specified scenario in OpenStack cloud, stores results and generates HTML report.

```
usage: shaker [-h] [--agent-dir AGENT_DIR]
              [--agent-join-timeout AGENT_JOIN_TIMEOUT]
              [--agent-loss-timeout AGENT_LOSS_TIMEOUT]
              [--artifacts-dir ARTIFACTS_DIR] [--book BOOK]
              [--cleanup-on-exit] [--config-dir DIR] [--config-file PATH]
              [--custom-user-opts CUSTOM_USER_OPTS] [--debug]
              [--dns-nameservers DNS_NAMESERVERS]
              [--external-net EXTERNAL_NET] [--flavor-name FLAVOR_NAME]
              [--image-name IMAGE_NAME] [--log-config-append PATH]
              [--log-date-format DATE_FORMAT] [--log-dir LOG_DIR]
              [--log-file PATH] [--matrix MATRIX] [--no-report-on-error]
              [--nocleanup-on-exit] [--nodebug] [--nono-report-on-error]
              [--noos-insecure] [--nouse-journal] [--nouse-json]
              [--nouse-syslog] [--nowatch-log-file] [--os-auth-url <auth-url>]
              [--os-cacert <auth-cacert>]
              [--os-identity-api-version <identity-api-version>]
              [--os-insecure] [--os-interface <os-interface>]
              [--os-password <auth-password>] [--os-profile <hmac-key>]
              [--os-project-domain-name <auth-project-domain-name>]
              [--os-project-name <auth-project-name>]
              [--os-region-name <auth-region-name>]
              [--os-tenant-name <auth-tenant-name>]
              [--os-user-domain-name <auth-user-domain-name>]
              [--os-username <auth-username>] [--output OUTPUT]
              [--polling-interval POLLING_INTERVAL] [--report REPORT]
              [--report-template REPORT_TEMPLATE]
              [--reuse-stack-name REUSE_STACK_NAME] [--scenario SCENARIO]
              [--scenario-availability-zone SCENARIO_AVAILABILITY_ZONE]
              [--scenario-compute-nodes SCENARIO_COMPUTE_NODES]
```

(continues on next page)

(continued from previous page)

```
              [--server-endpoint SERVER_ENDPOINT] [--stack-name STACK_NAME]
              [--subunit SUBUNIT] [--syslog-log-facility SYSLOG_LOG_FACILITY]
              [--use-journal] [--use-json] [--use-syslog] [--watch-log-file]

optional arguments:
  -h, --help            show this help message and exit
  --agent-dir AGENT_DIR
                        If specified, directs Shaker to write execution script
                        for the shell class in agent(s) instance defined
                        directory. Defaults to /tmp directory.
  --agent-join-timeout AGENT_JOIN_TIMEOUT
                        Timeout to treat agent as join failed in seconds,
                        defaults to env[SHAKER_AGENT_JOIN_TIMEOUT] (time
                        between stack deployment and start of scenario
                        execution).
  --agent-loss-timeout AGENT_LOSS_TIMEOUT
                        Timeout to treat agent as lost in seconds, defaults to
                        env[SHAKER_AGENT_LOSS_TIMEOUT]
  --artifacts-dir ARTIFACTS_DIR
                        If specified, directs Shaker to store there all its
                        artifacts (output, report, subunit and book). Defaults
                        to env[SHAKER_ARTIFACTS_DIR].
  --book BOOK           Generate report in ReST format and store it into the
                        specified folder, defaults to env[SHAKER_BOOK].
  --cleanup-on-exit     Clean up the heat-stack when exiting execution.
  --config-dir DIR      Path to a config directory to pull `*.conf` files
                        from. This file set is sorted, so as to provide a
                        predictable parse order if individual options are
                        over-ridden. The set is parsed after the file(s)
                        specified via previous --config-file, arguments hence
                        over-ridden options in the directory take precedence.
                        This option must be set from the command-line.
  --config-file PATH    Path to a config file to use. Multiple config files
                        can be specified, with values in later files taking
                        precedence. Defaults to None. This option must be set
                        from the command-line.
  --custom-user-opts CUSTOM_USER_OPTS
                        Set custom user option parameters for the scenario.
                        The value is specified in YAML, e.g. custom_user_opts
                        = { key1:value1, key2:value2} The values specified can
                        be referenced in the usual python way. e.g. {{
                        CONF.custom_user_opts['key1'] }}. This option is
                        useful to inject custom values into heat environment
                        files
  --debug, -d           If set to true, the logging level will be set to DEBUG
                        instead of the default INFO level.
  --dns-nameservers DNS_NAMESERVERS
                        Comma-separated list of IPs of the DNS nameservers for
                        the subnets. If no value is provided defaults to
                        Google Public DNS.
  --external-net EXTERNAL_NET
                        Name or ID of external network, defaults to
                        env[SHAKER_EXTERNAL_NET]. If no value provided then
                        Shaker picks any of available external networks.
  --flavor-name FLAVOR_NAME
                        Name of image flavor. The default is created by
                        shaker-image-builder.
```

(continues on next page)

(continued from previous page)

```
  --image-name IMAGE_NAME
                        Name of image to use. The default is created by
                        shaker-image-builder.
  --log-config-append PATH, --log-config PATH, --log_config PATH
                        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).
  --log-date-format DATE_FORMAT
                        Defines the format string for %(asctime)s in log
                        records. Default: None . This option is ignored if
                        log_config_append is set.
  --log-dir LOG_DIR, --logdir LOG_DIR
                        (Optional) The base directory used for relative
                        log_file paths. This option is ignored if
                        log_config_append is set.
  --log-file PATH, --logfile PATH
                        (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.
  --matrix MATRIX       Set the matrix of parameters for the scenario. The
                        value is specified in YAML format. E.g. to override
                        the scenario duration one may provide: "{time: 10}",
                        or to override list of hosts: "{host:[ping.online.net,
                        iperf.eenet.ee]}". When several parameters are
                        overridden all combinations are tested
  --no-report-on-error  Do not generate report for failed scenarios
  --nocleanup-on-exit   The inverse of --cleanup-on-exit
  --nodebug             The inverse of --debug
  --nono-report-on-error
                        The inverse of --no-report-on-error
  --noos-insecure       The inverse of --os-insecure
  --nouse-journal       The inverse of --use-journal
  --nouse-json          The inverse of --use-json
  --nouse-syslog        The inverse of --use-syslog
  --nowatch-log-file    The inverse of --watch-log-file
  --os-auth-url <auth-url>
                        Authentication URL, defaults to env[OS_AUTH_URL].
  --os-cacert <auth-cacert>
                        Location of CA Certificate, defaults to
                        env[OS_CACERT].
  --os-identity-api-version <identity-api-version>
                        Identity API version, defaults to
                        env[OS_IDENTITY_API_VERSION].
  --os-insecure         When using SSL in connections to the registry server,
                        do not require validation via a certifying authority,
                        defaults to env[OS_INSECURE].
  --os-interface <os-interface>
                        Interface type. Valid options are public, admin and
                        internal. defaults to env[OS_INTERFACE].
  --os-password <auth-password>
                        Authentication password, defaults to env[OS_PASSWORD].
```

(continues on next page)

(continued from previous page)

```
  --os-profile <hmac-key>
                        HMAC key for encrypting profiling context data,
                        defaults to env[OS_PROFILE].
  --os-project-domain-name <auth-project-domain-name>
                        Authentication project domain name. Defaults to
                        env[OS_PROJECT_DOMAIN_NAME].
  --os-project-name <auth-project-name>
                        Authentication project name. This option is mutually
                        exclusive with --os-tenant-name. Defaults to
                        env[OS_PROJECT_NAME].
  --os-region-name <auth-region-name>
                        Authentication region name, defaults to
                        env[OS_REGION_NAME].
  --os-tenant-name <auth-tenant-name>
                        Authentication tenant name, defaults to
                        env[OS_TENANT_NAME].
  --os-user-domain-name <auth-user-domain-name>
                        Authentication username. Defaults to
                        env[OS_USER_DOMAIN_NAME].
  --os-username <auth-username>
                        Authentication username, defaults to env[OS_USERNAME].
  --output OUTPUT       File for output in JSON format, defaults to
                        env[SHAKER_OUTPUT]. If it is empty, then output will
                        be saved to /tmp/shaker_<time_now>.json
  --polling-interval POLLING_INTERVAL
                        How frequently the agent polls server, in seconds
  --report REPORT       Report file name, defaults to env[SHAKER_REPORT].
  --report-template REPORT_TEMPLATE
                        Template for report. Can be a file name or one of
                        aliases: "interactive", "json". Defaults to
                        "interactive".
  --reuse-stack-name REUSE_STACK_NAME
                        Name of an existing Shaker heat stack to reuse. The
                        default is to not reuse an existing stack. Caution
                        should be taken to only reuse stacks meant for a
                        specific scenario. Also certain configs e.g. image-
                        name, flavor-name, stack-name, etc will be ignored
                        when reusing an existing stack.
  --scenario SCENARIO   Comma-separated list of scenarios to play. Each entity
                        can be a file name or one of aliases:
                        "misc/instance_metadata",
                        "openstack/cross_az/full_l2",
                        "openstack/cross_az/full_l3_east_west",
                        "openstack/cross_az/full_l3_north_south",
                        "openstack/cross_az/perf_l2",
                        "openstack/cross_az/perf_l3_east_west",
                        "openstack/cross_az/perf_l3_north_south",
                        "openstack/cross_az/udp_l2",
                        "openstack/cross_az/udp_l2_mss8950",
                        "openstack/cross_az/udp_l3_east_west",
                        "openstack/dense_l2", "openstack/dense_l3_east_west",
                        "openstack/dense_l3_north_south",
                        "openstack/external/dense_l3_north_south_no_fip",
                        "openstack/external/dense_l3_north_south_with_fip",
                        "openstack/external/full_l3_north_south_no_fip",
                        "openstack/external/full_l3_north_south_with_fip",
                        "openstack/external/perf_l3_north_south_no_fip",
```

(continues on next page)

(continued from previous page)

```
                      "openstack/external/perf_l3_north_south_with_fip",
                      "openstack/full_l2", "openstack/full_l3_east_west",
                      "openstack/full_l3_north_south", "openstack/perf_l2",
                      "openstack/perf_l3_east_west",
                      "openstack/perf_l3_north_south",
                      "openstack/qos/perf_l2", "openstack/udp_l2",
                      "openstack/udp_l3_east_west",
                      "openstack/udp_l3_north_south", "spot/ping",
                      "spot/tcp", "spot/udp". Defaults to
                      env[SHAKER_SCENARIO].
--scenario-availability-zone SCENARIO_AVAILABILITY_ZONE
                      Comma-separated list of availability_zone. If
                      specified this setting will override the
                      availability_zone accomodation setting in the scenario
                      test definition.Defaults to SCENARIO_AVAILABILITY_ZONE
--scenario-compute-nodes SCENARIO_COMPUTE_NODES
                      Number of compute_nodes. If specified this setting
                      will override the compute_nodes accomodation setting
                      in the scenario test definition. Defaults to
                      SCENARIO_COMPUTE_NODES
--server-endpoint SERVER_ENDPOINT
                      Address for server connections (host:port), defaults
                      to env[SHAKER_SERVER_ENDPOINT].
--stack-name STACK_NAME
                      Name of test heat stack. The default is a uniquely
                      generated name.
--subunit SUBUNIT     Subunit stream file name, defaults to
                      env[SHAKER_SUBUNIT].
--syslog-log-facility SYSLOG_LOG_FACILITY
                      Syslog facility to receive log lines. This option is
                      ignored if log_config_append is set.
--use-journal         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.
--use-json            Use JSON formatting for logging. This option is
                      ignored if log_config_append is set.
--use-syslog          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.
--watch-log-file      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.
```

4.2 shaker-spot

Executes specified scenario from the local node, stores results and generates HTML report.

```
usage: shaker-spot [-h] [--artifacts-dir ARTIFACTS_DIR] [--book BOOK]
                   [--config-dir DIR] [--config-file PATH]
                   [--custom-user-opts CUSTOM_USER_OPTS] [--debug]
                   [--log-config-append PATH] [--log-date-format DATE_FORMAT]
                   [--log-dir LOG_DIR] [--log-file PATH] [--matrix MATRIX]
                   [--no-report-on-error] [--nodebug] [--nono-report-on-error]
                   [--nouse-journal] [--nouse-json] [--nouse-syslog]
                   [--nowatch-log-file] [--output OUTPUT] [--report REPORT]
                   [--report-template REPORT_TEMPLATE] [--scenario SCENARIO]
                   [--scenario-availability-zone SCENARIO_AVAILABILITY_ZONE]
                   [--scenario-compute-nodes SCENARIO_COMPUTE_NODES]
                   [--subunit SUBUNIT]
                   [--syslog-log-facility SYSLOG_LOG_FACILITY] [--use-journal]
                   [--use-json] [--use-syslog] [--watch-log-file]

optional arguments:
  -h, --help            show this help message and exit
  --artifacts-dir ARTIFACTS_DIR
                        If specified, directs Shaker to store there all its
                        artifacts (output, report, subunit and book). Defaults
                        to env[SHAKER_ARTIFACTS_DIR].
  --book BOOK           Generate report in ReST format and store it into the
                        specified folder, defaults to env[SHAKER_BOOK].
  --config-dir DIR      Path to a config directory to pull `*.conf` files
                        from. This file set is sorted, so as to provide a
                        predictable parse order if individual options are
                        over-ridden. The set is parsed after the file(s)
                        specified via previous --config-file, arguments hence
                        over-ridden options in the directory take precedence.
                        This option must be set from the command-line.
  --config-file PATH    Path to a config file to use. Multiple config files
                        can be specified, with values in later files taking
                        precedence. Defaults to None. This option must be set
                        from the command-line.
  --custom-user-opts CUSTOM_USER_OPTS
                        Set custom user option parameters for the scenario.
                        The value is specified in YAML, e.g. custom_user_opts
                        = { key1:value1, key2:value2} The values specified can
                        be referenced in the usual python way. e.g. {{
                        CONF.custom_user_opts['key1'] }}. This option is
                        useful to inject custom values into heat environment
                        files
  --debug, -d           If set to true, the logging level will be set to DEBUG
                        instead of the default INFO level.
  --log-config-append PATH, --log-config PATH, --log_config PATH
                        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).
  --log-date-format DATE_FORMAT
                        Defines the format string for %(asctime)s in log
                        records. Default: None . This option is ignored if
                        log_config_append is set.
```

(continues on next page)

(continued from previous page)

```
  --log-dir LOG_DIR, --logdir LOG_DIR
                        (Optional) The base directory used for relative
                        log_file paths. This option is ignored if
                        log_config_append is set.
  --log-file PATH, --logfile PATH
                        (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.
  --matrix MATRIX       Set the matrix of parameters for the scenario. The
                        value is specified in YAML format. E.g. to override
                        the scenario duration one may provide: "{time: 10}",
                        or to override list of hosts: "{host:[ping.online.net,
                        iperf.eenet.ee]}". When several parameters are
                        overridden all combinations are tested
  --no-report-on-error  Do not generate report for failed scenarios
  --nodebug             The inverse of --debug
  --nono-report-on-error
                        The inverse of --no-report-on-error
  --nouse-journal       The inverse of --use-journal
  --nouse-json          The inverse of --use-json
  --nouse-syslog        The inverse of --use-syslog
  --nowatch-log-file    The inverse of --watch-log-file
  --output OUTPUT       File for output in JSON format, defaults to
                        env[SHAKER_OUTPUT]. If it is empty, then output will
                        be saved to /tmp/shaker_<time_now>.json
  --report REPORT       Report file name, defaults to env[SHAKER_REPORT].
  --report-template REPORT_TEMPLATE
                        Template for report. Can be a file name or one of
                        aliases: "interactive", "json". Defaults to
                        "interactive".
  --scenario SCENARIO   Comma-separated list of scenarios to play. Each entity
                        can be a file name or one of aliases:
                        "misc/instance_metadata",
                        "openstack/cross_az/full_l2",
                        "openstack/cross_az/full_l3_east_west",
                        "openstack/cross_az/full_l3_north_south",
                        "openstack/cross_az/perf_l2",
                        "openstack/cross_az/perf_l3_east_west",
                        "openstack/cross_az/perf_l3_north_south",
                        "openstack/cross_az/udp_l2",
                        "openstack/cross_az/udp_l2_mss8950",
                        "openstack/cross_az/udp_l3_east_west",
                        "openstack/dense_l2", "openstack/dense_l3_east_west",
                        "openstack/dense_l3_north_south",
                        "openstack/external/dense_l3_north_south_no_fip",
                        "openstack/external/dense_l3_north_south_with_fip",
                        "openstack/external/full_l3_north_south_no_fip",
                        "openstack/external/full_l3_north_south_with_fip",
                        "openstack/external/perf_l3_north_south_no_fip",
                        "openstack/external/perf_l3_north_south_with_fip",
                        "openstack/full_l2", "openstack/full_l3_east_west",
                        "openstack/full_l3_north_south", "openstack/perf_l2",
                        "openstack/perf_l3_east_west",
                        "openstack/perf_l3_north_south",
                        "openstack/qos/perf_l2", "openstack/udp_l2",
                        "openstack/udp_l3_east_west",
```

(continues on next page)

(continued from previous page)

```
                        "openstack/udp_l3_north_south", "spot/ping",
                        "spot/tcp", "spot/udp". Defaults to
                        env[SHAKER_SCENARIO].
  --scenario-availability-zone SCENARIO_AVAILABILITY_ZONE
                        Comma-separated list of availability_zone. If
                        specified this setting will override the
                        availability_zone accomodation setting in the scenario
                        test definition.Defaults to SCENARIO_AVAILABILITY_ZONE
  --scenario-compute-nodes SCENARIO_COMPUTE_NODES
                        Number of compute_nodes. If specified this setting
                        will override the compute_nodes accomodation setting
                        in the scenario test definition. Defaults to
                        SCENARIO_COMPUTE_NODES
  --subunit SUBUNIT     Subunit stream file name, defaults to
                        env[SHAKER_SUBUNIT].
  --syslog-log-facility SYSLOG_LOG_FACILITY
                        Syslog facility to receive log lines. This option is
                        ignored if log_config_append is set.
  --use-journal         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.
  --use-json            Use JSON formatting for logging. This option is
                        ignored if log_config_append is set.
  --use-syslog          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.
  --watch-log-file      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.
```

4.3 shaker-image-builder

Builds base image in OpenStack cloud. The image is based on Ubuntu cloud image distro and configured to run `shaker-agent`.

```
usage: shaker-image-builder [-h] [--cleanup-on-exit] [--config-dir DIR]
                            [--config-file PATH] [--debug]
                            [--dns-nameservers DNS_NAMESERVERS]
                            [--external-net EXTERNAL_NET]
                            [--flavor-disk FLAVOR_DISK]
                            [--flavor-name FLAVOR_NAME]
                            [--flavor-ram FLAVOR_RAM]
                            [--flavor-vcpus FLAVOR_VCPUS]
                            [--image-builder-distro IMAGE_BUILDER_DISTRO]
                            [--image-builder-mode IMAGE_BUILDER_MODE]
                            [--image-builder-template IMAGE_BUILDER_TEMPLATE]
                            [--image-name IMAGE_NAME]
                            [--log-config-append PATH]
```

(continues on next page)

(continued from previous page)

```
                            [--log-date-format DATE_FORMAT]
                            [--log-dir LOG_DIR] [--log-file PATH]
                            [--nocleanup-on-exit] [--nodebug]
                            [--noos-insecure] [--nouse-journal] [--nouse-json]
                            [--nouse-syslog] [--nowatch-log-file]
                            [--os-auth-url <auth-url>]
                            [--os-cacert <auth-cacert>]
                            [--os-identity-api-version <identity-api-version>]
                            [--os-insecure] [--os-interface <os-interface>]
                            [--os-password <auth-password>]
                            [--os-profile <hmac-key>]
                            [--os-project-domain-name <auth-project-domain-name>]
                            [--os-project-name <auth-project-name>]
                            [--os-region-name <auth-region-name>]
                            [--os-tenant-name <auth-tenant-name>]
                            [--os-user-domain-name <auth-user-domain-name>]
                            [--os-username <auth-username>]
                            [--reuse-stack-name REUSE_STACK_NAME]
                            [--stack-name STACK_NAME]
                            [--syslog-log-facility SYSLOG_LOG_FACILITY]
                            [--use-journal] [--use-json] [--use-syslog]
                            [--watch-log-file]

optional arguments:
  -h, --help            show this help message and exit
  --cleanup-on-exit     Clean up the heat-stack when exiting execution.
  --config-dir DIR      Path to a config directory to pull `*.conf` files
                        from. This file set is sorted, so as to provide a
                        predictable parse order if individual options are
                        over-ridden. The set is parsed after the file(s)
                        specified via previous --config-file, arguments hence
                        over-ridden options in the directory take precedence.
                        This option must be set from the command-line.
  --config-file PATH    Path to a config file to use. Multiple config files
                        can be specified, with values in later files taking
                        precedence. Defaults to None. This option must be set
                        from the command-line.
  --debug, -d           If set to true, the logging level will be set to DEBUG
                        instead of the default INFO level.
  --dns-nameservers DNS_NAMESERVERS
                        Comma-separated list of IPs of the DNS nameservers for
                        the subnets. If no value is provided defaults to
                        Google Public DNS.
  --external-net EXTERNAL_NET
                        Name or ID of external network, defaults to
                        env[SHAKER_EXTERNAL_NET]. If no value provided then
                        Shaker picks any of available external networks.
  --flavor-disk FLAVOR_DISK
                        Shaker image disk size in GB, defaults to
                        env[SHAKER_FLAVOR_DISK]
  --flavor-name FLAVOR_NAME
                        Name of image flavor. The default is created by
                        shaker-image-builder.
  --flavor-ram FLAVOR_RAM
                        Shaker image RAM size in MB, defaults to
                        env[SHAKER_FLAVOR_RAM]
  --flavor-vcpus FLAVOR_VCPUS
```

(continues on next page)

(continued from previous page)

```
                      Number of cores to allocate for Shaker image, defaults
                      to env[SHAKER_FLAVOR_VCPUS]
  --image-builder-distro IMAGE_BUILDER_DISTRO
                      Operating System Distribution for shaker image when
                      using diskimage-builder, defaults to ubuntu Allowed
                      values: ubuntu, centos7
  --image-builder-mode IMAGE_BUILDER_MODE
                      Image building mode: "heat" - using Heat template
                      (requires Glance v1 for base image upload); "dib" -
                      using diskimage-builder elements (requires qemu-utils
                      and debootstrap). If not set, switches to "dib" if
                      Glance v1 is not available. Can be specified as
                      env[SHAKER_IMAGE_BUILDER_MODE] Allowed values: heat,
                      dib
  --image-builder-template IMAGE_BUILDER_TEMPLATE
                      Heat template containing receipt of building the
                      image. Can be a file name or one of aliases: "centos",
                      "debian", "ubuntu". Defaults to "ubuntu".
  --image-name IMAGE_NAME
                      Name of image to use. The default is created by
                      shaker-image-builder.
  --log-config-append PATH, --log-config PATH, --log_config PATH
                      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).
  --log-date-format DATE_FORMAT
                      Defines the format string for %(asctime)s in log
                      records. Default: None . This option is ignored if
                      log_config_append is set.
  --log-dir LOG_DIR, --logdir LOG_DIR
                      (Optional) The base directory used for relative
                      log_file paths. This option is ignored if
                      log_config_append is set.
  --log-file PATH, --logfile PATH
                      (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.
  --nocleanup-on-exit   The inverse of --cleanup-on-exit
  --nodebug             The inverse of --debug
  --noos-insecure       The inverse of --os-insecure
  --nouse-journal       The inverse of --use-journal
  --nouse-json          The inverse of --use-json
  --nouse-syslog        The inverse of --use-syslog
  --nowatch-log-file    The inverse of --watch-log-file
  --os-auth-url <auth-url>
                      Authentication URL, defaults to env[OS_AUTH_URL].
  --os-cacert <auth-cacert>
                      Location of CA Certificate, defaults to
                      env[OS_CACERT].
  --os-identity-api-version <identity-api-version>
                      Identity API version, defaults to
```

(continues on next page)

(continued from previous page)

```
                      env[OS_IDENTITY_API_VERSION].
  --os-insecure       When using SSL in connections to the registry server,
                      do not require validation via a certifying authority,
                      defaults to env[OS_INSECURE].
  --os-interface <os-interface>
                      Interface type. Valid options are public, admin and
                      internal. defaults to env[OS_INTERFACE].
  --os-password <auth-password>
                      Authentication password, defaults to env[OS_PASSWORD].
  --os-profile <hmac-key>
                      HMAC key for encrypting profiling context data,
                      defaults to env[OS_PROFILE].
  --os-project-domain-name <auth-project-domain-name>
                      Authentication project domain name. Defaults to
                      env[OS_PROJECT_DOMAIN_NAME].
  --os-project-name <auth-project-name>
                      Authentication project name. This option is mutually
                      exclusive with --os-tenant-name. Defaults to
                      env[OS_PROJECT_NAME].
  --os-region-name <auth-region-name>
                      Authentication region name, defaults to
                      env[OS_REGION_NAME].
  --os-tenant-name <auth-tenant-name>
                      Authentication tenant name, defaults to
                      env[OS_TENANT_NAME].
  --os-user-domain-name <auth-user-domain-name>
                      Authentication username. Defaults to
                      env[OS_USER_DOMAIN_NAME].
  --os-username <auth-username>
                      Authentication username, defaults to env[OS_USERNAME].
  --reuse-stack-name REUSE_STACK_NAME
                      Name of an existing Shaker heat stack to reuse. The
                      default is to not reuse an existing stack. Caution
                      should be taken to only reuse stacks meant for a
                      specific scenario. Also certain configs e.g. image-
                      name, flavor-name, stack-name, etc will be ignored
                      when reusing an existing stack.
  --stack-name STACK_NAME
                      Name of test heat stack. The default is a uniquely
                      generated name.
  --syslog-log-facility SYSLOG_LOG_FACILITY
                      Syslog facility to receive log lines. This option is
                      ignored if log_config_append is set.
  --use-journal       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.
  --use-json          Use JSON formatting for logging. This option is
                      ignored if log_config_append is set.
  --use-syslog        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.
  --watch-log-file    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
```

(continues on next page)

(continued from previous page)

```
                        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.
```

4.4 shaker-agent

Client-side process that is run inside pre-configured image.

```
usage: shaker-agent [-h] [--agent-dir AGENT_DIR] [--agent-id AGENT_ID]
                    [--agent-socket-conn-retries AGENT_SOCKET_CONN_RETRIES]
                    [--agent-socket-recv-timeout AGENT_SOCKET_RECV_TIMEOUT]
                    [--agent-socket-send-timeout AGENT_SOCKET_SEND_TIMEOUT]
                    [--config-dir DIR] [--config-file PATH] [--debug]
                    [--log-config-append PATH] [--log-date-format DATE_FORMAT]
                    [--log-dir LOG_DIR] [--log-file PATH] [--nodebug]
                    [--nouse-journal] [--nouse-json] [--nouse-syslog]
                    [--nowatch-log-file] [--polling-interval POLLING_INTERVAL]
                    [--server-endpoint SERVER_ENDPOINT]
                    [--syslog-log-facility SYSLOG_LOG_FACILITY]
                    [--use-journal] [--use-json] [--use-syslog]
                    [--watch-log-file]

optional arguments:
  -h, --help            show this help message and exit
  --agent-dir AGENT_DIR
                        If specified, directs Shaker to write execution script
                        for the shell class in agent(s) instance defined
                        directory. Defaults to /tmp directory.
  --agent-id AGENT_ID   Agent unique id, defaults to MAC of primary interface.
  --agent-socket-conn-retries AGENT_SOCKET_CONN_RETRIES
                        Prior to exiting, the number of reconnects the Agent
                        will attempt with the server upon socket operation
                        errors.
  --agent-socket-recv-timeout AGENT_SOCKET_RECV_TIMEOUT
                        The amount of time the socket will wait for a response
                        from a sent message, in milliseconds.
  --agent-socket-send-timeout AGENT_SOCKET_SEND_TIMEOUT
                        The amount of time the socket will wait until a sent
                        message is accepted, in milliseconds.
  --config-dir DIR      Path to a config directory to pull `*.conf` files
                        from. This file set is sorted, so as to provide a
                        predictable parse order if individual options are
                        over-ridden. The set is parsed after the file(s)
                        specified via previous --config-file, arguments hence
                        over-ridden options in the directory take precedence.
                        This option must be set from the command-line.
  --config-file PATH    Path to a config file to use. Multiple config files
                        can be specified, with values in later files taking
                        precedence. Defaults to None. This option must be set
                        from the command-line.
  --debug, -d           If set to true, the logging level will be set to DEBUG
                        instead of the default INFO level.
  --log-config-append PATH, --log-config PATH, --log_config PATH
                        The name of a logging configuration file. This file is
```

(continues on next page)

(continued from previous page)

```
                      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).
  --log-date-format DATE_FORMAT
                      Defines the format string for %(asctime)s in log
                      records. Default: None . This option is ignored if
                      log_config_append is set.
  --log-dir LOG_DIR, --logdir LOG_DIR
                      (Optional) The base directory used for relative
                      log_file paths. This option is ignored if
                      log_config_append is set.
  --log-file PATH, --logfile PATH
                      (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.
  --nodebug           The inverse of --debug
  --nouse-journal     The inverse of --use-journal
  --nouse-json        The inverse of --use-json
  --nouse-syslog      The inverse of --use-syslog
  --nowatch-log-file  The inverse of --watch-log-file
  --polling-interval POLLING_INTERVAL
                      How frequently the agent polls server, in seconds
  --server-endpoint SERVER_ENDPOINT
                      Address for server connections (host:port), defaults
                      to env[SHAKER_SERVER_ENDPOINT].
  --syslog-log-facility SYSLOG_LOG_FACILITY
                      Syslog facility to receive log lines. This option is
                      ignored if log_config_append is set.
  --use-journal       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.
  --use-json          Use JSON formatting for logging. This option is
                      ignored if log_config_append is set.
  --use-syslog        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.
  --watch-log-file    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.
```

4.5 shaker-report

Generates report based on raw results stored in JSON format.

```
usage: shaker-report [-h] [--book BOOK] [--config-dir DIR]
                     [--config-file PATH] [--debug] [--input INPUT]
                     [--log-config-append PATH]
                     [--log-date-format DATE_FORMAT] [--log-dir LOG_DIR]
                     [--log-file PATH] [--nodebug] [--nouse-journal]
                     [--nouse-json] [--nouse-syslog] [--nowatch-log-file]
                     [--report REPORT] [--report-template REPORT_TEMPLATE]
                     [--subunit SUBUNIT]
                     [--syslog-log-facility SYSLOG_LOG_FACILITY]
                     [--use-journal] [--use-json] [--use-syslog]
                     [--watch-log-file]

optional arguments:
  -h, --help            show this help message and exit
  --book BOOK           Generate report in ReST format and store it into the
                        specified folder, defaults to env[SHAKER_BOOK].
  --config-dir DIR      Path to a config directory to pull `*.conf` files
                        from. This file set is sorted, so as to provide a
                        predictable parse order if individual options are
                        over-ridden. The set is parsed after the file(s)
                        specified via previous --config-file, arguments hence
                        over-ridden options in the directory take precedence.
                        This option must be set from the command-line.
  --config-file PATH    Path to a config file to use. Multiple config files
                        can be specified, with values in later files taking
                        precedence. Defaults to None. This option must be set
                        from the command-line.
  --debug, -d           If set to true, the logging level will be set to DEBUG
                        instead of the default INFO level.
  --input INPUT         File or list of files to read test results from,
                        defaults to env[SHAKER_INPUT].
  --log-config-append PATH, --log-config PATH, --log_config PATH
                        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).
  --log-date-format DATE_FORMAT
                        Defines the format string for %(asctime)s in log
                        records. Default: None . This option is ignored if
                        log_config_append is set.
  --log-dir LOG_DIR, --logdir LOG_DIR
                        (Optional) The base directory used for relative
                        log_file paths. This option is ignored if
                        log_config_append is set.
  --log-file PATH, --logfile PATH
                        (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.
  --nodebug             The inverse of --debug
  --nouse-journal       The inverse of --use-journal
  --nouse-json          The inverse of --use-json
  --nouse-syslog        The inverse of --use-syslog
```

(continues on next page)

(continued from previous page)

```
  --nowatch-log-file    The inverse of --watch-log-file
  --report REPORT       Report file name, defaults to env[SHAKER_REPORT].
  --report-template REPORT_TEMPLATE
                        Template for report. Can be a file name or one of
                        aliases: "interactive", "json". Defaults to
                        "interactive".
  --subunit SUBUNIT     Subunit stream file name, defaults to
                        env[SHAKER_SUBUNIT].
  --syslog-log-facility SYSLOG_LOG_FACILITY
                        Syslog facility to receive log lines. This option is
                        ignored if log_config_append is set.
  --use-journal         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.
  --use-json            Use JSON formatting for logging. This option is
                        ignored if log_config_append is set.
  --use-syslog          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.
  --watch-log-file      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.
```

4.6 shaker-cleanup

Removes base image from OpenStack cloud.

```
usage: shaker-cleanup [-h] [--cleanup] [--cleanup-on-exit] [--config-dir DIR]
                      [--config-file PATH] [--debug]
                      [--dns-nameservers DNS_NAMESERVERS]
                      [--external-net EXTERNAL_NET]
                      [--flavor-name FLAVOR_NAME] [--image-name IMAGE_NAME]
                      [--log-config-append PATH]
                      [--log-date-format DATE_FORMAT] [--log-dir LOG_DIR]
                      [--log-file PATH] [--nocleanup] [--nocleanup-on-exit]
                      [--nodebug] [--noos-insecure] [--nouse-journal]
                      [--nouse-json] [--nouse-syslog] [--nowatch-log-file]
                      [--os-auth-url <auth-url>] [--os-cacert <auth-cacert>]
                      [--os-identity-api-version <identity-api-version>]
                      [--os-insecure] [--os-interface <os-interface>]
                      [--os-password <auth-password>]
                      [--os-profile <hmac-key>]
                      [--os-project-domain-name <auth-project-domain-name>]
                      [--os-project-name <auth-project-name>]
                      [--os-region-name <auth-region-name>]
                      [--os-tenant-name <auth-tenant-name>]
                      [--os-user-domain-name <auth-user-domain-name>]
                      [--os-username <auth-username>]
```

(continues on next page)

(continued from previous page)

```
                      [--reuse-stack-name REUSE_STACK_NAME]
                      [--stack-name STACK_NAME]
                      [--syslog-log-facility SYSLOG_LOG_FACILITY]
                      [--use-journal] [--use-json] [--use-syslog]
                      [--watch-log-file]

optional arguments:
  -h, --help            show this help message and exit
  --cleanup             Cleanup the image and the flavor.
  --cleanup-on-exit     Clean up the heat-stack when exiting execution.
  --config-dir DIR      Path to a config directory to pull `*.conf` files
                        from. This file set is sorted, so as to provide a
                        predictable parse order if individual options are
                        over-ridden. The set is parsed after the file(s)
                        specified via previous --config-file, arguments hence
                        over-ridden options in the directory take precedence.
                        This option must be set from the command-line.
  --config-file PATH    Path to a config file to use. Multiple config files
                        can be specified, with values in later files taking
                        precedence. Defaults to None. This option must be set
                        from the command-line.
  --debug, -d           If set to true, the logging level will be set to DEBUG
                        instead of the default INFO level.
  --dns-nameservers DNS_NAMESERVERS
                        Comma-separated list of IPs of the DNS nameservers for
                        the subnets. If no value is provided defaults to
                        Google Public DNS.
  --external-net EXTERNAL_NET
                        Name or ID of external network, defaults to
                        env[SHAKER_EXTERNAL_NET]. If no value provided then
                        Shaker picks any of available external networks.
  --flavor-name FLAVOR_NAME
                        Name of image flavor. The default is created by
                        shaker-image-builder.
  --image-name IMAGE_NAME
                        Name of image to use. The default is created by
                        shaker-image-builder.
  --log-config-append PATH, --log-config PATH, --log_config PATH
                        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).
  --log-date-format DATE_FORMAT
                        Defines the format string for %(asctime)s in log
                        records. Default: None . This option is ignored if
                        log_config_append is set.
  --log-dir LOG_DIR, --logdir LOG_DIR
                        (Optional) The base directory used for relative
                        log_file paths. This option is ignored if
                        log_config_append is set.
  --log-file PATH, --logfile PATH
                        (Optional) Name of log file to send logging output to.
                        If no default is set, logging will go to stderr as
```

(continues on next page)

(continued from previous page)

```
                      defined by use_stderr. This option is ignored if
                      log_config_append is set.
  --nocleanup         The inverse of --cleanup
  --nocleanup-on-exit The inverse of --cleanup-on-exit
  --nodebug           The inverse of --debug
  --noos-insecure     The inverse of --os-insecure
  --nouse-journal     The inverse of --use-journal
  --nouse-json        The inverse of --use-json
  --nouse-syslog      The inverse of --use-syslog
  --nowatch-log-file  The inverse of --watch-log-file
  --os-auth-url <auth-url>
                      Authentication URL, defaults to env[OS_AUTH_URL].
  --os-cacert <auth-cacert>
                      Location of CA Certificate, defaults to
                      env[OS_CACERT].
  --os-identity-api-version <identity-api-version>
                      Identity API version, defaults to
                      env[OS_IDENTITY_API_VERSION].
  --os-insecure       When using SSL in connections to the registry server,
                      do not require validation via a certifying authority,
                      defaults to env[OS_INSECURE].
  --os-interface <os-interface>
                      Interface type. Valid options are public, admin and
                      internal. defaults to env[OS_INTERFACE].
  --os-password <auth-password>
                      Authentication password, defaults to env[OS_PASSWORD].
  --os-profile <hmac-key>
                      HMAC key for encrypting profiling context data,
                      defaults to env[OS_PROFILE].
  --os-project-domain-name <auth-project-domain-name>
                      Authentication project domain name. Defaults to
                      env[OS_PROJECT_DOMAIN_NAME].
  --os-project-name <auth-project-name>
                      Authentication project name. This option is mutually
                      exclusive with --os-tenant-name. Defaults to
                      env[OS_PROJECT_NAME].
  --os-region-name <auth-region-name>
                      Authentication region name, defaults to
                      env[OS_REGION_NAME].
  --os-tenant-name <auth-tenant-name>
                      Authentication tenant name, defaults to
                      env[OS_TENANT_NAME].
  --os-user-domain-name <auth-user-domain-name>
                      Authentication username. Defaults to
                      env[OS_USER_DOMAIN_NAME].
  --os-username <auth-username>
                      Authentication username, defaults to env[OS_USERNAME].
  --reuse-stack-name REUSE_STACK_NAME
                      Name of an existing Shaker heat stack to reuse. The
                      default is to not reuse an existing stack. Caution
                      should be taken to only reuse stacks meant for a
                      specific scenario. Also certain configs e.g. image-
                      name, flavor-name, stack-name, etc will be ignored
                      when reusing an existing stack.
  --stack-name STACK_NAME
                      Name of test heat stack. The default is a uniquely
                      generated name.
```

(continues on next page)

(continued from previous page)

```
  --syslog-log-facility SYSLOG_LOG_FACILITY
                        Syslog facility to receive log lines. This option is
                        ignored if log_config_append is set.
  --use-journal         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.
  --use-json            Use JSON formatting for logging. This option is
                        ignored if log_config_append is set.
  --use-syslog          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.
  --watch-log-file      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.
```

CHAPTER 5

Scenario Catalog

5.1 Scenarios

5.1.1 OpenStack instances metadata query

In this scenario Shaker launches ten instances on a single compute node and asks instances to retrieve the metadata. The scenario is used to load metadata processes.

To use this scenario specify parameter `--scenario misc/instance_metadata`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/misc/instance_metadata.yaml

5.1.2 OpenStack L2 Cross-AZ

In this scenario Shaker launches pairs of instances in the same tenant network. Every instance is hosted on a separate compute node, all available compute nodes are utilized. The primary and minion instances are in different availability zones. The scenario is used to test throughput between *nova* and *vcenter* zones. The traffic goes within the tenant network (L2 domain).

To use this scenario specify parameter `--scenario openstack/cross_az/full_l2`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/full_l2.yaml

5.1.3 OpenStack L3 East-West Cross-AZ

In this scenario Shaker launches pairs of instances, each instance on its own compute node. All available compute nodes are utilized. Instances are connected to one of 2 tenant networks, which plugged into single router. The traffic goes from one network to the other (L3 east-west). The primary and minion instances are in different availability zones. The scenario is used to test throughput between *nova* and *vcenter* zones.

To use this scenario specify parameter `--scenario openstack/cross_az/full_l3_east_west`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/full_l3_east_west.yaml

5.1.4 OpenStack L3 North-South Cross-AZ

In this scenario Shaker launches pairs of instances on different compute nodes. All available compute nodes are utilized. Instances are in different networks connected to different routers, primary accesses minion by floating ip. The traffic goes from one network via external network to the other network. The primary and minion instances are in different availability zones. The scenario is used to test throughput between *nova* and *vcenter* zones.

To use this scenario specify parameter `--scenario openstack/cross_az/full_l3_north_south`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/full_l3_north_south.yaml

5.1.5 OpenStack L2 Cross-AZ Performance

In this scenario Shaker launches 1 pair of instances in the same tenant network. Each instance is hosted on a separate compute node. The primary and minion instances are in different availability zones. The scenario is used to test throughput between *nova* and *vcenter* zones.

To use this scenario specify parameter `--scenario openstack/cross_az/perf_l2`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/perf_l2.yaml

5.1.6 OpenStack L3 East-West Cross-AZ Performance

In this scenario Shaker launches 1 pair of instances, each instance on its own compute node. Instances are connected to one of 2 tenant networks, which plugged into single router. The traffic goes from one network to the other (L3 east-west). The primary and minion instances are in different availability zones. The scenario is used to test throughput between *nova* and *vcenter* zones.

To use this scenario specify parameter `--scenario openstack/cross_az/perf_l3_east_west`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/perf_l3_east_west.yaml

5.1.7 OpenStack L3 North-South Cross-AZ Performance

In this scenario Shaker launches 1 pair of instances on different compute nodes. Instances are in different networks connected to different routers, primary accesses minion by floating ip. The traffic goes from one network via external network to the other network. The primary and minion instances are in different availability zones. The scenario is used to test throughput between *nova* and *vcenter* zones.

To use this scenario specify parameter `--scenario openstack/cross_az/perf_l3_north_south`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/perf_l3_north_south.yaml

5.1.8 OpenStack L2 Cross-AZ UDP

In this scenario Shaker launches pairs of instances in the same tenant network. Every instance is hosted on a separate compute node. The load is generated by UDP traffic. The primary and minion instances are in different availability zones. The scenario is used to test throughput between *nova* and *vcenter* zones.

To use this scenario specify parameter `--scenario openstack/cross_az/udp_l2`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/udp_l2.yaml

5.1.9 OpenStack L2 Cross-AZ UDP Jumbo

In this scenario Shaker launches pairs of instances in the same tenant network. Every instance is hosted on a separate compute node. The load is generated by UDP traffic and jumbo packets. The primary and minion instances are in different availability zones. The scenario is used to test throughput between *nova* and *vcenter* zones.

To use this scenario specify parameter `--scenario openstack/cross_az/udp_l2_mss8950`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/udp_l2_mss8950.yaml

5.1.10 OpenStack L3 East-West Cross-AZ UDP

In this scenario Shaker launches pairs of instances, each instance on its own compute node. Instances are connected to one of 2 tenant networks, which plugged into single router. The traffic goes from one network to the other (L3 east-west). The load is generated by UDP traffic. The primary and minion instances are in different availability zones. The scenario is used to test throughput between *nova* and *vcenter* zones.

To use this scenario specify parameter `--scenario openstack/cross_az/udp_l3_east_west`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/udp_l3_east_west.yaml

5.1.11 OpenStack L2 Dense

In this scenario Shaker launches several pairs of instances on a single compute node. Instances are plugged into the same tenant network. The traffic goes within the tenant network (L2 domain).

To use this scenario specify parameter `--scenario openstack/dense_l2`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/dense_l2.yaml

5.1.12 OpenStack L3 East-West Dense

In this scenario Shaker launches pairs of instances on the same compute node. Instances are connected to different tenant networks connected to one router. The traffic goes from one network to the other (L3 east-west).

To use this scenario specify parameter `--scenario openstack/dense_l3_east_west`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/dense_l3_east_west.yaml

5.1.13 OpenStack L3 North-South Dense

In this scenario Shaker launches pairs of instances on the same compute node. Instances are connected to different tenant networks, each connected to own router. Instances in one of networks have floating IPs. The traffic goes from one network via external network to the other network.

To use this scenario specify parameter `--scenario openstack/dense_l3_north_south`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/dense_l3_north_south.yaml

5.1.14 OpenStack L3 North-South Dense to external target

In this scenario Shaker launches instances on one compute node in a tenant network connected to external network. The traffic is sent to and from external host. The host name needs to be provided as command-line parameter, e.g. `--matrix "{host: 172.10.1.2}"`.

To use this scenario specify parameter `--scenario openstack/external/dense_l3_north_south_no_fip`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/external/dense_l3_north_south_no_fip.yaml

5.1.15 OpenStack L3 North-South Dense to external target with floating IP

In this scenario Shaker launches instances on one compute node in a tenant network connected to external network. All instances have floating IPs. The traffic is sent to and from external host. The host name needs to be provided as command-line parameter, e.g. `--matrix "{host: 172.10.1.2}"`.

To use this scenario specify parameter `--scenario openstack/external/dense_l3_north_south_with_fip`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/external/dense_l3_north_south_with_fip.yaml

5.1.16 OpenStack L3 North-South to external target

In this scenario Shaker launches instances in a tenant network connected to external network. Every instance is hosted on dedicated compute node. All available compute nodes are utilized. The traffic is sent to and from external host (L3 north-south). The host name needs to be provided as command-line parameter, e.g. `--matrix "{host: 172.10.1.2}"`.

To use this scenario specify parameter `--scenario openstack/external/full_l3_north_south_no_fip`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/external/full_l3_north_south_no_fip.yaml

5.1.17 OpenStack L3 North-South to external target with floating IP

In this scenario Shaker launches instances in a tenant network connected to external network. Every instance is hosted on dedicated compute node. All available compute nodes are utilized. All instances have floating IPs. The traffic is sent to and from external host (L3 north-south). The host name needs to be provided as command-line parameter, e.g. `--matrix "{host: 172.10.1.2}"`.

To use this scenario specify parameter `--scenario openstack/external/full_l3_north_south_with_fip`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/external/full_l3_north_south_with_fip.yaml

5.1.18 OpenStack L3 North-South Performance to external target

In this scenario Shaker launches instance in a tenant network connected to external network. The traffic is sent to and from external host. By default one of public iperf3 servers is used, to override this the target host can be provided as command-line parameter, e.g. `--matrix "{host: 172.10.1.2}"`.

To use this scenario specify parameter `--scenario openstack/external/perf_l3_north_south_no_fip`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/external/perf_l3_north_south_no_fip.yaml

5.1.19 OpenStack L3 North-South performance to external target with floating IP

In this scenario Shaker launches instance in a tenant network connected to external network. The instance has floating IP. The traffic is sent to and from external host. By default one of public iperf3 servers is used, to override this the target host can be provided as command-line parameter, e.g. `--matrix "{host: 172.10.1.2}"`.

To use this scenario specify parameter `--scenario openstack/external/perf_l3_north_south_with_fip`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/external/perf_l3_north_south_with_fip.yaml

5.1.20 OpenStack L2

In this scenario Shaker launches pairs of instances in the same tenant network. Every instance is hosted on a separate compute node, all available compute nodes are utilized. The traffic goes within the tenant network (L2 domain).

To use this scenario specify parameter `--scenario openstack/full_l2`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/full_l2.yaml

5.1.21 OpenStack L3 East-West

In this scenario Shaker launches pairs of instances, each instance on its own compute node. All available compute nodes are utilized. Instances are connected to one of 2 tenant networks, which plugged into single router. The traffic goes from one network to the other (L3 east-west).

To use this scenario specify parameter `--scenario openstack/full_l3_east_west`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/full_l3_east_west.yaml

5.1.22 OpenStack L3 North-South

In this scenario Shaker launches pairs of instances on different compute nodes. All available compute nodes are utilized. Instances are in different networks connected to different routers, primary accesses minion by floating ip. The traffic goes from one network via external network to the other network.

To use this scenario specify parameter `--scenario openstack/full_l3_north_south`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/full_l3_north_south.yaml

5.1.23 OpenStack L2 Performance

In this scenario Shaker launches 1 pair of instances in the same tenant network. Each instance is hosted on a separate compute node. The traffic goes within the tenant network (L2 domain).

To use this scenario specify parameter `--scenario openstack/perf_l2`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/perf_l2.yaml

5.1.24 OpenStack L3 East-West Performance

In this scenario Shaker launches 1 pair of instances, each instance on its own compute node. Instances are connected to one of 2 tenant networks, which plugged into single router. The traffic goes from one network to the other (L3 east-west).

To use this scenario specify parameter `--scenario openstack/perf_l3_east_west`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/perf_l3_east_west.yaml

5.1.25 OpenStack L3 North-South Performance

In this scenario Shaker launches 1 pair of instances on different compute nodes. Instances are in different networks connected to different routers, primary accesses minion by floating ip. The traffic goes from one network via external network to the other network.

To use this scenario specify parameter `--scenario openstack/perf_l3_north_south`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/perf_l3_north_south.yaml

5.1.26 OpenStack L2 QoS Performance

In this scenario Shaker launches 1 pair of instances in the same tenant network. Each instance is hosted on a separate compute node. The traffic goes within the tenant network (L2 domain). Neutron QoS feature is used to limit traffic throughput to 10 Mbit/s.

To use this scenario specify parameter `--scenario openstack/qos/perf_l2`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/qos/perf_l2.yaml

5.1.27 OpenStack L2 UDP

In this scenario Shaker launches pairs of instances in the same tenant network. Every instance is hosted on a separate compute node. The traffic goes within the tenant network (L2 domain). The load is generated by UDP traffic.

To use this scenario specify parameter `--scenario openstack/udp_l2`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/udp_l2.yaml

5.1.28 OpenStack L3 East-West UDP

In this scenario Shaker launches pairs of instances, each instance on its own compute node. Instances are connected to one of 2 tenant networks, which plugged into single router. The traffic goes from one network to the other (L3 east-west). The load is generated by UDP traffic.

To use this scenario specify parameter `--scenario openstack/udp_l3_east_west`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/udp_l3_east_west.yaml

5.1.29 OpenStack L3 North-South UDP

In this scenario Shaker launches pairs of instances on different compute nodes. Instances are in different networks connected to different routers, primary accesses minion by floating ip. The traffic goes from one network via external network to the other network. The load is generated by UDP traffic.

To use this scenario specify parameter `--scenario openstack/udp_l3_north_south`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/udp_l3_north_south.yaml

5.1.30 Ping

This scenario uses ping to measure the latency between the local host and the remote. The remote host can be provided via command-line, it defaults to 8.8.8.8. The scenario verifies SLA and expects the latency to be at most 30ms. The destination host can be overridden by command-line parameter, e.g. `--matrix "{host: 172.10.1.2}"`.

To use this scenario specify parameter `--scenario spot/ping`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/spot/ping.yaml

5.1.31 TCP bandwidth

This scenario uses iperf3 to measure TCP throughput between local host and ping.online.net (or against hosts provided via CLI). SLA check is verified and expects the speed to be at least 90Mbit and at most 20 retransmitts. The destination host can be overridden by command-line parameter, e.g. `--matrix "{host: 172.10.1.2}"`.

To use this scenario specify parameter `--scenario spot/tcp`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/spot/tcp.yaml

5.1.32 UDP bandwidth

This scenario uses iperf3 to measure UDP throughput between local host and ping.online.net (or against hosts provided via CLI). SLA check is verified and requires at least 10 000 packets per second. The destination host can be overridden by command-line parameter, e.g. `--matrix "{host: 172.10.1.2}"`.

To use this scenario specify parameter `--scenario spot/udp`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/spot/udp.yaml

5.1.33 Sample TCP Test with Advanced Iperf Arguments

This test definition demonstrates the use of advanced arguments with iperf. In this scenario Shaker launches pairs of instances in the same tenant network. Every instance is hosted on a separate compute node, 1 compute node is utilized. The traffic goes within the tenant network (L2 domain) and uses arguments not directly mapped by the iperf executor.

To use this scenario specify parameter `--scenario test/sample_with_advanced_iperf`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/test/sample_with_advanced_iperf.yaml

5.1.34 Sample TCP Test with Environment File

This test definition demonstrates the use of an environment file. In this scenario Shaker launches pairs of instances in the same tenant network. Every instance is hosted on a separate compute node, 1 compute node is utilized. The traffic goes within the tenant network (L2 domain)

To use this scenario specify parameter `--scenario test/sample_with_env`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/test/sample_with_env.yaml

5.1.35 Sample TCP Test with Support Stacks

This test definition demonstrates the use of support stacks In this scenario Shaker launches pairs of instances in the same tenant network. Each test VM is also connected to a previously launched support network. The support neworks are part of their own support heat stack. Every instance is hosted on a separate compute node, 1 compute node is utilized. The traffic goes within the tenant network (L2 domain)

To use this scenario specify parameter `--scenario test/sample_with_support_stacks`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/test/sample_with_support_stacks.yaml

5.1.36 Static agents

In this scenario Shaker runs tests in spot mode. The scenario can be used for Shaker integration testing.

To use this scenario specify parameter `--scenario test/spot`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/test/spot.yaml

5.1.37 Static agents

In this scenario Shaker runs tests on pre-deployed static agents. The scenario can be used for Shaker integration testing.

To use this scenario specify parameter `--scenario test/static_agent`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/test/static_agent.yaml

5.1.38 Paired static agents

In this scenario Shaker runs tests on pre-deployed pair of static agents. The scenario can be used for Shaker integration testing.

To use this scenario specify parameter `--scenario test/static_agents_pair`. Scenario source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/test/static_agents_pair.yaml

5.2 Heat Templates

5.2.1 misc/instance_metadata

Heat template creates a new Neutron network, a router to the external network, plugs instances into this network and assigns floating ips

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/misc/instance_metadata.hot

5.2.2 openstack/cross_az/l2

This Heat template creates a new Neutron network, a router to the external network and plugs instances into this new network. All instances are located in the same L2 domain.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/l2.hot

5.2.3 openstack/cross_az/l3_east_west

This Heat template creates a pair of networks plugged into the same router. Primary instances and minion instances are connected into different networks.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/l3_east_west.hot

5.2.4 openstack/cross_az/l3_north_south

This Heat template creates a new Neutron network plus a north_router to the external network. The template also assigns floating IP addresses to each instance so they are routable from the external network.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/cross_az/l3_north_south.hot

5.2.5 openstack/external/l3_north_south_no_fip

This Heat template creates a new Neutron network plugged into a router connected to the external network, and boots an instance in that network.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/external/l3_north_south_no_fip.hot

5.2.6 openstack/external/l3_north_south_with_fip

This Heat template creates a new Neutron network plugged into a router connected to the external network, and boots an instance in that network. The instance has floating IP.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/external/l3_north_south_with_fip.hot

5.2.7 openstack/l2

This Heat template creates a new Neutron network, a router to the external network and plugs instances into this new network. All instances are located in the same L2 domain.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/l2.hot

5.2.8 openstack/l3_east_west

This Heat template creates a pair of networks plugged into the same router. Primary instances and minion instances are connected into different networks.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/l3_east_west.hot

5.2.9 openstack/l3_north_south

This Heat template creates a new Neutron network plus a north_router to the external network. The template also assigns floating IP addresses to each instance so they are routable from the external network.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/l3_north_south.hot

5.2.10 openstack/qos/l2_qos

This Heat template creates a new Neutron network, a router to the external network and plugs instances into this new network. All instances are located in the same L2 domain.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/openstack/qos/l2_qos.hot

5.2.11 test/l2_with_env

This Heat template creates a new Neutron network, a router to the external network and plugs instances into this new network. All instances are located in the same L2 domain.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/test/l2_with_env.hot

5.2.12 test/templates/l2_with_support

This Heat template creates a new Neutron network, a router to the external network and plugs instances into this new network. All instances are located in the same L2 domain. The VMs are also connected to support networks that should exist before this template is spun up.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/test/templates/l2_with_support.hot

5.2.13 test/templates/support_network

This Heat template creates a new Neutron network. This is used to demonstrate a support stack in Shaker.

Template source is available at: https://opendev.org/performa/shaker/src/branch/master/shaker/scenarios/test/templates/support_network.hot

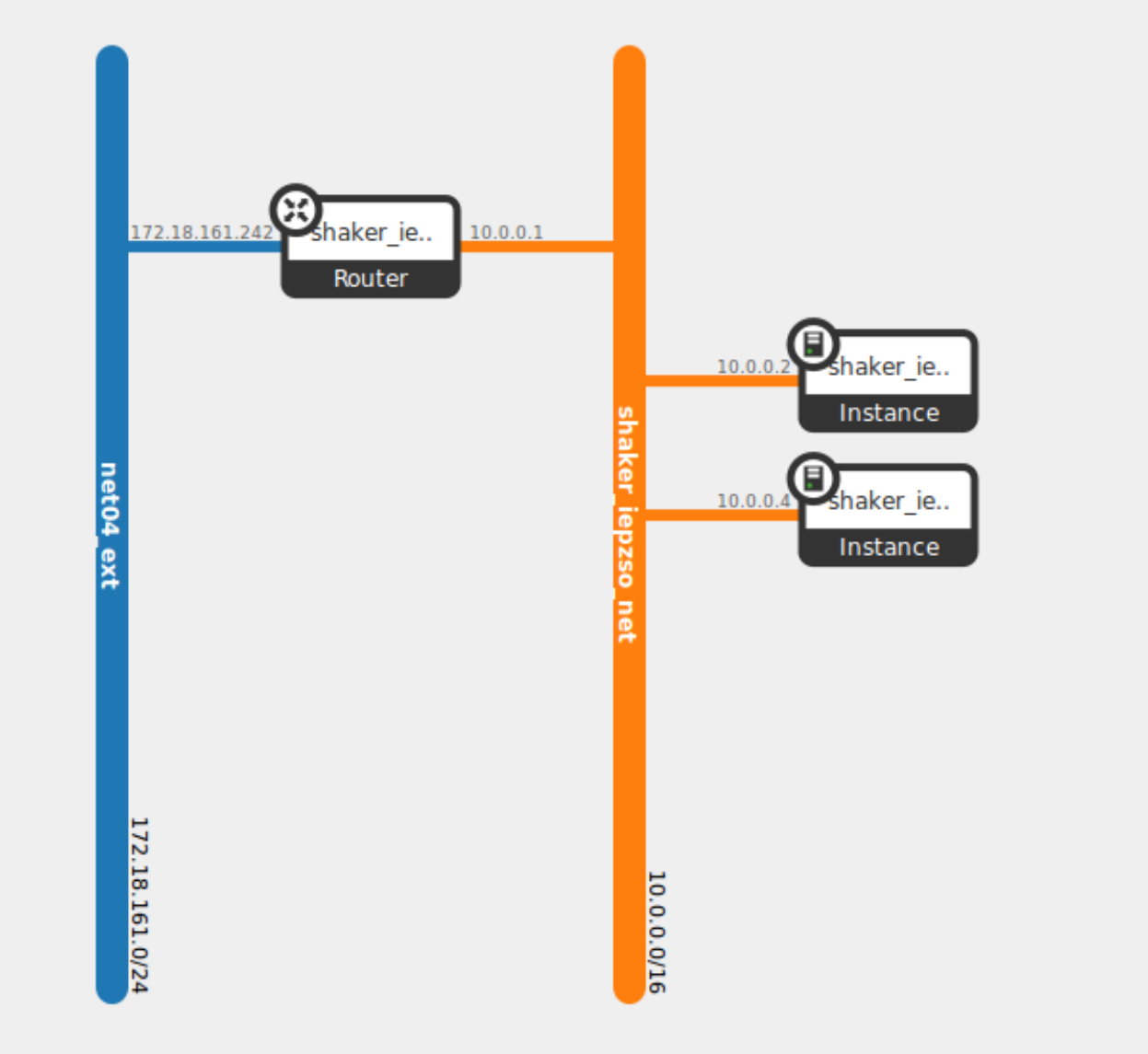
CHAPTER 6

OpenStack Scenarios

This section contains details for the most popular OpenStack scenarios. For the full list of Shaker scenarios please refer to *Scenario Catalog*.

6.1 L2 Same Domain

This scenario tests the bandwidth between pairs of instances in the same virtual network (L2 domain). Each instance is deployed on own compute node. The test increases the load from 1 pair until all available instances are used.



6.1.1 How To Run

```
shaker --server-endpoint <host:port> --scenario openstack/full_l2 --report <full_l2.
↪html>
```

6.1.2 Scenario

```
title: OpenStack L2

description:
  In this scenario Shaker launches pairs of instances in the same tenant
  network. Every instance is hosted on a separate compute node, all available
  compute nodes are utilized. The traffic goes within the tenant network
  (L2 domain).
```

(continues on next page)

(continued from previous page)

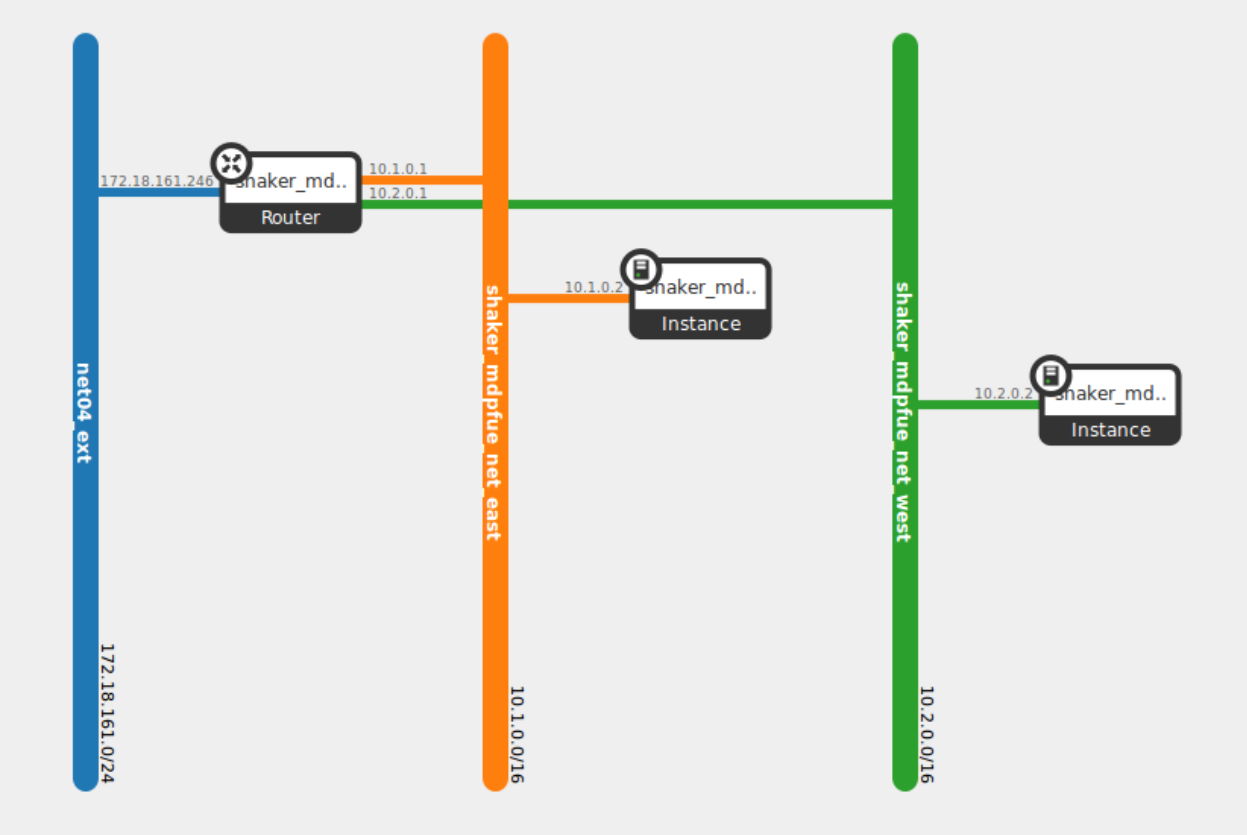
```
deployment:
  template: l2.hot
  accommodation: [pair, single_room]

execution:
  progression: quadratic
  tests:
  -
    title: Download
    class: flent
    method: tcp_download
  -
    title: Upload
    class: flent
    method: tcp_upload
  -
    title: Bi-directional
    class: flent
    method: tcp_bidirectional
```

6.1.3 Report

Example report collected at 20-nodes OpenStack cluster: *OpenStack L2*.

6.2 L3 East-West

This scenario tests the bandwidth between pairs of instances deployed in different virtual networks plugged into the same router. Each instance is deployed on its own compute node. The test increases the load from 1 pair pair until all available instances are used.



6.2.1 How To Run

```
shaker --server-endpoint <host:port> --scenario openstack/full_l3_east_west --report
↪<full_l3_east_west.html>
```

6.2.2 Scenario

```
title: OpenStack L3 East-West

description:
  In this scenario Shaker launches pairs of instances, each instance on its own
  compute node. All available compute nodes are utilized. Instances are
  connected to one of 2 tenant networks, which plugged into single router.
  The traffic goes from one network to the other (L3 east-west).

deployment:
  template: l3_east_west.hot
  accommodation: [pair, single_room]

execution:
  progression: quadratic
  tests:
  -
    title: Download
    class: flent
```

(continues on next page)

(continued from previous page)

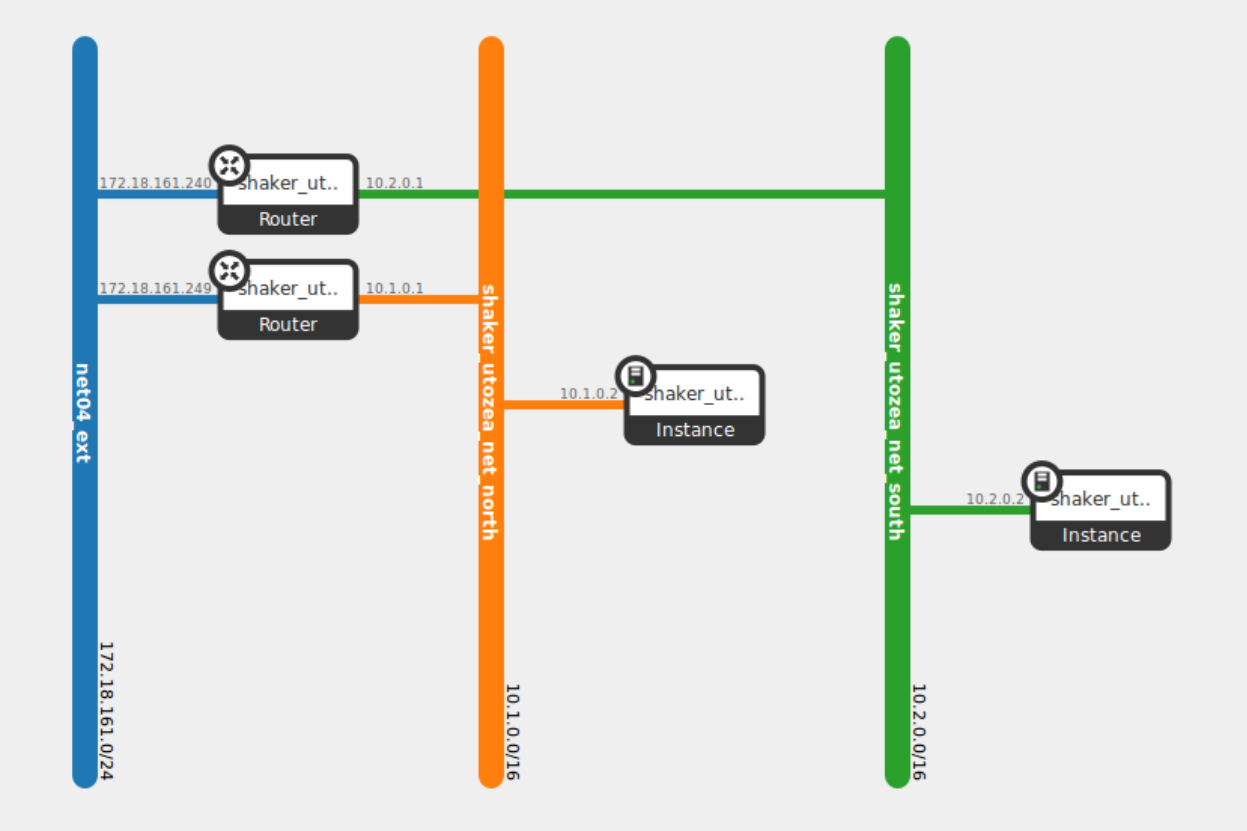
```
    method: tcp_download
  -
    title: Upload
    class: flent
    method: tcp_upload
  -
    title: Bi-directional
    class: flent
    method: tcp_bidirectional
```

6.2.3 Report

Example report collected at 20-nodes OpenStack cluster: *OpenStack L3 East-West*.

6.3 L3 North-South

This scenario tests the bandwidth between pairs of instances deployed in different virtual networks. Instances with primary agents are located in one network, instances with minion agents are reached via their floating IPs. Each instance is deployed on its own compute node. The test increases the load from 1 pair pair until all available instances are used.



6.3.1 How To Run

```
shaker --server-endpoint <host:port> --scenario networkingfull_l3_north_south --
↪report <full_l3_north_south.html>
```

6.3.2 Scenario

```
title: OpenStack L3 North-South

description:
  In this scenario Shaker launches pairs of instances on different compute
  nodes. All available compute nodes are utilized. Instances are in different
  networks connected to different routers, primary accesses minion by
  floating ip. The traffic goes from one network via external network to
  the other network.

deployment:
  template: l3_north_south.hot
  accommodation: [pair, single_room]

execution:
  progression: quadratic
  tests:
  -
    title: Download
    class: flent
    method: tcp_download
  -
    title: Upload
    class: flent
    method: tcp_upload
  -
    title: Bi-directional
    class: flent
    method: tcp_bidirectional
```

6.3.3 Report

Example report collected at 20-nodes OpenStack cluster: *OpenStack L3 North-South*.

CHAPTER 7

Spot Scenarios

Spot scenarios are executed between the local machine (where shaker runs) and the remote. Local machine must have all necessary tools installed, e.g. the following scenarios require iperf3 and flent utilities.

7.1 TCP

This scenario tests TCP bandwidth to the destination host. By default it sends traffic to one of public iperf3 servers. This can be overridden via parameter `--matrix "{host:<host>}"`. The scenario requires iperf3 to be installed locally.

7.1.1 How To Run

1. Run the scenario with defaults and generate interactive report into file `report.html`:

```
shaker-spot --scenario spot/tcp --report report.html
```

2. Run the scenario with overridden target host (10.0.0.2) and store raw result:

```
shaker-spot --scenario spot/tcp --matrix "{host:10.0.0.2}" --output report.
↪json
```

3. Run the scenario with overridden target host (10.0.0.2) and store SLA verification results in subunit stream file:

```
shaker-spot --scenario spot/tcp --matrix "{host:10.0.0.2}" --subunit report.
↪subunit
```

4. Run the scenario against the list of target hosts and store report:

```
shaker-spot --scenario spot/tcp --matrix "{host:[10.0.0.2, 10.0.0.3]}" --
↪output report.html
```

7.1.2 Scenario

```
title: TCP bandwidth

description: >
  This scenario uses iperf3 to measure TCP throughput between local host and
  ping.online.net (or against hosts provided via CLI). SLA check is verified
  and expects the speed to be at least 90Mbit and at most 20 retransmitts.
  The destination host can be overridden by command-line parameter,
  e.g. ``--matrix "{host: 172.10.1.2}"``.

execution:
  tests:
  -
    title: TCP
    class: iperf3
    host: ping.online.net
    time: 20
    sla:
    - "[type == 'agent'] >> (stats.bandwidth.avg > 90)"
    - "[type == 'agent'] >> (stats.retransmits.max < 20)"
```

7.2 UDP

This scenario tests UDP packets per second to the destination host. By default it sends traffic to one of public iperf3 servers. This can be overridden via parameter `--matrix "{host:<host>}"`. The scenario requires iperf3 to be installed locally.

7.2.1 How To Run

```
shaker-spot --scenario spot/udp --report report.html
```

7.2.2 Scenario

```
title: UDP bandwidth

description: >
  This scenario uses iperf3 to measure UDP throughput between local host and
  ping.online.net (or against hosts provided via CLI). SLA check is verified
  and requires at least 10 000 packets per second.
  The destination host can be overridden by command-line parameter,
  e.g. ``--matrix "{host: 172.10.1.2}"``.

execution:
  tests:
  -
    title: UDP
    class: iperf3
    host: ping.online.net
    udp: on
    time: 20
```

(continues on next page)

(continued from previous page)

```
    bandwidth: 1000M
    sla:
    - "[type == 'agent'] >> (stats.packets.avg > 10000)"
```

7.3 Ping

This scenario tests ICMP ping between the local machine and the remote. By default pings are sent to public 8.8.8.8 address. The remote address can be overridden via parameter `--matrix "{host: <host>}"`. The scenario requires flent to be installed locally.

7.3.1 How To Run

```
shaker-spot --scenario spot/ping --report report.html
```

7.3.2 Scenario

```
title: Ping

description: >
  This scenario uses ping to measure the latency between the local host and
  the remote. The remote host can be provided via command-line, it defaults
  to 8.8.8.8. The scenario verifies SLA and expects the latency to be at most
  30ms.
  The destination host can be overridden by command-line parameter,
  e.g. ``--matrix "{host: 172.10.1.2}"``.

execution:
  tests:
  -
    title: Ping
    class: flent
    host: 8.8.8.8
    method: ping
    time: 10
    sla:
    - "[type == 'agent'] >> (stats.ping_icmp.avg < 30)"
```

CHAPTER 8

Reports

All reports under this folder are collected in the following environment:

- 20 bare-metal nodes running KVM
- 10Gb tenant network
- 1Gb floating network
- Neutron ML2 plugin with VXLAN
- Neutron HA routers

To generate the report based on raw data:

```
shaker-report --input <raw data> --book <folder to store book into>
```

8.1 OpenStack L2

This scenario launches pairs of VMs in the same private network. Every VM is hosted on a separate compute node.

Scenario:

```
deployment:
  accommodation:
  - pair
  - single_room
  template: l2.hot
description: This scenario launches pairs of VMs in the same private network. Every
  VM is hosted on a separate compute node.
execution:
  progression: quadratic
  tests:
  - class: flent
    method: tcp_download
```

(continues on next page)

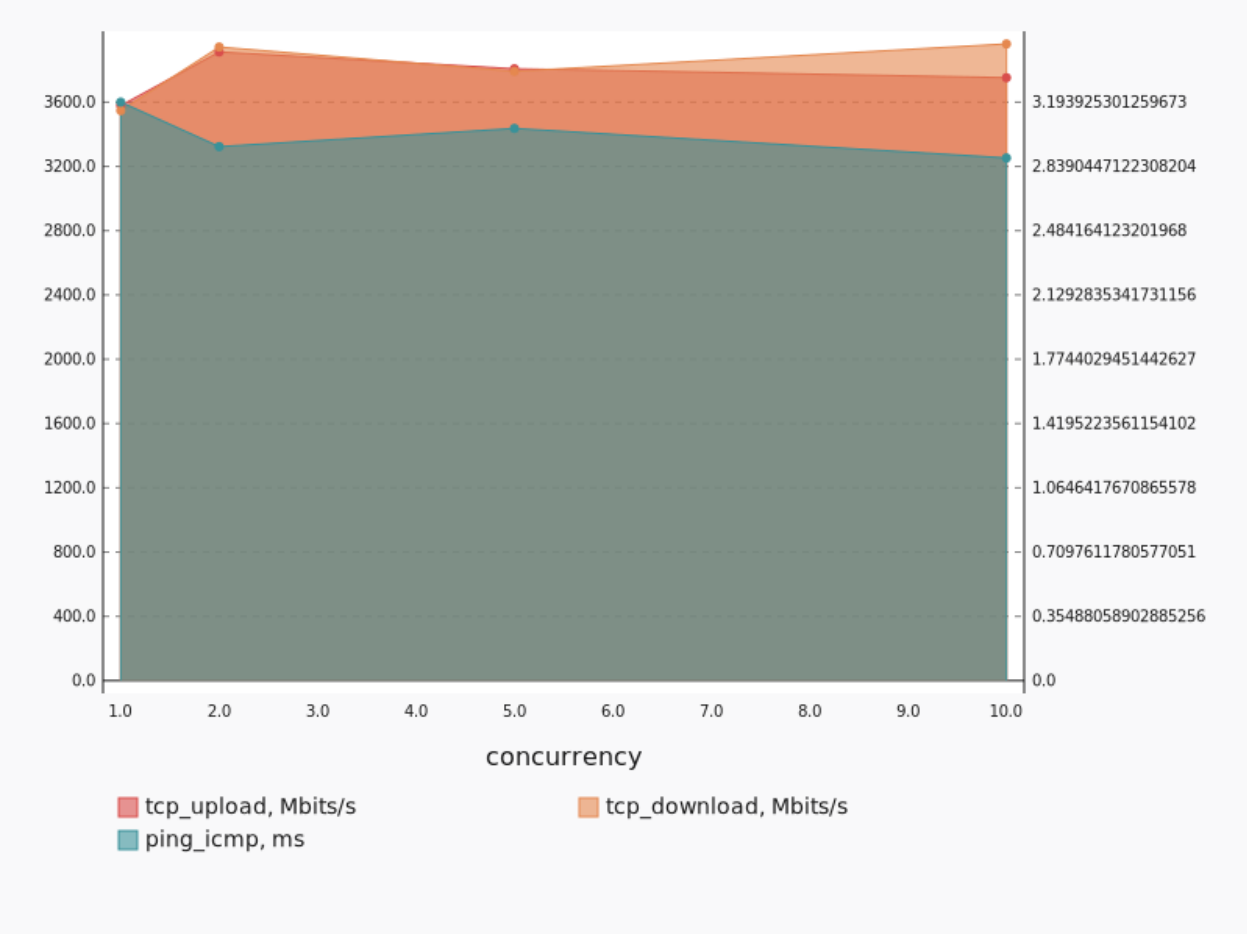
(continued from previous page)

```
    title: Download
  - class: flent
    method: tcp_upload
    title: Upload
  - class: flent
    method: tcp_bidirectional
    title: Bi-directional
file_name: /home/ishakhat/Work/shaker/shaker/scenarios/openstack/full_l2.yaml
title: OpenStack L2
```

8.1.1 Bi-directional

Test Specification:

```
class: flent
method: tcp_bidirectional
title: Bi-directional
```



Stats:

concurrency	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
1	3578.59	3.19	3547.75
2	3912.17	2.95	3942.74
5	3807.46	3.05	3791.78
10	3752.25	2.89	3962.02

Concurrency 1

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-8.domain.tld	3578.59	3.19	3547.75

Concurrency 2

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-7.domain.tld	3680.14	3.21	3711.57
node-8.domain.tld	4144.20	2.68	4173.90

Concurrency 5

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-11.domain.tld	3551.67	3.33	3544.93
node-18.domain.tld	3795.47	3.04	3811.38
node-4.domain.tld	3898.52	3.00	3882.67
node-7.domain.tld	3970.07	2.82	4005.72
node-8.domain.tld	3821.60	3.04	3714.18

Concurrency 10

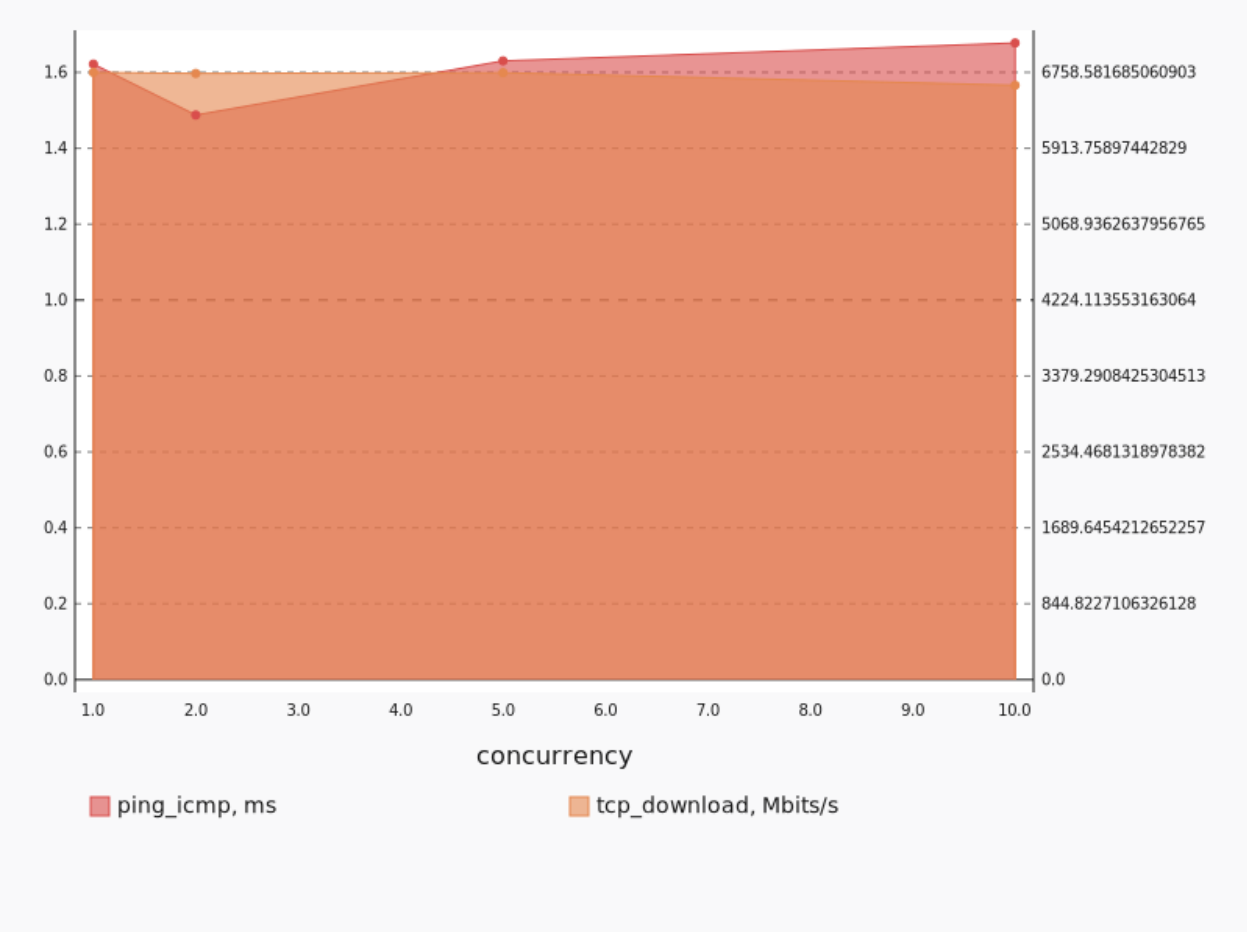
Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-11.domain.tld	4014.04	2.85	3878.48
node-13.domain.tld	3767.26	3.24	3651.51
node-15.domain.tld	3316.62	2.96	3861.89
node-17.domain.tld	3330.25	2.88	4175.01
node-18.domain.tld	4208.58	2.74	3639.62
node-20.domain.tld	3988.34	2.74	4112.45
node-4.domain.tld	3939.45	3.08	4057.85
node-5.domain.tld	3846.78	3.01	3784.39
node-7.domain.tld	3390.47	2.38	4657.64
node-8.domain.tld	3720.68	2.98	3801.36

8.1.2 Download

Test Specification:

```
class: flent
method: tcp_download
title: Download
```



Stats:

concurrency	ping_icmp, ms	tcp_download, Mbits/s
1	1.62	6758.58
2	1.49	6747.02
5	1.63	6755.12
10	1.68	6615.10

Concurrency 1

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-8.domain.tld	1.62	6758.58

Concurrency 2

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-7.domain.tld	1.50	6771.23
node-8.domain.tld	1.47	6722.80

Concurrency 5

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-11.domain.tld	1.52	6650.81
node-18.domain.tld	1.70	6870.23
node-4.domain.tld	1.74	6688.20
node-7.domain.tld	1.57	6741.27
node-8.domain.tld	1.63	6825.11

Concurrency 10

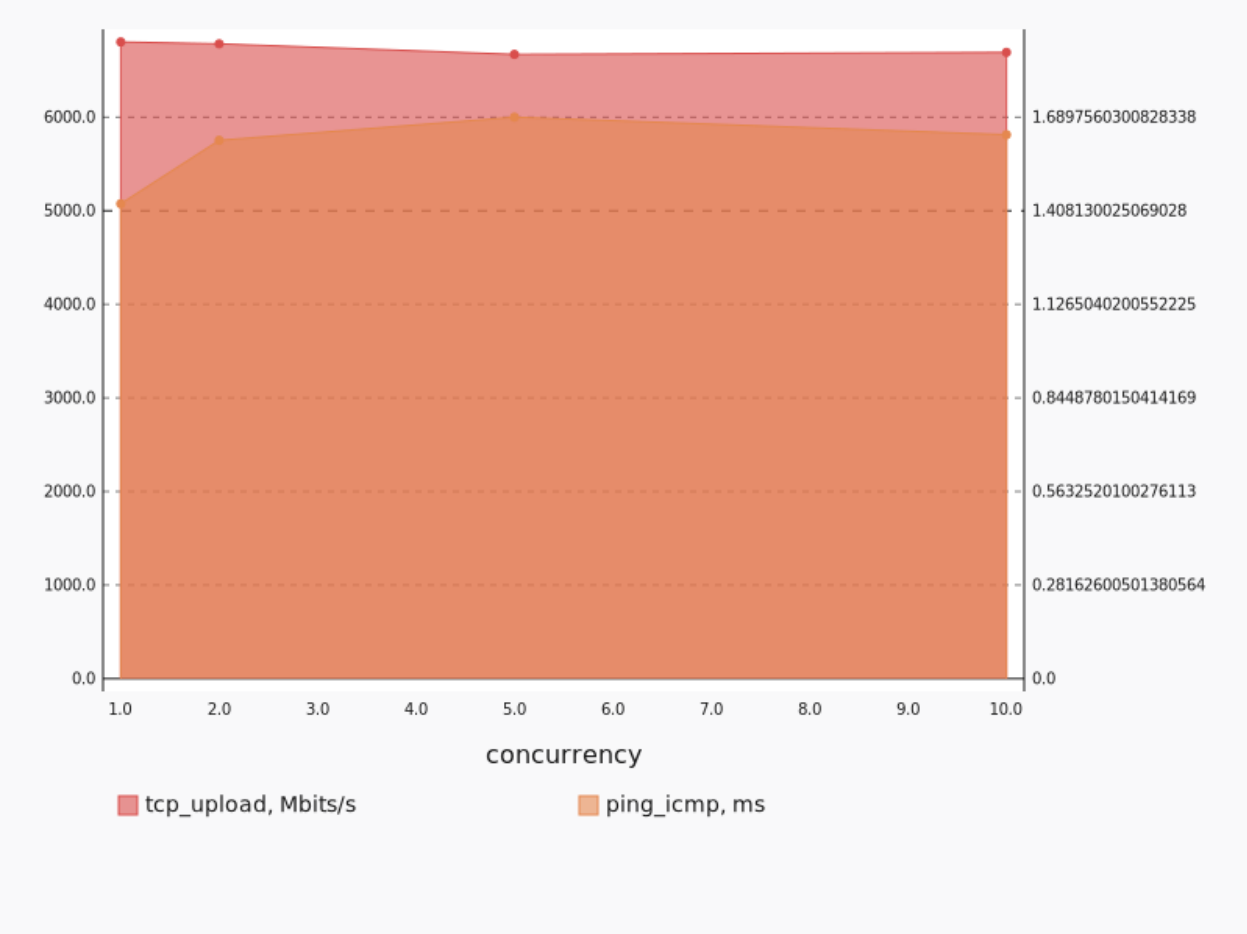
Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-11.domain.tld	1.43	6634.04
node-13.domain.tld	1.67	6769.58
node-15.domain.tld	1.60	6695.55
node-17.domain.tld	2.17	6145.54
node-18.domain.tld	1.64	6824.41
node-20.domain.tld	1.69	6786.08
node-4.domain.tld	1.70	6754.63
node-5.domain.tld	1.68	6572.60
node-7.domain.tld	1.80	6228.16
node-8.domain.tld	1.41	6740.39

8.1.3 Upload

Test Specification:

```
class: flent
method: tcp_upload
title: Upload
```



Stats:

concurrency	tcp_upload, Mbits/s	ping_icmp, ms
1	6804.07	1.43
2	6784.08	1.62
5	6671.28	1.69
10	6692.88	1.64

Concurrency 1

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-8.domain.tld	6804.07	1.43

Concurrency 2

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-7.domain.tld	6708.61	1.63
node-8.domain.tld	6859.54	1.61

Concurrency 5

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-11.domain.tld	6442.30	1.78
node-18.domain.tld	6514.95	1.47
node-4.domain.tld	7005.11	1.79
node-7.domain.tld	6682.03	1.58
node-8.domain.tld	6711.99	1.83

Concurrency 10

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-11.domain.tld	6701.87	1.75
node-13.domain.tld	6777.32	1.64
node-15.domain.tld	6620.17	1.68
node-17.domain.tld	6469.74	1.52
node-18.domain.tld	6709.92	1.65
node-20.domain.tld	6686.77	1.62
node-4.domain.tld	6687.55	1.55
node-5.domain.tld	6896.79	1.62
node-7.domain.tld	6686.20	1.58
node-8.domain.tld	6692.50	1.75

8.2 OpenStack L3 East-West

This scenario launches pairs of VMs in different networks connected to one router (L3 east-west)

Scenario:

```
deployment:
  accommodation:
  - pair
  - single_room
  template: l3_east_west.hot
description: This scenario launches pairs of VMs in different networks connected to
  one router (L3 east-west)
execution:
  progression: quadratic
  tests:
  - class: flent
    method: tcp_download
```

(continues on next page)

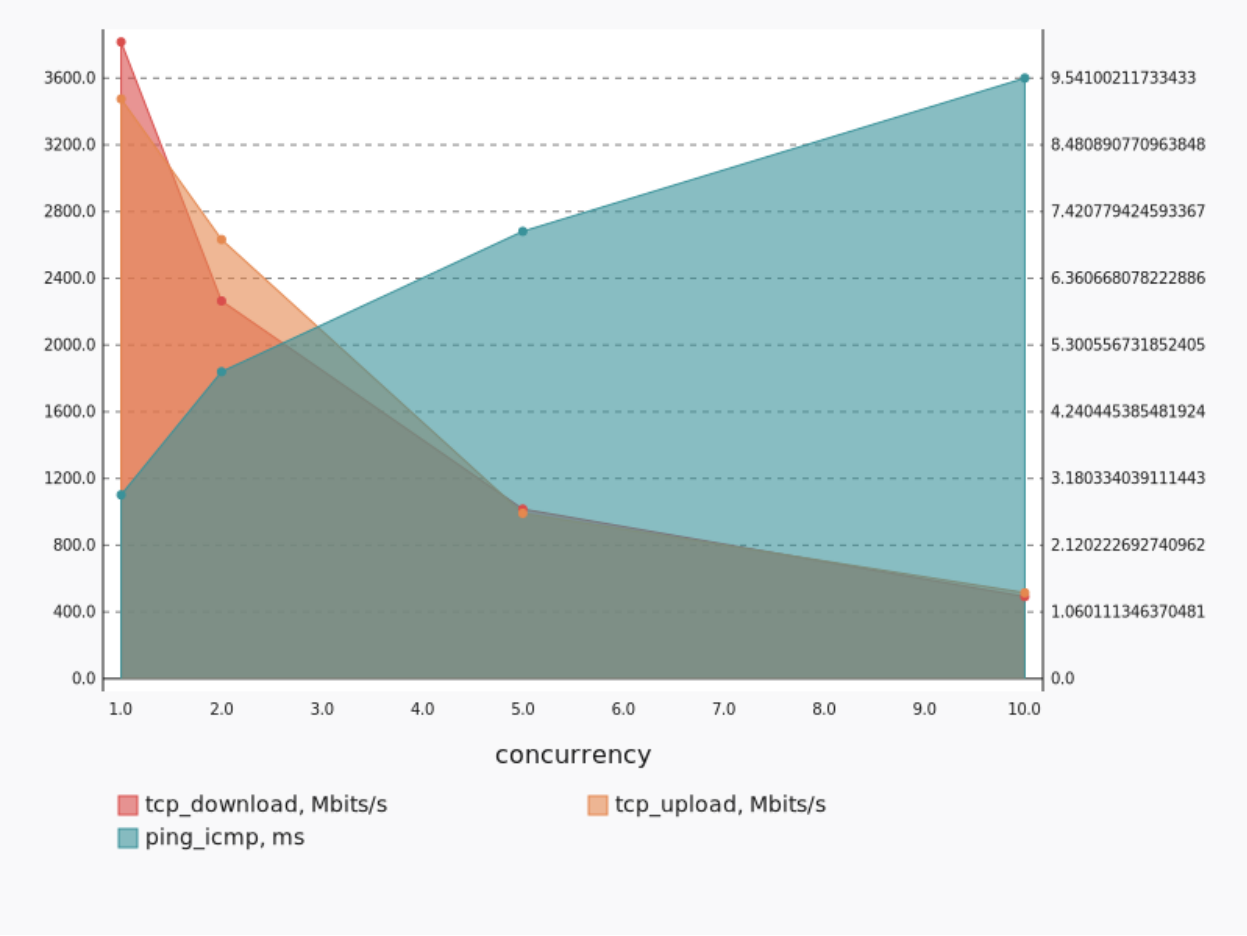
(continued from previous page)

```
    title: Download
  - class: flent
    method: tcp_upload
    title: Upload
  - class: flent
    method: tcp_bidirectional
    title: Bi-directional
file_name: /home/ishakhat/Work/shaker/shaker/scenarios/openstack/full_l3_east_west.
↪yaml
title: OpenStack L3 East-West
```

8.2.1 Bi-directional

Test Specification:

```
class: flent
method: tcp_bidirectional
title: Bi-directional
```



Stats:

concurrency	tcp_download, Mbits/s	tcp_upload, Mbits/s	ping_icmp, ms
1	3816.62	3474.93	2.92
2	2264.43	2632.60	4.88
5	1016.47	991.04	7.11
10	491.52	514.84	9.54

Concurrency 1

Stats:

node	tcp_download, Mbits/s	tcp_upload, Mbits/s	ping_icmp, ms
node-13.domain.tld	3816.62	3474.93	2.92

Concurrency 2

Stats:

node	tcp_download, Mbits/s	tcp_upload, Mbits/s	ping_icmp, ms
node-13.domain.tld	2423.30	2639.19	6.56
node-15.domain.tld	2105.57	2626.00	3.20

Concurrency 5

Stats:

node	tcp_download, Mbits/s	tcp_upload, Mbits/s	ping_icmp, ms
node-13.domain.tld	971.69	839.75	10.07
node-15.domain.tld	1490.89	948.82	6.33
node-20.domain.tld	758.93	889.14	5.69
node-4.domain.tld	786.01	1125.13	7.69
node-5.domain.tld	1074.82	1152.35	5.75

Concurrency 10

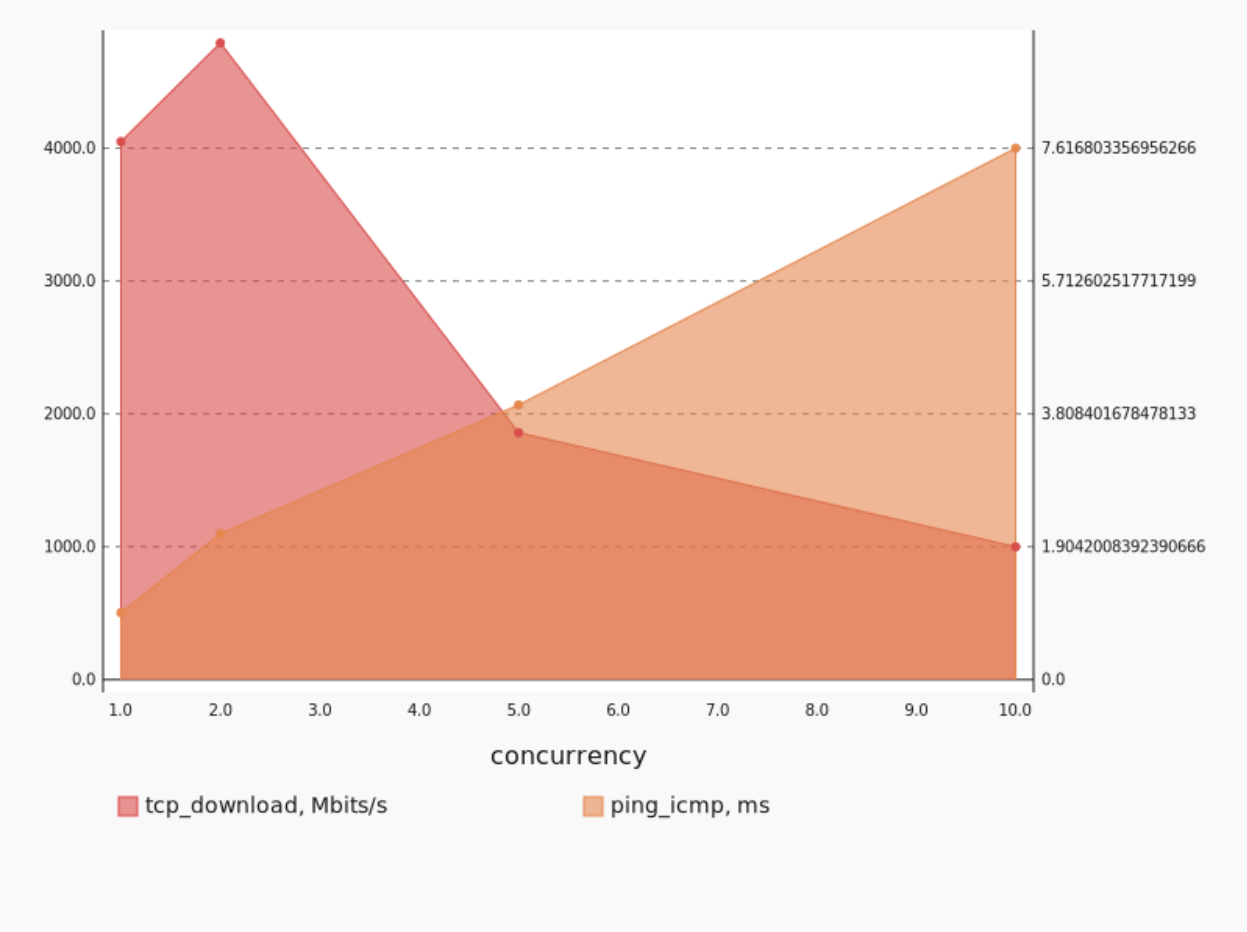
Stats:

node	tcp_download, Mbits/s	tcp_upload, Mbits/s	ping_icmp, ms
node-11.domain.tld	752.08	763.63	9.52
node-13.domain.tld	320.14	935.47	13.50
node-15.domain.tld	354.13	506.37	5.85
node-17.domain.tld	902.35	346.84	13.27
node-18.domain.tld	790.13	358.23	13.42
node-20.domain.tld	378.52	360.62	5.99
node-4.domain.tld	346.47	437.56	9.38
node-5.domain.tld	367.27	706.91	5.70
node-7.domain.tld	347.72	392.19	9.47
node-8.domain.tld	356.42	340.56	9.33

8.2.2 Download

Test Specification:

```
class: flent
method: tcp_download
title: Download
```



Stats:

concurrency	tcp_download, Mbits/s	ping_icmp, ms
1	4049.22	0.96
2	4792.05	2.09
5	1858.96	3.94
10	999.79	7.62

Concurrency 1

Stats:

node	tcp_download, Mbits/s	ping_icmp, ms
node-13.domain.tld	4049.22	0.96

Concurrency 2

Stats:

node	tcp_download, Mbits/s	ping_icmp, ms
node-13.domain.tld	5126.86	2.81
node-15.domain.tld	4457.24	1.38

Concurrency 5

Stats:

node	tcp_download, Mbits/s	ping_icmp, ms
node-13.domain.tld	1475.56	4.33
node-15.domain.tld	1486.69	7.91
node-20.domain.tld	2385.87	2.15
node-4.domain.tld	2470.58	3.87
node-5.domain.tld	1476.10	1.42

Concurrency 10

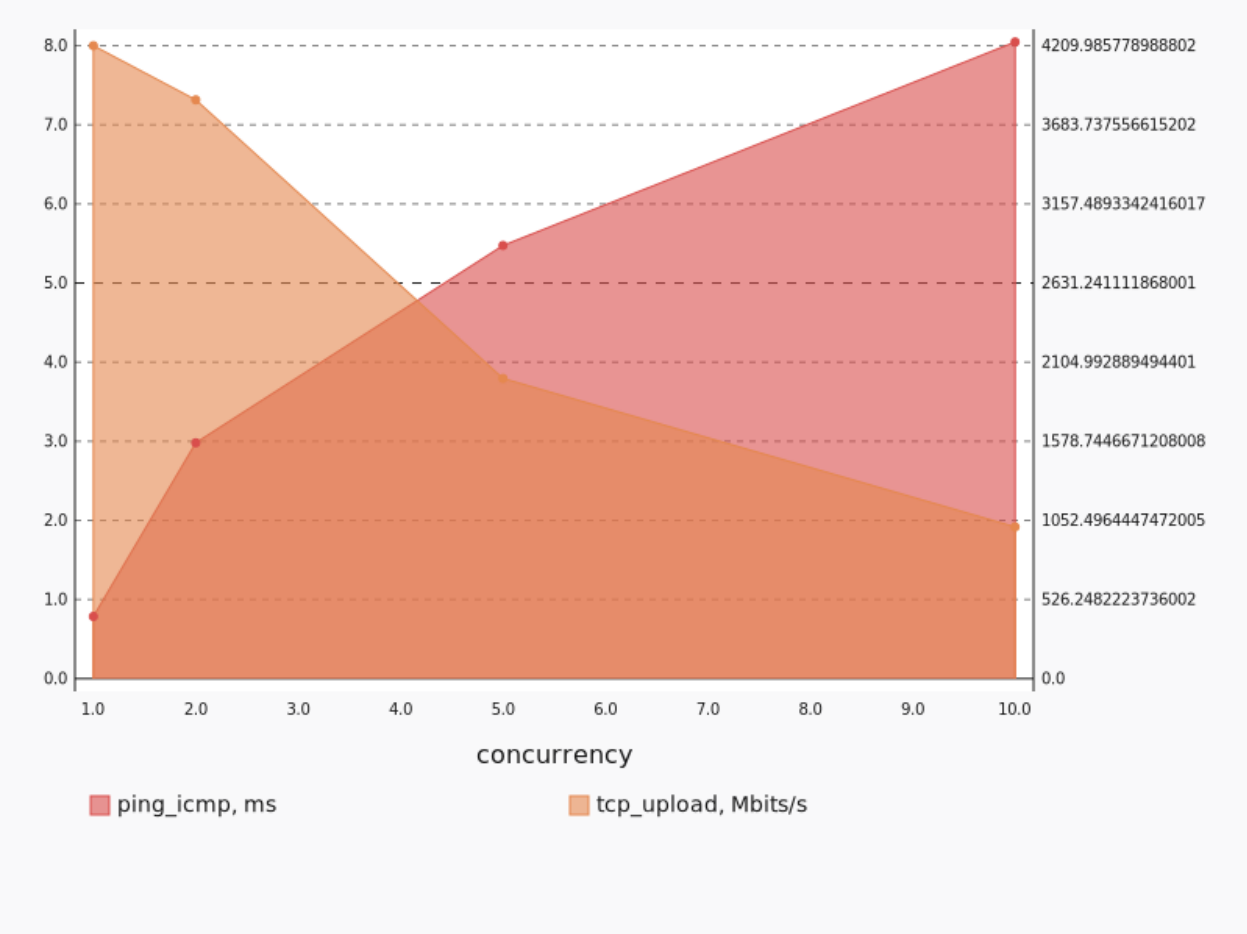
Stats:

node	tcp_download, Mbits/s	ping_icmp, ms
node-11.domain.tld	842.15	7.68
node-13.domain.tld	1180.86	8.50
node-15.domain.tld	1496.95	6.76
node-17.domain.tld	1018.10	8.80
node-18.domain.tld	979.22	8.77
node-20.domain.tld	893.75	6.47
node-4.domain.tld	846.17	7.52
node-5.domain.tld	822.03	6.59
node-7.domain.tld	866.79	7.42
node-8.domain.tld	1051.91	7.65

8.2.3 Upload

Test Specification:

```
class: flent
method: tcp_upload
title: Upload
```



Stats:

concurrency	tcp_upload, Mbits/s	ping_icmp, ms
1	4209.99	0.79
2	3849.74	2.98
5	1996.74	5.47
10	1009.21	8.05

Concurrency 1

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-13.domain.tld	4209.99	0.79

Concurrency 2

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-13.domain.tld	4086.94	2.07
node-15.domain.tld	3612.54	3.89

Concurrency 5

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-13.domain.tld	2053.60	9.05
node-15.domain.tld	1525.48	3.71
node-20.domain.tld	1463.32	3.94
node-4.domain.tld	3485.97	6.73
node-5.domain.tld	1455.31	3.96

Concurrency 10

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-11.domain.tld	830.32	8.19
node-13.domain.tld	720.02	11.14
node-15.domain.tld	807.43	4.96
node-17.domain.tld	956.33	11.02
node-18.domain.tld	926.50	11.21
node-20.domain.tld	1272.34	5.04
node-4.domain.tld	1371.94	8.07
node-5.domain.tld	1306.22	4.91
node-7.domain.tld	906.63	7.85
node-8.domain.tld	994.41	8.08

8.3 OpenStack L3 North-South

This scenario launches pairs of VMs on different compute nodes. VMs are in the different networks connected via different routers, primary accesses minion by floating ip

Scenario:

```
deployment:
  accommodation:
  - pair
  - single_room
  template: l3_north_south.hot
description: This scenario launches pairs of VMs on different compute nodes. VMs are
  in the different networks connected via different routers, primary accesses minion
  by floating ip
execution:
  progression: quadratic
  tests:
```

(continues on next page)

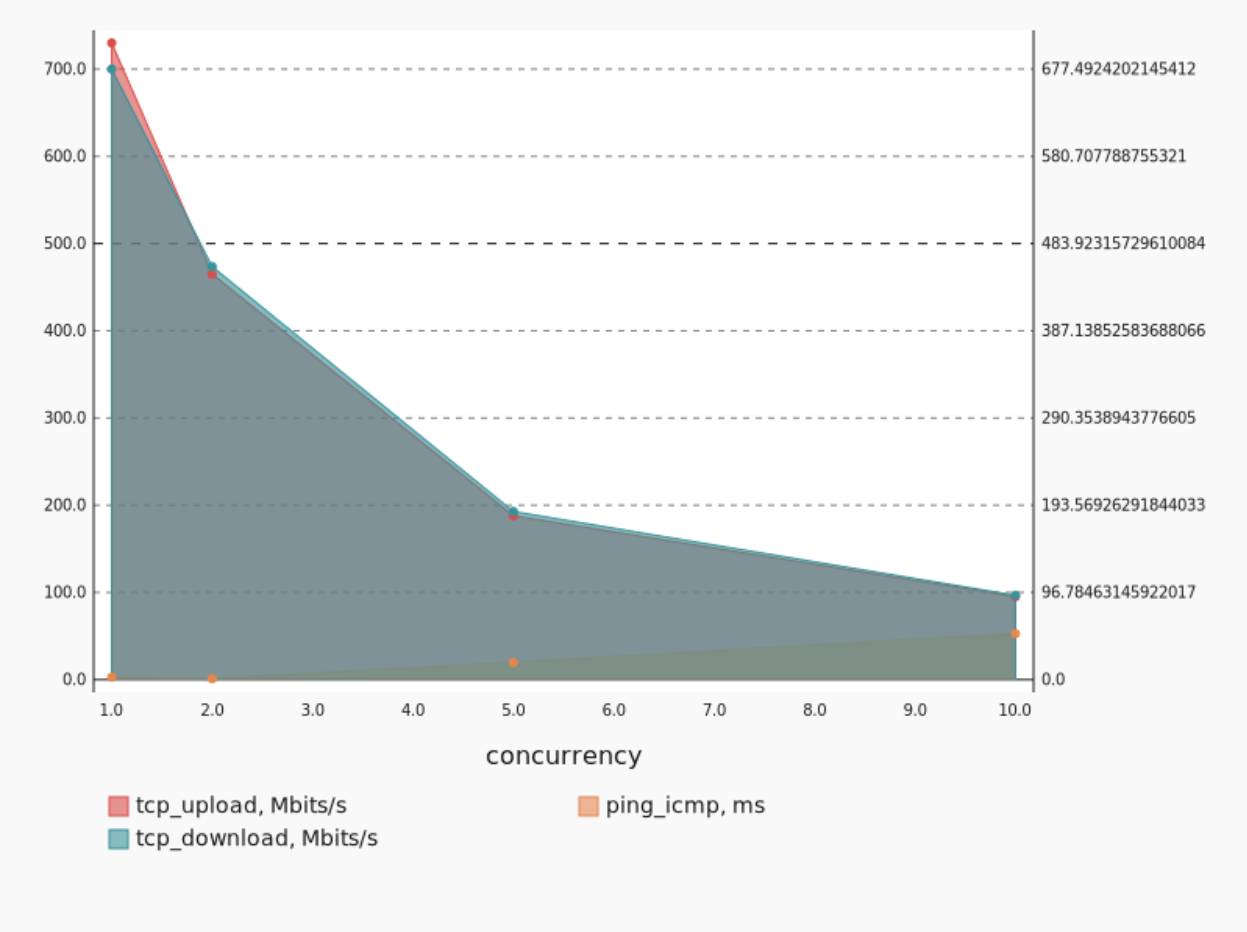
(continued from previous page)

```
  - class: flent
    method: tcp_download
    title: Download
  - class: flent
    method: tcp_upload
    title: Upload
  - class: flent
    method: tcp_bidirectional
    title: Bi-directional
file_name: /home/ishakhat/Work/shaker/shaker/scenarios/openstack/full_l3_north_south.
↪yaml
title: OpenStack L3 North-South
```

8.3.1 Bi-directional

Test Specification:

```
class: flent
method: tcp_bidirectional
title: Bi-directional
```



Stats:

concurrency	tcp_download, Mbits/s	tcp_upload, Mbits/s	ping_icmp, ms
1	677.49	730.02	2.83
2	458.31	464.96	1.10
5	186.56	188.01	19.69
10	93.53	95.16	52.70

Concurrency 1

Stats:

node	tcp_download, Mbits/s	tcp_upload, Mbits/s	ping_icmp, ms
node-7.domain.tld	677.49	730.02	2.83

Concurrency 2

Stats:

node	tcp_download, Mbits/s	tcp_upload, Mbits/s	ping_icmp, ms
node-7.domain.tld	463.71	358.63	1.17
node-8.domain.tld	452.91	571.29	1.04

Concurrency 5

Stats:

node	tcp_download, Mbits/s	tcp_upload, Mbits/s	ping_icmp, ms
node-17.domain.tld	131.38	126.00	1.17
node-18.domain.tld	174.60	248.76	23.30
node-4.domain.tld	218.45	174.13	48.85
node-7.domain.tld	252.50	247.47	1.25
node-8.domain.tld	155.87	143.68	23.88

Concurrency 10

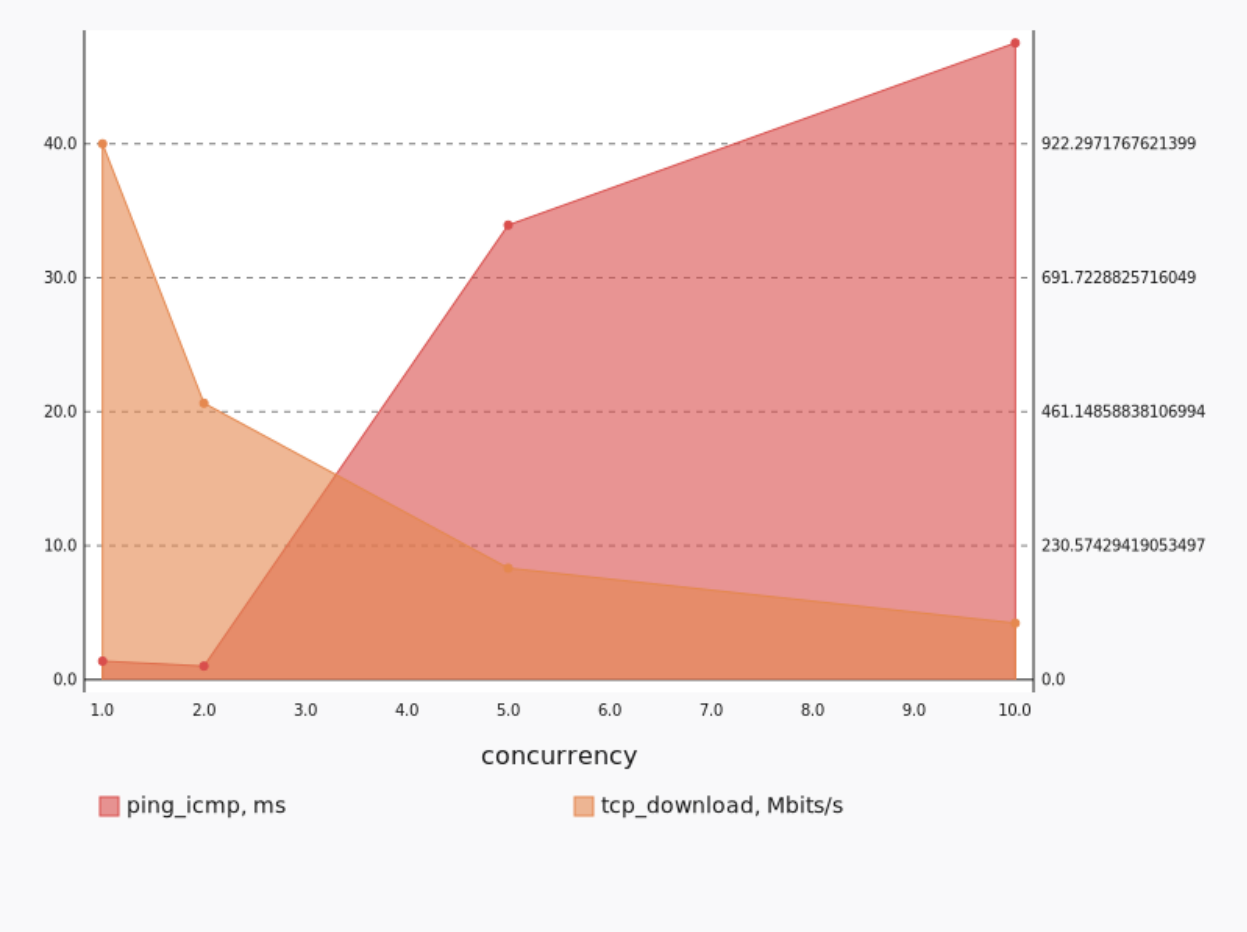
Stats:

node	tcp_download, Mbits/s	tcp_upload, Mbits/s	ping_icmp, ms
node-11.domain.tld	70.72	105.89	32.10
node-13.domain.tld	41.09	87.66	58.91
node-15.domain.tld	50.97	66.22	49.67
node-17.domain.tld	134.96	107.46	49.53
node-18.domain.tld	195.38	73.91	57.20
node-20.domain.tld	47.33	109.20	64.02
node-4.domain.tld	93.19	130.02	69.01
node-5.domain.tld	160.04	84.94	36.94
node-7.domain.tld	80.14	53.36	50.13
node-8.domain.tld	61.44	132.92	59.52

8.3.2 Download

Test Specification:

```
class: flent
method: tcp_download
title: Download
```



Stats:

concurrency	tcp_download, Mbits/s	ping_icmp, ms
1	922.30	1.38
2	475.85	1.01
5	191.92	33.93
10	97.23	47.53

Concurrency 1

Stats:

node	tcp_download, Mbits/s	ping_icmp, ms
node-7.domain.tld	922.30	1.38

Concurrency 2

Stats:

node	tcp_download, Mbits/s	ping_icmp, ms
node-7.domain.tld	472.46	1.12
node-8.domain.tld	479.23	0.91

Concurrency 5

Stats:

node	tcp_download, Mbits/s	ping_icmp, ms
node-17.domain.tld	192.51	39.78
node-18.domain.tld	189.76	41.85
node-4.domain.tld	189.54	45.34
node-7.domain.tld	189.81	41.66
node-8.domain.tld	198.01	1.04

Concurrency 10

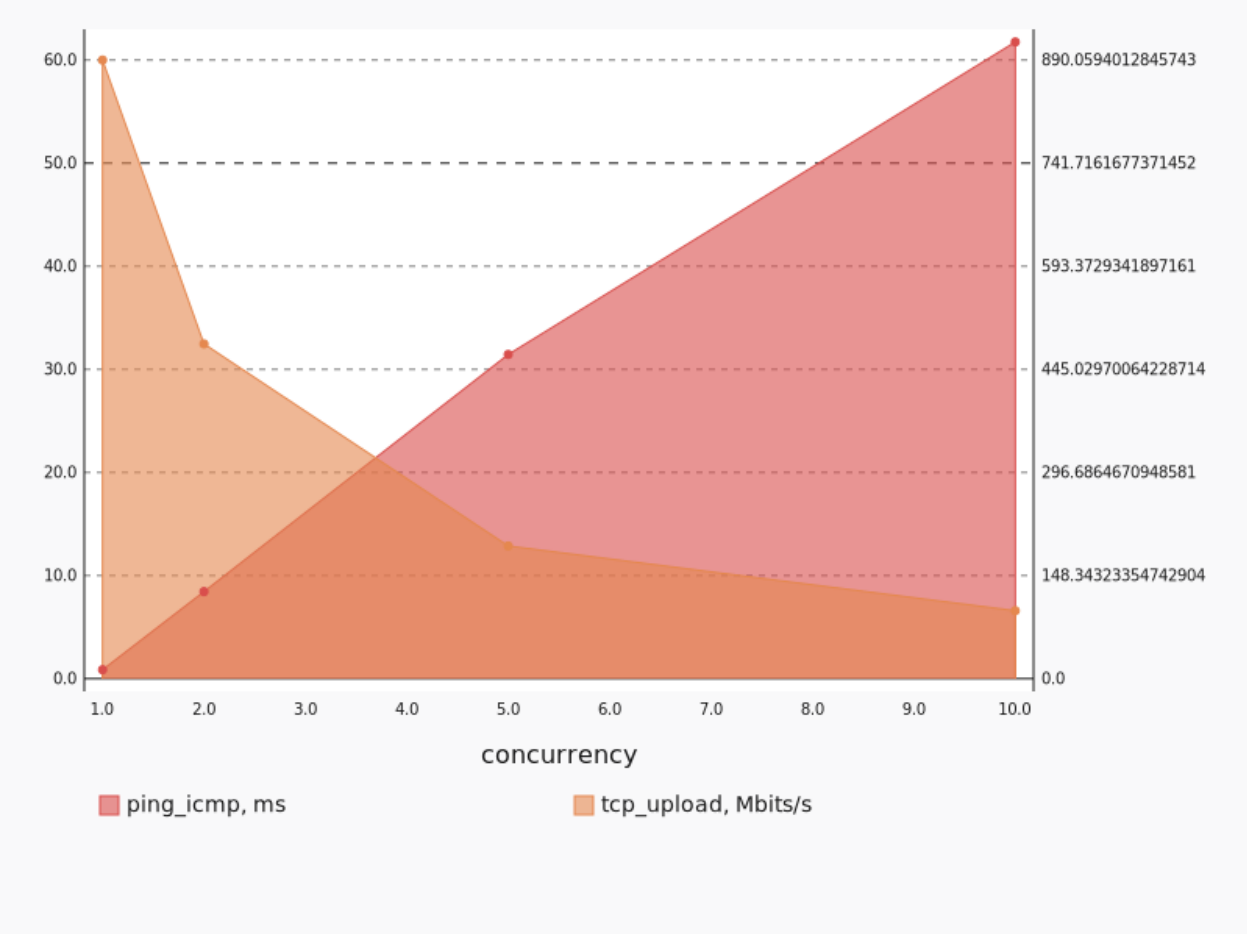
Stats:

node	tcp_download, Mbits/s	ping_icmp, ms
node-11.domain.tld	161.82	50.27
node-13.domain.tld	66.99	51.33
node-15.domain.tld	83.39	54.02
node-17.domain.tld	62.38	54.22
node-18.domain.tld	77.17	54.20
node-20.domain.tld	51.60	54.22
node-4.domain.tld	97.86	50.46
node-5.domain.tld	53.75	0.98
node-7.domain.tld	158.17	54.30
node-8.domain.tld	159.16	51.26

8.3.3 Upload

Test Specification:

```
class: flent
method: tcp_upload
title: Upload
```



Stats:

concurrency	tcp_upload, Mbits/s	ping_icmp, ms
1	890.06	0.86
2	481.63	8.44
5	190.86	31.44
10	97.73	61.75

Concurrency 1

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-7.domain.tld	890.06	0.86

Concurrency 2

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-7.domain.tld	476.55	0.75
node-8.domain.tld	486.72	16.13

Concurrency 5

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-17.domain.tld	192.28	41.43
node-18.domain.tld	190.41	0.87
node-4.domain.tld	189.01	38.76
node-7.domain.tld	190.01	36.40
node-8.domain.tld	192.59	39.75

Concurrency 10

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-11.domain.tld	138.34	62.15
node-13.domain.tld	138.37	64.57
node-15.domain.tld	63.27	63.77
node-17.domain.tld	72.49	63.56
node-18.domain.tld	137.22	58.73
node-20.domain.tld	56.73	64.66
node-4.domain.tld	76.95	60.73
node-5.domain.tld	68.55	59.09
node-7.domain.tld	87.67	59.11
node-8.domain.tld	137.68	61.18

8.4 OpenStack L2 Performance

This scenario launches 1 pair of VMs in the same private network on different compute nodes.

Scenario:

```
deployment:
  accommodation:
  - pair
  - single_room
  - compute_nodes: 2
  template: l2.hot
description: This scenario launches 1 pair of VMs in the same private network on␣
↪different
  compute nodes.
execution:
  tests:
  - class: flent
```

(continues on next page)

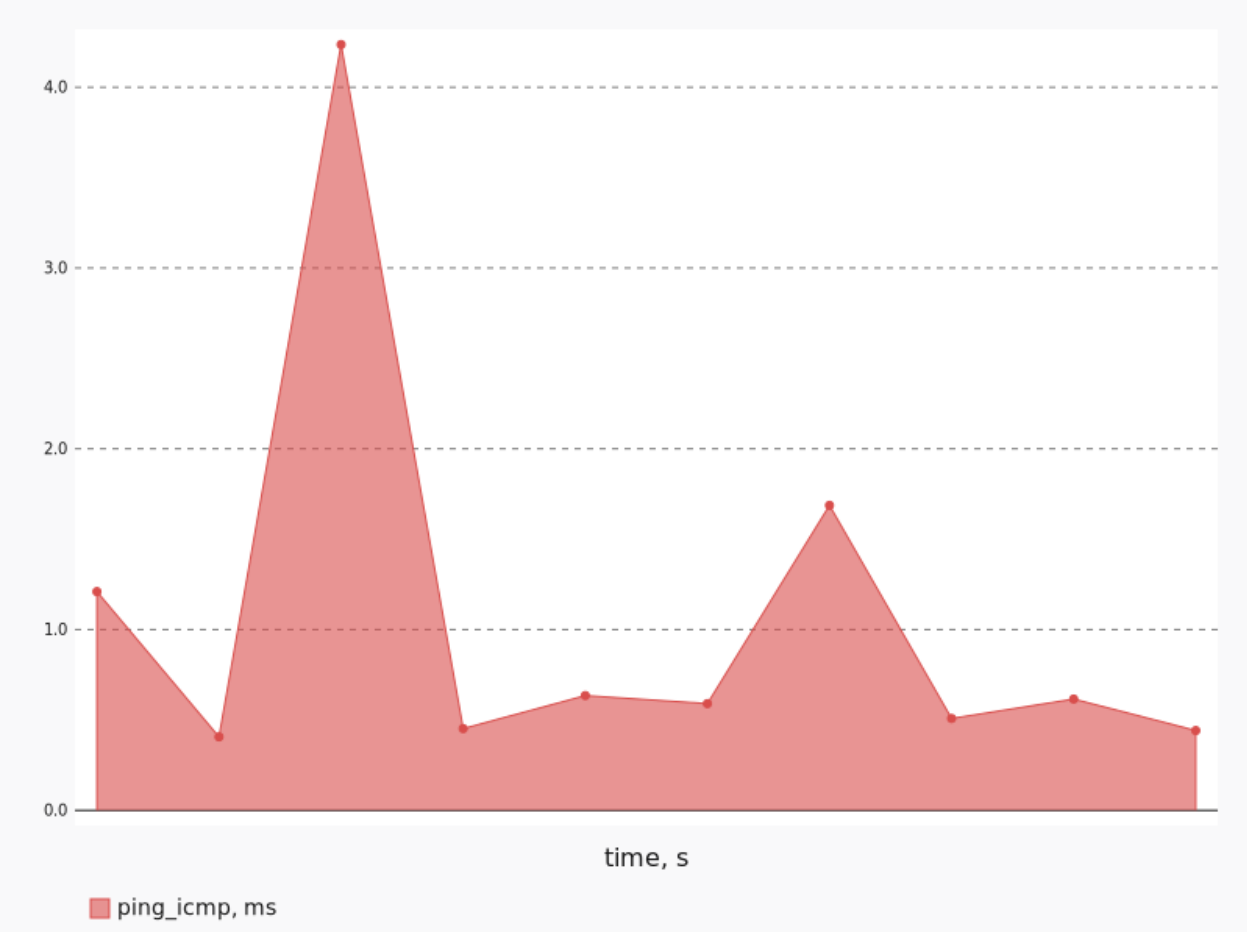
(continued from previous page)

```
    method: ping
    sla:
    - '[type == ''agent''] >> (stats.ping_icmp.avg < 0.5)'
    time: 10
    title: Ping
  - class: iperf3
    sla:
    - '[type == ''agent''] >> (stats.bandwidth.avg > 5000)'
    - '[type == ''agent''] >> (stats.retransmits.max < 10)'
    title: TCP
  - bandwidth: 0
    class: iperf3
    datagram_size: 32
    sla:
    - '[type == ''agent''] >> (stats.packets.avg > 100000)'
    title: UDP
    udp: true
file_name: /home/ishakhat/Work/shaker/shaker/scenarios/openstack/perf_l2.yaml
title: OpenStack L2 Performance
```

8.4.1 Ping

Test Specification:

```
class: flent
method: ping
sla:
- '[type == ''agent''] >> (stats.ping_icmp.avg < 0.5)'
time: 10
title: Ping
```



Stats:

```
ping_icmp:
  max: 4.236238930666339
  avg: 1.0783260741090341
  min: 0.4065897760580819
  unit: ms
```

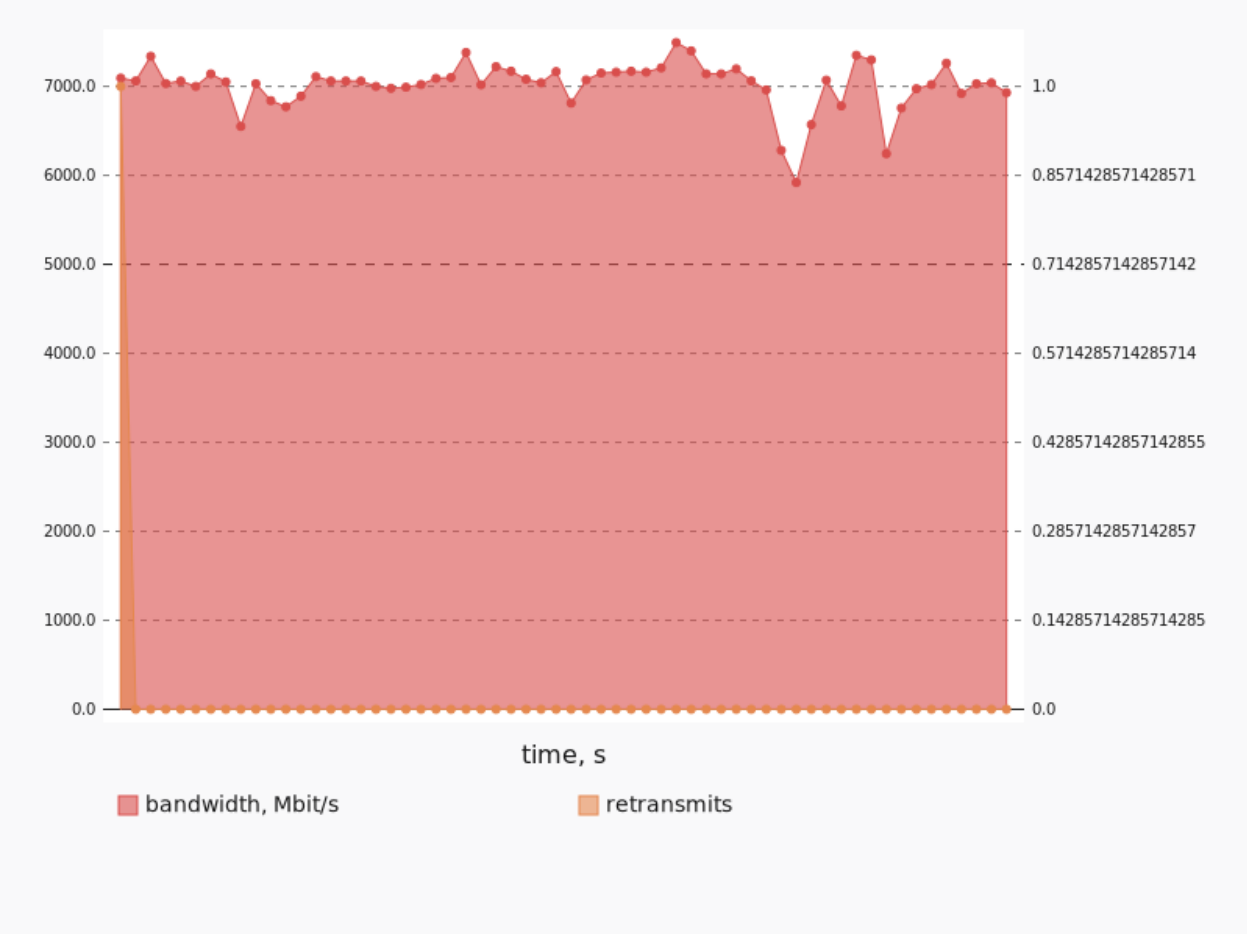
SLA:

Expression	Concurrency	Node	Result
stats.ping_icmp.avg < 0.5	1	node-9.domain.tld	FAIL

8.4.2 TCP

Test Specification:

```
class: iperf3
interval: 1
sla:
- '[type == ''agent''] >> (stats.bandwidth.avg > 5000)'
- '[type == ''agent''] >> (stats.retransmits.max < 10)'
title: TCP
```



Stats:

```
bandwidth:
  max: 7492.275238037109
  avg: 7015.98030573527
  min: 5919.618606567383
  unit: Mbit/s
retransmits:
  max: 1
  avg: 1.0
  min: 1
  unit: ''
```

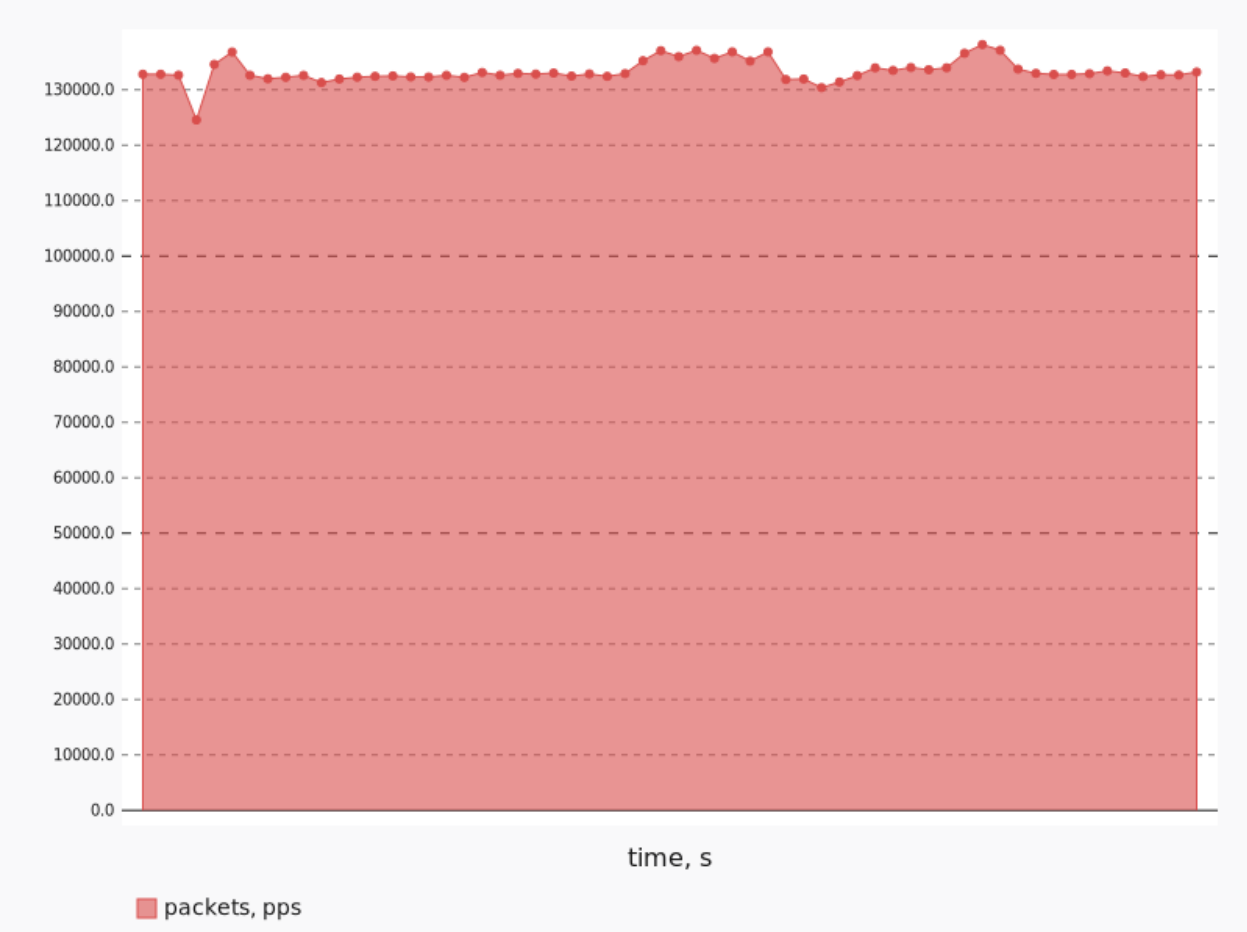
SLA:

Expression	Concurrency	Node	Result
stats.bandwidth.avg > 5000	1	node-9.domain.tld	OK
stats.retransmits.max < 10	1	node-9.domain.tld	OK

8.4.3 UDP

Test Specification:

```
bandwidth: 0
class: iperf3
datagram_size: 32
interval: 1
sla:
- '[type == ''agent''] >> (stats.packets.avg > 100000)'
title: UDP
udp: true
```



Stats:

```
packets:
  max: 138160
  avg: 133338.5
  min: 124560
  unit: pps
```

SLA:

Expression	Concurrency	Node	Result
stats.packets.avg > 100000	1	node-9.domain.tld	OK

8.5 OpenStack L3 East-West Performance

This scenario launches 1 pair of VMs in different networks connected to one router (L3 east-west). VMs are hosted on different compute nodes

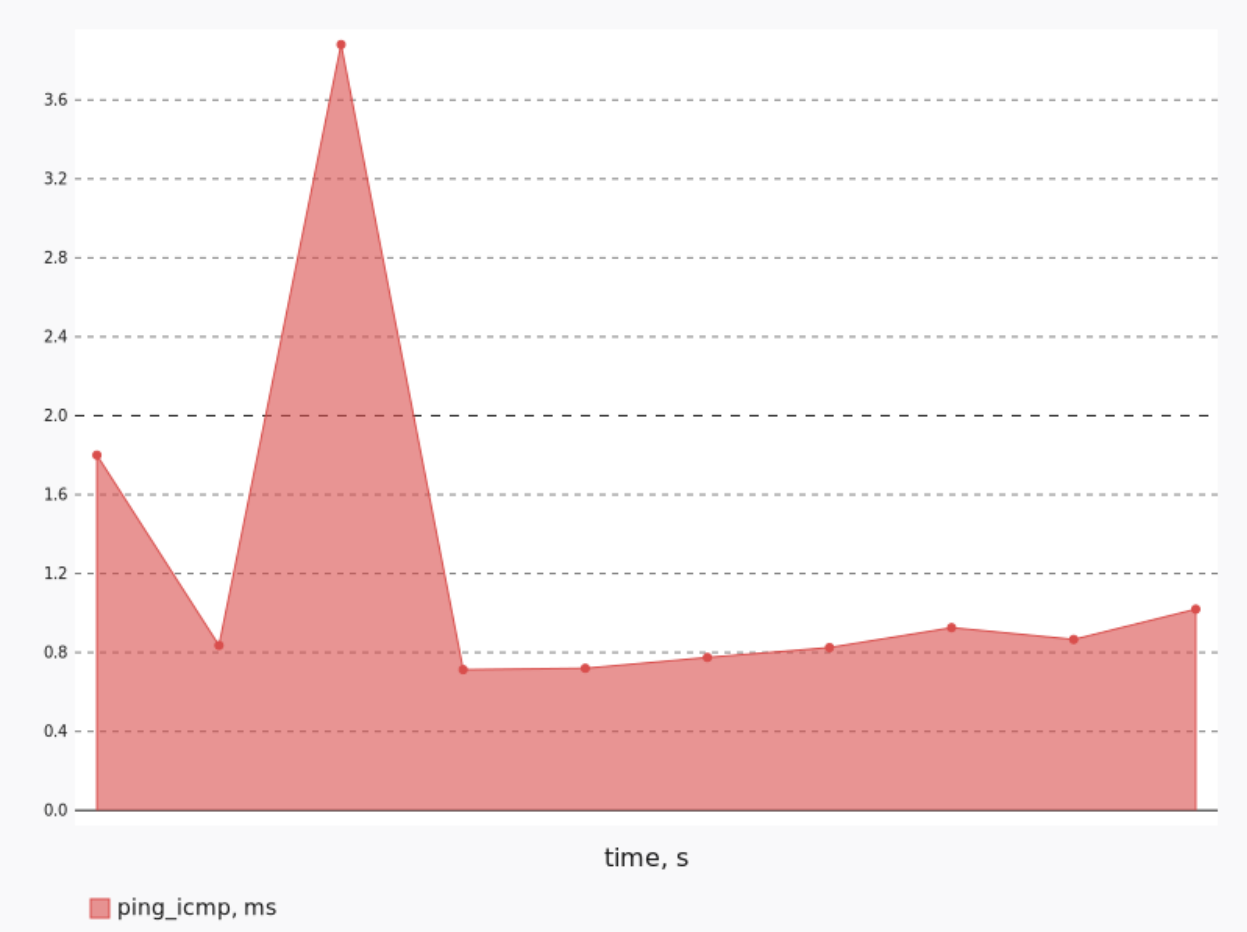
Scenario:

```
deployment:
  accommodation:
  - pair
  - single_room
  - compute_nodes: 2
  template: l3_east_west.hot
description: This scenario launches 1 pair of VMs in different networks connected
  to one router (L3 east-west). VMs are hosted on different compute nodes
execution:
  tests:
  - class: flent
    method: ping
    sla:
    - '[type == ''agent''] >> (stats.ping_icmp.avg < 2.0)'
    time: 10
    title: Ping
  - class: iperf3
    sla:
    - '[type == ''agent''] >> (stats.bandwidth.avg > 5000)'
    - '[type == ''agent''] >> (stats.retransmits.max < 10)'
    title: TCP
  - bandwidth: 0
    class: iperf3
    datagram_size: 32
    sla:
    - '[type == ''agent''] >> (stats.packets.avg > 100000)'
    title: UDP
    udp: true
file_name: /home/ishakhat/Work/shaker/shaker/scenarios/openstack/perf_l3_east_west.
↪yaml
title: OpenStack L3 East-West Performance
```

8.5.1 Ping

Test Specification:

```
class: flent
method: ping
sla:
- '[type == ''agent''] >> (stats.ping_icmp.avg < 2.0)'
time: 10
title: Ping
```



Stats:

```
ping_icmp:
  max: 3.880741082830054
  avg: 1.23610103398376
  min: 0.7130612739715825
  unit: ms
```

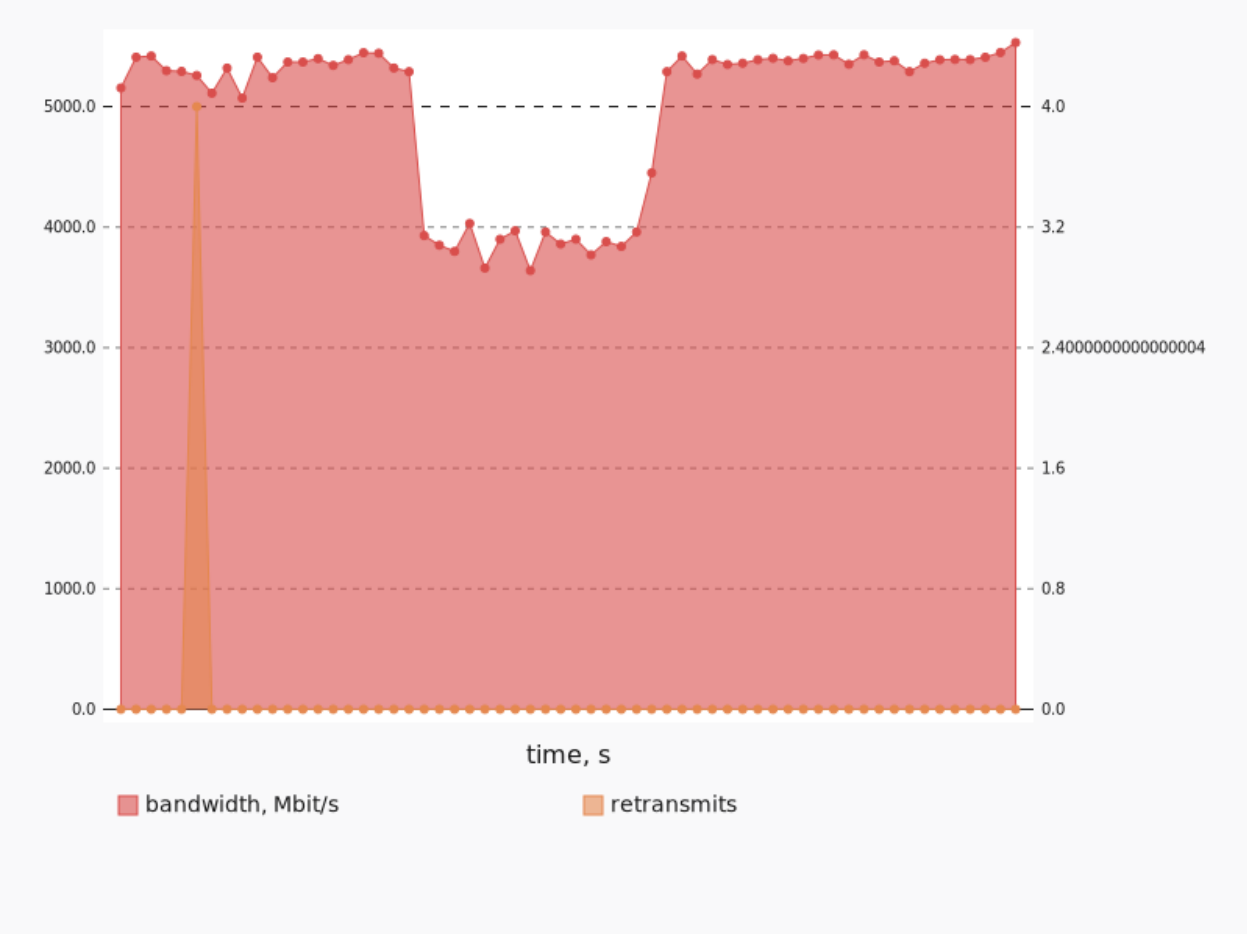
SLA:

Expression	Concurrency	Node	Result
stats.ping_icmp.avg < 2.0	1	node-19.domain.tld	OK

8.5.2 TCP

Test Specification:

```
class: iperf3
interval: 1
sla:
- '[type == ''agent''] >> (stats.bandwidth.avg > 5000)'
- '[type == ''agent''] >> (stats.retransmits.max < 10)'
title: TCP
```



Stats:

```
bandwidth:
  max: 5531.473159790039
  avg: 4966.737230682373
  min: 3640.0222778320312
  unit: Mbit/s
retransmits:
  max: 4
  avg: 4.0
  min: 4
  unit: ''
```

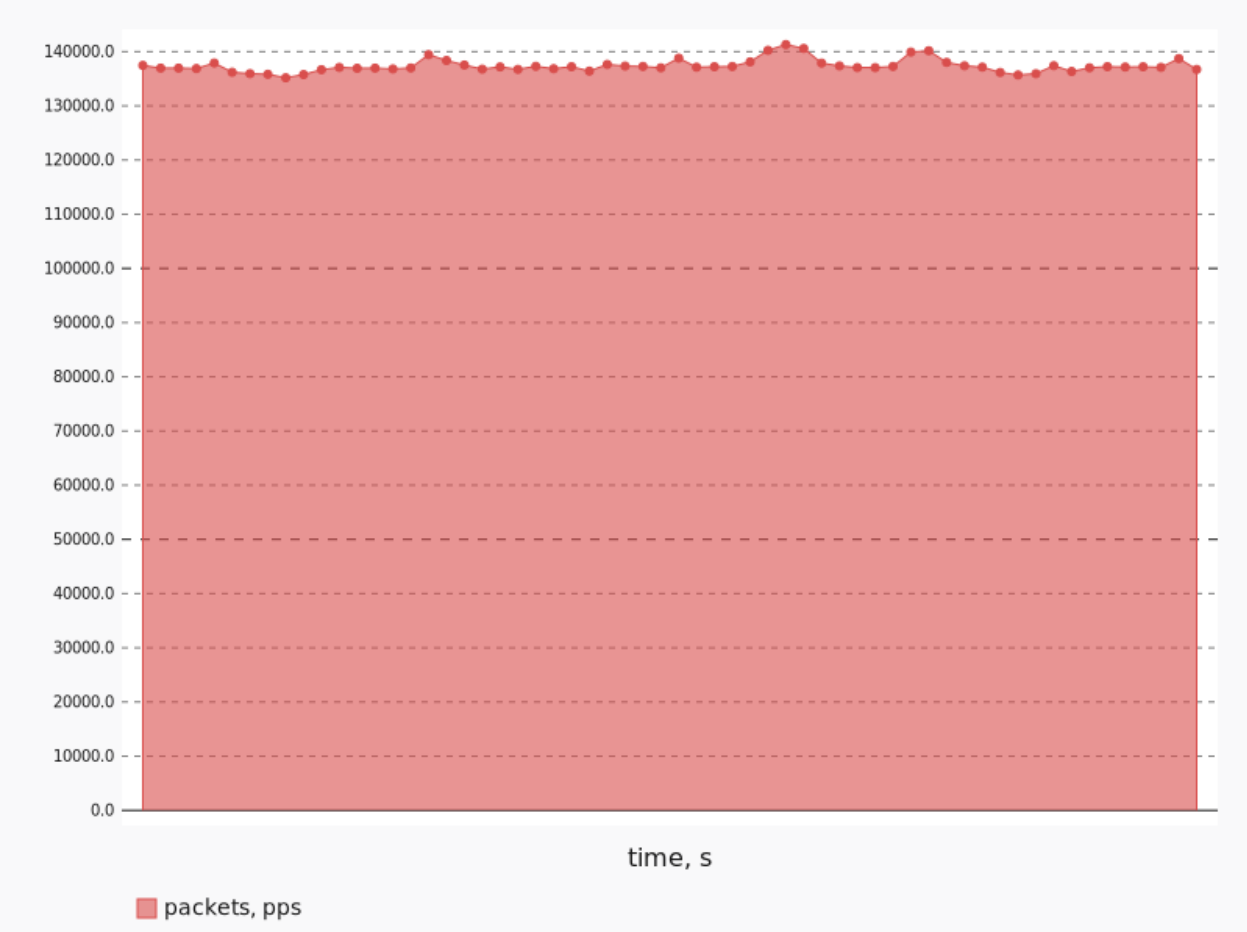
SLA:

Expression	Concurrency	Node	Result
stats.bandwidth.avg > 5000	1	node-19.domain.tld	FAIL
stats.retransmits.max < 10	1	node-19.domain.tld	OK

8.5.3 UDP

Test Specification:

```
bandwidth: 0
class: iperf3
datagram_size: 32
interval: 1
sla:
- '[type == ''agent''] >> (stats.packets.avg > 100000)'
title: UDP
udp: true
```



Stats:

```
packets:
  max: 141310
  avg: 137370.33333333334
  min: 135180
  unit: pps
```

SLA:

Expression	Concurrency	Node	Result
stats.packets.avg > 100000	1	node-19.domain.tld	OK

8.6 OpenStack L3 North-South Performance

This scenario launches 1 pair of VMs on different compute nodes. VMs are in the different networks connected via different routers, primary accesses minion by floating ip

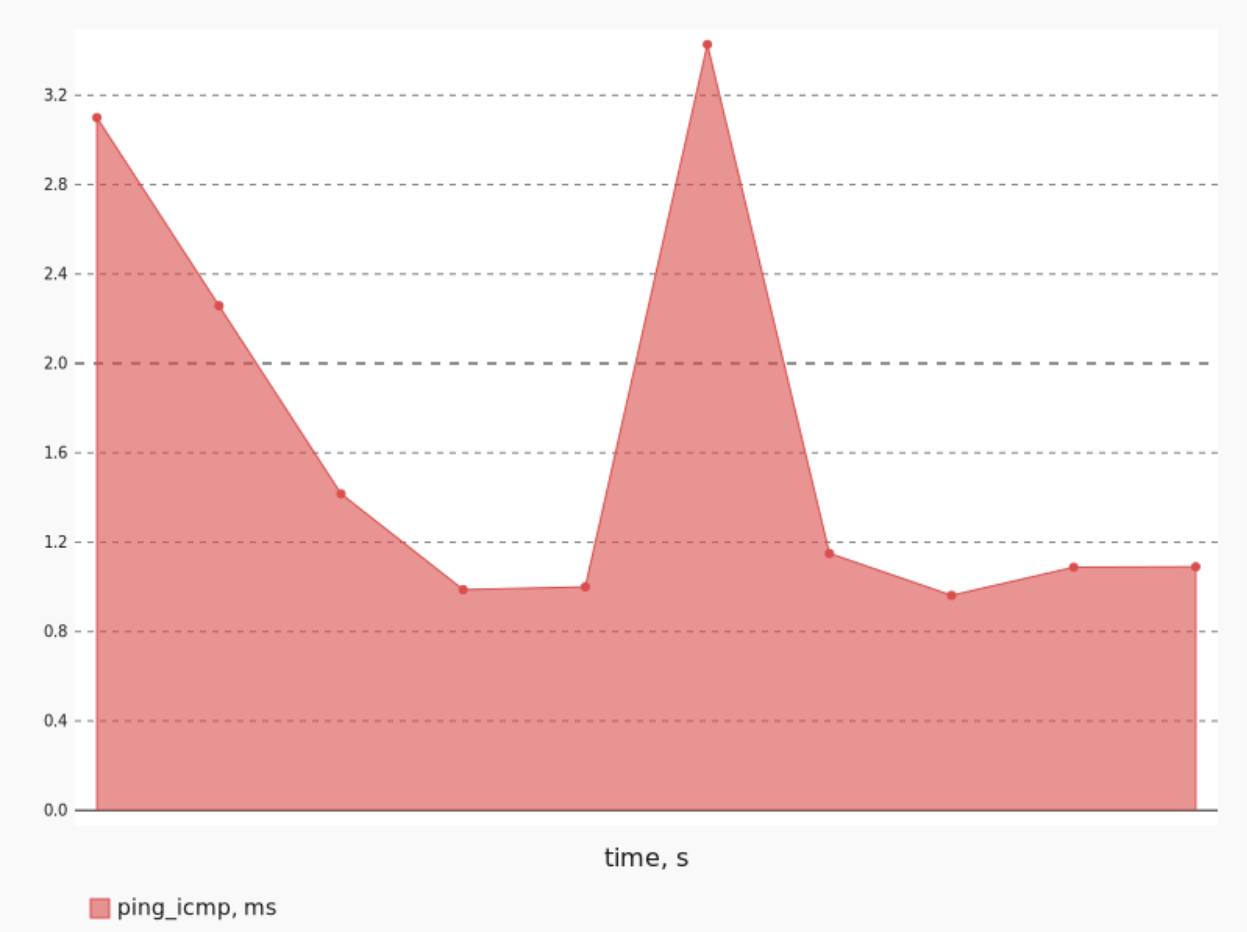
Scenario:

```
deployment:
  accommodation:
  - pair
  - single_room
  - compute_nodes: 2
  template: l3_north_south.hot
description: This scenario launches 1 pair of VMs on different compute nodes. VMs
  are in the different networks connected via different routers, primary accesses␣
↪minion
  by floating ip
execution:
  tests:
  - class: flent
    method: ping
    sla:
    - '[type == ''agent''] >> (stats.ping_icmp.avg < 2.0)'
    time: 10
    title: Ping
  - class: iperf3
    sla:
    - '[type == ''agent''] >> (stats.bandwidth.avg > 5000)'
    - '[type == ''agent''] >> (stats.retransmits.max < 10)'
    title: TCP
  - bandwidth: 0
    class: iperf3
    datagram_size: 32
    sla:
    - '[type == ''agent''] >> (stats.packets.avg > 100000)'
    title: UDP
    udp: true
file_name: /home/ishakhat/Work/shaker/shaker/scenarios/openstack/perf_l3_north_south.
↪yaml
title: OpenStack L3 North-South Performance
```

8.6.1 Ping

Test Specification:

```
class: flent
method: ping
sla:
- '[type == ''agent''] >> (stats.ping_icmp.avg < 2.0)'
time: 10
title: Ping
```



Stats:

```
ping_icmp:
  max: 3.4270406725254006
  avg: 1.6479111172469332
  min: 0.9622029103967339
  unit: ms
```

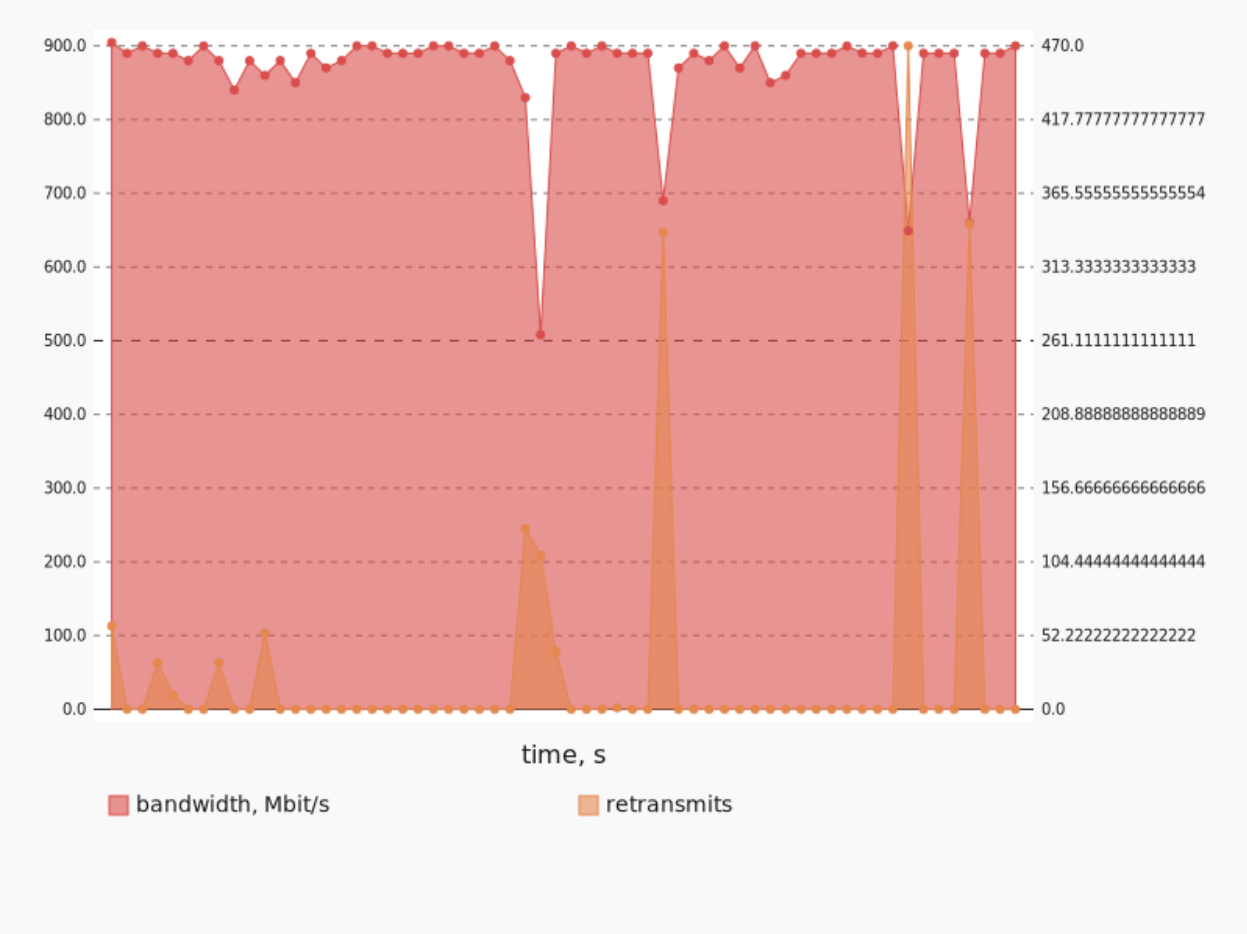
SLA:

Expression	Concurrency	Node	Result
stats.ping_icmp.avg < 2.0	1	node-11.domain.tld	OK

8.6.2 TCP

Test Specification:

```
class: iperf3
interval: 1
sla:
- '[type == ''agent''] >> (stats.bandwidth.avg > 5000)'
- '[type == ''agent''] >> (stats.retransmits.max < 10)'
title: TCP
```



Stats:

```
bandwidth:
  max: 904.4981002807617
  avg: 868.6801114400228
  min: 508.1815719604492
  unit: Mbit/s
retransmits:
  max: 470
  avg: 135.0
  min: 1
  unit: ''
```

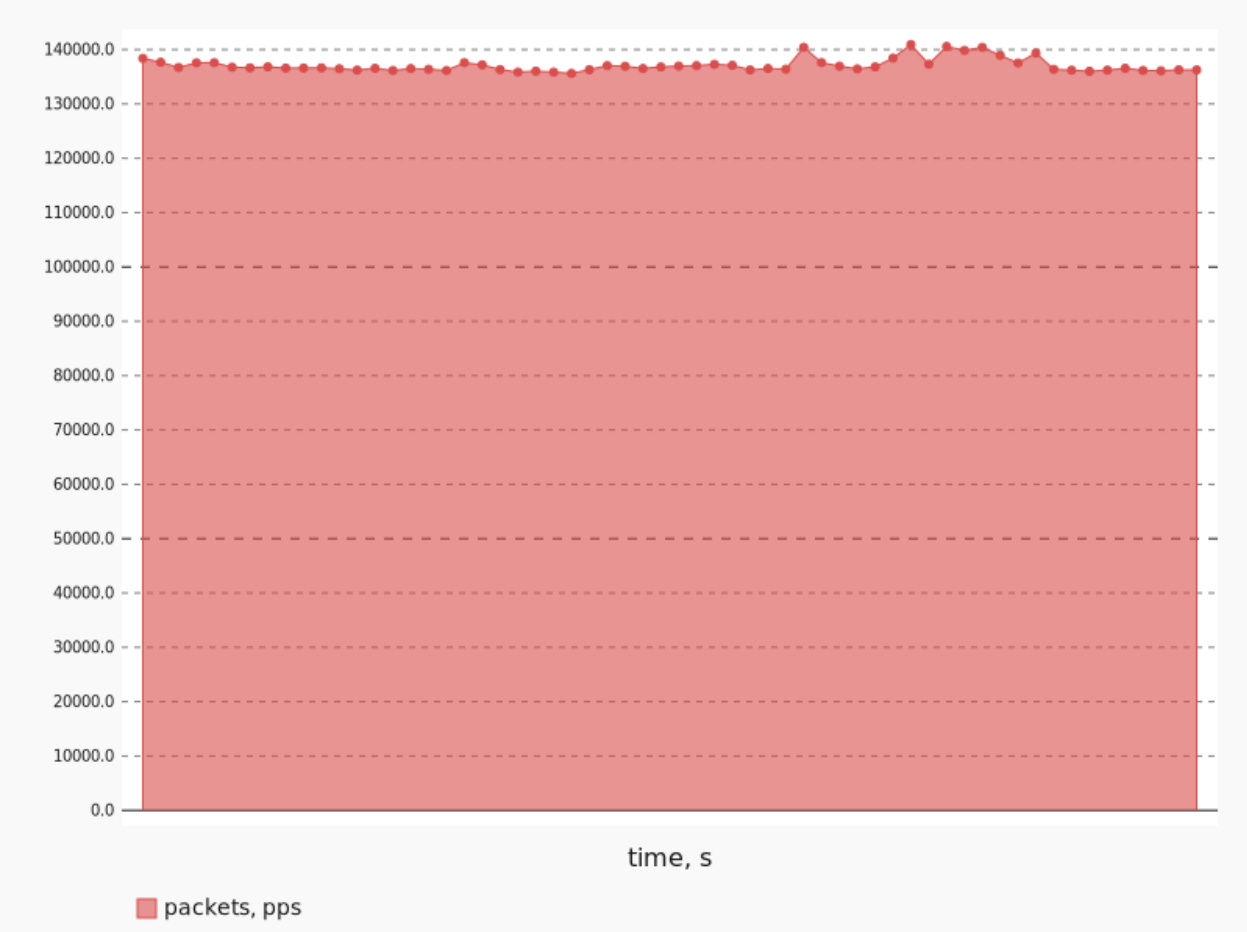
SLA:

Expression	Concurrency	Node	Result
stats.bandwidth.avg > 5000	1	node-11.domain.tld	FAIL
stats.retransmits.max < 10	1	node-11.domain.tld	FAIL

8.6.3 UDP

Test Specification:

```
bandwidth: 0
class: iperf3
datagram_size: 32
interval: 1
sla:
- '[type == ''agent''] >> (stats.packets.avg > 100000)'
title: UDP
udp: true
```



Stats:

```
packets:
  max: 140930
  avg: 137099.0
  min: 135620
  unit: pps
```

SLA:

Expression	Concurrency	Node	Result
stats.packets.avg > 100000	1	node-11.domain.tld	OK

8.7 OpenStack L2 Dense

This scenario launches several pairs of VMs on the same compute node. VM are plugged into the same private network. Useful for testing performance degradation when the number of VMs grows.

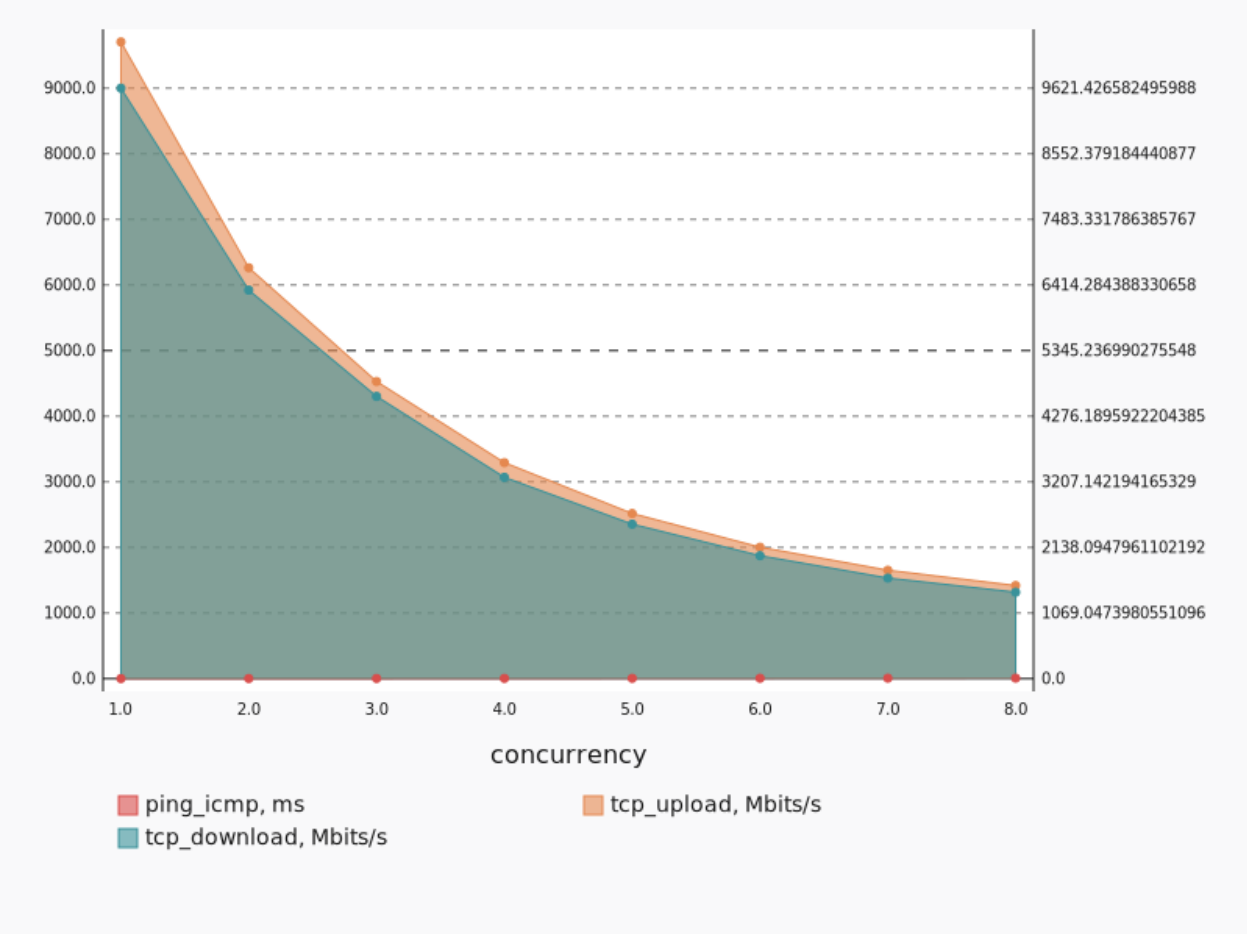
Scenario:

```
deployment:
  accommodation:
  - pair
  - double_room
  - density: 8
  - compute_nodes: 1
  template: l2.hot
description: This scenario launches several pairs of VMs on the same compute node.
  VM are plugged into the same private network. Useful for testing performance␣
↪degradation
  when the number of VMs grows.
execution:
  progression: linear
  tests:
  - class: flent
    method: tcp_download
    title: Download
  - class: flent
    method: tcp_upload
    title: Upload
  - class: flent
    method: tcp_bidirectional
    title: Bi-directional
file_name: /home/ishakhat/Work/shaker/shaker/scenarios/openstack/dense_l2.yaml
title: OpenStack L2 Dense
```

8.7.1 Bi-directional

Test Specification:

```
class: flent
method: tcp_bidirectional
title: Bi-directional
```



Stats:

concurrency	ping_icmp, ms	tcp_download, Mbits/s	tcp_upload, Mbits/s
1	1.20	9621.43	9704.36
2	1.87	6330.36	6262.75
3	2.55	4598.51	4529.14
4	3.52	3279.71	3291.72
5	4.55	2516.36	2516.94
6	5.71	2002.73	2003.24
7	6.97	1638.64	1652.10
8	7.81	1408.17	1419.22

Concurrency 1

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s	tcp_upload, Mbits/s
node-6.domain.tld	1.20	9621.43	9704.36

Concurrency 2

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s	tcp_upload, Mbits/s
node-6.domain.tld	1.86	6294.84	6204.99
node-6.domain.tld	1.88	6365.88	6320.52

Concurrency 3

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s	tcp_upload, Mbits/s
node-6.domain.tld	2.39	4557.23	4428.49
node-6.domain.tld	2.64	4670.00	4664.19
node-6.domain.tld	2.63	4568.32	4494.73

Concurrency 4

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s	tcp_upload, Mbits/s
node-6.domain.tld	3.68	3259.31	3287.13
node-6.domain.tld	3.26	3298.23	3314.15
node-6.domain.tld	3.83	3257.17	3226.80
node-6.domain.tld	3.33	3304.13	3338.81

Concurrency 5

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s	tcp_upload, Mbits/s
node-6.domain.tld	5.04	2550.88	2583.93
node-6.domain.tld	4.14	2486.48	2480.28
node-6.domain.tld	3.97	2520.54	2515.50
node-6.domain.tld	4.82	2483.47	2484.11
node-6.domain.tld	4.81	2540.44	2520.88

Concurrency 6

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s	tcp_upload, Mbits/s
node-6.domain.tld	5.90	1961.10	1984.38
node-6.domain.tld	4.99	2052.38	2051.06
node-6.domain.tld	6.02	1990.23	1965.51
node-6.domain.tld	5.19	1986.60	1964.58
node-6.domain.tld	6.02	1982.95	2006.11
node-6.domain.tld	6.15	2043.14	2047.81

Concurrency 7

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s	tcp_upload, Mbits/s
node-6.domain.tld	7.39	1683.33	1700.30
node-6.domain.tld	5.99	1614.44	1628.19
node-6.domain.tld	6.22	1631.46	1648.62
node-6.domain.tld	7.12	1615.92	1620.92
node-6.domain.tld	7.22	1624.42	1648.09
node-6.domain.tld	7.10	1609.21	1646.56
node-6.domain.tld	7.72	1691.71	1672.05

Concurrency 8

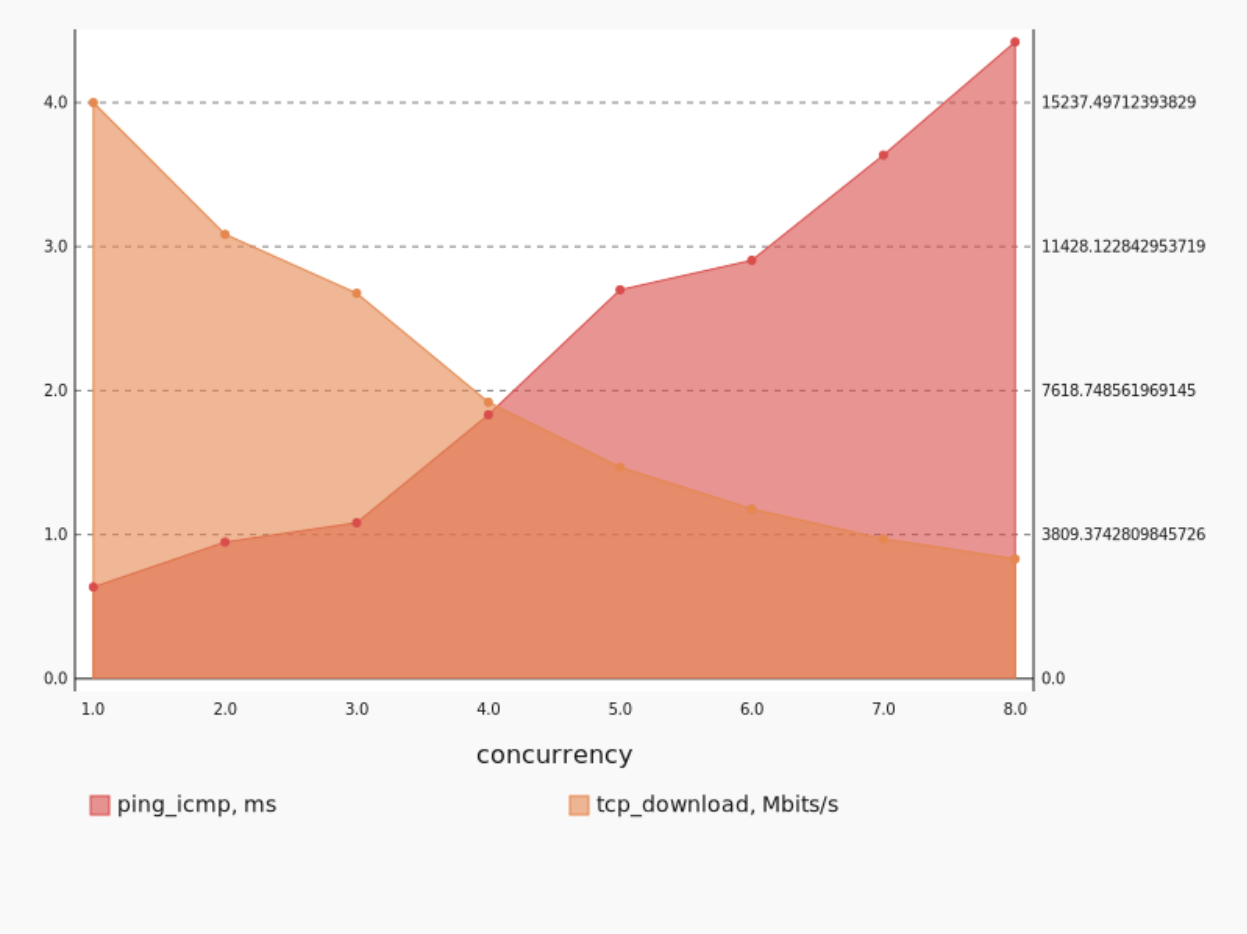
Stats:

node	ping_icmp, ms	tcp_download, Mbits/s	tcp_upload, Mbits/s
node-6.domain.tld	7.86	1381.55	1380.70
node-6.domain.tld	8.10	1360.85	1354.82
node-6.domain.tld	8.00	1629.02	1659.45
node-6.domain.tld	7.36	1403.67	1401.41
node-6.domain.tld	8.19	1362.26	1367.91
node-6.domain.tld	7.74	1395.07	1399.40
node-6.domain.tld	7.06	1377.46	1421.64
node-6.domain.tld	8.13	1355.44	1368.43

8.7.2 Download

Test Specification:

```
class: flent
method: tcp_download
title: Download
```



Stats:

concurrency	ping_icmp, ms	tcp_download, Mbits/s
1	0.64	15237.50
2	0.95	11753.03
3	1.08	10193.87
4	1.83	7311.93
5	2.70	5592.60
6	2.90	4488.04
7	3.64	3696.83
8	4.42	3166.11

Concurrency 1

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-6.domain.tld	0.64	15237.50

Concurrency 2

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-6.domain.tld	0.96	11632.38
node-6.domain.tld	0.94	11873.68

Concurrency 3

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-6.domain.tld	1.07	10284.54
node-6.domain.tld	1.18	10014.04
node-6.domain.tld	0.99	10283.04

Concurrency 4

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-6.domain.tld	1.90	7257.45
node-6.domain.tld	1.84	7282.47
node-6.domain.tld	1.72	7416.10
node-6.domain.tld	1.88	7291.69

Concurrency 5

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-6.domain.tld	2.60	5518.59
node-6.domain.tld	2.61	5753.13
node-6.domain.tld	2.38	5560.52
node-6.domain.tld	3.24	5583.56
node-6.domain.tld	2.67	5547.21

Concurrency 6

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-6.domain.tld	2.68	4458.91
node-6.domain.tld	2.94	4565.03
node-6.domain.tld	2.83	4493.59
node-6.domain.tld	2.82	4502.03
node-6.domain.tld	3.30	4430.72
node-6.domain.tld	2.85	4477.96

Concurrency 7

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-6.domain.tld	3.06	3685.12
node-6.domain.tld	4.15	3789.90
node-6.domain.tld	3.56	3668.97
node-6.domain.tld	3.19	3606.68
node-6.domain.tld	3.25	3753.06
node-6.domain.tld	4.08	3707.98
node-6.domain.tld	4.15	3666.12

Concurrency 8

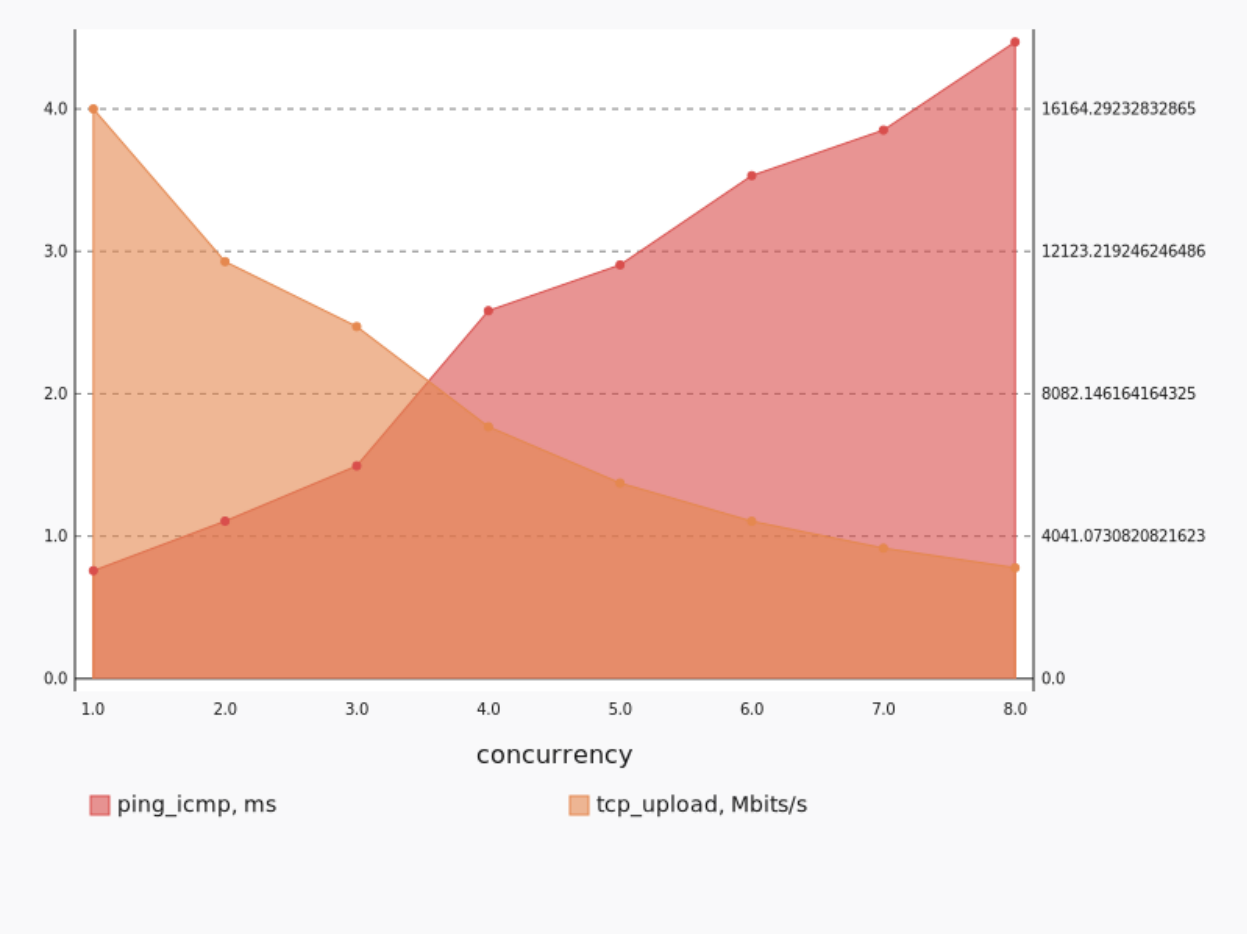
Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-6.domain.tld	4.45	3188.59
node-6.domain.tld	3.68	3129.72
node-6.domain.tld	4.80	3081.13
node-6.domain.tld	4.02	3093.75
node-6.domain.tld	4.72	3209.73
node-6.domain.tld	4.52	3068.88
node-6.domain.tld	4.28	3107.04
node-6.domain.tld	4.89	3450.02

8.7.3 Upload

Test Specification:

```
class: flent
method: tcp_upload
title: Upload
```



Stats:

concurrency	ping_icmp, ms	tcp_upload, Mbits/s
1	0.76	16164.29
2	1.11	11832.46
3	1.49	9988.86
4	2.58	7146.27
5	2.90	5548.76
6	3.53	4465.03
7	3.85	3701.96
8	4.47	3145.42

Concurrency 1

Stats:

node	ping_icmp, ms	tcp_upload, Mbits/s
node-6.domain.tld	0.76	16164.29

Concurrency 2

Stats:

node	ping_icmp, ms	tcp_upload, Mbits/s
node-6.domain.tld	1.11	11898.27
node-6.domain.tld	1.11	11766.64

Concurrency 3

Stats:

node	ping_icmp, ms	tcp_upload, Mbits/s
node-6.domain.tld	1.69	10005.98
node-6.domain.tld	1.54	9859.36
node-6.domain.tld	1.26	10101.24

Concurrency 4

Stats:

node	ping_icmp, ms	tcp_upload, Mbits/s
node-6.domain.tld	2.66	7042.02
node-6.domain.tld	2.77	7181.58
node-6.domain.tld	2.44	7203.51
node-6.domain.tld	2.47	7157.96

Concurrency 5

Stats:

node	ping_icmp, ms	tcp_upload, Mbits/s
node-6.domain.tld	2.87	5610.24
node-6.domain.tld	2.60	5423.45
node-6.domain.tld	2.71	5540.39
node-6.domain.tld	3.38	5503.63
node-6.domain.tld	2.97	5666.08

Concurrency 6

Stats:

node	ping_icmp, ms	tcp_upload, Mbits/s
node-6.domain.tld	3.33	4583.27
node-6.domain.tld	3.79	4437.25
node-6.domain.tld	3.01	4497.67
node-6.domain.tld	3.47	4516.93
node-6.domain.tld	3.71	4490.94
node-6.domain.tld	3.89	4264.11

Concurrency 7

Stats:

node	ping_icmp, ms	tcp_upload, Mbits/s
node-6.domain.tld	4.72	3699.14
node-6.domain.tld	3.39	3684.00
node-6.domain.tld	3.57	3694.32
node-6.domain.tld	3.58	3778.59
node-6.domain.tld	3.62	3667.92
node-6.domain.tld	3.80	3658.24
node-6.domain.tld	4.28	3731.53

Concurrency 8

Stats:

node	ping_icmp, ms	tcp_upload, Mbits/s
node-6.domain.tld	4.42	3313.16
node-6.domain.tld	4.45	3090.43
node-6.domain.tld	4.58	3049.20
node-6.domain.tld	3.67	3099.69
node-6.domain.tld	4.30	3217.62
node-6.domain.tld	4.92	3086.23
node-6.domain.tld	4.62	3131.54
node-6.domain.tld	4.80	3175.52

8.8 OpenStack L3 East-West Dense

This scenario launches pairs of VMs in different networks connected to one router (L3 east-west)

Scenario:

```
deployment:
  accommodation:
  - pair
  - double_room
  - density: 8
  - compute_nodes: 1
  template: l3_east_west.hot
description: This scenario launches pairs of VMs in different networks connected to
```

(continues on next page)

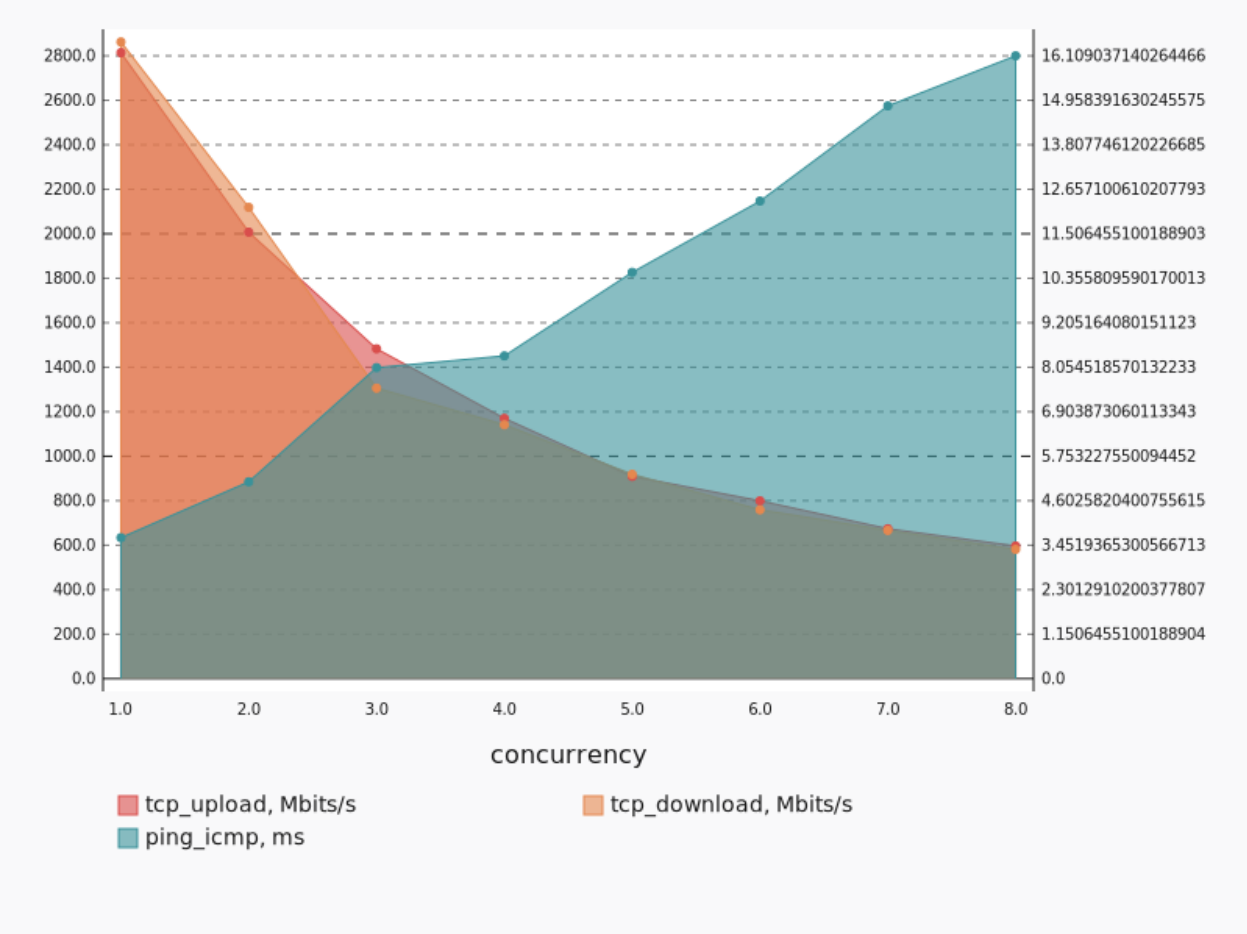
(continued from previous page)

```
  one router (L3 east-west)
execution:
  progression: linear
  tests:
  - class: flent
    method: tcp_download
    title: Download
  - class: flent
    method: tcp_upload
    title: Upload
  - class: flent
    method: tcp_bidirectional
    title: Bi-directional
file_name: /home/ishakhat/Work/shaker/shaker/scenarios/openstack/dense_l3_east_west.
↪yaml
title: OpenStack L3 East-West Dense
```

8.8.1 Bi-directional

Test Specification:

```
class: flent
method: tcp_bidirectional
title: Bi-directional
```



Stats:

concurrency	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
1	2814.04	3.65	2862.18
2	2007.10	5.09	2118.44
3	1482.64	8.04	1305.91
4	1170.08	8.35	1141.41
5	909.19	10.51	918.53
6	799.28	12.35	759.03
7	673.86	14.81	666.51
8	596.48	16.11	581.02

Concurrency 1

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	2814.04	3.65	2862.18

Concurrency 2

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	2080.99	4.29	2483.08
node-5.domain.tld	1933.21	5.89	1753.80

Concurrency 3

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	1238.39	10.64	1045.61
node-5.domain.tld	2016.54	5.48	1768.01
node-5.domain.tld	1192.99	8.02	1104.12

Concurrency 4

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	1177.99	7.54	1289.99
node-5.domain.tld	1135.60	8.45	1112.07
node-5.domain.tld	1204.90	9.21	1025.01
node-5.domain.tld	1161.82	8.19	1138.58

Concurrency 5

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	937.75	10.72	859.38
node-5.domain.tld	984.40	9.69	999.06
node-5.domain.tld	884.40	12.42	892.02
node-5.domain.tld	878.76	10.17	986.22
node-5.domain.tld	860.63	9.55	855.98

Concurrency 6

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	800.83	14.16	800.62
node-5.domain.tld	907.79	12.76	774.30
node-5.domain.tld	789.24	12.71	751.34
node-5.domain.tld	778.34	11.16	790.35
node-5.domain.tld	778.92	10.96	769.99
node-5.domain.tld	740.54	12.37	667.58

Concurrency 7

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	719.54	16.54	660.84
node-5.domain.tld	722.22	14.58	625.52
node-5.domain.tld	626.60	14.66	726.26
node-5.domain.tld	684.59	13.92	682.97
node-5.domain.tld	682.67	13.97	728.80
node-5.domain.tld	649.98	15.72	552.49
node-5.domain.tld	631.41	14.30	688.73

Concurrency 8

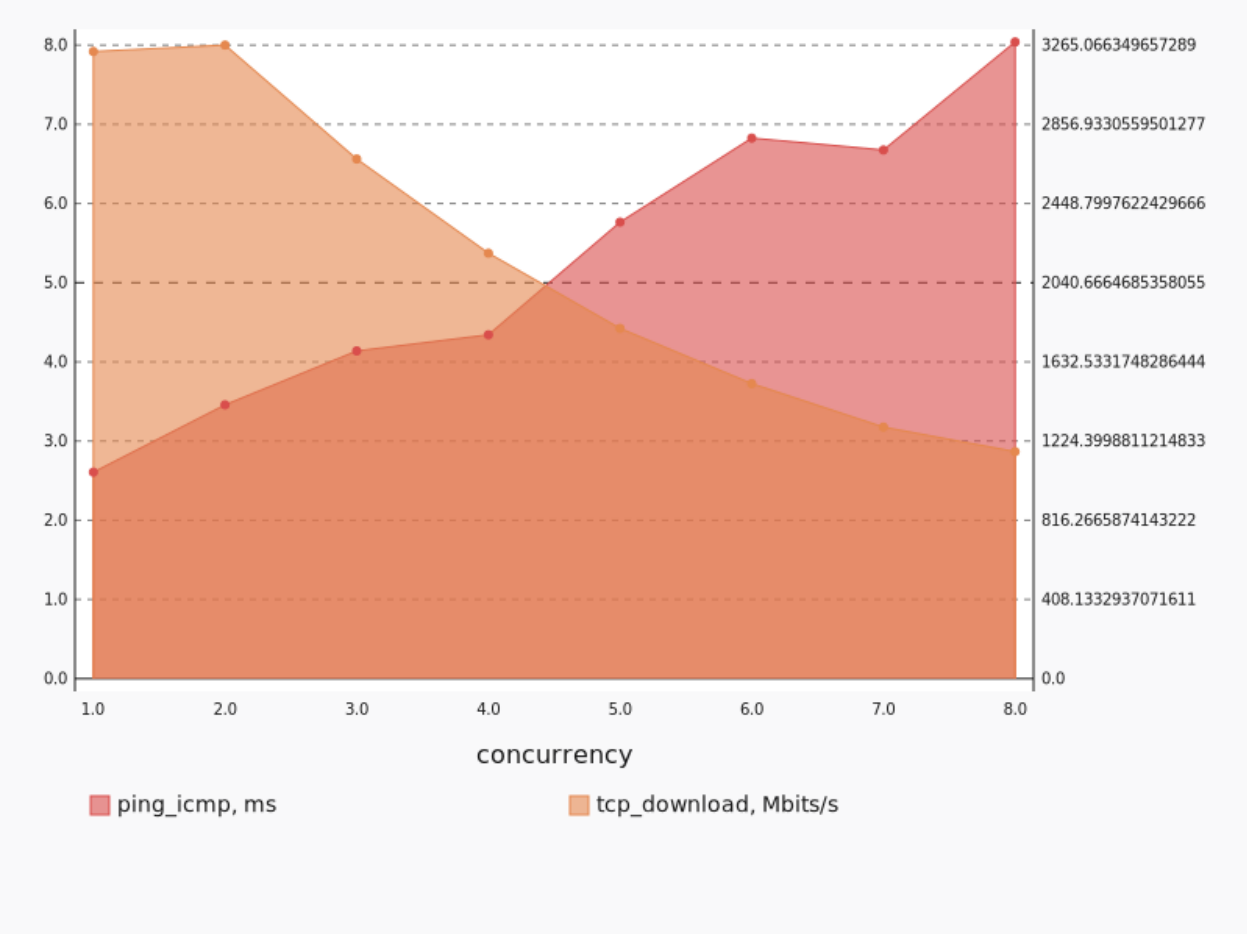
Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	572.87	14.97	607.17
node-5.domain.tld	558.98	15.34	631.26
node-5.domain.tld	589.19	17.86	583.32
node-5.domain.tld	595.93	15.09	537.40
node-5.domain.tld	619.96	16.15	549.46
node-5.domain.tld	566.98	17.50	585.90
node-5.domain.tld	628.83	15.26	582.33
node-5.domain.tld	639.13	16.70	571.30

8.8.2 Download

Test Specification:

```
class: flent
method: tcp_download
title: Download
```



Stats:

concurrency	ping_icmp, ms	tcp_download, Mbits/s
1	2.61	3232.05
2	3.46	3265.07
3	4.14	2678.01
4	4.34	2192.83
5	5.77	1805.04
6	6.83	1520.49
7	6.68	1296.37
8	8.04	1169.80

Concurrency 1

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	2.61	3232.05

Concurrency 2

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	3.50	3145.52
node-5.domain.tld	3.41	3384.62

Concurrency 3

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	4.10	2752.96
node-5.domain.tld	3.57	2717.00
node-5.domain.tld	4.75	2564.08

Concurrency 4

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	4.79	2105.32
node-5.domain.tld	4.27	2252.28
node-5.domain.tld	4.76	2144.97
node-5.domain.tld	3.55	2268.76

Concurrency 5

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	6.57	1742.67
node-5.domain.tld	5.39	1868.02
node-5.domain.tld	5.24	1697.80
node-5.domain.tld	6.39	1952.90
node-5.domain.tld	5.24	1763.82

Concurrency 6

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	6.80	1347.71
node-5.domain.tld	7.98	1406.02
node-5.domain.tld	6.81	1546.89
node-5.domain.tld	5.43	1662.43
node-5.domain.tld	7.36	1513.16
node-5.domain.tld	6.58	1646.74

Concurrency 7

Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	5.44	1524.59
node-5.domain.tld	6.32	985.88
node-5.domain.tld	6.65	1551.91
node-5.domain.tld	7.44	1444.54
node-5.domain.tld	6.60	1492.27
node-5.domain.tld	7.01	965.67
node-5.domain.tld	7.26	1109.73

Concurrency 8

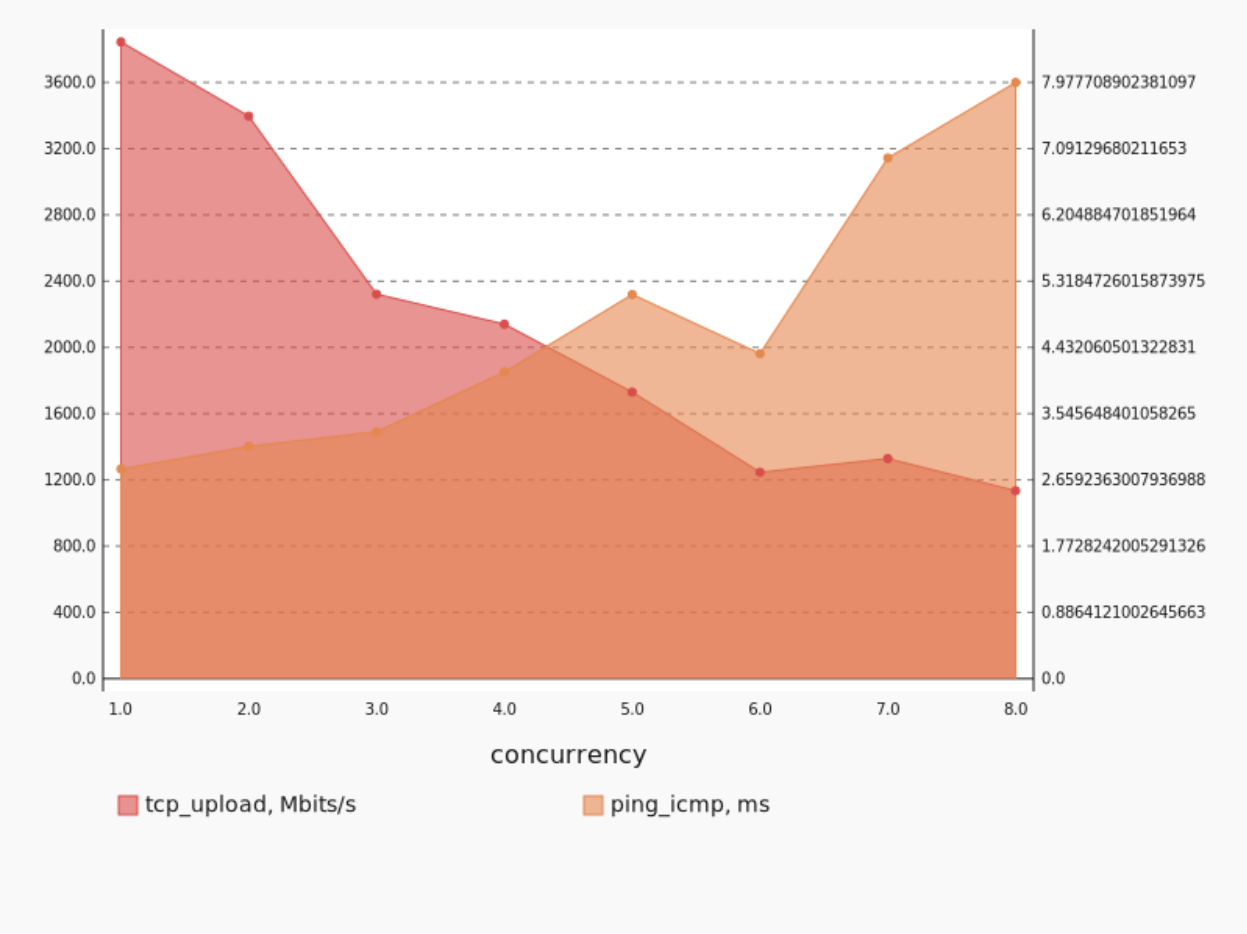
Stats:

node	ping_icmp, ms	tcp_download, Mbits/s
node-5.domain.tld	6.66	1361.59
node-5.domain.tld	7.88	1041.82
node-5.domain.tld	8.44	1263.24
node-5.domain.tld	8.40	1052.99
node-5.domain.tld	9.14	1218.77
node-5.domain.tld	7.72	1166.68
node-5.domain.tld	6.83	1189.83
node-5.domain.tld	9.23	1063.47

8.8.3 Upload

Test Specification:

```
class: flent
method: tcp_upload
title: Upload
```



Stats:

concurrency	tcp_upload, Mbits/s	ping_icmp, ms
1	3844.43	2.81
2	3396.30	3.11
3	2321.55	3.30
4	2140.43	4.10
5	1730.21	5.14
6	1246.42	4.35
7	1329.00	6.97
8	1134.45	7.98

Concurrency 1

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-5.domain.tld	3844.43	2.81

Concurrency 2

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-5.domain.tld	3482.66	2.78
node-5.domain.tld	3309.94	3.44

Concurrency 3

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-5.domain.tld	2942.33	2.80
node-5.domain.tld	2025.66	3.07
node-5.domain.tld	1996.67	4.05

Concurrency 4

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-5.domain.tld	1833.08	3.68
node-5.domain.tld	2506.52	4.41
node-5.domain.tld	2223.73	3.82
node-5.domain.tld	1998.38	4.49

Concurrency 5

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-5.domain.tld	1527.11	4.09
node-5.domain.tld	1877.01	3.86
node-5.domain.tld	1851.41	4.48
node-5.domain.tld	1944.21	6.07
node-5.domain.tld	1451.29	7.21

Concurrency 6

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-5.domain.tld	755.12	14.41
node-5.domain.tld	2021.84	2.26
node-5.domain.tld	928.22	1.26
node-5.domain.tld	2076.70	3.16
node-5.domain.tld	848.13	1.59
node-5.domain.tld	848.49	3.42

Concurrency 7

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-5.domain.tld	1330.81	8.47
node-5.domain.tld	1497.74	5.40
node-5.domain.tld	1297.62	6.61
node-5.domain.tld	1207.32	7.11
node-5.domain.tld	1388.78	8.44
node-5.domain.tld	1210.06	6.73
node-5.domain.tld	1370.67	6.01

Concurrency 8

Stats:

node	tcp_upload, Mbits/s	ping_icmp, ms
node-5.domain.tld	1131.88	8.76
node-5.domain.tld	1058.38	7.68
node-5.domain.tld	1067.14	7.80
node-5.domain.tld	1350.97	7.68
node-5.domain.tld	985.73	6.97
node-5.domain.tld	1060.46	7.20
node-5.domain.tld	1117.55	9.80
node-5.domain.tld	1303.53	7.92

CHAPTER 9

Shaker config parameters

9.1 DEFAULT

agent_dir

Type string

Default <None>

If specified, directs Shaker to write execution script for the shell class in agent(s) instance defined directory. Defaults to /tmp directory.

server_endpoint

Type unknown type

Default <None>

Address for server connections (host:port), defaults to env[SHAKER_SERVER_ENDPOINT].

polling_interval

Type integer

Default 10

How frequently the agent polls server, in seconds

os_auth_url

Type string

Default u''

Authentication URL, defaults to env[OS_AUTH_URL].

os_tenant_name

Type string

Default u''

Authentication tenant name, defaults to env[OS_TENANT_NAME].

os_project_name

Type string

Default `u''`

Authentication project name. This option is mutually exclusive with –os-tenant-name. Defaults to env[OS_PROJECT_NAME].

os_project_domain_name

Type string

Default `Default`

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

Authentication project domain name. Defaults to env[OS_PROJECT_DOMAIN_NAME].

os_username

Type string

Default `u''`

Authentication username, defaults to env[OS_USERNAME].

os_user_domain_name

Type string

Default `u''`

Authentication username. Defaults to env[OS_USER_DOMAIN_NAME].

os_identity_api_version

Type string

Default `3`

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

Identity API version, defaults to env[OS_IDENTITY_API_VERSION].

os_password

Type string

Default `u''`

Authentication password, defaults to env[OS_PASSWORD].

os_cacert

Type string

Default `u''`

Location of CA Certificate, defaults to env[OS_CACERT].

os_insecure

Type boolean

Default `false`

When using SSL in connections to the registry server, do not require validation via a certifying authority, defaults to env[OS_INSECURE].

os_region_name

Type string

Default `RegionOne`

Authentication region name, defaults to env[OS_REGION_NAME].

os_interface

Type string

Default `u''`

Interface type. Valid options are public, admin and internal. defaults to env[OS_INTERFACE].

os_profile

Type string

Default `u''`

HMAC key for encrypting profiling context data, defaults to env[OS_PROFILE].

external_net

Type string

Default `<None>`

Name or ID of external network, defaults to env[SHAKER_EXTERNAL_NET]. If no value provided then Shaker picks any of available external networks.

dns_nameservers

Type list

Default `8.8.8.8,8.8.4.4`

Comma-separated list of IPs of the DNS nameservers for the subnets. If no value is provided defaults to Google Public DNS.

image_name

Type string

Default `shaker-image`

Name of image to use. The default is created by shaker-image-builder.

flavor_name

Type string

Default `shaker-flavor`

Name of image flavor. The default is created by shaker-image-builder.

stack_name

Type string

Default `<None>`

Name of test heat stack. The default is a uniquely generated name.

reuse_stack_name

Type string

Default `<None>`

Name of an existing Shaker heat stack to reuse. The default is to not reuse an existing stack. Caution should be taken to only reuse stacks meant for a specific scenario. Also certain configs e.g. image-name, flavor-name, stack-name, etc will be ignored when reusing an existing stack.

cleanup_on_exit

Type boolean

Default `true`

Clean up the heat-stack when exiting execution.

scenario

Type list

Default `<None>`

Comma-separated list of scenarios to play. Each entity can be a file name or one of aliases: "misc/instance_metadata", "openstack/cross_az/full_l2", "openstack/cross_az/full_l3_east_west", "openstack/cross_az/full_l3_north_south", "openstack/cross_az/perf_l2", "openstack/cross_az/perf_l3_east_west", "openstack/cross_az/perf_l3_north_south", "openstack/cross_az/udp_l2", "openstack/cross_az/udp_l2_mss8950", "openstack/cross_az/udp_l3_east_west", "openstack/dense_l2", "openstack/dense_l3_east_west", "openstack/dense_l3_north_south", "openstack/external/dense_l3_north_south_no_fip", "openstack/external/dense_l3_north_south_with_fip", "openstack/external/full_l3_north_south_no_fip", "openstack/external/full_l3_north_south_with_fip", "openstack/external/perf_l3_north_south_no_fip", "openstack/external/perf_l3_north_south_with_fip", "openstack/full_l2", "openstack/full_l3_east_west", "openstack/full_l3_north_south", "openstack/perf_l2", "openstack/perf_l3_east_west", "openstack/perf_l3_north_south", "openstack/qos/perf_l2", "openstack/udp_l2", "openstack/udp_l3_east_west", "openstack/udp_l3_north_south", "spot/ping", "spot/tcp", "spot/udp". Defaults to env[SHAKER_SCENARIO].

matrix

Type unknown type

Default `<None>`

Set the matrix of parameters for the scenario. The value is specified in YAML format. E.g. to override the scenario duration one may provide: "{time: 10}", or to override list of hosts: "{host:[ping.online.net, iperf.eenet.ee]}". When several parameters are overridden all combinations are tested

output

Type string

Default `u''`

File for output in JSON format, defaults to env[SHAKER_OUTPUT]. If it is empty, then output will be saved to /tmp/shaker_<time_now>.json

artifacts_dir

Type string

Default `<None>`

If specified, directs Shaker to store there all its artifacts (output, report, subunit and book). Defaults to env[SHAKER_ARTIFACTS_DIR].

no_report_on_error

Type boolean

Default `false`

Do not generate report for failed scenarios

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

scenario_availability_zone

Type list

Default `<None>`

Comma-separated list of availability_zone. If specified this setting will override the availability_zone accomodation setting in the scenario test definition.Defaults to SCENARIO_AVAILABILITY_ZONE

scenario_compute_nodes

Type integer

Default `<None>`

Number of compute_nodes. If specified this setting will override the compute_nodes accomodation setting in the scenario test definition. Defaults to SCENARIO_COMPUTE_NODES

custom_user_opts

Type unknown type

Default `<None>`

Set custom user option parameters for the scenario. The value is specified in YAML, e.g. custom_user_opts = { key1:value1, key2:value2} The values specified can be referenced in the usual python way. e.g. {{ CONF.custom_user_opts['key1'] }}. This option is useful to inject custom values into heat environment files

agent_loss_timeout

Type integer

Default `60`

Timeout to treat agent as lost in seconds, defaults to env[SHAKER_AGENT_LOSS_TIMEOUT]

agent_join_timeout

Type integer

Default `600`

Timeout to treat agent as join failed in seconds, defaults to env[SHAKER_AGENT_JOIN_TIMEOUT] (time between stack deployment and start of scenario execution).

report_template

Type string

Default `interactive`

Template for report. Can be a file name or one of aliases: "interactive", "json". Defaults to "interactive".

report

Type string

Default `<None>`

Report file name, defaults to env[SHAKER_REPORT].

subunit

Type string

Default <None>

Subunit stream file name, defaults to env[SHAKER_SUBUNIT].

book

Type string

Default <None>

Generate report in ReST format and store it into the specified folder, defaults to env[SHAKER_BOOK].

input

Type list

Default <None>

File or list of files to read test results from, defaults to env[SHAKER_INPUT].

agent_id

Type string

Default <None>

Agent unique id, defaults to MAC of primary interface.

agent_socket_recv_timeout

Type integer

Default <None>

The amount of time the socket will wait for a response from a sent message, in milliseconds.

agent_socket_send_timeout

Type integer

Default <None>

The amount of time the socket will wait until a sent message is accepted, in milliseconds.

agent_socket_conn_retries

Type integer

Default 10

Prior to exiting, the number of reconnects the Agent will attempt with the server upon socket operation errors.

image_builder_template

Type string

Default ubuntu

Heat template containing receipt of building the image. Can be a file name or one of aliases: “centos”, “debian”, “ubuntu”. Defaults to “ubuntu”.

flavor_ram

Type integer

Default `512`

Shaker image RAM size in MB, defaults to env[SHAKER_FLAVOR_RAM]

`flavor_vcpus`

Type integer

Default `1`

Number of cores to allocate for Shaker image, defaults to env[SHAKER_FLAVOR_VCPUS]

`flavor_disk`

Type integer

Default `3`

Shaker image disk size in GB, defaults to env[SHAKER_FLAVOR_DISK]

`image_builder_mode`

Type string

Default `<None>`

Valid Values heat, dib

Image building mode: "heat" - using Heat template (requires Glance v1 for base image upload); "dib" - using diskimage-builder elements (requires qemu-utils and debootstrap). If not set, switches to "dib" if Glance v1 is not available. Can be specified as env[SHAKER_IMAGE_BUILDER_MODE]

`image_builder_distro`

Type string

Default `ubuntu`

Valid Values ubuntu, centos7

Operating System Distribution for shaker image when using diskimage-builder, defaults to ubuntu

`cleanup`

Type boolean

Default `true`

Cleanup the image and the flavor.

`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	log-config
DEFAULT	log_config

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 setto "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 [%(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 %(tenant)s %(domain)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,amqplib=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.

CHAPTER 10

Contributing

10.1 Contribute to Shaker

Shaker follows standard OpenStack contribution workflow as described at https://docs.openstack.org/infra/manual/developers.html

10.1.1 Start working

1. Clone the repo:

```
$ git clone https://opendev.org/performa/shaker
```

2. From the root of your workspace, check out a new branch to work on:

```
$ git checkout -b <TOPIC-BRANCH>
```

3. Implement your code

10.1.2 Before Commit

4. Make sure your code works by running the tests:

```
$ tox
```

By default tox executes the same set of tests as configured in Jenkins, i.e.: py34 and py27 unit tests, pep8 style check and documentation build.

5. If there are any changes in config parameters, also do:

```
$ tox -egenconfig
```

This job updates sample config file as well as documentation on CLI utils.

10.1.3 Submit Review

6. Commit the code:

```
$ git commit -a
```

Commit message should indicate what the change is, for a bug fix commit it needs to contain reference to Launchpad bug number.

7. Submit the review:

```
$ git review
```

8. If the code is approved with a +2 review, Gerrit will automatically merge your code.

10.2 Bug Tracking

Bugs are tracked at Launchpad:

https://bugs.launchpad.net/shaker

10.3 Developer's Guide of OpenStack

If you would like to contribute to the development of OpenStack, you must follow the steps in this page:

https://docs.openstack.org/infra/manual/developers.html

Once those steps have been completed, changes to OpenStack should be submitted for review via the Gerrit tool, following the workflow documented at:

https://docs.openstack.org/infra/manual/developers.html#development-workflow

Note that the primary repo is https://opendev.org/performa/shaker/ Repos located at GitHub are mirrors and may be out of sync.

Project bug tracker is Launchpad:

https://launchpad.net/shaker