

OVS statistic collectd plugin tests.

- Table #1. OVS statistics collectd plugin tests results.

OVS events collectd plugin tests.

- Table #2. OVS events collectd plugin tests results.

OVS statistic SNMP tests.

- Table #3. SNMP: OVS statistic tests results.

OVS events multi instance collectd plugin tests.

- Table #4. OVS events multi instance collectd plugin tests results.

OVS stats multi instance collectd plugin tests.

- Table #5. OVS stats multi instance collectd plugin tests results.

OVS statistic collectd plugin tests.

Test Environment details:

- Bare Metal, Ubuntu 16.04.1 LTS
- Kernel version: 4.4.0-43-generic

Repo/branch used:

- collectd/master

Tests precondition:

OVS version - 2.5.2.

OpenvSwitch configuration:

- Start ovs: service openvswitch-switch start
- Add new bridge: ovs-vsctl add-br br0
- Allow connection to ovssdb-server: ovs-vsctl set-manager ptcp:6640

Ovs_stat configuration:

- Open collectd config file: vim /install_folder/etc/collectd.conf
- Uncomment LoadPlugin "ovs_stat" fields.
- Set up IP address, port, bridges (Address/port or OVS DB Socket):

```
<Plugin ovs_stats>
Port "6640"
Address "127.0.0.1"
Socket "/var/run/openvswitch/db.sock"
Bridges "br0"
</Plugin>
```

Table #1. OVS statistics collectd plugin tests results.

#	Test summary	Steps	Actual result	Expected result	Status - PASS/FAIL	Automated status	Comments
1	Verify user can setup simple configuration for ovs_stat plugin.	<ol style="list-style-type: none"> 1. Start collectd daemon : /install_folder/sbin/collectd 2. Navigate to: /tmp/hostname_folder/ 3. Check that separate files for all available interface monitoring statistic are present in folder : ls /ovs_stats-br0.br0 	<p>Folder "ovs_stats-br0.br0" was created</p> <p>Collectd started. Files in format \$monitoring_metrics-YYYY-MM-DD were created</p>	<p>Folder "ovs_stats-br0.br0" was created</p> <p>Collectd started. Files in format \$monitoring_metrics-YYYY-MM-DD were created</p>	PASS	Yes	
2	Check ovs_stat doesn't collect data when openvswitch is stopped	<ol style="list-style-type: none"> 1. Stop ovs: service openvswitch-switch stop 2. Start collectd daemon : /install_folder/sbin/collectd 3. Navigate to: /tmp/hostname_folder/ 4. Start ovs: service openvswitch-switch start 5. Navigate to: /tmp/hostname_folder/ovs_stats-br0.br0/ 6. Stop ovs: service openvswitch-switch stop 	<p>Folder "ovs_stats-br0.br0" wasn't created</p> <p>Files in format \$monitoring_metrics-YYYY-MM-DD were created</p> <p>New entry with data doesn't appear</p>	<p>Folder "ovs_stats-br0.br0" wasn't created</p> <p>Files in format \$monitoring_metrics-YYYY-MM-DD were created</p> <p>New entry with data appears with last collected value</p>	PASS (HAA-1118)	Yes	

		7. Execute command: tail -f /tmp/localhost/ovs_stats-br0.br0/if_packet s-YYYY-MM-DD					
3	Verify user configurations ovs_stats with unsupported parameter s collectd raise an error	<p>1. Open collectd.conf file: vim /install_folder/etc/collectd.conf</p> <p>2. Set up OVS Server field in ovs_stats plugin section to \$test_data</p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p> <p>4. Verify that collectd doesn't start and error raised.</p> <p>test_data = ["fsdfs"; 1146546, 1.5; \$%\$]</p>	Error raises for all \$test_data and collectd doesn't start	Error raises for all \$test_data and collectd doesn't start (need to be verified expected behavior)	PASS (HAA-1119)	Yes	
4	Check new data are collected by ovs_stats with a defined interval	<p>1. Start collectd daemon : /install_folder/sbin/collectd</p> <p>2. Execute command: tail -f /tmp/localhost/ovs_stats-br0.br0/if_packet s-2016-10-26</p> <p>3. Verify that collected values were renewed and written every 10 second (default value).</p>	<p>Every 10 second new data writes into file</p> <p>Every 1 second new</p>	<p>Every 10 second new data writes into file</p> <p>Every 1 second new</p>	PASS	Yes	

		<p>4. Change Interval from 10 to 1 in /install_folder/etc/collectd.conf file</p> <p>5. Restart collectd: service collectd restart</p> <p>6. Execute command: tail -f /tmp/localhost/ovs_stats-br0.br0/if_packet_s-YYYY-MM-DD</p> <p>7. Verify that collected values were renewed and written every 1 second.</p>	data writes into file	data writes into file			
5	Check if LoadPlugin section collectd.conf is commented out, warning message appears	<p>1. Open collectd.conf file: vim /install_folder/etc/collectd.conf</p> <p>2. Configure ovs_stats: Comment out #LoadPlugin ovs_stats</p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p>	<p>1. Warning message appears: "Loadplugin Found a configuration for the 'ovs_stats' plugin, but the plugin isn't loaded or didn't register a configuration callback."</p> <p>2. Collectd have started.</p>	<p>1. Warning message appears: "Loadplugin Found a configuration for the 'ovs_stats' plugin, but the plugin isn't loaded or didn't register a configuration callback."</p> <p>2. Collectd have started.</p>	PASS	Yes	
6	Verify if Address, Port, Bridges configuration is empty in collectd.conf	<p>1. Open collectd.conf file: vim /install_folder/etc/collectd.conf</p> <p>2. Configure ovs_stats : leave</p>	Collectd started with default parameters.	Collectd started with default parameters.	PASS	Yes	

	<i>nf default parameters take place</i>	<p>Bridges section blank</p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p> <p>4. Configure ovs_stats : leave OVS Address, Port, Bridges section blank</p> <p>5. Start collectd daemon : /install_folder/sbin/collectd</p>					
7	<i>Verify if in ovs_stats section is present extra keyword with value, collectd fails and error message appears</i>	<p>1. Open collectd.conf file: vim /install_folder/etc/collectd.conf</p> <p>2. Configure ovs_stats : add extra key - value: Interval 1</p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p>	<p>1. Message: ovs_stats: option 'Interval' not allowed here</p> <p>2. ovs_stats and collectd failed</p>	<p>1. Message: ovs_stats: option 'Interval' not allowed here</p> <p>2. ovs_stats and collectd failed</p>	PASS	Yes	
8	<i>Check if LoadPlugin configuration section in collectd.conf is commented out, ovs_stats using default configuration values</i>	<p>1. Open collectd.conf file: vim /install_folder/etc/collectd.conf</p> <p>2. Comment out: #<Plugin ovs_stats> # Address "127.0.0.1" #Port "6640" # Bridges "br0" #</Plugin></p> <p>3. Start collectd daemon :</p>	<p>Files in format \$monitoring_metrics-YYYY-MM-DD were created. Default values are used for OVS.</p> <p><u>monitoring metrics</u></p> <p>if_collisions if_dropped if_errors</p>	<p>Files in format \$monitoring_metrics-YYYY-MM-DD were created. Default values are used for OVS.</p> <p><u>monitoring metrics</u></p> <p>if_collisions if_dropped if_errors</p>	PASS	Yes	

		/install_folder/sbin/collectd 4. Check that separate files for all available interface monitoring statistic are present in folder : ls /ovs_stats-br0.br0	if_packets if_rx_errors-crc if_rx_errors-frame if_rx_errors-over if_rx_octets if_tx_octets	if_packets if_rx_errors-crc if_rx_errors-frame if_rx_errors-over if_rx_octets if_tx_octets			
9	Verify that ovs_stats supports complex (more than 1 bridge) configuration	1. Add new openvswitch instance: ovs-vsctl add-br br1 2. Add created switch into ovs_stats configuration section: Bridges "br0" "br1" 3. Start collectd daemon : /install_folder/sbin/collectd 4. Check that separate folders were created for each bridge in config. 5. Move to each created folder.	Folders: ovs_stats-br0.br0 , ovs_stats-br1.br1 were created Files in format \$monitoring_metrics-YYYY-MM-DD were created into each folder	Folders: ovs_stats-br0.br0 , ovs_stats-br1.br1 were created Files in format \$monitoring_metrics-YYYY-MM-DD were created into each folder	PASS	Yes	
10	Check that there are no double entries in data created files	1. Start collectd daemon : /install_folder/sbin/collectd 2. Execute command: cat /tmp/hostname_folder/ovs_stats-br0.br0/if_dropped-YYYY-MM-DD 3. Verify that there is only one entry for collected	One entry per time interval	One entry per time interval.	PASS (HAA-1117)	Yes	

		test data per interval.					
1 1	Verify that ovs_stats collectd data from new bridges (bridge interfaces) when they were added after collectd and ovs_stats have started	<p>1. Start collectd daemon : /install_folder/sbin/collectd</p> <p>2. Add new bridge: ovs-vsctl add-br new_bridge</p> <p>3. Navigate to: /tmp/hostname_folder/</p> <p>4. Check that separate files for all available interface monitoring statistic are present in folder : ls /ovs_stats-new_bridge.new_bridge</p> <p>5. Add port to br0: ovs-vsctl add-port br0 test_port1 -- set interface test_port1 type=internal</p> <p>6. Navigate to: /tmp/hostname_folder/</p> <p>7. Check that separate files for all available interface monitoring statistic are present in folder : ls /ovs_stats-br0.test_port1</p>	<p>Folder "ovs_stats-new_bridge.new_bridge" was created</p> <p>Files in format \$monitoring_metrics-YYYY-MM-DD were created into folder</p> <p>Folder "ovs_stats-br0.test_port1" was created</p> <p>Files in format \$monitoring_metrics-YYYY-MM-DD were created into each folder</p>	<p>Folder "ovs_stats-new_bridge.new_bridge" was created</p> <p>Files in format \$monitoring_metrics-YYYY-MM-DD were created into folder</p> <p>Folder "ovs_stats-br0.test_port1" was created</p> <p>Files in format \$monitoring_metrics-YYYY-MM-DD were created into each folder</p>	PASS	Yes	
1 2	Check that ovs_stats collects data from ovsdb.	1. Start python socket which emulates ovsdb replies.	\$test_data was written into appropriate files	\$test_data was written into appropriate files	PASS	Yes	

		<p>2.Start collec d daemon : /install_fold er/sbin/colle ctd</p> <p>3. Send ovsdb reply with \$test_data.</p> <p>4. Check that sended \$test_data was written in appropriate files fro br0.br0 interface: cat /ovs_stats- br0.br0/\$monit oring_metrics files</p> <p>test_data:</p> <p>if_collisions 50 if_dropped 100 if_errors 125 if_packets 300 if_rx_errors-crc 512 if_rx_errors- frame 345 if_rx_errors-over 800 if_rx_octets 93660 if_tx_octets 45646</p>					
1 3	Verify plugin stops collecting data for specific bridge when bridge is removed from ovs	<p>1. Start collec d daemon : /install_fold er/sbin/colle ctd</p> <p>2. Check that ovs_stats collecting data: cat /ovs_stats- br0.br0/\$monitor ing_metrics files</p> <p>3. Delete ovs: ovs-vsctl del-br br0.</p>	<p>New entry with data appears</p> <p>New entry with data doesn't appear</p>	<p>New entry with data appears</p> <p>New entry with data doesn't appear</p>	PASS	Yes	

		4. Check that ovs_stats stop collecting data: cat /ovs_stats-br0.br0/\$monitoring_metrics files					
1 4	Check plugin stops collecting data for specific port when port is removed from bridge	<p>1. Start collectd daemon : /install_folder/sbin/collectd</p> <p>2. Add port to br0: ovs-vsctl add-port br0 test_port1 -- set interface test_port1 type=internal</p> <p>3. Delete port: ovs-vsctl del-port test_port1</p> <p>4. Check that ovs_stats stop collecting data: cat /ovs_stats-br0.test_port1/\$monitoring_metrics files</p>	<p>Folder "ovs_stats-br0.test_port1" was created</p> <p>New entry with data doesn't appear</p>	<p>Folder "ovs_stats-br0.test_port1" was created</p> <p>New entry with data doesn't appear</p>	PASS	Yes	
1 5	Ovs_stats plugin: overflow stats counter	<p>1. Start python socket which emulates ovsdb replies.</p> <p>2. Set if_rx_errors-over value > 2^63</p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p> <p>4. Check handled data.</p>	<p>New entry with value 2^63-1 appears into if_rx_errors-over file. Errors or exceptions weren't raised.</p> <p>Counter's values should wrap around (max value, zero, one, so on)</p>	<p>New entry with value 2^63-1 appears into if_rx_errors-over file. Errors or exceptions weren't raised.</p> <p>Counter stuck at max value.</p>	PASS	Yes	

1 6	Verify OVS stats IP address field could be set up to work in either IPv4 or IPv6	Set up Interface filed in collectd.conf file to IPv4. Configure ovs-manager for listening IPv4. Start collectd. Set up Interface filed in collectd.conf file to IPv6. Configure ovs-manager for listening IPv6. Start collectd.	New entries are created for ovs_stats files. New entries are created for ovs_stats files.	New entries are created for ovs_stats files. New entries are created for ovs_stats files.	PASS	Yes	
1 7	Verify OVS stats updated only for bridges are written to the collectd.conf	Create new OVS bridge. Configure collectd.conf. Start collectd.	Done. Collectd is started. OVS stats only updated for bridges set in collectd.conf .	Done. Collectd is started. OVS stats only updated for bridges set in collectd.conf .	PASS	Yes	
1 8	Verify OVS stats statistic is updated over socket created by address/port pair	Stop openvswitch service. Configure collectd.conf to work via address/port. Start collectd. Emulate OVS server starting from zero values.	Success. Success. OVS stats counters and CSV write plugin data are the same	Success. Success. OVS stats counters and CSV write plugin data are the same	PASS	Yes	
1 9	Verify OVS stats statistic Rx packets and bytes counters sent from Ixia and read from	Add linux ports to OVS bridge. Start collectd. Configure streams, clear statistic, send packets.	Collect is running. Verify clear traffic run through OVS. Verify sent packets/bytes and OVS stats values	Collect is running. Verify clear traffic run through OVS. Verify sent packets/bytes and OVS stats values	PASS	Yes	

	CSV plugin.		are the same. Verify OVS stats and OVS plugin values (using CSV plugin) are the same.	are the same. Verify OVS stats and OVS plugin values (using CSV plugin) are the same.			
20	Verify bad CRC packets correct counting by OVS plugin.	Add linux ports to OVS bridge. Start collectd. Configure bad CRC streams, clear statistic, send packets. Stop sending packets.	Collectd is running. Bad CRC counter is being increased. Verify bad CRC counters sent from Ixia and in OVS stats are the same. Verify bad CRC counters in OVS stats and OVS stats plugin (using CSV plugin) are the same.	Collectd is running. Bad CRC counter is being increased. Verify bad CRC counters sent from Ixia and in OVS stats are the same. Verify bad CRC counters in OVS stats and OVS stats plugin (using CSV plugin) are the same.	PASS	Yes	
21	Verify rx dropped packets correct counting by OVS plugin.	Add linux ports to OVS bridge. Start collectd. Configure bad CRC streams, clear statistic, send packets with 100% rate (10Gb). Stop sending packets.	Collectd is running. Packets got dropped. Verify rx dropped counters in OVS stats and OVS stats plugin (using CSV plugin) are the same.	Collectd is running. Packets got dropped. Verify rx dropped counters in OVS stats and OVS stats plugin (using CSV plugin) are the same.	PASS	Yes	

OVS events collectd plugin tests.

Test Environment details:

- Bare Metal, Ubuntu 16.04.1 LTS
- Kernel version: 4.4.0-43-generic

Repo/branch used:

- collectd/master

Tests precondition:

OpenvSwitch configuration:

- Start ovs: service openvswitch-switch start
- Add new bridge: ovs-vsctl add-br br0
- Allow connection to ovsdb-server: ovs-vsctl set-manager ptcp:6640

Ovs_events configuration:

- Open collectd config file: vim /install_folder/etc/collectd.conf
- Uncomment LoadPlugin "ovs_events" fields.
- Set up Port "6640", Address "127.0.0.1", Interfaces "br0"

Ovs_events configuration:

- Open collectd config file: vim /install_folder/etc/collectd.conf
- Uncomment LoadPlugin "ovs_events" fields.
- Set up Port "6640", Address "127.0.0.1", Interfaces "br0"

Exec plugin configuration:

- Create non root user (example "test") and put bash file:

```
#!/bin/bash
rm -f /tmp/notifications
while read x y
do
echo $x$y >> /tmp/notifications
done
```

- Open collectd config file: vim /install_folder/etc/collectd.conf
- Configure exec plugin: <LoadPlugin exec>
<Plugin exec>
Exec "user_name:user_group" "path_to_bash_script"
NotificationExec "user_name:user_group" "path_to_bash_script"
</Plugin>

Syslog plugin configuration:

- Configure:

```
<Plugin syslog>
LogLevel debug
NotifyLevel "OKAY"
</Plugin>
```

Table #2. OVS events collectd plugin tests results.

#	Test summary	Steps	Actual result	Expected result	Status - PASS/FAIL	Automated status	Comments
1	Verify user can setup simple configuration for ovs_events plugin.	1. Start collectd daemon : /install_folder/sbin/collectd 2. Navigate to: /tmp/hostname_folder/ 3. Check that file with link status was created: ls /ovs_events-br0	Folder "ovs_events-br0" was created. File in format "gauge-link_status-YYYY-MM-DD" was created	Folder "ovs_events-br0" was created. File in format "gauge-link_status-YYYY-MM-DD" was created	PASS	YES	
2	Check ovs_events doesn't collect data when openvswitch is stopped	1. Stop ovs: service openvswitch-switch stop 2. Start collectd daemon : /install_folder/sbin/collectd 3. Navigate to: /tmp/hostname_folder/ 4. Start ovs: service openvswitch-switch start 5. Navigate to: /tmp/hostname_folder/ovs_events-br0/ 6. Stop ovs: service openvswitch-switch stop 7. Execute command: tail -f /tmp/localhost/ovs_events-br0/gauge-link_status-YYYY-MM-DD	Folder "ovs_events-br0" wasn't created File in format "gauge-link_status-YYYY-MM-DD" was created New entry with data	Folder "ovs_events-br0" wasn't created File in format "gauge-link_status-YYYY-MM-DD" was created New entry with data	PASS	YES	

			appears with last collected value	appears with last collected value			
3	Check new data are collected by ovs_events with a defined interval	<p>1. Start collectd daemon : /install_folder/sbin/collectd</p> <p>2. Execute command: tail -f /tmp/localhost/ovs_events-br0/gauge-link_status-YYYY-MM-DD</p> <p>3. Verify that collected values were renewed and written every 10 second (default value).</p> <p>4. Change Interval from 10 to 1 in /install_folder/etc/collectd.conf file</p> <p>5. Restart collectd: service collectd restart</p> <p>6. Execute command: tail -f /tmp/localhost/ovs_events-br0/gauge-link_status-YYYY-MM-DD</p> <p>7. Verify that collected values were renewed and written every 1 second.</p>	<p>Every 10 second new data writes into file</p> <p>Every 1 second new data writes into file</p>	<p>Every 10 second new data writes into file</p> <p>Every 1 second new data writes into file</p>	PASS	YES	
4	Check if Plugin configuration section in collectd.conf is commented out, ovs_events using default configuration values	<p>1. Open collectd.conf file: vim /install_folder/etc/collectd.conf</p> <p>2. Comment out: #<Plugin ovs_events> # Port "6640" # Address "127.0.0.1" # Socket "/var/run/openvswitch/db.sock" # Interfaces "br0" # SendNotification false #</Plugin></p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p> <p>4. Check that file for interface is present in folder : ls /ovs_events-br0</p>	<p><u>default values:</u> Port "6640" Address "localhost" Interface empty SendNotification false</p>	<p><u>default values:</u> Port "6640" Address "localhost" Interface empty SendNotification false</p>	PASS	YES	

			File in format "gauge-link_status-YYYY-MM-DD" was created	File in format "gauge-link_status-YYYY-MM-DD" was created			
5	Verify that ovs_events supports complex (more than 1 bridge) configuration	<p>1. Add new openvswitch instance: ovs-vsctl add-br br1</p> <p>2. Add created switch into ovs_events configuration section: Bridges "br0" "br1"</p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p> <p>4. Check that separate folders were created for each bridge in config.</p> <p>5. Move to each created folder.</p>	<p>Folders: ovs_events-br0, ovs_events-br1 were created</p> <p>File in format "gauge-link_status-YYYY-MM-DD" was created for each folder</p>	<p>Folders: ovs_events-br0, ovs_events-br1 were created</p> <p>File in format "gauge-link_status-YYYY-MM-DD" was created for each folder</p>	PASS	YES	
6	Check that there are no double entries in data created files	<p>1. Start collectd daemon : /install_folder/sbin/collectd</p> <p>2. Execute command: cat /tmp/hostname_folder/ovs_events-br0/gauge-link_status-YYYY-MM-DD</p> <p>3. Verify that there are only one entry for collected test data per interval.</p>	One entry per time interval	One entry per time interval	PASS	YES	
7	Verify that ovs_events collectd data from new	<p>1. Start collectd daemon : /install_folder/sbin/collectd</p> <p>2. Add new bridge: ovs-vsctl add-br new_bridge</p> <p>3. Navigate to: /tmp/hostname_folder/</p>	Folder "ovs_events-new_bridge" was created	Folder "ovs_events-new_bridge" was created	PASS	YES	

	bridges (bridge interfaces) when they were added after collectd and ovs_eventstst have started	<p>4. Check that file with link status was created: ls /ovs_events-new_bridge</p> <p>5. Add port to br0: ovs-vsctl add-port br0 test_port1 -- set interface test_port1 type=internal</p> <p>6. Navigate to: /tmp/hostname_folder/</p> <p>7. Check that file with link status was created: ls /ovs_events-test_port1</p>	<p>File in format "gauge-link_status-YYYY-MM-DD" was created</p> <p>Folder "ovs_events-test_port1" was created</p> <p>File in format "gauge-link_status-YYYY-MM-DD" was created</p>	<p>File in format "gauge-link_status-YYYY-MM-DD" was created</p> <p>Folder "ovs_events-test_port1" was created</p> <p>File in format "gauge-link_status-YYYY-MM-DD" was created</p>			
8	Check that ovs_events collect data from ovsdb	<p>1. Start python socket which emulates ovsdb replies.</p> <p>2. Start collectd daemon : /install_folder/sbin/collectd</p> <p>3. Send ovsdb reply with "link_state":"up".</p> <p>4. Check that sent data was written in appropriate files from br0 interface: cat /ovs_events-br0/gauge-link_status-YYYY-MM-DD</p>	<p>Link status was changed from 0 to 1</p>	<p>Link status was changed from 0 to 1</p>	PASS	-	
9	Verify plugin stops collecting data for specific bridge when bridge is removed	<p>1. Start collectd daemon : /install_folder/sbin/collectd</p> <p>2. Check that ovs_events collecting data: cat /ovs_events-br0/gauge-link_status-YYYY-MM-DD</p> <p>3. Delete ovs: ovs-vsctl del-br br0.</p> <p>4. Check that ovs_events stop collecting data: cat</p>	<p>New entry with data appears</p> <p>New entry with data</p>	<p>New entry with data appears</p> <p>New entry with data</p>	PASS	YES	

	d from ovs	/ovs_events-br0/gauge-link_status-YYYY-MM-DD	doesn't appear	doesn't appear			
10	Check plugin stops collecting data for specific port when port is removed from bridge	<p>1. Start collectd daemon : /install_folder/sbin/collectd</p> <p>2. Add port to br0: ovs-vsctl add-port br0 test_port1 -- set interface test_port1 type=internal</p> <p>3. Delete port: ovs-vsctl del-port test_port1</p> <p>4. Check that ovs_events stop collecting data: cat /ovs_events-br0.test_port1/gauge-link_status-YYYY-MM-DD</p>	<p>Folder "ovs_events-test_port1" was created</p> <p>New entry with data doesn't appear</p>	<p>Folder "ovs_events-test_port1" was created</p> <p>New entry with data doesn't appear</p>	PASS	-	
11	Verify that user can set up into Address : network hostname, IPv4 number s-and-dots notation or IPv6 hexadecimal string format	<p>1. Start collectd daemon : /install_folder/sbin/collectd</p> <p>2. Check that ovs_events collecting data: cat /ovs_events-br0.test_port1/gauge-link_status-YYYY-MM-DD</p> <p>3. Set up Address "\$hostname", where hostname is name of linux machine (ex: silpixa00397261)</p> <p>4. Restart collectd: service collectd restart</p> <p>5. Check that ovs_events collecting data: cat /ovs_events-br0.test_port1/gauge-link_status-YYYY-MM-DD</p> <p>6. Allow connection to ovsdb-server using ipv6: ovs-vsctl set-manager tcp:::6640:::1</p> <p>7. Set up Address "::<"</p> <p>8. Restart collectd: service collectd restart</p> <p>9. Check that ovs_events collecting data: cat</p>	<p>New entry with data appears</p>	<p>New entry with data appears</p>	PASS	YES	

		/ovs_events-br0.test_port1/gauge-link_status-YYYY-MM-DD					
1 2	Check if Socket path is set, Address and Port options are ignored	<p>1. Set up Address "192.168.0.1", Port "8956" and Socket "/var/run/openvswitch/db.sock"</p> <p>2. Start collectd daemon : /install_folder/sbin/collectd</p> <p>3. Check that ovs_events collecting data: cat /ovs_events-br0.test_port1/gauge-link_status-YYYY-MM-DD</p>	<p>Set up for Address and Port was ignored and ovs_events using Socket options.</p> <p>New entry with data appears</p>	<p>Set up for Address and Port was ignored and ovs_events using Socket options.</p> <p>New entry with data appears</p>	PAS S	YES	
1 3	Verify if SendNotification disabled ovs_events doesn't send any notifications	<p>1. Set up SendNotification false.</p> <p>2. Set up NotifyLevel OKAY for syslog plugin.</p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p> <p>4. Run command: tail -f /var/log/syslog</p> <p>5. Enable br0 interface: ifconfig br0 up</p>	<p>Notification message doesn't appear into syslog file</p>	<p>Notification message doesn't appear into syslog file</p>	PAS S	YES	
1 4	Check if SendNotification enabled ovs_events send notifications when - ovs interface (bridge) link status is changed -	<p>1. Set up SendNotification true.</p> <p>2. Set up NotifyLevel OKAY for syslog plugin.</p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p> <p>4. Run command: tail -f /var/log/syslog</p> <p>5. Enable br0 interface: ifconfig br0 up</p> <p>6. Stop ovs: service openvswitch-switch stop</p>	<p>Notification message about link status appears into syslog file</p>	<p>Notification message about link status appears into syslog file</p>	PAS S	YES	

	connecti on to OVS databas e (ovsdb- server daemon) has been lost.		Notificati on message about ovsdb connecti on appears into syslog file	Notificati on message about ovsdb connecti on appears into syslog file			
1 5	Check if "Interfac e" is commen ted out all availabl e interfac es are monitor ed	1. Add new openwswitch instance: ovs-vsctl add-br br1 2. Comment out #Interfaces and set SendNotification true 3. Enable exec plugin. 4. Run command: tail -f /var/log/syslog 5. Start collecd deamon : /install_folder/sbin/collect 6. Check that ovs_events collecting link status and metadata for br0 and br1 : cat path_to_exec_write_file.	Notificati on message about link status appears into syslog file for br0 and br1 interface s Uuid and vm-uuid are present in meta field New entry with data appears	Notificati on message about link status appears into syslog file for br0 and br1 interface s Uuid and vm-uuid are present in meta field New entry with data appears	PAS S	YES	
1 6	Verify that only specifie d interfac e names in Interfac e field are monitor ed by	1. Add new openwswitch instance: ovs-vsctl add-br br1 2. Set Interface "br0" 3. Start collecd deamon : /install_folder/sbin/collect 4. Check that ovs_events collecting link status and metadata only for br0 :cat path_to_exec_write_file.	Uuid and vm-uuid are present in meta field New entry with data appears Folder/fil e for br1	Uuid and vm-uuid are present in meta field New entry with data appears Folder/fil e for br1	PAS S	YES	

	ovs_events plugin	<p>5. Add port br1 into Interface section.</p> <p>6. Restart collectd.</p> <p>7. Check that ovs_events collecting link status and metadata for br0 and br1 : cat path_to_exec_write_file.</p>	<p>weren't created</p> <p>Uuid and vm-uuid are present in meta field New entry with data appears</p>	<p>weren't created</p> <p>Uuid and vm-uuid are present in meta field New entry with data appears</p>			
17	Check that metadata are present when notification raises	<p>1. Set SendNotification true.</p> <p>2. Enable exec plugin for handling metadata.</p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p> <p>4. Execute: ifconfig br0 up</p> <p>5. Check that notification raises.</p> <p>6. Verify metadata collected by exec. Execute: ovs-vsctl list interface br0</p>	<p>Notification message about link status appears into syslog file br0 interfaces</p> <p>_uuid = uuid collected by exec, vm-uuid = vm_uuid collected by exec</p>	<p>Notification message about link status appears into syslog file br0 interfaces</p> <p>_uuid = uuid collected by exec, vm-uuid = vm_uuid collected by exec</p>	PASS	YES	
18	Verify that ovs_events collecting data from ovs	<p>1. Add new openvswitch instance: ovs-vsctl add-br br1</p> <p>2. Add patch interfaces to br0 and br1: ovs-vsctl add-port br0 test_1 – set Interface test_1</p>			PASS	YES	

	patch interfaces	<p>type=patch options:peer=test_2</p> <p>ovs-vsctl add-port br1 test_2 – set Interface test_2 type=patch options:peer=test_1</p> <p>3. Add br1, test_1, test2 to collectd.conf</p> <p>4. Start collectd daemon : /install_folder/sbin/collect</p> <p>5. Check that separate folders were created for each bridge in config.</p> <p>6. Move to each created folder.</p>	<p>Folders: ovs_events-br0, ovs_events-br1, ovs_events-test1, ovs_events - test_2 were created</p> <p>File in format "gauge-link_statuses-YYYY-MM-DD" was created for each folder</p>	<p>Folders: ovs_events-br0, ovs_events-br1, ovs_events-test1, ovs_events - test_2 were created</p> <p>File in format "gauge-link_statuses-YYYY-MM-DD" was created for each folder</p>			
19	Check ovs_events handle data from ovs dpdk interfaces	<p>1. Configure ovs with dpdk (https://github.com/jrfastab/OVS/blob/master/INSTALL.DPK.md)</p> <p>2. Add 2 dpdk ports into ovs.</p> <p>3. Start collectd daemon : /install_folder/sbin/collect</p> <p>4. Check that separate folders were created for each bridge in config.</p> <p>5. Move to each created folder.</p>	<p>Folders: ovs_events-br0, ovs_events-br1, ovs_events-dpdk0, ovs_events-dpdk1 were created</p> <p>File in format "gauge-link_statuses-YYYY-MM-DD" was created for each folder</p>	<p>Folders: ovs_events-br0, ovs_events-br1, ovs_events-dpdk0, ovs_events-dpdk1 were created</p> <p>File in format "gauge-link_statuses-YYYY-MM-DD" was created for each folder</p>	PASS	YES	
20	Verify if Dispatch false.	<p>1. Set up DispatchValues false.</p>			PASS	YES	

	<i>hValues disabled ovs_events does not handle data about link status</i>	<p>2. Start collectd daemon : /install_folder/sbin/collect.</p> <p>3. Check that link status from ovsdb does not handle.</p> <p>4. Set up DispatchValues true.</p> <p>5. Restart collectd: service collectd restart</p>	Data about link status doesn't collect	Data about link status doesn't collect			
2 1	Check ovs_events handle data about link status and events from 30 openvswitch bridges	<p>1. Add 30 ovs-bridges.</p> <p>2. Start collectd daemon : /install_folder/sbin/collect.</p> <p>3. Check that link status from ovsdb handle by ovs_events.</p> <p>4. Change link status to up.</p>	<p>Link status for all 30 bridges was handled</p> <p>Link status for all 30 bridges was changed to "Up"</p> <p>30 events appeared in syslog</p>	<p>Link status for all 30 bridges was handled</p> <p>Link status for all 30 bridges was changed to "Up"</p> <p>30 events appeared in syslog</p>	PASS	YES	
2 2	Verify ovs_events handle data about link status every 0.005 sec	<p>1. Set Interval = 0.005</p> <p>2. Start collectd daemon : /install_folder/sbin/collect.</p> <p>3. Check that link status from ovsdb handle by ovs_events every 0.005 sec.</p>	Data about link status appears every 0.005 sec	Data about link status appears every 0.005 sec	PASS	YES	
2 3	Verify ovs_events does	1. Set up Socket to invalid link			PASS	YES	

	not handle data when invalid socket path is set	"/var/test/openvswitch/db.sock" 2. Start collectd daemon : /install_folder/sbin/collectd	Folders and file for ovs_events were not created Collectd have started	Folders and file for ovs_events were not created Collectd have started			
24	Verify ovs_events event time less or equal to 0.005 seconds	1. Add linux port to ovs 2. Change port status to UP 3. Compare result from kernel log and raised event. 4. Set up port status DOWN 5. Compare result from kernel log and raised event.	Event raised time less or equal 0.005 sec Event raised time less or equal 0.005 sec	Event raised time less or equal 0.005 sec Event raised time less or equal 0.005 sec	PASS	YES	event time ~ 0.001-0.002 sec
25	Verify that ovs_events handle data from virtual function interfaces	1. Create max VF for interfaces with 'i40e' and 'ixgbe' 2. Change port status to UP 3. Set up port status DOWN	Port status for selected VM changed to 1 Port status for selected VM changed to 0	Port status for selected VM changed to 1 Port status for selected VM changed to 0	PASS	YES	
26	Check if NotifyInterfaceAdded set to False notification message	1. Add new port to ovs bridge: ovs-vsctl add-port br0 test_port 2. Using exec plugin handle notification. 3. Remove test_port from ovs.	Notification message with test_port state	Notification message with test_port state	PASS	NO	

	<i>e after port/inte rface added didn't appear</i>	4. Set NotifyInterfaceAdd false. 5. Restart collectd: service collectd restart	changed to "DOWN" appears Notificati on message didn't appear	changed to "DOWN" appears Notificati on message didn't appear			
--	---	---	---	---	--	--	--

OVS statistic SNMP tests.

Important:

OvS stats can be read directly from OvS using standard SNMP MIBs (IF_MIB) and this does not require OvS plugin availability.

Tests precondition:

1. SNMP installed and configured.
2. OvS installed and running.
3. Standard SNMP MIBS are installed.
4. Important NOTE: No need in OvS plugin.

Useful information:

1. IF-MIB starting OID
.1.3.6.1.2.1.2
2. Reading list of interfaces from Standard IF_MIB
snmpwalk -v2c -cpublic localhost .1.3.6.1.2.1.2.2.1.2
3. Actual OID for OvS interfaces will be stable during current system run, but will differs between different system runs. For example:

.....

iso.3.6.1.2.1.2.2.1.2.311 = STRING: "br0"
iso.3.6.1.2.1.2.2.1.2.314 = STRING: "test_port1"

4. List of Counters means simple counters, not detailed.
For example:

iso.3.6.1.2.1.2.2.1.10.311 = Counter32: 0
iso.3.6.1.2.1.2.2.1.11.311 = Counter32: 0
iso.3.6.1.2.1.2.2.1.12.311 = Counter32: 0
iso.3.6.1.2.1.2.2.1.13.311 = Counter32: 6524
iso.3.6.1.2.1.2.2.1.14.311 = Counter32: 0
iso.3.6.1.2.1.2.2.1.15.311 = Counter32: 0

iso.3.6.1.2.1.2.2.1.16.311 = Counter32: 6025
iso.3.6.1.2.1.2.2.1.17.311 = Counter32: 48
iso.3.6.1.2.1.2.2.1.18.311 = Counter32: 0
iso.3.6.1.2.1.2.2.1.19.311 = Counter32: 0
iso.3.6.1.2.1.2.2.1.20.311 = Counter32: 0

Table #3. SNMP: OVS statistic tests results.

#	Test case title	Priority	Steps	Expected result	Actual result	Status	Environment	Is automated	Automation result	Comments
1	Verify that SNMP will return all interfaces including OvS interfaces by SNMP WALK standard IF_MIB	High	Read list of interfaces from IF_MIB Start OvS and add an interfaces to it Read list of interfaces from IF_MIB	OvS interfaces are available in output	OvS interfaces are available in output	PASS	TBD	No	NA	
2	Verify that SNMP can GET interface info of OvS interfaces from standard IF_MIB	High	Start OvS, add interfaces Find interfaces number in IF_MIB	Interface name, counters can be get by appropriate OID number	Interface name, counters returned by appropriate OID number	PASS	TBD	No	NA	

			Get Interface name Get interface counters							
3	Verify that reading SNMP value using GETNEXT behaves correctly	High	Start OvS, add interfaces Try GetNext on previous OID Try GetNext on interface's OID	Value next after specified is returned	Value next after specified is returned	PASS	TBD	No	NA	
4	Verify that SET doesn't work on counters.	High	Start OvS, add interfaces Try to set counter's value for interface	Attempt failed with correct error message	Attempt failed	PASS	TBD	No	NA	
5	Verify that counters can	High	Start OvS, add multi	Appropriate values for	Appropriate values for	PASS	TBD	No	NA	

	be read if multiple Ovs interfaces are available		ply interfaces Try get info for different interfaces	expected interface is returned	expected interface is returned					
--	--	--	--	--------------------------------	--------------------------------	--	--	--	--	--

OVS events multi instance collectd plugin tests.

Ovs_events configuration:

- Open collectd config file: vim /install_folder/etc/collectd.conf
- Uncomment LoadPlugin "ovs_events" fields.
- Uncomment

```
<Plugin ovs_events>
<Instance hostname>
DispatchValues true
</Instance>
SendNotification true
</Plugin>
```

OpenvSwitch configuration:

- Start ovs: service openvswitch-switch start
- Add new bridge: ovs-vsctl add-br br0
- Allow connection to ovsdb-server: ovs-vsctl set-manager tcp:6640

Table #4. OVS events multi instance collectd plugin tests results.

#	Test summary	Steps	Actual result	Expected result	Status - PASS/ FAIL	Automated status	Comments
1	Verify user can setup simple configuration	1. Start collectd daemon : /install_folder/sbin/collectd	Folder "ovs_events-br0" was created.	Folder "ovs_events-br0" was created.	PASS		

#	Test summary	Steps	Actual result	Expected result	Status - PASS/ FAIL	Automated status	Comments
	of ovs_events for DUT	2. Navigate to: /tmp/hostname_folder/ 3. Check that file with link status was created: ls /ovs_events-br0	File in format "gauge-link_status-YYYY-MM-DD" was created	File in format "gauge-link_status-YYYY-MM-DD" was created			
2	Check ovs_events handle data from remote DUT	1. Add Address = "address of remote DUT" 2. On remote DUT add ovs bridge br0. 3. Start collectd daemon : /install_folder/bin/collectd 4. Navigate to: /tmp/hostname_folder/ 5. Check that file with link status was created: ls /ovs_events-br0 6. Change link status of br0 to UP.	Folder "ovs_events-br0" was created File in format "gauge-link_status-YYYY-MM-DD" was created New entry with data appears with last collected value 1	Folder "ovs_events-br0" was created File in format "gauge-link_status-YYYY-MM-DD" was created New entry with data appears with last collected value 1	PASS		
3	Verify user can setup multi instance configuration for	1. Add new instance for remote DUT: <Instance remote_host> DispatchValues true Address	Folders with appropriate	Folders with appropriate	PASS		

#	Test summary	Steps	Actual result	Expected result	Status - PASS/ FAIL	Automated status	Comments
	ovs_events plugin	<p>"remote IP"</p> <p></Instance></p> <p>2. On remote DUT add ovs bridge br0.</p> <p>3. Start collectd daemon : /install_folder/bin/collectd</p> <p>4. Navigate to: /tmp/hostname_folder/</p> <p>5. Change link status of br0 on DUT and remote DUT to UP.</p>	<p>files in /tmp/hostname_folder/</p> <p>and /tmp/remote_host were created</p> <p>New entry with data appears with last collected value 1 for local and remote DUT</p>	<p>files in /tmp/hostname_folder/</p> <p>and /tmp/remote_host were created</p> <p>New entry with data appears with last collected value 1 for local and remote DUT</p>			
4	Check ovs_events plugin multi instance configuration handle notification events	<p>1. Add new instance for remote DUT:</p> <p><Instance remote_host></p> <p>DispatchValues true</p> <p>Address "remote IP"</p> <p></Instance></p> <p>2. On remote DUT add ovs bridge br0.</p> <p>3. Start collectd daemon : /install_folder/bin/collectd</p> <p>4. Change link status of br0 on DUT to UP.</p>	<p>Notification message with data from local DUT appears</p> <p>Notification message with data from remote DUT appears</p>	<p>Notification message with data from local DUT appears</p> <p>Notification message with data from remote DUT appears</p>	PASS		

#	Test summary	Steps	Actual result	Expected result	Status - PASS/FAIL	Automated status	Comments
		<p>5. Using exec plugin handle notification.</p> <p>6. Change link status of br0 on remote DUT to UP.</p>					
5	Verify user can not create duplicate instances for ovs_events plugin	<p>1. Add new instance for remote DUT same as local DUT:</p> <pre><Instance hostname> DispatchValues true Address "remote IP" </Instance></pre> <p>2. Start collectd daemon : /install_folder/sbin/collectd</p>	<p>Error message appears in syslog. Collectd does not start</p>	<p>Error message appears in syslog. Collectd does not start</p>	PASS		
6	Check user can setup multi instance configuration for 30 ovs instances for ovs_events plugin	<p>1. Add 30 ovs docker containers.</p> <p>2. Add ovs bridge and port to each container.</p> <p>3. Add new instances in collectd conf:</p> <pre><Instance docker_host_0 > DispatchValues true Port XXXX </Instance></pre>	<p>Folders with appropriate files in /tmp/docker_folder/</p>	<p>Folders with appropriate files in /tmp/docker_folder/</p>	PASS		

#	Test summary	Steps	Actual result	Expected result	Status - PASS/FAIL	Automated status	Comments
		<p><Instance docker_host_29> DispatchValues true Port XXXY </Instance></p> <p>4. Start collectd daemon : /install_folder/sbin/collectd</p> <p>5. Change link status of interface on all ovs instances UP.</p> <p>6. Using exec plugin handle notification.</p>	<p>for each instance were created</p> <p>New entry with data appears with last collected value 0 for all ovs instances</p> <p>New entry with data appears with last collected value 1 for all ovs instances</p> <p>Notification message with data from each instance appears</p>	<p>for each instance were created</p> <p>New entry with data appears with last collected value 0 for all ovs instances</p> <p>New entry with data appears with last collected value 1 for all ovs instances</p> <p>Notification message with data from each instance appears</p>			
7	Verify ovs_events handle data about link status every 0.005 sec	<p>1. Set Interval = 0.005</p> <p>2. Add new instance for remote DUT: <Instance remote_host> DispatchValues true Address "remote IP" </Instance></p> <p>3. Start collectd daemon : /install_folder/sbin/collect.</p> <p>4. Check that link status from</p>	<p>Data about link status appears every 0.005 sec</p>	<p>Data about link status appears every 0.005 sec</p>	PASS		

#	Test summary	Steps	Actual result	Expected result	Status - PASS/ FAIL	Automated status	Comments
		ovsdb handle by ovs_events every 0.005 sec.					
8	Check ovs_events plugin multiple default configuration	1. Comment Plugin ovs_events section in collectd.conf. 2. Start collectd daemon : /install_folder/sbin/collect. 3. Change link status of br0 on DUT to UP. 4. Using exec plugin handle notification.	Notification message with data from local DUT appears Data about link status didn't handle by ovs_events plugin	Notification message with data from local DUT appears Data about link status didn't handle by ovs_events plugin	PASS		
9	Verify ovs_events handle data about link status from 12 instances with 1 VF and 4 local port into each instance	1. Create 12 VF from active interface. 2. Add VFs to ovs-instances and 4 local port. 3. Add new instances to collectd plugin: <Instance remote_host> DispatchValues true Port "XXXX" </Instance> 4. Start collectd daemon : /install_folder/sbin/collect.	Folders with appropriate files in /tmp/docker_folder/ for each instance were created New entry with data appears with last collected value 0 for all ovs instances	Folders with appropriate files in /tmp/docker_folder/ for each instance were created New entry with data appears with last collected value 0 for all ovs instances	PASS		

#	Test summary	Steps	Actual result	Expected result	Status - PASS/FAIL	Automated status	Comments
		<p>5. Check that link status from ovsdb handle by ovs_events every 0.005 sec.</p> <p>6. Change link status of interface on all ovs instances UP.</p> <p>7. Using exec plugin handle notification.</p>	<p>Data about link status appears every 0.005 sec</p> <p>New entry with data appears with last collected value 1 for all ovs instances</p> <p>Notification message with data from each instance appears</p>	<p>Data about link status appears every 0.005 sec</p> <p>New entry with data appears with last collected value 1 for all ovs instances</p> <p>Notification message with data from each instance appears</p>			
10	Check ovs_events plugin multi instance with physical interfaces	<p>1. Get names of available NIC (except management interface).</p> <p>2. Add 1 NIC port 4 local (internal) ovs ports to ovs-instance.</p> <p>3. Add new instances to collectd plugin:</p> <pre><Instance remote_host> DispatchValues true Port "XXXX" </Instance></pre> <p>4. Start collectd daemon:</p>	<p>Folders with appropriate files in /tmp/docker_folder/</p> <p>for each instance were created</p> <p>New entry with data appears with last collected value 0 for all ovs instances</p>	<p>Folders with appropriate files in /tmp/docker_folder/</p> <p>for each instance were created</p> <p>New entry with data appears with last collected value 0 for all ovs instances</p>	PASS		

#	Test summary	Steps	Actual result	Expected result	Status - PASS/FAIL	Automated status	Comments
		/install_folder/bin/collect. 5. Check that link status from ovsdb handle by ovs_events every 0.005 sec. 6. Change link status of interface on all ovs instances UP. 7. Using exec plugin handle notification.	Data about link status appears every 0.005 sec New entry with data appears with last collected value 1 for all ovs instances Notification message with data from each instance appears	Data about link status appears every 0.005 sec New entry with data appears with last collected value 1 for all ovs instances Notification message with data from each instance appears			
1 1	Check ovs_events plugin multi instance start/stop ovs instance	1. Add 10 ovs containers. 2. Add ovs bridge and port to each container. 3. Add new instances in collectd conf: <Instance docker_host_0 > DispatchValues true Port XXXX </Instance> <Instance docker_host_9 > DispatchValues true	New entry with data appears with last collected value 1 for all ovs instances	New entry with data appears with last collected value 1 for all ovs instances	PASS		

#	Test summary	Steps	Actual result	Expected result	Status - PASS/FAIL	Automated status	Comments
		Port XXXY </Instance> 4. Start collectd daemon : /install_folder/sbin/collectd 5. Change link status of interface on all ovs instances UP. 6. Stop 5 ovs containers. 7. Start 5 ovs containers.	ovs_events stop collecting data from stopped ovs instances ovs_events collecting data from started ovs instances	ovs_events stop collecting data from stopped ovs instances ovs_events collecting data from started ovs instances			

OVS stats multi instance collectd plugin tests.

Ovs_stats configuration:

- Open collectd config file: vim /install_folder/etc/collectd.conf
- Uncomment LoadPlugin "ovs_stats" fields.
- Uncomment

```
<Plugin ovs_stats>
<Instance hostname>
</Instance>
</Plugin>
```

OpenvSwitch configuration:

- Start ovs: service openvswitch-switch start
- Add new bridge: ovs-vsctl add-br br0
- Allow connection to ovsdb-server: ovs-vsctl set-manager tcp:6640

Table #5. OVS stats multi instance collectd plugin tests results.

#	Test summary	Steps	Actual result	Expected result	Status - PASS/FAIL	Automated status	Comments
1	Verify ovs_stats when extra keyword present in collectd.conf	<p>1. Configure ovs_stats section and add extra values: <Plugin ovs_stats> <Instance hostname> Port "6640" Address "127.0.0.1" DispatchValue true </Instance> </Plugin></p> <p>2. Start collectd</p>	Collectd crashes after start		PASS HAA-1809		
2	Verify user can setup multi instance configuration for ovs_stats plugin	<p>1. Add new instance for remote DUT: <Instance remote_host> Address "remote IP" </Instance></p> <p>2. On remote DUT add ovs bridge br0.</p> <p>3. Start collectd daemon : /install_folder/sbin/collectd</p> <p>4. Navigate to: /tmp/hostname_folder/</p> <p>5. Change link status of br0 on DUT and remote DUT to UP.</p>	<p>Folders with appropriate files in /tmp/hostname_folder/ and /tmp/remote_host were created</p> <p>New entry with data appears with last collected value 1 for local and remote DUT</p>	<p>Folders with appropriate files in /tmp/hostname_folder/ and /tmp/remote_host were created</p> <p>New entry with data appears with last collected value 1 for local and remote DUT</p>	PASS HAA-1812		

#	Test summary	Steps	Actual result	Expected result	Status - PASS/ FAIL	Automated status	Comments
3	Verify user can not create duplicate instances for ovs_stats plugin	<p>1. Add new instance for remote DUT same as local DUT:</p> <pre><Instance hostname> Address "remote IP" </Instance></pre> <p>2. Start collectd daemon : /install_folder/s bin/collectd</p>	Error message appears in syslog. Collectd does not start	Error message appears in syslog. Collectd does not start	PASS		
4	Check ovs_stats plugin multi instance default configuration	<p>1. Comment Plugin ovs_stats section in collectd.conf.</p> <p>2. Start collectd daemon : /install_folder/s bin/collect.</p>	Plugin start handling data from available ovs_instance . Collectd started	Plugin start handling data from available ovs_instance . Collectd started	PASS		
5	Verify ovs_stats handle data from ovs-dpdk ports	<p>1. Add dpdk port to open virtual bridge.</p> <p>2. Start collectd daemon : /install_folder/s bin/collect.</p>	Plugin start handling data from dpdk port. Files for all available dpdk statistics are present.	Plugin start handling data from dpdk port. Files for all available dpdk statistics are present.	PASS		OvS Stats Plugin
6	Check that ovs_stats handle data after openvswitchd	1. Stop openvswitchd: systemctl stop openvswitch-switch.service.	Ovs-stats plugin stop collecting data from stopped ovsswitchd.	Ovs-stats plugin stop collecting data from stopped ovsswitchd.	PASS		

#	Test summary	Steps	Actual result	Expected result	Status - PASS/ FAIL	Auto mated status	Com ments
	<i>stop/start /restart</i>	<p>2. Start openvswitch: systemctl start openvswitch-switch.service.</p> <p>3.Restart openvswitch: systemctl restart openvswitch-switch.service.</p>	<p>Ovs-stats plugin start collecting data after time Interval.</p> <p>Ovs-stats plugin stop and than start collecting data after time Interval.</p>	<p>Ovs-stats plugin start collecting data after time Interval.</p> <p>Ovs-stats plugin stop and than start collecting data after time Interval.</p>			
7	<i>Verify ovs_stats plugin monitors 20 ovs-instances with 50 ovs-internal ports each</i>	<p>1. Create 20 ovs instances (separate docker containers with ovs).</p> <p>2. Add 50 ports to each ovs instance.</p> <p>3. Start collectd.</p>	<p>Folders with appropriate files were created for each ovs instance and for all 50 internal ports</p>	<p>Folders with appropriate files were created for each ovs instance and for all 50 internal ports</p>	PASS		