Snapshot Examples

The following sections are **partial** examples that identify the options required or useful when discovering specific features. These examples focus on requirements and anomalies. They do not represent all the possible tasks and CLI options. All examples assume you have defined the environment variables and called other tasks in snapshot.txt, and that you have properly configured the snapshot.inc file for all other tasks.

Collecting Segment Routing LSPs

Step 1 In snapshot.txt, confirm that the following tasks are enabled (uncommented):

- LOGIN_FIND_IGP_DB
- SNMP_FIND_NODES
- SNMP_FIND_INTERFACES
- IMPORT_PCEP_LSPS
- GET_CONFIGS
- PARSE_CONFIGS
- SNMP_FIND_RSVP

Step 2 In snapshot.inc, add or set the following options:

Name	Required Value
<LOGIN_FIND_IGP_DB_CMD_OPT>	
get-segments	true
<SNMP_FIND_RSVP_CMD_OPT>	
keep-pcep-paths	true
get-pcep-paths	false
<PARSE_CONFIGS_CMD_OPT>	
include-object	BASE, RSVP, SR_LSPS

For example:

```
<LOGIN_FIND_IGP_DB_CMD_OPT>
Name    Value
```

```
out-file        $(work_dir)/$(unique).txt
seed-router     $(seed_router)
backup-router   $(backup_router)
igp-protocol    $(igp)
isis-level      $(isis_level)
ospf-area       $(ospf_area)
get-segments    true
session-type    $(session_type)
database-file   $(debug_dir)/$(unique)-$(igp)_db.txt
verbosity       $(cmd_verbosity)
log-file        $(log_dir)/$(unique)-log-login_find_igp_db.log

<SNMP_FIND_RSVP_CMD_OPT>
Name    Value
out-file        $(work_dir)/$(unique).txt
plan-file       $(work_dir)/$(unique).txt
net-recorder    $(net_recorder)
net-record-file $(data_dir)/$(unique)-record-snmp_find_rsvp.txt
verbosity       $(cmd_verbosity)
log-file        $(log_dir)/$(unique)-log-snmp_find_rsvp.log
use-signaled-name       true
get-backup-paths        true
keep-pcep-paths true
get-pcep-paths  false

<PARSE_CONFIGS_CMD_OPT>
Name    Value
igp-protocol    $(igp)
isis-level      $(isis_level)
include-object  BASE,RSVP,SR_LSPS,LAG,SRLG,RSVP,VPN
out-file        $(work_dir)/$(unique).txt
data-dir        $(cfg_dir)
log-level       $(log_level)
log-file        $(log_dir)/$(unique)-log-parse_configs.log
```

Insert Data into External Archive

This example shows how to insert data into an external archive where the information is available for all applications to use.

Step 1 In the snapshot.txt, point the default `archive_dir` to point to the external archive. Best practice is to keep the default.

Example: `archive_dir $(home_dir)/archives`

Step 2 In snapshot.txt, enable the ARCHIVE_INSERT task (uncomment it).

Step 3 In snapshot.inc, use `archive_insert` to insert WAE Live plan files into the external archive during the collection process.

Name	Required Value
<ARCHIVE_INSERT>	
cmd	$(cariden_home)/bin/archive_insert
cmd_opt	ARCHIVE_INSERT_CMD_OPT
<ARCHIVE_INSERT_CMD_OPT>	
plan-file	$(work_dir)/$(unique).pln
archive	$(archive_dir)/$(unique)-archive
time	$(start_time)

Collecting BGP LS

This example provides the workflow for enabling WAE Collector to get BGP LS from the Open SDN Controller (OSC).

Prerequisites:

- BGP LS must be properly configured on the router. For an example of how to do this, refer to the command reference guide for the Cisco IOS XR router, which you can find here: http://www.cisco.com/c/en/us/support/routers/carrier-routing-system/products-command-reference-list.html.
- For BGP LS to be collected, the Automation server must have the wae-osc, wae-core, and wae-db services running. It is also recommended to have wae-messaging service running.

Step 1 Configure OSC to collect BGP LS. For information, see the *BGP LS PCEP User Guide*:

https://wiki.opendaylight.org/view/BGP_LS_PCEP:User_Guide

Step 2 Configure the snapshot.txt file to specify an environment variable for the BGP LS server URL and to turn on the BGL LS Discovery task.

a. Specify the `bgpls_url` environment variable. The default BGP LS server port on which it listens is 8181.

Example: `bgpls_url http://localhost:8181`

b. Uncomment the FIND_BGPLS task in the snapshot.txt file.

Example:

```
<DISCOVERY_TASKS>
#SAM_GETPLAN
#SNMP_FIND_OSPF_DB
#LOGIN_FIND_IGP_DB
FIND_BGPLS
SNMP_FIND_NODES
SNMP_FIND_INTERFACES
```

Step 3 Configure the snapshot.inc file to specify how you want to collect the BGP LS data. The default is to collect OSPF, area 0. This example demonstrates how to collect using both IS-IS levels.

Name	Required Value
<FIND_BGPLS>	
cmd	$(cariden_home)/bin/find_bgpls
cmd_opt	FIND_BGPLS_CP_CMD_OPT
<SNMP_FIND_BGPLS_CMD_OPT>	
url	$(bgpls_url)
out-file	$(work_dir)/$(unique).txt
igp-protocol	isis
isis-level	2
log-file	$(log_dir)/$(unique)-log-find_bgpls.log

Step 4 On the Automation server, start or restart the wae-osc service.

```
service wae-osc start
service wae-osc restart
```

Collecting BGP Peers

Step 1 In snapshot.txt, confirm that the following tasks are enabled (uncommented):

- LOGIN_FIND_IGP_DB (or SNMP_FIND_OSPF_DB)
- SNMP_FIND_NODES
- SNMP_FIND_INTERFACES
- FIND_BGP

For example:

```
<DISCOVERY_TASKS>
Task
#SAM_GETPLAN
SNMP_FIND_OSPF_DB
#LOGIN_FIND_IGP_DB
#FIND_BGPLS
SNMP_FIND_NODES
SNMP_FIND_INTERFACES
#IMPORT_PCEP_LSPS
#GET_CONFIGS
#PARSE_CONFIGS
FIND_BGP
#SNMP_FIND_RSVP
#SNMP_FIND_VPN
```

Collect eBGP Peers by MAC Address

This example shows how to discover and poll eBGP peers by MAC address using manual snapshots. This feature provides more granular traffic collection for networks that establish BGP peering with a large number of ASNs through switch interfaces at public Internet exchange points (IXPs).

Note MAC accounting must be enabled on the routers.

Step 1 In snapshot.inc, use `find-bgp` with the `-get-mac-address` option set to `true`. This enables discovery of eBGP peers by MAC addresses.

Name	Required Value
<FIND_BGP>	
cmd	$(cariden_home)/bin/find_bgp
cmd_opt	FIND_BGP_CMD_OPT
<FIND_BGP_CMD_OPT>	
get-mac-address	true

Step 2 In snapshot.inc, use `snmp_poll` with the `-poll-function` option set to a value that specifies both `interface` and `mac`. This collects interface traffic statistics by MAC addresses.

Name	Required Value
<SNMP_POLL>	
cmd	$(cariden_home)/bin/snmp_poll
cmd_opt	SNMP_POLL_CMD_OPT
<SNMP_POLL_CMD_OPT>	
poll-function	interfaces, mac
polling-interval	interfaces=60, mac=60
number-of-samples	interfaces=1,mac=1

Step 3 In snapshot.txt, ensure that <FIND_BGP> and <SNMP_POLL> are enabled.

Collect Data for WAE Live

For Explore and Analytics components, this example shows how to set up the collection of the statistics that are put into the data store.

Step 1 In snapshot.txt, ensure the following are enabled:

- <SNMP_FIND_NODES>

- Either <SNMP_POLL> or <SNMP_POLL_INTERFACES>, depending on which has its `-perf-data` option set to true in the snapshot.inc file.

Step 2 In snapshot.inc, set the `-perf-data` option to true for `snmp_find_nodes`.

Name	Required Value
<SNMP_FIND_NODES>	
cmd	$(cariden_home)/bin/snmp_find_nodes
cmd_opt	SNMP_FIND_NODES_CMD_OPT
<SNMP_FIND_NODES_CMD_OPT>	
perf-data	true

Step 3 In snapshot.inc, set the `-perf-data` option to true for either `snmp_poll` or `snmp_poll_interfaces`.

	Either...	Or...
	<SNMP_POLL>	**<SNMP_POLL_INTERFACES>**
cmd	$(cariden_home)/bin/snmp_poll	$(cariden_home)/bin/snmp_poll_interfaces
cmd_opt	SNMP_POLL_CMD_OPT	SNMP_POLL_INTERFACES CMD_OPT
	<SNMP_POLL_CMD_OPT>	**<SNMP_POLL_INTERFACES_CMD_OPT>**
perf-data	true	true

Step 4 If analyzing LAGs in WAE Live, set the `snmp_find_interfaces -lag` option to `true`. See Collect LAG Membership and Traffic.

Manually Insert WAE Live Data

This example shows how to insert data directly into the WAE data store and Map archive, rather than storing on a server or in an external archive.

Insert Data into Data Store

Step 1 In snapshot.txt, enable the ML_INSERT task (uncomment it).

Step 2 In snapshot.inc, use <ML_INSERT> to insert plan files into the WAE Live data store during the collection process.

Name	Required Value
<ML_INSERT>	
cmd	$(cariden_home)/bin/ml_insert_plan
cmd_opt	ML_INSERT_CMD_OPT
<ML_INSERT_CMD_OPT>	
plan-file	$(work_dir)/$(unique).pln
time	$(start_time_direct)

Insert Data into Map Archive

This is only applicable if using `ml_insert_plan` and if using the WAE Live Map component. The location specified must be the location of the Map archive directory. This is not the same as the external archive.

Step 1 In the snapshot.txt, create an environment variable that specifies the location of the Map archive.

Example: `map_archive_dir $(home_dir)/data/mldata/archive`

Step 2 In snapshot.txt, add an MAP_ARCHIVE_INSERT task.

Step 3 In snapshot.inc, add <MAP_ARCHIVE_INSERT> to insert WAE Live plan files into the internal Map archive during the collection process.

Name	Required Value
<MAP_ARCHIVE_INSERT>	
cmd	$(cariden_home)/bin/archive_insert
cmd_opt	MAP_ARCHIVE_INSERT_CMD_OPT
<MAP_ARCHIVE_INSERT_CMD_OPT>	
plan-file	$(work_dir)/$(unique).pln
archive	$(map_archive_dir)/archive
time	$(start_time)

Collect LAG Membership and Traffic

Step 1 In snapshot.txt, ensure both <SNMP_FIND_INTERFACES> and <SNMP_POLL> are enabled.

Step 2 In snapshot.inc, use `snmp_find_interfaces` to discover LAG ports with the `-lag true` option. This populates the <Ports> and <PortCircuits> tables. The latter is based on a best-match rule according to ascending port names and numbers.

Use the `-lag-port-match` option to specify how ports are matched in port circuits. Here, the `complete` value is used to tell WAE Collector that if a port is up, match it deterministically based on LACP, and if a port is down, use the `guess` mode to match as many ports as possible.

Name	Required Value
<SNMP_FIND_INTERFACES>	
cmd	$(cariden_home)/bin/snmp_find_interfaces
cmd_opt	SNMP_FIND_INTERFACES_CMD_OPT
<SNMP_FIND_INTERFACES_CMD_OPT>	
lag	true
lag-port-match	complete

Step 3 In snapshot.inc, use `snmp_poll` to poll all LAG and bundle ports for traffic measurements with the `-poll-function ports` option. Ports are polled with the same parameters as interfaces.

Name	Required Value
<SNMP_POLL>	
cmd	$(cariden_home)/bin/snmp_poll
cmd_opt	SNMP_POLL_CMD_OPT
<SNMP_POLL_CMD_OPT>	
poll-function	interfaces, ports
polling-interval	interfaces=60
number-of-samples	interfaces=1

Collect QoS and Traffic

Step 1 In snapshot.txt, ensure both <SNMP_FIND_NODES> and <SNMP_POLL> are enabled.

Step 2 In snapshot.inc, use `snmp_find_nodes` to discover interface queues with the `-read-qos-queues true` option.

Name	Required Value
<SNMP_FIND_NODES>	
cmd	$(cariden_home)/bin/snmp_find_nodes
cmd_opt	SNMP_FIND_NODES_CMD_OPT
<SNMP_FIND_NODES_CMD_OPT>	
read-qos-queues	true

Step 3 In `snapshot.inc`, use `snmp_poll` to poll all interface queues with the `-qos-queues '*'` option.

Name	Required Value
<SNMP_POLL>	
cmd	$(cariden_home)/bin/snmp_poll
cmd_opt	SNMP_POLL_CMD_OPT
<SNMP_POLL_CMD_OPT>	
poll-function	interfaces
polling-interval	interfaces=60
number-of-samples	interfaces=1
qos-queues	‘*’