



CHAPTER 8

Subscriber Manager Integration - Configuration

Revised: August 22, 2012, OL-24194-06

Introduction

This chapter describes how to configure the DHCP Lease Query Login Event Generator (LEG) on the subscriber manager and how to use the command-line utilities (CLUs).



Note

This module is relevant if you are using the subscriber manager server on your network. You should install the DHCP Lease Query LEG on the Subscriber Manager (SM) server and not on the service control engine (SCE) device.

Information About Configuring the DHCP Lease Query LEG

The DHCP Lease Query LEG on the SM is configured using two configuration files: **leaseq.cfg** (general configuration) and **dhcp_pkg.cfg** (dynamic package association), which reside in the **~pcube/sm/server/root/config** directory.

The configuration files consist of sections headed by a bracketed section title; for example, **[DHCP-Lease-Query-LEG]**. Each section consists of several parameters having the format **parameter=value**. The number sign (“#”) at the beginning of a line signifies that it is a remark.

- [Configuring the DHCP Lease Query LEG, page 8-2](#)
- [Configuring Policy Association, page 8-5](#)
- [Applying the Configuration on the SM, page 8-8](#)
- [Information About the DHCP Lease Query LEG CLU, page 8-9](#)

Configuring the DHCP Lease Query LEG

The following is a description of the configuration variables of **leaseq.cfg**.

The **[DHCP-Lease-Query-LEG]** section contains the following parameters:

- **start**
Defines whether the SM runs the DHCP Lease Query LEG at startup.
Possible values for this parameter are **yes** and **no**. The default value is **no**.
To run the LEG, this parameter must be set to **yes**.
- **max_concurrent_sessions**
Defines the number of concurrent sessions the LEG should support. This parameter limits the resources used by this module.
Possible values for this parameter are integers. The default value is 256.
- **server_mode**
Defines whether the specified DHCP servers are in the **active_standby** mode or **active_active** mode. In the **active_active** mode, the lease query request will be sent to all the specified DHCP servers. However, in the **active_standby** mode, the lease query request will be sent to one of the active servers.
- **dhcp_servers**
Defines to which DHCP servers the LEG can send requests.
You must enter the IP addresses or hostnames of the DHCP servers separated by commas.
- **server_port**
Defines the UDP port to which the DHCP servers listen and the Lease Query messages are sent. It is recommended to use 9067 when working with the DHCP Forwarder. The default value is 9067.
- **listening_port**
Defines the UDP port to which the LEG listens and the Lease Query replies are sent. It is recommended to use 8067 when working with the DHCP Forwarder. The default value is 8067.
- **client_port**

Defines the UDP port that the LEG uses when sending Lease Query messages to the DHCP servers. It is recommended to use 8068 when working with the DHCP Forwarder. The default value is 8068.

- `client_ip_address`

Defines the source IP address of the lease-query packets sent to the DHCP servers. The `giaddr` field of the DHCP packet also uses this IP address. This parameter is useful for machines with multiple network interfaces.

The default value is the loopback IP address of the machine.

- `support_auto_logout`

Defines whether the LEG should query the DHCP servers whenever the autologout mechanism identifies an expired lease.

Possible values for this parameter are **true** and **false**. The default value is **false**.

- `use_forwarder`

Defines whether the LEG utilizes the DHCP Forwarder application on the local machine.

Possible values for this parameter are **true** and **false**. The default value is **true**.

- `fail_over_criteria`

Defines the number of consecutive request failures (timeouts) that triggers a fail-over. Since the queries are not answered when the server fails, these queries will time out. The consecutive timed-out queries are counted and when they reach this threshold, the second server is set as the active server. The default value is 3.



Note

The `session_timeout` parameter affects how long it takes to detect a failed server. Only when the configured amount of queries fail will the fail-over process be triggered.

- `log_timed_out_queries`

Controls log messages regarding timed-out queries.

Possible values for this parameter are **true** or **false**. The default value is **true**.

- `log_failed_queries`

Controls log messages regarding queries that are not sent.

Possible values for this parameter are **true** or **false**. The default value is **true**.

- `log_all_queries`

Controls log messages regarding each query sent and any reply received.

Possible values for this parameter are **true** or **false**. The default value is **false**.

Use this parameter only for troubleshooting.

- `log_login_failures`

Controls log messages regarding replies that did not result in the login of a subscriber to the SM.

Possible values for this parameter are **true** or **false**. The default value is **true**.

The **[Subscriber ID]** section defines the functionality of how the LEG handles the subscriber ID. The subscriber ID can be taken from a DHCP option, with the ability to fallback to using the allocated IP address as the subscriber ID. This section contains the following parameters:

- **dhcp_option**
Defines which DHCP option to use as the subscriber ID. The format of this parameter is the option number itself. For DHCP options that have sub-options, the format is the DHCP option and sub-option type, separated by a colon. For example: 43:123 or 61. The default value is 82:2 (Relay-Agent-Information using the Remote-ID information).
- **dhcp_option_type**
Defines the format type of the DHCP option defined by the **dhcp_option** parameter.
Optional values are **binary**, indicating a binary string converted to an ASCII hexadecimal string; or **string**, indicating an ASCII string. The default value is **binary**.
- **default_id**
Defines whether in cases where the **dhcp_option** is not found in the DHCP packet, the LEG should fall back to a different way of defining the subscriber ID. The supported fallbacks are:
 - **ip**—Use the allocated IP address to create a subscriber ID in the format of: IP_aaa.bbb.ccc.ddd.
 - Not setting this parameter—No fallback. No login is performed.
 By default, this parameter is not set.

**Note**

Once configured, the **default_id** parameter is applied for each DHCP transaction; after which, for a certain subscriber the **dhcp_option** must be present in *all* DHCP transactions, or the **dhcp_option** must be omitted from *all* DHCP transactions. If the DHCP transactions contain a mixture of transactions with and without the **dhcp_option**, two subscribers with different Subscriber IDs are created in the SM database.

The [**DHCP-Lease-Query-Ids**] section contains the message type numbers of the different Lease Query transaction message types. This is necessary because the DHCP Lease Query definition is an IETF draft. This section contains the following parameters:

- **lease_query**
Defines the DHCPLEASEQUERY message type value. The default value is 13.
- **lease_active**
Defines the DHCPLEASEACTIVE message type value. The default value is 16.

The following is a sample configuration file:

```
[DHCP-Lease-Query-LEG]
start=yes
dhcp_servers = 198.1.2.3, 198.5.6.7
fail_over_criteria=10
session_timeout=10
log_timed_out_queries=true
log_failed_queries=true
log_all_queries=true
log_login_failures=true
[Subscriber ID]
dhcp_option=44
dhcp_option_type=binary
[DHCP-Lease-Query-Ids]
lease_query=13
lease_active=16
```

Configuring Policy Association



Note

The configuration described in this section is optional.

The subscriber policy configuration in the DHCP Lease Query LEG can be handled in one of the following ways:

- Dynamic assignment of policy information using information extracted from the DHCP packet. See [Dynamic Assignment of Policy Information, page 8-5](#).
- Static assignment of a constant package ID for all subscribers who log on via the DHCP Lease Query LEG. See [Static Assignment of Policy Information, page 8-8](#).

Dynamic Assignment of Policy Information

Dynamic assignment of policy information is supported when policy information is submitted in the DHCP packets. The LEG concatenates the desired options and creates a *policy-name*. It is possible to map, using the configuration, between the *policy-names* and the application policy parameters such as package IDs and Virtual-links. The DHCP Lease Query LEG can support multiple policies.

To extract the policy information data from the DHCP packet, use the **dhcp_pkg.cfg** configuration file to define the option types that contain the policy information and define the conversion map of the *policy-names* to the package IDs (or any other policy) of the Service Control Application for Broadband (SCA BB).

The LEG is able to add additional data to the login operation based on the LEG configuration. This data is added as a key-value pair. Other modules in the login chain can use this data, such as the SOAP LEG (see the [SOAP LEG](#) part of this guide). This data can be created by concatenating the data of several DHCP options and can be given a user-defined label.

The **[DHCP.Policy.XXX]** sections contain the following parameters:

- **options_order_for_policy_name**
 Defines the DHCP options that contain the policy association information and defines the order of concatenation of the data. The DHCP header field called giaddr (Relay-Agent IP) is also supported; it requires the use of the type integer in the **option_type** parameter.
 This parameter has no default value.
 The format is: **option[:subtype],option[:subtype],giaddr**
- **options_type**
 Defines the format type of the DHCP options and fields defined by the **options_order_for_policy_name** parameter.
 Possible values for this parameter are **binary** (a binary string that is converted to an ASCII hexadecimal string), **string** (an ASCII string), or **integer** (a 4-byte integer converted to an IP address string in dotted notation). Order the list in the same way as **options_order_for_policy_name**.
 This parameter has no default value.
- **name_seperator_value**
 Defines the separator character to use between two options when concatenating them to each other to create the policy name. Any character is accepted. The default value is '_'.

- **use_default**
Determines whether to use a default policy when no policy information can be extracted from the DHCP data, such as the configurable options are missing or no options were configured.
Possible values for this parameter are **true** or **false**. The default value is **false**.
- **default_policy**
Defines the default policy ID to use if no policy information is extracted from the DHCP data. This parameter is relevant only if the **use_default** parameter is set to **true**.
Possible values for this parameter are any integer number. This parameter has no default value.
- **allow_login_with_no_policy**
Defines whether to perform a login without policy information when no policy information can be extracted from the DHCP data and the **use_default** parameter is set to **false**.
This parameter is relevant only if the **use_default** parameter is set to **false**.
Possible values for this parameter are **true** or **false**. The default value is **true**.
- **ignore_policy_list**
Defines a list of indexes separated by commas where each index represents a policy or package value. On login operation, if the current subscriber policy is one of the values defined in this attribute, a login operation occurs without changing the subscriber policy value.
- **policy_property_name**
Defines the name of the application property that contains the policy information. This parameter has no default value.

**Note**

The **policy_property_name** parameter is case sensitive and must be written exactly as defined by the SCA BB Console. For example, **packageId**, **monitor**, **upVlinkId**, or **downVlinkId**.

- **log_all**
Defines whether to write detailed user-log messages for all policy association events.
Possible values for this parameter are **true** or **false**. The default value is **false**.
- **log_default_assignment**
Defines whether to write a user-log message for every assignment of the default value (as defined by the **default_policy** parameter).
Possible values for this parameter are **true** or **false**. The default value is **false**.
- **mapping_table.<policy_name>**
Multiple entries containing the information to convert from the policy information as it appears in the DHCP packet to the policy property value to be used by the SCA BB application.
These entries do not have default values.

**Note**

The **policy_name** is case sensitive and must be written exactly as it exists in the DHCP packets.

The **[Additional Data]** section of the configuration file contains the following parameters:

- **label_options**
Defines which DHCP option to extract to add to the login operation.
Possible values are the option number or, in the case of DHCP options with sub-options, the option and sub-option separated by a colon. For example, 43:123 or 61.
There is no default value for this parameter.
- **label_keys**
Defines the keys that should mark the DHCP options defined by the **label_options** parameter.
There is no default value for this parameter.
- **label_options_type**
Defines the format type of the DHCP option defined by the **label_options** parameter.
Possible values for this parameter are **binary** (a binary string that is converted to an ASCII hexadecimal string) or **string** (an ASCII string).
The default value is **binary**.

Dynamic Assignment of Policy Information Example

Suppose that the policy information appears inside option 43 (Vendor Specific Option) of the DHCP packet and that both subtypes, 102 and 101, are in use. Configure the **options_order_for_policy_name** parameter as follows:

```
options_order_for_policy_name=43:102,43:101
```

Suppose that option 43 with subtype 102 contains the type of package (gold, silver, or bronze), and that option 43 with subtype 101 contains domain information (the package type has a different meaning in different domains). If the separator value is configured to the default value, configure the **mapping_table** entries as follows:

```
mapping_table.gold_domain1=11
mapping_table.gold_domain2=12
mapping_table.silver_domain1=13
mapping_table.silver_domain2=14
```

This configuration means that if the DHCP packet contains the value 'gold' inside option 43 with subtype 102, and the value 'domain1' inside option 43 with subtype 101, the package ID that are associated to the subscriber in the SM will have the value 11.

The following configuration describes how to add the data of the Relay-Agent Circuit-Id option as additional data to the login operation:

```
[Additional Data]
label_options=82:1
label_keys=PORT_ID
label_options_type=string
```

The following is an example of the entire configuration file:

```
[DHCP.Policy.Package]
options_order_for_policy_name=43:102,43:101
name_separator_value=_
use_default=true
default_policy=1
policy_property_name=packageId
allow_login_with_no_policy=false
log_all=false
log_default_assignment=false
mapping_table.gold_domain1=11
mapping_table.gold_domain2=12
mapping_table.silver_domain1=13
mapping_table.silver_domain2=14
[Additional Data]
label_options=82:1
label_keys=PORT_ID
label_options_type=string
```

Static Assignment of Policy Information

If the installation does not require dynamic assignment of package information, the configuration file **dhcp_pkg.cfg** should define the default package ID and default Virtual-Link to be assigned to all the subscribers, as shown in the following example:

```
[DHCP.Policy.Package]
policy_property_name=packageId
allow_login_with_no_policy=false
use_default=true
default_policy=1
[DHCP.Policy.VirtualLinkDownstream]
policy_property_name=downVLinkId
allow_login_with_no_policy=false
use_default=true
default_policy=0
[DHCP.Policy.VirtualLinkUpstream]
policy_property_name=upVLinkId
allow_login_with_no_policy=false
use_default=true
default_policy=0
```

All other configuration parameters should not be set.

Applying the Configuration on the SM

After editing the relevant configuration files, use the following p3sm command-line utility to load the configuration file:

```
>p3sm --load-config
```


Information About the DHCP Lease Query LEG CLU

The `p3leasequery` command-line utility (CLU) displays the DHCP Lease Query LEG configuration, status, and statistics. The command format is `p3leasequery <operation>`.

Table 8-1 lists the `p3leasequery` operations.

Table 8-1 *p3leasequery Operations*

Operation	Description
<code>--show</code>	Displays all of DHCP Lease Query LEG configurations and status
<code>--show-statistics</code>	Displays counters of DHCP messages handled and number of logon operations performed
<code>--reset-statistics</code>	Resets counters of DHCP messages handled and number of logon operations performed
<code>--show-version</code>	Displays the version number of the DHCP Lease Query LEG
<code>--help</code>	Displays a list of available operations and arguments, with a short explanation of their meanings.

- [Viewing the DHCP Lease Query LEG Status, page 8-9](#)
- [Viewing the DHCP Lease Query LEG Statistics, page 8-10](#)
- [Viewing the DHCP Lease Query LEG with Multiple DHCP Servers Status, page 8-10](#)
- [Viewing the DHCP Lease Query LEG with Multiple DHCP Servers Statistics, page 8-10](#)
- [Viewing the DHCP Lease Query LEG Version, page 8-11](#)

Viewing the DHCP Lease Query LEG Status

The following is an example using the `p3leasequery` CLU with the `show` operation:

```
>p3leasequery --show
DHCP Lease-Query LEG:
=====
Active:    true
DHCP Servers:
Active:    10.1.2.3
Standby:   N/A
Session timeout: 20
Fail over criteria: 3
Subscriber ID:
Option:    82:2
Format:    binary
Fallback:  none
Command terminated successfully
>
```

Viewing the DHCP Lease Query LEG Statistics

The following is an example of the `p3leasequery` CLU using the `show-statistics` operation:

```
>p3leasequery --show-statistics
DHCP Lease-Query LEG Statistics:
=====
Lease-Queries Sent:          16
Lease-Queries Replied:      16
Active Lease Replies:       16
Non-Active Lease Replies:    0
Total timed-out sessions:   0
Consecutive timed-out sessions: 0
Number of fail-overs        0
Invalid Replies:            0
Sessions in process:        0
Max-Concurrent sessions:    3
Command terminated successfully
>
```

Viewing the DHCP Lease Query LEG with Multiple DHCP Servers Status

The following is an example using the `p3leasequery` CLU with the `show` operation:

```
>p3leasequery --show
DHCP Lease-Query LEG:
=====
Active:      true
Mode :      active_active
DHCP Servers: 1.0.2.42,1.0.4.11
Session timeout: 20
Subscriber ID:
Option:      82:2
Format:      binary
Fallback: none
Command terminated successfully
-bash-3.2$
```

Viewing the DHCP Lease Query LEG with Multiple DHCP Servers Statistics

The following is an example of the `p3leasequery` CLU using the `show-statistics` operation:

```
>p3leasequery --show-statistics
DHCP Lease-Query LEG Statistics:
=====
Max-Concurrent sessions : 1
Sessions in process      : 0
Anonymous replies       : 0
Configured Servers      : 1.0.2.41,1.0.2.42,1.0.2.46
Server-Wise Statistics   :
=====
SERVER ADDRESS          : 1.0.2.41
Lease-Queries Sent      : 1
Lease-Queries Replied   : 1
Active Lease Replies    : 1
Non-Active Lease Replies: 0
Total timed-out sessions: 0
Invalid Replies         : 0
Failed log-in operations: 0
Failed log-out operations: 0
```

```
SERVER ADDRESS           : 1.0.2.42
Lease-Queries Sent      : 1
Lease-Queries Replied   : 1
Active Lease Replies     : 0
Non-Active Lease Replies : 0
Total timed-out sessions : 0
Invalid Replies         : 1
Failed log-in operations : 0
Failed log-out operations : 0

SERVER ADDRESS           : 1.0.2.46
Lease-Queries Sent      : 1
Lease-Queries Replied   : 0
Active Lease Replies     : 0
Non-Active Lease Replies : 0
Total timed-out sessions : 0
Invalid Replies         : 0
Failed log-in operations : 0
Failed log-out operations : 0
Command terminated successfully
```

Viewing the DHCP Lease Query LEG Version

The following is an example of the `p3leasequery` CLU using the `show-version` operation:

```
>p3leasequery --show-version
DHCP LEASE QUERY LEG 3.7.5 Build 303
>
```

