



APPENDIX **A**

Configuration File Options

This module describes in detail all the parameters that can be configured by using the Subscriber Manager (SM) configuration file. The shorter description of SM configuration given in [Configuration and Management, page 4-1](#) is more oriented toward the routine configuration tasks that can be performed online using the CLU.

Information About the Configuration File Options

The SM can be configured only by using its configuration file (the CLU is used for displaying, not configuring). The SM has one configuration file, **p3sm.cfg**, which is located under `~pcube/sm/server/root/config/`. To change any configuration parameter, edit the configuration file using a standard text editor, and then use the CLU to reload it (see [Reloading the SM Configuration \(p3sm\)](#), page 4-14).

You can use the **p3sm.cfg** configuration file for setting parameters for the following:

- SM configuration
- Radius Listener configuration
- Redundancy (cluster/standalone) configuration
- Domain configuration
- SCE configuration
- Cable adapter configuration
- PRPC port configuration
- FTP port configuration
- HTTP port configuration
- Database configuration

Description of the Configuration File Options

The following sections correspond to the sections of the SM configuration file, **p3sm.cfg**.

For an explanation of the terms and concepts used in these sections, see [How to Configure a Subscriber Management Solution](#), page 4-3.

- [SM General Section](#), page A-3
- [SM High Availability Setup Section](#), page A-4
- [SM-LEG Failure Handling Section](#), page A-5
- [LEG-Domains Association Section](#), page A-6
- [Domain.XXX Section](#), page A-7
- [Default Domains Configuration Section](#), page A-8
- [Auto Logout Section](#), page A-9
- [Inactive Subscriber Removal Section](#), page A-10
- [Radius Listener Section](#), page A-11
- [Radius.NAS.XXX Section](#), page A-12
- [Radius.Subscriber ID Section](#), page A-13
- [Radius.Property.Package Section](#), page A-14
- [RPC.Server Section](#), page A-16
- [MPLS-VPN Section](#), page A-16
- [SCE.XXX Section](#), page A-17
- [FTP Section](#), page A-17
- [HTTP Tech-IF Section](#), page A-18
- [RDR Server Section](#), page A-18
- [Cable Adapter Section](#), page A-19
- [Data Repository Section](#), page A-19

SM General Section

The [SM General] section contains the following parameters:

- `introduction_mode`

Defines whether the SM introduces the subscribers to the SCE devices immediately after a login operation (Push mode), or only when the SCE requests the subscriber specifically (Pull mode).

Possible values for this parameter are **push** and **pull**. The default value is **push**.

The following is an example of assigning a value to this parameter:

```
introduction_mode=push
```

- `application_subscriber_lock`

Defines whether to lock subscriber-related operations (login, logout, etc.) at the application level. Set this flag to **true** only if several LEG applications can update simultaneously the same parameters of a subscriber.

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

The following is an example of assigning a value to this parameter:

```
application_subscriber_lock=true
```

- `force_subscriber_on_one_sce`

Defines whether the SM supports the solution when a Cisco 7600/6500 Router is used for load-balancing among several SCE platforms. In this solution when one SCE platform fails, subscriber traffic is redistributed to a different SCE platform. The SM must remove subscribers from the failed SCE platform and send the relevant subscriber information to the new SCE platform. This parameter is relevant only in the pull introduction mode.

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



Note Changing this value requires a restart of the SM process.

- The following is an example of assigning a value to this parameter:

```
force_subscriber_on_one_sce=false
```

- `logon_logging_enabled`

Defines whether to enable the logging of subscriber logon events.



Note Setting this flag to **true** might cause performance degradation.

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

The following is an example of assigning a value to this parameter:

```
logon_logging_enabled=false
```

- `subscriber_id_case_sensitivity`

Optional parameter that defines whether subscriber IDs are case sensitive or not. When this flag is set to **no**, all subscriber IDs in the subscriber database are set to be lower case. For example, 'JohnSmith' is converted to 'johnsmith'.

**Note**

Setting this flag to **no** when the subscriber database is not empty is not allowed. An error message will be shown and the configuration will not be loaded. To overcome this limitation, you can do the following:

1. Export the subscriber database to an external file
2. Clear the subscriber database
3. Change the configuration by setting the flag to **no** and load the new configuration
4. Import the subscriber database from the external file

Possible values for this parameter are **yes** and **no**. The default value is **yes**.

The following is an example of assigning a value to this parameter:

```
subscriber_id_case_sensitivity=yes
```

SM High Availability Setup Section

The [**SM High Availability Setup**] section contains the following parameter:

- topology

Defines in what kind of topology the SM should work. Possible values for this parameter are **standalone** and **cluster**. The default value is **standalone**.

The following is an example of assigning a value to this parameter:

```
topology=standalone
```

SM-LEG Failure Handling Section

The [SM-LEG Failure Handling] section contains parameters that affect the discovery of an SM-LEG connection failure event and the actions taken by SM upon a connection failure event. A network problem or a severe failure (such as reboot) of the SM or the LEG can cause an SM-LEG connection failure event.

The [SM-LEG Failure Handling] section contains the following parameters:

- `clear_all_mappings`

Defines the behavior of the SM in case of LEG-SM connection failure. This parameter is relevant only for cases where the SM and LEG are running on *different* machines.



Note This parameter defines a behavior that is similar for all LEG applications connected to the SM.

If this parameter is set to **true** and a SM-LEG connection failure occurs that is not recovered within the defined **timeout**, the mappings of all subscribers in the domains defined in the [LEG-Domains Association Section, page A-6](#) for the LEG that was disconnected will be removed.



Note **Important** : If you set the **clear_all_mappings** parameter to **true**, you must also set the **LEG-Domains Association** parameter to clear the mappings in the SM if an SM-LEG connection failure occurs.

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

The following is an example of assigning a value to this parameter:

```
clear_all_mappings=false
```

- `timeout`

Defines the time in seconds from a SM-LEG connection failure until clearing the mappings in the SM database.



Note It takes several seconds for the SM to detect an SM-LEG connection failure. You must add this time to the value of the **timeout** parameter when calculating how long it will take the SM to react in case of an SM-LEG connection failure. For example, if **timeout** is set to 80, it will take 80 seconds **from the failure detection time** until the SM clears the subscribers database.

The default value for this parameter is 60 (seconds).

The following is an example of assigning a value to this parameter:

```
timeout=60
```

LEG-Domains Association Section

The [LEG Domains Association] section defines the domains for which the mapping of all subscribers that belong to them will be cleared on SM-LEG connection failure. This section is relevant only if the `clear_all_mappings` parameter has been set in the [SM-LEG Failure Handling Section, page A-5](#).



Note

Even though you set the **LEG-Domains Association** parameter, you must also set the `clear_all_mappings` parameter to **true** to actually clear the mappings in the SM if an SM-LEG connection failure occurs.

This section contains a list of LEG-Domain associations, each item in a separate line. Each LEG-Domain association is specified as shown for the following parameter:

- `<LEG name>=domain_name1[,domain_name2,...]`

Defines the domains whose subscriber mapping will be cleared on an SM-LEG connection failure. The key is the **<LEG name>**.

To determine which value or values to use for the **<LEG name>** key, consult the documentation of the LEG that you use. The **<LEG name>** is usually divided into two parts: **<hostname>.<common suffix>**. The first part is a general LEG identifier. The second part is extracted from the machine on which the LEG is running. Alternatively, you can use the CLU command **p3rpc --show-client-names**.

A **<LEG name>** of " *" specifies all LEGs. The (comma-separated) values are the domains (**domain_name**) to be cleared in the event of a network link failure (connection failure) between the specified LEG and the SM. A value of "*" for the **domain_name** specifies all subscriber domains in the system.



Note

The LEG name is *case sensitive* .

By default, there are no LEG domain mappings.

The following is an example of assigning values to this parameter:

```
10.1.12.76.NB.SM-API.J=home_users
10.1.12.77.B.SM-API.C=office_users
```

The following is an example of specifying all subscriber domains:

```
10.1.12.31.CNR.LEG=*
```

The following is an example of specifying all LEGs and all subscriber domains:

```
*=*
```

Domain.XXX Section

Each **[Domain.XXX]** section specifies one domain, where *XXX* represents the domain name.

This section contains the following parameters:

- `elements=logical_name1[,logical_name2,...]`

Specifies the name or names of the SCE platforms that are part of the domain.



Note Each name must be the *exact* "XXX" name (case sensitive) that appears in the [SCE.XXX Section, page A-17](#).

The following is an example of assigning a value to this parameter:

```
elements=se0,se1
```

- `aliases=alias_name1[,alias_name2,...]`

Defines domain aliases. When subscriber information is received from the LEG with one of the aliases (for example, *alias1*), the information is distributed to the domain that matches this alias (for example, *domain_name1*). A typical alias could be a network access device IP address, where, for example, each string in the values can be the IP address of a NAS or a CMTS.



Note Each alias can appear in only one domain section.

By default, there are no domain aliases.

The following is an example of assigning a value to this parameter:

```
aliases=10.10.88.99,10.10.88.98
```

- `property.name1=value1[,property.name2=value2,...]`

Defines the default policy properties values for a domain. Unless the LEG/API overrides these defaults when it logs in the subscriber to the SM, the subscriber policy is set according to the default policy properties values of its domain.

The policy format is a comma-separated list of **property_name=property_value**, where each property value is an integer.



Note To learn more about policy specification, see the *Cisco Service Control Application for Broadband (SCA BB) User Guide*.

- The following is an example of assigning a value to this parameter:

```
property.packageId=1
```

Default Domains Configuration Section

The **[Default Domains Configuration]** section defines the default policy for the domain. It is used for those domain properties that were not defined in the domain policy configuration (see [Domain.XXX Section, page A-7](#)), and for properties of domains for which no policy whatsoever was defined.

This section contains the following parameter:

- `property.name1=value1[,property.name2=value2,...]`

Defines the default policy properties values for all domains. Unless the LEG/API overrides these defaults when it logs in the subscriber to the SM, or unless they are overwritten by the default policy property values of the subscriber domain, the subscriber policy is set according to the global default policy property values defined in this section.

The policy format is a comma-separated list of **property_name=property_value**.



Note To learn more about policy specification, see the *Cisco Service Control Application for Broadband (SCA BB) User Guide*.

The following is an example of assigning a value to this parameter:

```
property.packageId=1
```


Auto Logout Section

The [**Auto Logout**] section defines the parameters for the Auto Logout feature, which is relevant mainly for cable environments. This feature is relevant to automatic integrations where the LEG/API cannot provide logout indications. In such a case, you can turn on the automatic logout mechanism, which instructs the SM to log out a subscriber automatically after a certain period of time. Note that a login event for a subscriber resets the subscriber logout timer.

**Note**

Not using the Auto Logout feature in the scenario described above (a provisioning system that can provide subscriber login events to the SM but cannot provide subscriber logout events) might result in exhausting the SCE resources, because subscribers are logged in but are never logged out.

This section contains the following parameters:

- `auto_logout_interval`

Configures the interval value, in seconds, of the SM auto-logout mechanism. Every interval, the SM checks for which subscriber IP addresses the lease time has expired, and begins to automatically remove these IP addresses from the system.

Lease time is the timeout defined by the LEG during the login operation per each IP address. All subscriber login events will start a timer of lease-time seconds. When the timer expires and the grace period (see below) has also expired, the subscriber IP addresses are removed, causing the subscriber to be removed from the SCE platform database. Any login event by the subscriber with an existing IP address during the timer countdown period resets the timer, causing it to restart.

Setting the interval value to zero (0) disables the SM auto-logout mechanism.

Setting the interval to a value greater than zero enables the SM auto-logout mechanism.

**Note**

The interval should be smaller, but of the order of the lease time used in the system. It is recommended that during a lease time the auto-logout task will run several times.

The default value for this parameter is 0 (seconds), meaning the auto-logout mechanism is disabled.

The following is an example of assigning a value to this parameter:

```
auto_logout_interval=600
```

The following is an example of deactivating the Auto Logout feature:

```
auto_logout_interval=0
```

- `grace_period`

Defines the grace period, in seconds, for each subscriber. After a subscriber auto-logout timeout has expired, the subscriber IP address is logged out automatically after the grace period has also expired.

The default value for this parameter is 10 (seconds).

The following is an example of assigning a value to this parameter:

```
grace_period=10
```

- `max_rate`

Defines the maximum rate (logouts per second) that the auto-logout task is allowed to perform logouts from the system. This limit spreads out the load of the logout operations over time, reducing the performance impact on other operations.

Calculate the value for this parameter to spread the logouts over at least half of the **auto_logout_interval** time. The default value is 50.

The following is an example of assigning a value to this parameter:

```
max_rate=50
```



Note Use the lowest rate possible to reduce the influence of the auto-logout process on other operations. However, as a guideline, calculate the value so that the auto-logout process takes about half of the **auto_logout_interval** and be similar to the maximum login rate to the SM.

Inactive Subscriber Removal Section

The [**Inactive Subscriber Removal**] section defines the parameters for the Inactive Subscriber Removal feature. This feature facilitates the removal of subscribers who have been logged out of the SM and are not mapped to any network-Id for a configurable time period. Effective use of this mechanism can keep the size of the SM database relatively small and close to the number of active subscribers.

A task runs intermittently to remove the inactive subscribers. The time interval between running of the task is defined automatically based on the configured inactivity time.



Note This feature is applicable only to subscribers that were logged in or out using an IP address/range. It can also be used to remove subscribers that have unsubscribed from a customer network that has no mechanism for removing such subscribers.

This section contains the following parameters:

- **start**
Defines whether or not to remove inactive subscribers.
Possible values for this parameter are **yes**, **no**, **true**, or **false**.
The default value is **no**.
- **inactivity_timeout**
Defines the time period after which subscribers will be removed from the SM database if they have not been assigned any network-Id.
Possible values for this parameter are "**X minutes**", "**X hours**", "**X days**", or "**X weeks**" where X is a decimal number. The allowed range is a minute to a year.
The default value is **1 hour**.
- **max_removal_rate**
Defines the maximum number of subscribers that the removal task can remove per second.
Possible values for this parameter are integer numbers between 1 and 1000.
The default value is **10**.
- **log_removals**
Defines whether to write user-log messages for each subscriber record removal.
Possible values for this parameter are **true** or **false**.
The default value is **true**.

Radius Listener Section

Use the [**Radius Listener**] section for integrating with the RADIUS Listener LEG.

**Note**

For additional RADIUS Listener configuration information, see the *Cisco SCMS SM RADIUS Listener LEG Reference Guide*.

This section contains the following parameters:

- **start**

Defines whether the SM should run the RADIUS Listener at startup.

Possible values for this parameter are **yes** and **no**. The default value is **no**.

The following is an example of assigning a value to this parameter:

```
start=no
```

- **accounting_port**

Defines the RADIUS Listener's accounting port number.

The default value is 1813.

The following is an example of assigning a value to this parameter:

```
accounting_port=1813
```

- **ip**

(Optional) Defines the IP address to where the RADIUS Listener should bind. Use this parameter in cluster setups or when the machine local-host IP is not the IP to where the RADIUS messages are sent.

By default, this parameter is not set.

The following is an example of assigning a value to this parameter:

```
ip=192.56.21.200
```

Radius.NAS.XXX Section

Each [**Radius.NAS.XXX**] section specifies a single Network Access System (NAS), where XXX represents the NAS name.

**Note**

The RADIUS Listener LEG refers to all of its RADIUS clients as NAS devices, even though they might be RADIUS servers.

This section contains the following parameters:

- **domain**
Specifies the Cisco Service Control subscriber domain name.
The following is an example of assigning a value to this parameter:
`domain=my_domain`
- **IP_address**
Specifies the IP address in dotted notation (xxx.xxx.xxx.xxx).
The following is an example of assigning a value to this parameter:
`IP_address= 1.1.1.1`
- **NAS_identifier**
Specifies the name of the NAS that exists in the **NAS-ID** attribute. For information about the use of this parameter, see the *Cisco SCMS SM RADIUS Listener LEG Reference Guide* .
The following is an example of assigning a value to this parameter:
`NAS_identifier=RedHat37`
- **secret**
Specifies a secret key defined in the NAS for this connection.
The following is an example of assigning a value to this parameter:
`secret=mysecret`

Radius.Subscriber ID Section

The [**Radius.Subscriber ID**] section defines the RADIUS attribute on which to base the subscriber ID association. Default association is based on the **User-Name** attribute.



Note

This section is commented out in the configuration file. If you want to base subscriber ID association on an attribute other than **User-Name**, you should uncomment this section.

This section contains the following parameters:

- **radius_attribute**

Specifies the RADIUS protocol attribute number. For multiple attributes use commas between the attribute numbers or use the following format for Vendor Specific Attributes (VSA): **26(vendor-id;sub-attribute)**.

The default value for this parameter is **1**.

The following is an example of assigning a value to this parameter:

```
radius_attribute=26(10415;1)
```

- **radius_attribute_type**

Specifies the RADIUS attribute type.

Possible values for this parameter are **integer** and **string**. The default value is **integer**.

The following is an example of assigning a value to this parameter:

```
radius_attribute_type=string
```

- **radius_attribute_separator**

Defines the character that should be used when concatenating several attributes. This parameter has no default value.

- **strip_type**

Defines how to manipulate the RADIUS attribute value. If multiple attributes are defined for the **radius_attribute** parameter, the same number of comma separated values must exist for this parameter.

There are three possible values for the **strip_type** parameter:

- **remove_suffix**—Strip the suffix from the attribute beginning from a certain character (defined by the **strip_character** parameter).
- **remove_prefix**—Remove the prefix of the attribute ending at a certain character (defined by the **strip_character** parameter).
- **dont_strip**—Leave the attribute value as is (default).

The following is an example of assigning a value to this parameter:

```
strip_type=dont_strip
```

- `strip_character`

Defines how to manipulate the RADIUS attribute value. If multiple attributes are defined for the **radius_attribute** parameter, the same number of comma separated values must exist for this parameter.

The following is an example of assigning a value to this parameter:

```
strip_character=@
```

Radius.Property.Package Section

The [**Radius.Property.Package**] section defines the RADIUS attribute from which a subscriber package is retrieved.



Note

The configuration described in this section is optional.

This section contains the following parameters:

- `radius_attribute`

Specifies the RADIUS protocol attribute number. Use the value of **26** for Vendor Specific Attributes (VSA).

The following is an example of assigning a value to this parameter:

```
radius_attribute=26
```

Use the following two parameters only if the association is based on the vendor specific attribute (VSA).

- `radius_attribute_vendor_id`

This parameter defines the vendor ID number for the subscriber ID classification.

This parameter has no default value.

- `radius_sub_attribute`

This parameter defines the sub attribute within the vendor specific attribute that is used for subscriber ID classification.

This parameter has no default value.

- `radius_attribute_type`

Specifies the RADIUS attribute type.

Possible values for this parameter are **integer** and **string**. The default value is **integer**.

The following is an example of assigning a value to this parameter:

```
radius_attribute_type=integer
```

- `strip_type`

Defines how to manipulate the RADIUS attribute value.

There are three possible values for the **strip_type** parameter:

- **remove_suffix**—Strip the suffix from the attribute beginning from a certain character (defined by the **strip_character** parameter).
- **remove_prefix**—Remove the prefix of the attribute ending at a certain character (defined by the **strip_character** parameter).
- **dont_strip**—Leave the attribute value as is (default).

The following is an example of assigning a value to this parameter:

```
strip_type=dont_strip
```

- `strip_character`

Defines how to manipulate the RADIUS attribute value.

The following is an example of assigning a value to this parameter:

```
strip_character=@
```

- `use_default`

Defines whether to use a default value if the attribute was not found.

Possible values for this parameter are **true** and **false**.

The default value is **true**.

- `default`

Defines the default value to use if the attribute was not found.

There is no default value.



Note This parameter must be set if the **use_default** parameter is set to **true**.

- `mapping_table`

Defines a conversion table between the result of the attribute manipulation (e.g. after stripping), and the property value. The conversion is mandatory if the **radius_attribute_type** is string. The format of the `mapping_table` is **<attribute-value>=<property-value>**.

There is no default value.

RPC.Server Section

The **[RPC.Server]** section represents the PRPC server configuration.

This section contains the following parameters:

- port

Defines the PRPC server port. The default is 14374.

The following is an example of assigning a value to this parameter:

```
port=14374
```

- security_level

Defines whether the PRPC server forces authentication on all connections (**full**), authenticates connections that support authentication while still accepting connections that do not (**semi**), or not to enforce authentication (**none**). When clients attempt to connect to the SM, they are authenticated if configured correctly.

Possible values for this parameter are **full**, **semi**, and **none**. The default value is **semi**.



Note

Starting from version 3.0.5 SCA BB Console supports authentication with the SM PRPC Server; therefore, it can be used in conjunction with all **security_level** values.



Note

From version 3.0.5, the SM Java API, SM C/C++ API, and CNR LEG do not support authentication with the SM PRPC Server; therefore, if installed, the security level cannot be configured to **full**.

The following is an example of assigning a value to this parameter:

```
security_level=semi
```

MPLS-VPN Section

The **[MPLS-VPN]** section contains configuration parameters that are relevant to MPLS/VPN installations. See the *Cisco SCMS SM MPLS/VPN BGP LEG Reference Guide* for a description of Subscriber Management in MPLS/VPN networks.

This section contains the following parameters:

- vpn_id

Defines the BGP attribute to use to identify the VPN subscribers.

Possible values for this parameter are **RD** or **RT**. The default value is **RT**.

The following is an example of assigning a value to this parameter:

```
vpn_id=RD
```

- log_all

Defines the logging level of the BGP LEG.

Possible values for this parameter are **true** or **false**. The default value is **false**. If set to **true** the SM logs all BGP packets that were received by it and is useful during the integration and testing phase.

The following is an example of assigning a value to this parameter:

```
log_all=false
```


SCE.XXX Section

Each [SCE.XXX] section represents a single SCE platform, where XXX represents the SCE logical name.

This section contains the following parameters:

- ip

Defines the IP address of a standalone SCE device or the two IP addresses of a cascade SCE pair.

The following is an example of assigning a value to this parameter:

```
ip=11.12.13.1
```

The following is an example of assigning a value to this parameter for a cascade SCE pair:

```
ip=11.12.13.1,11.12.13.2
```



Note For a cascade SCE pair, the SM adds two SCEs to the SM with the names XXX_cascade1 and XXX_cascade2. Therefore, the names _cascade1 and _cascade2 are reserved suffixes for the SCE names.

- port

Defines the port through which to connect to the SCE platform. The default is 14374.

The following is an example of assigning a value to this parameter:

```
port=14374
```

FTP Section

The SM manages an internal FTP server for various purposes.

The [FTP] section contains the following parameters:

- start

Defines whether the SM should run the FTP server at startup.

Possible values are **yes** and **no**. The default is **no**.

The following is an example of assigning a value to this parameter:

```
start=yes
```

- port

Defines the FTP server port. The default is 21000.

The following is an example of assigning a value to this parameter:

```
port=21000
```

The following parameters define the TCP ports range when the FTP server works in passive mode:

- #first_passive_port = 21001
- #last_passive_port = 21100



Note Uncomment these parameters when working with the FTP server via a firewall.

HTTP Tech-IF Section

The SM manages an internal HTTP adapter server that is a technician interface.

The **[HTTP Tech-IF]** section contains the following parameters:

- start

Defines whether the SM should run the HTTP server at startup.



Note This is a technician interface and normally should not be used.

Possible values are **yes** and **no**. The default is **no**.

The following is an example of assigning a value to this parameter:

```
start=yes
```

- port

Defines the HTTP server port. The default is 8082.

The following is an example of assigning a value to this parameter:

```
port=8082
```

RDR Server Section

The SM manages an internal RDR server that is used to receive RDR messages from the SCE.

The **[RDR Server]** section contains the following parameters:

- start

Defines whether the SM should run the RDR server at startup.



Note This interface should be used when installing the SCE-Sniffer LEGs on the SM.

Possible values are **yes** and **no**. The default is **no**.

The following is an example of assigning a value to this parameter:

```
start=yes
```

- port

Defines the RDR server port. The default is 33001.

The following is an example of assigning a value to this parameter:

```
port=33001
```

- max_connections

Defines the maximum number of connections accepted by the server. The default is 10.

The following is an example of assigning a value to this parameter:

```
max_connections=10
```

Cable Adapter Section

The SM manages a Cable Adapter, a special cable support module that is a translator between the cable world (DHCP events) and the SM. For additional information, see [CPE as Subscriber in Cable Environment, page C-1](#).

The [**Cable Adapter**] section contains the following parameter:

- `allow_dynamic_CM`

Defines whether to allow logins from cable modems (CM) that are not in the SM database.

Possible values are **yes** and **no**. The default is **no**.

The following is an example of assigning a value to this parameter:

```
allow_dynamic_CM=no
```

Data Repository Section

The [**Data Repository**] section defines the SM operation with the TimesTen In-Memory Database.



Note

Some of the parameters in this section are discarded on regular configuration loading, and resetting them requires restarting the SM.

The [**Data Repository**] section contains the following parameters:

- `support_ip_ranges`

Defines whether the SM supports IP-Ranges. Disabling this support provides better performance.



Note Resetting this parameter requires restarting the SM. This parameter is discarded on regular configuration loading (using CLU).

Possible values are **yes** and **no**. The default is **no**.

The following is an example of assigning a value to this parameter:

```
support_ip_ranges=yes
```

- `checkpoint_interval_in_seconds`

Defines the interval, in seconds, for calling the TimesTen checkpoints. Reducing the value affects performance, increasing the value increases vulnerability to power-down.

The default value is 60 (seconds).

The following is an example of assigning a value to this parameter:

```
checkpoint_interval_in_seconds=60
```

- `max_range_size`

Determines the maximum IP range size used in the system.

This parameter is used for improving performance of the SM in Pull mode when the Data Repository section is configured with `support_ip_ranges=yes`.



Note Defining this parameter with too low a value may cause incorrect operation in handling pull requests.

The default value is 256.

The following is an example of assigning a value to this parameter:

```
max_range_size=256
max_number_of_subscribers
```

Determines the maximum number of subscribers the database supports. The limit is 20 million subscribers for Solaris and two million subscribers for Linux.



Note You cannot decrease the value for this parameter without destroying the replicated datastore.



Note Changing this parameter requires a restart of the SM process.



Note To achieve better database performance, it is recommended (when possible) to destroy the subscriber database after changing this parameter and then to rebuild the database.

The default value is 200000.

The following is an example of assigning a value to this parameter:

```
max_number_of_subscribers=1000000
```

- `cache_size`

Determines the subscriber cache size. The cache is used to optimize the database transactions. It should be set to the number of active subscribers that are updated at a high frequency by the management systems.



Note Changing this parameter requires a restart of the SM process.

The default value is 100000.

The following is an example of assigning a value to this parameter:

```
cache_size=1600000
```

- `max_number_of_vpns`

Determines the maximum number of VPNs supported by the SM. This parameter is used to create database indices and to determine database table sizes.



Note You cannot decrease the value for this parameter without destroying the replicated datastore.



Note Changing this parameter requires a restart of the SM process.

The default value is 2048.

The following is an example of assigning a value to this parameter:

```
max_number_of_vpns=3000
```

- `vpn_cache_size`

Determines the VPN cache size. The cache is used to optimize the database transactions. It should be set to the number of active VPNs that are updated at a high frequency by the management systems.



Note Changing this parameter requires a restart of the SM process.

The default value is 2048.

The following is an example of assigning a value to this parameter:

```
vpn_cache_size=3000
```

