CHAPTER 14

Scheduler Management Functions

This chapter provides information about the following scheduler management functions:

- listScheduler, page 14-1
- registerDiscoveryScheduler, page 14-2
- registerScheduler, page 14-2
- registerServerStatusListener, page 14-3
- unregisterDiscoveryScheduler, page 14-4
- unregisterScheduler, page 14-4

listScheduler

Synopsis

```
Schedule[] listScheduler(UserToken token) throws RemoteException;
```

Description

This is a simple scheduling function that allows you to list a scheduled task from the scheduler.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	Token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.

Return

This function returns an array of schedule objects if a schedule has been set. It returns null if a schedule has not been set.

Error and Exception

If a system error prevents the operation from completing, a RemoteException is thrown.

registerDiscoveryScheduler

Synopsis

```
Status registerDiscoveryScheduler(UserToken token, DiscoverySchedule shced) throws
RemoteException;
```

Description

This function allows you to schedule the time, frequency, and DiscoverySetting[] to be performed. The DiscoverySetting[] contains network IP address, netmask, group, and authentication data that is retrieved from getDiscoveryAuthInfo. You can modify the data and set it in DiscoverySchedule.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	Token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
discoverysched	DiscoverySchedule	—	DiscoverySchedule object containing details on when to execute through the scheduler, frequency of execution, and DiscoverySetting[] for the subnet and authentication data.

Return

This function returns a Status object.

Error and Exception

If a system error prevents the operation from completing, a RemoteException is thrown.

When an error occurs, the status object returns the none SUCCESS error code and error message.

registerScheduler

Synopsis

```
registerScheduler(UserToken token, ClmTask taskName, Schedule schedule) throws
RemoteException;
```

Description

This function allows you to schedule the time, frequency and the specific Cisco License Manager action to be performed.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	Token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
taskName			ClmTask object containing Cisco License Manager supported task string that is to be executed through the scheduler.
sched	Schedule	—	DiscoverySchedule object containing details on when to execute through the scheduler, frequency of execution, and DiscoverySetting[] for the subnet and authentication data.

Return

This function returns a Status object.

Error and Exception

If a system error prevents the operation from completing, a RemoteException is thrown.

When an error occurs, the status object returns the none SUCCESS error code and error message.

registerServerStatusListener

Synopsis

```
registerServerStatusListener(UserToken token,StatusListener listener) throws
RemoteException;
```

Description

This function allows the client program to register a listener for the server status. When the connection to the server is lost, the onStatus() method in the listener object will be invoked.

Input Parameters

Parameter	Type	Value	Description
token	UserToken,mandatory	—	Token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
listener	String, mandatory	—	Object that implements the StatusListener interface.

Return

This function returns true if the registration is successful, or false otherwise.

Error and Exception

If a system error prevents the operation from completing, a RemoteException is thrown.

unregisterDiscoveryScheduler

Synopsis

```
Status unregisterDiscoveryScheduler(UserToken token) throws RemoteException;
```

Description

This function allows you to remove a scheduled discovery task from the scheduler.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	Token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.

Return

This function returns a Status object.

Error and Exception

If a system error prevents the operation from completing, a RemoteException is thrown.

When an error occurs, the status object returns the none SUCCESS error code and error message.

unregisterScheduler

Synopsis

```
Status unregisterScheduler(UserToken token, Schedule.ClmTask taskName) throws
RemoteException;
```

Description

This is a simple scheduling function that allows you to remove a scheduled task from the scheduler.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	Token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
taskName	Schedule.ClmTask, mandatory	—	Schedule.ClmTask object containing a Cisco License Manager-supported task string to be executed through the scheduler.

Return

This function returns a Status object.

Error and Exception

If a system error prevents the operation from completing, a RemoteException is thrown.

When an error occurs, the status object returns the none SUCCESS error code and error message.