



CHAPTER 16

User Management Functions

This chapter provides information about the following user management functions:

- [add_user_to_device_access_list](#), page 16-1
- [add_user_to_group_access_list](#), page 16-2
- [add_user_to_pak_access_list](#), page 16-3
- [create_user](#), page 16-4
- [delete_user](#), page 16-5
- [get_associated_actions](#), page 16-5
- [get_user_access_list_from_device](#), page 16-6
- [get_user_access_list_from_group](#), page 16-6
- [get_user_access_list_from_pak](#), page 16-7
- [list_all_users](#), page 16-8
- [read_user_info](#), page 16-8
- [remove_access_list_from_device](#), page 16-9
- [remove_access_list_from_group](#), page 16-9
- [remove_user_from_device_access_list](#), page 16-10
- [remove_user_from_group_access_list](#), page 16-11
- [remove_user_from_pak_access_list](#), page 16-12
- [write_user_info](#), page 16-12

add_user_to_device_access_list

Synopsis

```
add_user_to_device_access_list ($token, $dev, [@usr_ids])
```

Description

This function provides user access control and associates a device with a list of user IDs.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
dev	String, mandatory	—	The device ID.
user_ids	Array of string, mandatory	Name is a string containing up to 64 ASCII characters in the range from x21 to x7A	Array of user IDs to be added to a device access list.

Return

This function returns a Cisco::CLM::Common::DeviceStatus object.

Error and Exception

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

When an operation error occurs, the information is contained in the returned DeviceStatus object. To inspect the status of an individual element, you must traverse the DeviceStatusItem array within DeviceStatus. Each DeviceStatusItem contains the Device object, error code, and error message.

add_user_to_group_access_list

Synopsis

```
add_user_to_group_access_list ($token, $group, [@usr_ids])
```

Description

This function provides user access control. It adds user IDs to the access list of a given group.

Input Parameters

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

Parameter	Type	Value	Description
group	String, mandatory	—	The group ID.
user_ids	Array of string, mandatory	Name is a string containing up to 64 ASCII characters in the range from x21 to x7A	An array of user IDs to be added to a device group access list.

Return

This function returns a Cisco::CLM::Common::GroupStatus object.

Error and Exception

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

This function returns the GroupStatus object that contains none SUCCESS error code and error message when operation error occurs and array of GroupStatusItems. Each GroupStatusItem contains individual Group object and error code and error message.

add_user_to_pak_access_list

Synopsis

```
add_user_to_pak_access_list($token, $pak, [@usr_ids])
```

Description

This function adds an array of user IDs into the access list of a given product authorization key (PAK) name. This function supports user access control.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
pak	String, mandatory	—	The PAK ID.
user_ids	Array of string, mandatory	Name is a string containing up to 64 ASCII characters in the range from x21 to x7A	An array of user IDs to be added to the access list.

Return

This function returns a Cisco::CLM::Common::PAKStatus object. PAKStatus contains an array of PAKStatusItem, and each PAKStatusItem contains the PAK ID, error code, and error message.

Error and Exception

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

The `PAKStatus` contains overall operation status. When a system-level error occurs and the operation cannot be completed, the `PAKStatus` error code returns `none SUCCESS` and an error message. For each individual ID, the `PAKStatus` contains `PAKStatusItem []`. Each `PAKStatusItem` contains a `PAK` object, error code, and error message that shows the individual status.

create_user

Synopsis

```
create_user ($token, $username, $password, $role)
```

Description

This function creates a new user. The user must log in using the administrator role to have permission to invoke this function. If a role is not entered, a default role is assigned.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
username	String, mandatory	Up to 64 ASCII characters in the range from x21 to x7A	The username.
password	String, mandatory	Up to 64 ASCII characters in the range from x21 to x7A	The user password.
role	Role, mandatory	ADMIN, INVENTORYMGR, PAKMGR, LICENSEMGR, REPORTMGR	Defines the role for a user.

Return

This function returns a `Cisco::CLM::Common::UserInfo` object if the operation is successful, are null otherwise.

Error and Exception

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

When an error occurs, this function returns null.

delete_user

Synopsis

```
delete_user ($token, $username)
```

Description

This function deletes a user from the system. The user must log in using the administrator role to have permission to invoke this function.

Input Parameters

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

Return

This function returns Boolean true if the operation is successful and false if it is unsuccessful.

Error and Exception

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

When an error occurs, this function returns false.

get_associated_actions

Synopsis

```
get_associated_actions ($token, $username)
```

Description

This function returns a list of strings representing the actions that can be performed.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
username	String, mandatory	Up to 64 ASCII characters in the range from x21 to x7A	The username.

Return

This function returns an array of String.

Error and Exception

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

When an error occurs, this function returns null.

get_user_access_list_from_device

Synopsis

```
get_user_access_list_from_device ($token, $dev_id)
```

Description

This function returns the user access list of a given device ID as an array of string. Each string is a user ID.

Input Parameters

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

Return

This function returns an array of string.

Error and Exception

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

When an error occurs, this function returns null.

get_user_access_list_from_group

Synopsis

```
get_user_access_list_from_group ($token, $group)
```

Description

This function returns the user access list of a given group as an array of string. Each string is a user ID.

Input Parameters

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

Return

This function returns an array of string.

Error and Exception

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

When an error occurs, this function returns null.

get_user_access_list_from_pak

Synopsis

```
get_user_access_list_from_pak ($token, $pak_id) throws RemoteException;
```

Description

This function returns the user access list of a given PAK as an array of string. Each string is a user ID.

Input Parameters

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

Return

This function returns an array of string.

Error and Exception

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

When an error occurs, this function returns null.

list_all_users

Synopsis

```
list_all_users ($token)
```

Description

This function returns a list of usernames in the system. The user must log in using the administrator role to have permission to invoke this function.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A 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 string representing the names of users.

Error and Exception

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

When an error occurs, this function returns null.

read_user_info

Synopsis

```
read_user_info ($token)
```

```
read_user_info ($token, $username)
```

Description

This function retrieves user information. If token is the only input, it retrieves the user information associated with token. If both token and username are present, it retrieves the information associated with username. The latter can only be called by a user with the Administrator role.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A 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 `Cisco::CLM::Common::UserInfo` object.

Error and Exception

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

When an error occurs, this function returns null.

remove_access_list_from_device

Synopsis

```
remove_access_list_from_device ($token, $dev_id)
```

Description

This function removes an access list from the specified device.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
dev_id	String, mandatory	Name is a string containing up to 64 ASCII characters in the range from x21 to x7A	The device ID of the access list to be removed.

Return

This function returns a `Cisco::CLM::Common::Status` object. If an error occurs, the Status object contains an error code with none `ClmErrors.SUCCESS` and an error message regarding the error.

Error and Exception

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

remove_access_list_from_group

Synopsis

```
remove_access_list_from_group ($token, $group_id)
```

Description

This function removes an access list from the specified group.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
group_id	String, mandatory	Name is a string containing up to 64 ASCII characters in the range from x21 to x7A	The group ID of the access list to be removed.

Return

The function returns a Cisco::CLM::Common::Status object. If an error occurs, the Status object contains the error code none CImErrors.SUCCESS and an error message regarding the error.

Error and Exception

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

remove_user_from_device_access_list

Synopsis

```
remove_user_from_device_access_list ($token, $dev, [@usr_ids])
```

Description

This function provides user access control. It removes a list of user IDs from the access list of a given device.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
dev	String, mandatory	—	Device ID.
user_ids	Array of string, mandatory	Name is a string containing up to 64 ASCII characters in the range from x21 to x7A	An array of user IDs to be removed from a device access list.

Return

This function returns a Cisco::CLM::Common::DeviceStatus object.

Error and Exception

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

When an operation error occurs, information about the error is contained in the returned `DeviceStatus` object. To inspect the individual element status, you must traverse the `DeviceStatusItem` array within `DeviceStatus`. Each `DeviceStatusItem` contains the `Device` object, error code, and error message.

remove_user_from_group_access_list

Synopsis

```
remove_user_from_group_access_list ($token, $group, [@usr_ids])
```

Description

This function provides user access control. It removes a list of user IDs from the access list of a given group.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
group	String, mandatory	Name is a string containing up to 64 ASCII characters in the range from x21 to x7A	The group ID.
user_ids	Array of string, mandatory	Name is a string containing up to 64 ASCII characters in the range from x21 to x7A	An array of user IDs to be removed from a group access list.

Return

This function returns a `Cisco::CLM::Common::GroupStatus` object.

Error and Exception

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

This function returns a `GroupStatus` object that contains none `SUCCESS` error code and error message when an operation error occurs and also an array of `GroupStatusItems`. Each `GroupStatusItem` contains an individual `GROUP` object, error code, and error message.

remove_user_from_pak_access_list

Synopsis

```
remove_user_from_pak_access_list ($token, $pak, [@usr_ids])
```

Description

This function removes an array of user IDs from the access list of a given PAK ID.

Input Parameters

Parameter	Type	Value	Description
token	UserToken, mandatory	—	A token that represents your authorization pass, which is obtained after you invoke the login function and are authenticated by the back-end server.
pak	String, mandatory	ID is a string containing up to 64 ASCII characters in the range from x21 to x7A	The PAK ID.
user_ids	Array of String, mandatory	Name is a string containing up to 64 ASCII characters in the range from x21 to x7A	An array of user IDs to be removed from PAK access list. Each user ID can contain up to 64 printable characters.

Return

This function returns a Cisco::CLM::Common::PakStatus object.

Error and Exception

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

The PakStatus contains overall operation status. When a system-level error occurs and the operation cannot be completed, the PakStatus error code returns none SUCCESS and is accompanied by an error message (if the cause of failure is known). For each individual ID, PakStatus contains PakStatusItem[]. Each PakStatusItem contains a PAK object, error code, and error message that shows the individual status.

write_user_info

Synopsis

```
write_user_info ($token, $user)
```

Description

This function writes the given UserInfo object into the inventory. The input UserInfo object must exist and be retrieved from the inventory by the read_user_info function.

Input Parameters

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

Return

This function returns Boolean true if the operation is successful and false if it is unsuccessful.

Error and Exception

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

When an error occurs, this function returns false.

■ write_user_info