



# APPENDIX **B**

## Sample Client Program in Perl

---

The following text is a client program in Perl. When you connect to Cisco License Manager Server, you must provide the following information:

- IP address of the Cisco License Manager host
- Login name, password, port, idle-timeout and EulaInfo

```
package main;

use strict;
use warnings;

use Cisco::CLM; ## should be the first 'use' statement
use Cisco::CLM::SDK;
use Cisco::CLM::Common;

my $eula_info;
my $lic_manager;
my $token;

## Read Eula
$eula_info = Cisco::CLM::Common::EulaInfo->new;
print "\n Eula Text: ".$eula_info->get_eula_statement;

##Accept EULA
$eula_info->set_eula_accepted(1);
$eula_info->set_do_not_ask_again(1);

$lic_manager = Cisco::CLM::SDK::LicenseManager->new;

## login to the server
$token =lic_manager->login("admin", "password", "localhost", 1099, 0, $eula_info);

#####
# create_devices_by_ip_address
#####

my @dev_auth_info;
my @ips;
my @transport;

push @transport, $Cisco::CLM::Common::Device::TransportMethod::HTTP;
push @ips, "192.19.199.27";
my $auth_info=Cisco::CLM::Common::DeviceAuthentication>new("admin", "pwd", "pwd");

push @dev_auth_info, $auth_info;
my $dev_status=
```

```

$lic_manager-> create_devices_by_ip_addr($token,[@ips], "grp",[@dev_auth_info]);
my $dev_status_item = $dev_status->get_device_status_items;
my $dev_id = $$dev_status_item[0]->get_device_id;

#####
# create_paks
#####
my @pak_id;
push @pak_id, "3XXXREE4B64";

my $pak_status = $licMang->create_paks($token,[@pak_id]);
my $pak_status_items = $pak_status->get_pak_status_items;
my @pak_ids;

## get internal pak id
foreach my $x (@{$pak_status_items}){
    my $pak = $x->get_pak;
    my $id = $pak->get_pak_id;
    push @pak_ids, $id;
}
#####
# download_pak_info
#####

my $id_status = $lic_manager->download_pak_info($token,[@pak_ids]);

## wait until download is complete
##NOTE: download_pak_info is a blocking call. It blocks until the IDStatus is returned by
the server.

my $id_status_items = $id_status->get_id_status_items;

#####
# obtain_license
#####

my @lic_requests;
my $lic_request = Cisco::CLM::Common::LicenseRequest-> new ;
push @lic_requests, $lic_request;

my $pak_1 = $$pak_status_items[0]->get_pak;
my $sku_list = $pak_1->get_sku_list;
my @skus = @{$sku_list};
$lic_requests[0]->set_sku_selection(\@skus);
$lic_requests[0]->set_device_id($dev_id);

### obtain license and deploy it
my $status = $lic_manager->obtain_license($token,[@lic_requests],1);

### Wait for operation to complete
### .....
###.....
##NOTE: obtain_license is a blocking call. It blocks until the IDStatus is returned by the
server.

$lic_manager->logout($token);

```

# Sample Code for Error Code/Messages

The following section describes the procedure for obtaining error codes/messages contained in the following "Status" objects:

- [DeviceStatus, page B-3](#)
- [GroupStatus, page B-3](#)
- [IDStatus, page B-4](#)
- [LicenseStatus, page B-4](#)
- [LicLineStatus, page B-5](#)
- [PAKStatus, page B-5](#)
- [PolicyStatus, page B-6](#)
- [RehostInfoStatus, page B-6](#)
- [ReportStatus, page B-7](#)

## DeviceStatus

The following example shows how to read error code and message from DeviceStatus object:

```
# The general error code of the operation.
my $err_code = $device_status->get_error_code;

# The general error message of the operation.
my $err_msg = $device_status->get_error_message;

# A list of status for each individual element in the
# bulk operation.
my $device_status_items = $id_status->get_device_status_items;

# Iterate through the list to get individual status.
foreach my $k (@{$device_status_items}){

# Get the individual Device object returned by the operation.
my $device = $k->get_device;

#Get the individual error code corresponding to
# the Device
my $item_err_code = $k->get_error_code;

# Get the individual error message corresponding to
# the Device.
my $item_err_msg = $k->get_error_message;
}
```

## GroupStatus

The following example shows how to read error code and message from GroupStatus object:

```
# The general error code of the operation.
my $err_code = $group_status->get_error_code;
```

```

# The general error message of the operation.
my $err_msg = = $group_status->get_error_message;

# A list of status for each individual element in the
# bulk operation.
my $group_status_items = $group_status->get_group_status_items;

# Iterate through the list to get individual status.
foreach my $k (@{$group_status_items}){

    #Get the individual error code corresponding to
    # the object ID.
    my $item_err_code = $k->get_error_code;

    # Get the individual error message corresponding to
    # the object ID.
    my $item_err_msg = $k->get_error_message;
}

```

## IDStatus

The following example shows how to read error code and message from IDStatus object:

```

# The general error code of the operation.
my $err_code = $id_status->get_error_code;

# The general error message of the operation.
my $err_msg = = $id_status->get_error_message;

# A list of status for each individual element in the
# bulk operation.
my $id_status_items = $id_status->get_id_status_items;

# Iterate through the list to get individual status.
foreach my $k (@{$id_status_items}){

    # Get the individual object ID returned by the operation.
    my $id = $k->get_id;

    #Get the individual error code corresponding to
    # the object ID.
    my $item_err_code = $k->get_error_code;

    # Get the individual error message corresponding to
    # the object ID.
    my $item_err_msg = $k->get_error_message;
}

```

## LicenseStatus

The following example shows how to read error code and message from LicenseStatus object:

```

# The general error code of the operation.
my $err_code = $license_status->get_error_code;

# The general error message of the operation.

```

```

my $err_msg = = $license_status->get_error_message;

# A list of status for each individual element in the
# bulk operation.
my $license_status_items = $license_status->get_license_status_items;

# Iterate through the list to get individual status.
foreach my $k (@{$license_status_items}){

    # Get the individual License object returned by the operation.
    my $license = $k->get_license;

    #Get the individual error code corresponding to
    # the License object.
    my $item_err_code = $k->get_error_code;

    # Get the individual error message corresponding to
    # the License object.
    my $item_err_msg = $k->get_error_message;
}

```

## LicLineStatus

The following example shows how to read error code and message from LicLineStatus object:

```

# The general error code of the operation.
my $err_code = $lic_line_status->get_error_code;

# The general error message of the operation.
my $err_msg = = $lic_line_status->get_error_message;

# A list of status for each individual element in the
# bulk operation.
my $lic_line_status_items = $lic_line_status->get_lic_line_status_items;

# Iterate through the list to get individual status.
foreach my $k (@{$lic_line_status_items}){

    # Get the individual LicenseLine object returned by the operation.
    my $lic_line = $k->get_lic_line;

    #Get the individual error code corresponding to
    # the object ID.
    my $item_err_code = $k->get_error_code;

    # Get the individual error message corresponding to
    # the object ID.
    my $item_err_msg = $k->get_error_message;
}

```

## PAKStatus

The following example shows how to read error code and message from PAKStatus object:

```

# The general error code of the operation.
my $err_code = $pak_status->get_error_code;

```

```

# The general error message of the operation.
my $err_msg = = $pak_status->get_error_message;

# A list of status for each individual element in the
# bulk operation.
my $pak_status_items = $pak_status->get_pak_status_items;

# Iterate through the list to get individual status.
foreach my $k (@{$pak_status_items}){

    # Get the individual PAK object returned by the operation.
    my $pak = $k->get_pak;

    #Get the individual error code corresponding to
    # the PAK object
    my $item_err_code = $k->get_error_code;

    # Get the individual error message corresponding to
    # the PAK object.
    my $item_err_msg = $k->get_error_message;
}

```

## PolicyStatus

The following example shows how to read error code and message from PolicyStatus object:

```

# Get the policy object
my $policy = $policy_status->get_policy;

# The general error code of the operation.
my $err_code = $policy_status->get_error_code;

# The general error message of the operation.
my $err_msg = = $policy_status->get_error_message;

```

## RehostInfoStatus

The following example shows how to read error code and message from RehostInfoStatus object:

```

# The general error code of the operation.
my $err_code = $rehost_info_status->get_error_code;

# The general error message of the operation.
my $err_msg = = $rehost_info_status->get_error_message;

# A list of status for each individual element in the
# bulk operation.
my $rehost_info_status_items = $rehost_info_status->get_rehost_info_status_items;

# Iterate through the list to get individual status.
foreach my $k (@{$rehost_info_status_items}){

    # Get the individual RehostInfo object returned by the operation.
    my $rehost_info = $k->get_rehost_info;
}

```

```
#Get the individual error code corresponding to
# the RehostInfo object.
my $item_err_code = $k->get_error_code;

# Get the individual error message corresponding to
# the RehostInfo object.
my $item_err_msg = $k->get_error_message;
}
```

## ReportStatus

The following example shows how to read error code and message from ReportStatus object:

```
# The general error code of the operation.
my $err_code = $report_status->get_error_code;

# The general error message of the operation.
my $err_msg = $report_status->get_error_message;

# Get the content of the report
my $content = $report_status->get_content
```

