Lenovo xClarity Administrator Ansible Documentation

This project contains Ansible Playbooks, Roles and Modules for LXCA and can be used collectively to implement various use cases. Project contains following Ansible Roles

- Inventory : Role to get all inventory from LXCA.
- Configuration : Role to do config operation, firmware update, apply patterns,os deploy.

Pre-requisite

Ansible Role requires LXCA Python Client and LXCA Ansible module installed.

- Ansible version 2.4.2 or later (Ansible installation documentation)
- Python Client for Lenovo xClarity Administratr.(LXCA Python Client v2.1.0)

```
pip install pylxca
```

Installation

ansible-galaxy install lenovo.lxca-inventory
ansible-galaxy install lenovo.lxca-config

Ansible Role: Lenovo LXCA Inventory

Ansible Role to collect get inventory data of managed elements from Lenovo xClarity Administrator.

Role Variables

Available variables are listed below, along with description:

Variable	Description
uuid	uuid of device
id	id of job
update_list	update list for tasks
discover_ip	discover specific ip

Supported Tags

Supported tags are listed below, along with description:

tags	Description
chassis	chassis details
cmms	cmms details
nodes	nodes details
discover	discover using slp
fans	fans details
fanmuxes	fanmux details
jobs	job details
lxcalog	lxca log

tags	Description
powersupplies	powersupply details
scalablesystem	scalablesystem details
switches	switch details
tasks	task details and update
users	user details
ffdc	ffdc for uuid

Variable	Description
uuid	UUID of a managed element
id	ID of a resource
endpoint_ip	IP address of Endpoint
user	user name
password	password
recovery_password	recovery password for resource
jobid	job id of background job
mode	operation mode for config action
lxca_action	operation for config
server	compute node details
switch	switch details
storage	storage details
cmm	cmm details
force	force flag for config action
policy_info	policy detail
policy_name	policy name
policy_type	policy type
repo_key	repository key
machine_type	machine type
scope	operation scope
fixids	firmware image id
file_type	type of file in config action
endpoint	target managed endpoint for config action
restart	config action
config_pattern_name	name of config patteren
config_profile_name	name of config profile
powerdown	power operation
resetimm	action reset imm
pattern_update_dict	dictionary of category pattern information
pattern_from_file	file path for pattern data
includeSettings	flag for reading config data
osimages_info	information about os images
osimages_dict	dictionary of information about os images

Variable	Description
files	files location
update_key	key for firmware update
status	status of managed element
uuid_list	list of UUID

Dependencies

Connectivity with Lenovo xClarity Administrator.

Example Playbook

To execute an Ansible playbook, use the following command:

```
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217" inventory.yml -vvvv
```

-vvv is an optional verbos command that helps identify what is happening during playbook execution.

```
- name: get nodes data from LXCA
  pylxca_module:
    login_user: "{{ lxca_user }}"
    login_password: "{{ lxca_password }}"
    auth_url: "{{ lxca_url }}"
    command_options: nodes
  register: rslt
  tags:
     nodes
```

Ansible Role: Lenovo LXCA Configuration

Ansible Role for various configuration operation (like manage, update firmware, config policy, os deploy) on managed elements from Lenovo xClarity Administrator.

Mandatory Variables

Available variables are listed below, along with description:

Variable	Description
lxca_user	lxca user to connect to lxca
lxca_password	password of user
lxca_url	lxca url to connect

Role Variables

Available variables are listed below, along with description:

Variable	Description
uuid	UUID of a managed element
id	ID of a resource
endpoint_ip	IP address of Endpoint
user	user name
password	password
recovery_password	recovery password for resource
storedcredential_id	id of stored credential
jobid	job id of background job
mode	operation mode for config action
lxca_action	operation for config
server	compute node details
switch	switch details
storage	storage details
cmm	cmm details
force	force flag for config action
policy_info	policy detail
policy_name	policy name
policy_type	policy type
repo_key	repository key
machine_type	machine type
scope	operation scope
fixids	firmware image id
file_type	type of file in config action
endpoint	target managed endpoint for config action
restart	config action
config_pattern_name	name of config patteren
config_profile_name	name of config profile
powerdown	power operation
resetimm	action reset imm
pattern_update_dict	dictionary of category pattern information
pattern_from_file	file path for pattern data
includeSettings	flag for reading config data
osimages_info	information about os images
osimages_dict	dictionary of information about os images
files	files location
update_key	key for firmware update
status	status of managed element
uuid_list	list of UUID

Supported Tags

Supported tags are listed below, along with description:

tags	Description
get_configpatterns	return all server patterns
get_particular_configpattern	get config pattern by name or id
import_configpatterns	import config pattern to lxca
apply_configpatterns	apply config pattern to device
get_configstatus	get config status of device by uuid
configprofiles	config profiles operations
configtargets	get config targets details
manage	perform manage operation on discovered device
manage_status	check completion status of manage operation
unmanage	perform unmanage operation
unmanage_status	check completetion status of unmanage operation
osimages	Perform osimage and os deployment opertion
updaterepo	update repository operation
update_firmware	update firmware for specific device with specified mt and fixids
update_firmware_all	update firmware for device with list of uuid with assigned policy
update_firmware_query_status	check status of firmware update
update_firmware_query_comp	list updateable components
get_managementserver_pkg	get packages detail management server
update_managementserver_pkg	update management server packages
import_managementserver_pkg	import management server from local system
updatepolicy	update compliance policy operations
get_storedcredentials	get stored credentials
create_storedcredentials	create new stored credentials
update_storedcredentials	update existing stored credentials
delete_storedcredentials	delete stored credentails

Dependencies

Connectivity with Lenovo xClarity Administrator.

Example Playbook

To execute an Ansible playbook, you need to choose one of tag specified above, also supply variable used by tag use the following command:

manage

manage uses following additional variable user, password or storedcredential_id

```
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
```

```
lxca_url=https://10.240.29.217 endpoint_ip=10.240.72.172 user=USERID
password=CME44ibm recovery_password=CME55ibm force=True" test.yml -vvvv --tag
manage

ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217 endpoint_ip=10.240.72.172 storedcredential_id=21
force=True" test.yml -vvvv --tag
manage
```

unmanage

unmanage uses following additional variable endpoint_ip which is ip_address,uuid,device_type device_type can have following value Chassis ThinkServer Storage Rackswitch Rack-Tower"

```
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217
endpoint_ip=10.240.72.172;46920C143355486F97C19A34ABC7D746;Chassis force=True"
test.yml -v --tag unmanage
```

Updatepolicy and Update Firmware

Following additional variable are used with these tag

policy_info: description: - used with command updatepolicy following values are possible - "FIRMWARE - Get List of Applicable Frimware policies" - "RESULTS - List the persisted compare result for servers to which a policy is assigned" - "COMPARE_RESULTS -Check compliant with the assigned compliance policy using the job or task ID that was returned when the compliance policy was assigned." - "NAMELIST - Returns the available compliance policies"

policy_name: description: used with command updatepolicy, name of policy to be applied

policy_type: description: - used with command updatepolicy, policy applied to value specified it can have following value - CMM - Chassis Management Module - IOSwitch - Flex switch - RACKSWITCH - RackSwitch switch - STORAGE - Lenovo Storage system - SERVER - Compute node or rack server

uuid_list: description: - used with command update_all_firmware_withpolicy. Apply firmware to uuid in list - if uuid_list is empty firmware is updated for all updateable components with policy. - "example ['38D9D7DBCB713C12A210E60C74A0E931','00000000000010008000542AA2D3CB00']"

mode: description: - "used with command update_firmware, update_all_firmware_withpolicy Indicates when to activate the update. This can be one of the following values." - "immediate - Uses Immediate Activaton mode when applying firmware updates to the selected endpoints." - "delayed - Uses Delayed Activaton mode when applying firmware updates to the selected endpoints."

```
List all  policy
----------------
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217" playbooks/config/config.yml -vvvv --tag
```

```
updatepolicy

Get List of Applicable Frimware policies
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 policy_info=FIRMWARE" playbooks/config/config.yml
-vvvv --tag updatepolicy

List  the persisted compare result for servers to which a policy is assigned
----------------

ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 update_policy_info=RESULTS"
playbooks/config/config.yml -vvvv --tag updatepolicy

Check compliant with the assigned compliance policy using the job or task ID that
was returned when the compliance policy was assigned.
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 update_policy_info=COMPARE_RESULTS
uuid=EF362CF0FB4511E397AB40F2E9AF01D0 jobid=2" playbooks/config/config.yml -vvvv
--tag updatepolicy

Assign policy to Endpoint
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 policy_name=x220_imm2 policy_type=SERVER
uuid=7C5E041E3CCA11E18B715CF3FC112D8A" playbooks/config/config.yml -vvvv --tag
updatepolicy
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 jobid=16" playbooks/config/config.yml -vvvv --tag
updatepolicy

Update endpoint Firmware
================

Query Updatable components
----------------
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217" playbooks/config/config.yml -v --tag
query_update_comp

Query Firmware Update Status
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220" playbooks/config/config.yml -vvvv --tag
query_update_status

Applying Firmware with policy
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 mode=immediate lxca_action=apply
server='7C5E041E3CCA11E18B715CF3FC112D8A,IMM2 (Primary)'"
playbooks/config/config.yml -vvvv --tag update_firmware

Applying Firmware with policy for specified updateable components
----------------
```

```
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217 mode=immediate lxca_action=apply
uuid_list=['38D9D7DBCB713C12A210E60C74A0E931','00000000000010008000542AA2D3CB00']"
playbooks/config/config.yml -vvvv --tag update_all_firmware_withpolicy

Applying Firmware with policy for all updateable components
---------------
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217 mode=immediate lxca_action=apply uuid_list=[]"
playbooks/config/config.yml -vvvv --tag update_all_firmware_withpolicy
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217 mode=immediate lxca_action=apply"
playbooks/config/config.yml -vvvv --tag update_all_firmware_withpolicy
```

Config Patterns, Config Profile and Config Targets operations

Following additional variables are used for this tag

endpoint: description: - used with command configprofiles, apply_configpatterns and get_configstatus, - its uuid of deivce for node, rack, tower - endpointdid for flex

restart: description: - used with command configprofiles and apply_configpatterns - when to activate the configurations. This can be one of the following values - defer - Activate IMM settings but do not restart the server. - immediate - Activate all settings and restart the server immediately - pending - Manually activate the server profile and restart the server. this can be used with apply_configpatterns only. choices: - defer - immediate - pending

type: description: - used with apply_configpatterns valid values are choices: - node - rack - tower - flex

powerdown: description: used with command configprofiles to power down server

resetimm: description: used with command configprofiles to reset imm

pattern_update_dict: description: used with command import_configpatterns to import pattern specified in this variable as dict.

includeSettings: description: used with command get_configpatterns to get detailed settings of configpattern set this to 'True'

```
Get All config patterns
---------------
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217" playbooks/config/config.yml -vvvv --tag
get_configpatterns

Get specified config pattern with id
---------------
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217 id=52" playbooks/config/config.yml -vvvv --tag
get_particular_configpattern
```

```
Get specified config pattern for id with includeSettings
----------------
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217 id=52 includeSettings=True"
playbooks/config/config.yml -vvvv --tag get_particular_configpattern

Apply pattern to endpoint
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 id=65 endpoint=B918EDCA1B5F11E2803EBECB82710ADE
restart=pending type=node" playbooks/config/config.yml -vvvv --tag
apply_configpatterns
using name
----------
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217 config_pattern_name=From_abcdef
endpoint=B918EDCA1B5F11E2803EBECB82710ADE restart=pending type=node"
playbooks/config/config.yml -vvvv --tag apply_configpatterns
Import SystemInfo pattern
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220
pattern_update_dict={'template_type':'SystemInfo','template':{'contact':'contact','
description':'Pattern created by test API
','location':'location','name':'Learned-System_Info-99','systemName':{'autogen':'Di
sable','hyphenChecked':False},'type':'SystemInfo','uri':'\/config\/template\/61','u
serDefined':True}}" playbooks/config/config.yml -vvvv --tag import_configpatterns

Import Pattern from file
------------------------
Read config pattern data from config_pattern_import.yml file in vars folder of
config
roles
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 pattern_file=config_pattern_import.yml
pattern_from_file=true" playbooks/config/config.yml -vvvv --tag
import_configpatterns

config status
------------
get config status
ansible-playbook -e "lxca_user=USERID lxca_password=CME44ibm
lxca_url=https://10.240.29.217 endpoint=B918EDCA1B5F11E2803EBECB82710ADE
status=True" playbooks/config/config.yml -vvvv --tag get_configstatus
```

Config Profile operations

```
get all profiles
----------------
ansible-playbook -e "lxca_user=USERID lxca_password=Passw0rd
lxca_url=https://10.240.29.217 " playbooks/config/config.yml -v --tag
configprofiles

get specified profile with id
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
```

```
lxca_url=https://10.240.29.220 id=69 " playbooks/config/config.yml -vvvv --tag
configprofiles

Change profile name of id
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 id=69 config_profile_name='changed name 3' "
playbooks/config/config.yml -vvvv --tag configprofiles

Activate profile for endpoint
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 id=69 endpoint=B918EDCA1B5F11E2803EBECB82710ADE
restart=immediate " playbooks/config/config.yml -vvvv --tag configprofiles

Unassign profile
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 id=69 lxca_action=unassign "
playbooks/config/config.yml -vvvv --tag configprofiles

Delete profile
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 id=69 lxca_action=delete"
playbooks/config/config.yml -vvvv --tag configprofiles
```

Update Repostory commands

Following additional variables are used for this tag

machine_type: description: - used with command updaterepo - its string with value like '7903'

fixids: description: - used with command updaterepo , get_particular_managementserver_pkg and update_managementserver_pkg - its string with value like 'lnvgy_sw_lxca-fw-repository-pack_1-1.0.1_anyos_noarch'

scope: description: - used with command updaterepo, following are possible values - all - When lxca_action=refresh is specified, this parameter returns information about all versions of all firmware updates that are available for all supported devices. - latest - When lxca_action=refresh is specified, this parameter returns information about the most current version of all firmware updates for all supported devices. - payloads - When lxca_action=acquire is specified, this parameter returns information about specific firmware updates.

file_type: description: - used with command updaterepo, When lxca_action=delete or lxca_action=export is specified, this parameter is used. You can specify one of the following values - all - Deletes selected update-package files (payload, change history, readme, and metadata files) - payloads - Deletes only the selected payload image files

```
Queries
----------------
```

```
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 repo_key=importDir" playbooks/config/config.yml
-vvvv --tag updaterepo
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 repo_key=lastRefreshed" playbooks/config/config.yml
-vvvv --tag updaterepo
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 repo_key=publicKeys" playbooks/config/config.yml
-vvvv --tag updaterepo
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 repo_key=size" playbooks/config/config.yml -vvvv
--tag updaterepo
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 repo_key=supportedMts" playbooks/config/config.yml
-vvvv --tag updaterepo
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 repo_key=updates" playbooks/config/config.yml -vvvv
--tag updaterepo
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 repo_key=updatesByMt" playbooks/config/config.yml
-vvvv --tag updaterepo
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 repo_key=updatesByMtByComp"
playbooks/config/config.yml -vvvv --tag updaterepo

Action
----------------
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 lxca_action=read" playbooks/config/config.yml -vvvv
--tag updaterepo
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 lxca_action=refresh machine_type=7903"
playbooks/config/config.yml -vvvv --tag updaterepo
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 lxca_action=delete machine_type=7903
file_type=payloads fixids=ibm_fw_imm2_1aoo78j-6.20_anyos_noarch"
playbooks/config/config.yml -vvvv --tag updaterepo
ansible-playbook -e "lxca_user=TEST lxca_password=CME44ibm
lxca_url=https://10.240.29.220 lxca_action=acquire machine_type=7903 scope=payloads
fixids=ibm_fw_imm2_1aoo78j-6.20_anyos_noarch" playbooks/config/config.yml -vvvv
--tag updaterepo
```

managementserver operations

Following additional variables are used for this tag

fixids: description: - used with command updaterepo , get_particular_managementserver_pkg and update_managementserver_pkg - its string with value like 'lnvgy_sw_lxca-fw-repository-pack_1-1.0.1_anyos_noarch'

update_key: description: - Used with managementserver commands following are valid options Returns the specified type of update. This can be one of the following values. - all - Returns all information. This is the default value. - currentVersion - Returns the current version of Lenovo XClarity Administrator. - history Returns the history of management-server updates. - importDir

Returns the directory for the management-server updates repository. - size - Returns the repository size (in bytes). - updates - Returns information about all updates packages. - updatedDate - Returns the date when the last update was performed.

files: description: - Used with managementserver commands to import files to LXCA file can be specified as comma separated string - example - '/home/naval/updates/updates/lnvgy_sw_lxca_thinksystemrepo1-1.3.2_anyos_noarch.txt, /home/naval/updates/updates/lnvgy_sw_lxca_thinksystemrepo1-1.3.2_anyos_noarch.chg, /home/naval/updates/updates/lnvgy_sw_lxca_thinksystemrepo1-1.3.2_anyos_noarch.xml'

```
get managementserver with different key options
-----------------------------------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','update_key':'importDir'}"
playbooks/config/config.yml -vvvv --tag get_managementserver_pkg

ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','update_key':'size'}"
playbooks/config/config.yml -vvvv --tag get_managementserver_pkg

ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','update_key':'updates'}"
playbooks/config/config.yml -vvvv --tag get_managementserver_pkg

get particular details with fixids
----------------------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','update_key':'filetypes',
'fixids':'lnvgy_sw_lxca-fw-repository-pack_1-1.0.1_anyos_noarch'}"
playbooks/config/config.yml -vvvv --tag get_particular_managementserver_pkg

nsible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217',
'fixids':'lnvgy_sw_lxca-fw-repository-pack_1-1.0.1_anyos_noarch','type':'readme'}"
playbooks/config/config.yml -vvvv --tag get_particular_managementserver_pkg

Update options for managementserver
-----------------------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217', 'lxca_action':'refresh'}"
playbooks/config/config.yml -vvvv --tag update_managementserver_pkg

ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217',
'fixids':'lnvgy_sw_lxca-fw-repository-pack_1-1.0.1_anyos_noarch','lxca_action':'acq
uire'}" playbooks/config/config.yml -vvvv --tag update_managementserver_pkg

ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217',
'fixids':'lnvgy_sw_lxca-fw-repository-pack_1-1.0.1_anyos_noarch','lxca_action':'del
ete'}" playbooks/config/config.yml -vvvv --tag update_managementserver_pkg

Import local files to managementserver
--------------------------------------
```

```
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','lxca_action':'import',
'files':'/home/naval/updates/updates/lnvgy_sw_lxca_thinksystemrepo1-1.3.2_anyos_noa
rch.txt,/home/naval/updates/updates/lnvgy_sw_lxca_thinksystemrepo1-1.3.2_anyos_noar
ch.chg,/home/naval/updates/updates/lnvgy_sw_lxca_thinksystemrepo1-1.3.2_anyos_noarc
h.xml'}" playbooks/config/config.yml -vvvv --tag import_managementserver_pkg

files specified with relative to playbook file
----------------------------------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','lxca_action':'import',
'files':'../../files/updates/lnvgy_sw_lxca_thinksystemrepo1-1.3.2_anyos_noarch.xml'
}" playbooks/config/config.yml -vvvv --tag import_managementserver_pkg
```

osimages operations

Following additional variables are used for this tag

osimages_info: description: - Used with osimage it can have following values - globalSettings - Setting global values used in os deployment - hostPlatforms - Used for deploying os images - remoteFileServers - Used for remote ftp, http server operations

osimages_dict: type: dict description: Used with osimage it is used for setting osimage and os deployment parameters.

```
get all osimages
-----------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217'}" playbooks/config/config.yml -v --tag osimages

get globalSetting for osimages
------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','osimages_info':'globalSettings'}"
playbooks/config/config.yml -vvvv --tag osimages

import osimage file from remote server
--------------------------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217', 'osimages_imagetype_dict':{'imageType':'OS'},
'import_dict':{'imageType':'OS','os':'rhels','imageName':'fixed','path':'iso/rhel73.
iso','serverId':'1'}}" playbooks/config/config.yml -vvvv --tag import_osimages

get hostplatforms detail for osimages
-------------------------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','osimages_info':'hostPlatforms'}"
playbooks/config/config.yml -vvvv --tag osimages

deploy osimage to node
----------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','osimages_info':'hostPlatforms','osimages_dict':
```

```
{'networkSettings':{'dns1':
'10.240.0.10','dns2':'10.240.0.11','gateway':'10.240.28.1','ipAddress':'10.240.29.2
26','mtu':1500,'prefixLength':64,'selectedMac':'AUTO','subnetMask':'255.255.252.0',
'vlanId':521},'selectedImage':'rhels7.3|rhels7.3-x86_64-install-Minimal','storageSe
ttings':{'targetDevice':'localdisk'},'uuid':'B918EDCA1B5F11E2803EBECB82710ADE'}}"
playbooks/config/config.yml -vvvv --tag osimages

get all remoteFileServers
-------------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','osimages_info':'remoteFileServers'}"
playbooks/config/config.yml -vvvv --tag osimages

Get Specific remoteFileServers
------------------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','osimages_info':'remoteFileServers',
'osimages_dict':{'id':'1'}}" playbooks/config/config.yml -vvvv --tag osimages

Delete Specific Remote File Server
----------------------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','osimages_info':'remoteFileServers',
'osimages_dict':{'deleteid':'2'}}" playbooks/config/config.yml -vvvv --tag osimages

Add Remote File Server
----------------------
ansible-playbook -e "{'lxca_user':'USERID', 'lxca_password':'Passw0rd',
'lxca_url':'https://10.240.29.217','osimages_info':'remoteFileServers',
'osimages_dict':{'displayName':'TEST99','address': '192.168.1.10','keyPassphrase':
'Passw0rd','keyType': 'RSA-2048','port': 8080,'protocol': 'HTTPS'}}"
playbooks/config/config.yml -vvvv --tag osimages
```

-vvv is an optional verbose command that helps identify what is happening during playbook execution.

```
- name: get configtargets data from LXCA
  pylxca_module:
    login_user: "{{ lxca_user }}"
    login_password: "{{ lxca_password }}"
    auth_url: "{{ lxca_url }}"
    id: "{{ id }}"
    command_options: configtargets
  register: rslt
  tags:
     configtargets
```

unittest cases

```
It uses mock and nose modules. For coverage it uses coverage module.
run unittest from root folder of this repo

nosetests -v -s  test/test_pylxca_module.py
```

```
nosetests -v -s  test/test_pylxca_module.py --with-coverage
nosetests -v -s  test/test_pylxca_module.py --with-coverage --cover-package=pylxca
```