Post Installation Tasks

This chapter contains the following sections:

Note It is recommended that you ignore the `do not edit` message, if you want to modify the cloud-init day-0 configuration, file.

ESC Lifecycle Notifications During Login

When logging in to ESC through SSH, the message of the day may include an End-of-Life (EoL) or End-of-Support (EoS) notice. Note that each version of ESC is EoL on the release date of the next version, however this does not mean that the software is now defunct but rather that Cisco are providing a newer version. In fact, as the message advises, support and releases of software maintenance patches continue as normal. For example:

```
$ ssh admin@ESC_IP

*** ESC NOTICE: ESC 5.7 EOL on 29 July 2022 ***
*** Please note that this version continues to be supported until 29 January 2026 ***

admin@ESC_IP$
```

A further notice appears when the EoS date is 30 days away:

```
*** ESC NOTICE: Support for ESC 5.7 ends on 29 January 2026 ***
*** Please contact your Cisco Account Manager for details on how to upgrade ***
```

These notices ensure that you upgrade in a timely manner and have the latest feature set and security patches installed.

Changing the ESC Password

You will be forced to change the default password on the first time login. Portal will not let you bypass this step and will keep returning you to this page until you change the default password. After the first time password change, you can change your password using the procedures described in this section. Also, if the user has multiple browsers or tabs or the SAME user is logged on by 2 or more computers and one of the user changes the password then everyone will be logged off and asked to re-enter the new password. The user session has an expiry of 1 hour so if the user is inactive on the portal for an hour then portal will expire the session and the user will have to re-login. If you forgot your password, you can also update or randomly generate the password.

This section discusses how to change the passwords.

Example for REST:

```
sudo escadm rest set --username {USERNAME} --password {PASSWORD}
```

Example for ETSI:

```
sudo escadm etsi set --rest_user {USERNAME:PASSWORD}
```

Changing the ConfD Netconf/CLI Administrator Password Using the Command Line Interface

After you install ESC, to change the Confd admin password, do the following:

You cannot execute the confd commands, such as confd_cli. The *confd_cli -u admin* is replaced with the *ssh admin@localhost -p 2024* command.

For information on Installing ESC, see Installing Cisco Elastic Services Controller Using the QCOW Image.

To access the confD cli for an admin account:

```
admin@esc$ ssh admin@localhost -p 2024
admin@localhost's password: *******

admin connected from 127.0.0.1 using ssh on esc
admin@esc>
```

Procedure

Step 1 Log in to the ESC VM.

```
$ ssh USERNAME@ESC_IP
```

Step 2 Switch to the admin user.

```
[admin@esc-ha-0 esc]$ sudo bash
[sudo] password for admin:
```

Step 3 Load the ConfD CLI:

```
$ /opt/cisco/esc/confd/bin/ssh admin@localhost -p 2024
```

Step 4 Set the new admin password:

```
$ configure
$ set aaa authentication users user admin password <new password>
```

Step 5 Save the changes.

```
$ commit
```

Creating Readonly User Group for ConfD in ESC

ConfD in ESC is enhanced with the introduction of a new group named readonly. If you are a member of readonly group, you can only retrieve the information and you cannot modify the permissions.

You can use 'readonly' as the role name with bootvm. The following example shows how to create two users in ConfD. One is admin and the other is readonly:

```
# bootvm.py name-500-105-100 --user_confd_pass admin:admin --user_confd_pass
readonly:readonly::readonly --user_pass admin:admin --image ESC-5_0_0_105 --net network
```

For HA A/A, you can use 'readonly' as the group name in aa-day0.yaml Following is the example:

```
confd:
  init_aaa_users:
  - group: readonly
    name: admin
    passwd: $6$rounds=4096$Ps1JIjKihRTF$fo8XPBxwEHJWWfNiXDnO269rlhAxAhWBc
PBfGnZxy1gM3QMxcN8jJ6guWt9Bu.ZkWdPt3hr0OghO73Wr3iDHb0
```

You can also create a confd readonly user after ESC vm is deployed. The following steps create a confd readonly user named 'test' with password 'test':

```
[root@name-500-155 admin]#  /opt/cisco/esc/confd/bin/ssh admin@localhost -p 2024
admin connected from 127.0.0.1 using console on name-500-155
admin@name-500-155> configure
Entering configuration mode private
[ok][2019-12-06 18:17:39]
[edit]
admin@name-500-155% set aaa authentication users user test uid 9000 gid 9000 password $0$test
 homedir /var/confd/homes/test ssh_keydir /var/confd/homes/test/.ssh
[ok][2019-12-06 18:19:15]
[edit]
admin@name-500-155% set nacm groups group readonly user-name test
[ok][2019-12-06 18:19:41]
[edit]
admin@name-500-155% commit
Commit complete.
[ok][2019-12-06 18:19:47]
[edit]
admin@name-500-155%
```

As a readonly user, you can also access ConfD remotely:

```
name@my-server-39:~$ ssh -p 2024 readonly@172.29.0.57
readonly@172.29.0.57's password:
readonly connected from 172.16.103.46 using ssh on name-500-156
readonly@name-500-156> configure
Entering configuration mode private
[ok][2019-12-13 16:15:33]
[edit]
readonly@name-500-156% show esc_datamodel
tenants {
    tenant admin {
        description      "Built-in Admin Tenant";
        managed_resource false;
        vim_mapping      true;
    }
}
[ok][2019-12-13 16:15:38]
[edit]
```

ESC in ConfD sends access-denied error if you fall under readonly ConfD group, and require modify permissions. Following is the example of the access-denied error message:

```
$ esc_nc_cli --user readonly --password ****** edit-config dep.xml
Configure
/opt/cisco/esc/confd/bin/netconf-console --port=830 --host=127.0.0.1 --user=readonly
--password=****** --edit-config=/tmp/d.xml
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <rpc-error>
    <error-type>application</error-type>
    <error-tag>access-denied</error-tag>
    <error-severity>error</error-severity>
  </rpc-error>
</rpc-reply>
```

If ESC is configured to use PAM/IDM. The groups in IDM servers are directly mapped to the groups in ConfD. Hence, the readonly user must be mapped in the IDM group 'readonly'.

For example:

```
$ ipa group-find  --all readonly
---------------
1 group matched
---------------
  dn: cn=readonly,cn=groups,cn=accounts,dc=linuxsysadmins,dc=local
  Group name: readonly
  GID: 5003
  Member users: readonly
  ipantsecurityidentifier: S-1-5-21-2222126199-2113948134-574478857-1003
  ipauniqueid: 858b8cda-0d34-11ea-bca8-525400b29c19
  objectclass: top, groupofnames, nestedgroup, ipausergroup, ipaobject, posixgroup,
ipantgroupattrs
----------------------------
Number of entries returned 1
----------------------------
```

Restricting ESC ConfD CLI access

Access to the ESC ConfD CLI is disabled by default for new ESC deployments starting from ESC 5.6 release, and also to the upgrades from earlier version of ESC to ESC 5.6 version. This restriction is introduced as a level of protection from user using the ConfD CLI command without knowing its implications.

When ConfD CLI access is enabled, you can login to the ConfD CLI using one of the following ways:

- `esc_nc_cli`

```
[admin@esc-test1606-confd-instance ~]$ esc_nc_cli cli
ssh -o StrictHostKeyChecking=no -p 2024 admin@127.0.0.1
admin@127.0.0.1's password:

admin connected from 127.0.0.1 using ssh on esc-test1606-confd-instance.novalocal
admin@esc-test1606-confd-instance>
```

- `confd_cli`

```
[admin@esc-test1606-confd-instance ~]$ sudo -i

   ###################################################################
   #         ESC on esc-test1606-confd-instance.novalocal
   ###################################################################

[root@esc-test1606-confd-instance ~]#
[root@esc-test1606-confd-instance ~]# source /opt/cisco/esc/confd/confdrc
[root@esc-test1606-confd-instance ~]# confd_cli -u admin -C

admin connected from 127.0.0.1 using console on esc-test1606-confd-instance.novalocal
esc-test1606-confd-instance#
```

Access to the ESC ConfD CLI using `esc_nc_cli` or `confd_cli` commands is disabled by default starting ESC 5.6 release. Use the following commands to enable or disable the access to ConfD CLI:

1. Command to enable ConfD CLI access

```
esc_nc_cli cli enable
```

Note Once the ConfD CLI access is enabled, access privilege will be purely based on Network Configuration Access Control Model(NACM) rules list defined in ESC. Refer RFC8341 for more details on NACM.

2. Command to disable ConfD CLI access

```
esc_nc_cli cli disable
```

Changing Linux Account Password

Procedure

Step 1 Log in to ESC VM.

```
$ ssh USERNAME@ESC_IP
```

Step 2 To update or generate a random password, use the following command:

```
/usr/bin/pwqcheck
/usr/bin/pwqgen
```

Changing the ESC Portal Password

The user can update or reset the default admin password.

Procedure

Step 1 Log in to ESC VM.

Step 2 Switch to the root user.

Step 3 To update the default admin password or randomly generate a password, use one of the following method:

- Using escadm utility:

  To update the default admin password (admin/********):

  ```
  [root@anyname-v44-52 admin]# sudo escadm portal set --username admin --password ********
  Successfully updated password for username admin
  ```

  To generate a random password:

  ```
  [root@anyname-v44-52 admin]# sudo escadm portal set --username admin
  Would you like to use the generated password: "Accent5omit&Wide"?[y|n]y
  Successfully updated password for username admin
  ```

  The *--must_change* variable will ask the user to change their password at the next login.

  The *--must_change* variable is not applicable for REST users.

  ```
  [root@anyname-v44-52 admin]# sudo escadm portal set --username admin --must_change
  Would you like to use the generated password: "Rainy4Dozen&Behave"?[y|n]y
  Successfully reset password for username admin. User must change the password at the
  next login.
  ```

- To reset to a specific password:

  ```
  [root@anyname-v44-52 admin]# sudo escadm portal set --username admin --password *******
   --must_change
  Successfully reset password for username admin. User must change the password at the
  next login.
  ```

- Using the bootvm command line:

  ```
  --user_portal_pass admin:<new password>
  ```

- Using the ESC Portal:

  a. Log in to ESC portal using your username and password.

  b. Choose **Accounts Setting** on the Navigation menu.

  c. Enter the old password in the Old password field, then enter a new password in the New Password and Confirm Password fields.

  d. Click **Update Password**.

Configuring Pluggable Authentication Module (PAM) Support for Cisco Elastic Services Controller

You can configure the ESC services to use Pluggable Authentication Modules (PAM) for user authentication in ESC. With Cisco Elastic Services supporting PAM, you can also enable LDAP authentication in ESC. If PAM is not configured, ESC will continue to use the default authentication method for each ESC service. The following table lists the commands to enable PAM authentication for each ESC service.

Table 1: Configuring PAM for ESC Services

ESC Service/Component	Command to configure PAM authentication
ESCManager (REST interface)	`sudo escadm escmanager set --auth PAM:<pam_service_name>`
ESC Monitor (Health API)	`sudo escadm monitor set --auth PAM:<pam_service_name>`
Confd	`sudo escadm confd set --auth PAM:<pam_service_name>`
Portal	`sudo escadm portal set --auth PAM:<pam_service_name>`
ETSI	`sudo escadm etsi set --pam_service <pam_service_name>`

Note

- The SSHD service that runs inside the ESC VM already uses PAM authentication by default.
- If any component sets PAM authentication without specifying the PAM service, ESC defaults to the PAM service 'system-auth'.

PAM Authentication Service Configurations and User Groups

Each ESC service (listed above) has an associated PAM authentication configuration and a user group to provide specific access control. The user group is defined in the `/etc/group` file. The admin user is a member of all the groups.

Table 2: PAM Authentication Service Configurations and User Groups

/etc/group	/etc/pam.d
portal-user:x:1002:admin	portal-auth
rest-user:x:1003:admin	rest-auth
confd-user:x:1004:admin	confd-auth

/etc/group	/etc/pam.d
etsi-user:x:1005:admin	etsi-auth
health-user:x:1006:admin	health-auth

For example, to configure PAM authentication for the health API and restrict access only to the health user group, run the following command:

```
$ sudo escadm monitor set --auth PAM:health-auth
ESC configuration was changed and saved automatically. They will take effect once you restart
 ESC service by running "sudo escadm restart"
```

For adding PAM user to ESC component, see Adding PAM User to an ESC Service/Component, on page 8.

Adding PAM User to an ESC Service/Component

You can add the PAM user to the following groups of ESC service:

- rest-user
- confd-user
- portal-user
- etsi-user
- health-user

Perform the following steps to add the PAM user to an ESC service/component:

Procedure

Step 1 Log in to the ESC VM.

Step 2 Add a PAM user, by using the following command:

```
sudo passwd pamuser
             Changing password for user pamuser.
             New password:
             Retype new password:
             passwd: all authentication tokens updated successfully.
```

Step 3 Add a PAM user to an ESC service/component group, by using the following command:

```
sudo usermod -a -G <ESC Service Group> pamuser
```

Note The PAM user must be added to the admin or readonly group for the Confd service

Configuring Cisco Elastic Services Controller as Identity Management Client

Prerequisites

- Ensure that the Identity Management Client (IDM) server is up and running.
- Ensure that the DNS server in ESC is in working state. So that, ESC interacts with the IDM server using the host name.

The following example shows how ESC (esc-client-500.linuxsysadmins.local) reaches the IDM server (idmns.linuxsysadmins.local).

```
[root@esc-client-500 admin]# ping idmns
PING idmns.linuxsysadmins.local (192.168.222.176) 56(84) bytes of data.
64 bytes from idmns.linuxsysadmins.local (192.168.221.176): icmp_seq=1 ttl=64 time=0.492
ms
64 bytes from idmns.linuxsysadmins.local (192.168.221.176): icmp_seq=2 ttl=64 time=0.457
ms
64 bytes from idmns.linuxsysadmins.local (192.168.221.176): icmp_seq=3 ttl=64 time=0.645
ms
```

You can configure IDM through sssd. ESC provides a PAM configuration file, point to `/etc/pam.d/syste-auth` to `system-auth-esc-sssd` to start configuring services in ESC to work with the IDM server.

```
# cd /etc/pam.d
# ln -sf system-auth-esc-sssd  system-auth
# ls -al /etc/pam.d/system-auth
lrwxrwxrwx. 1 root root 20 Nov 13 00:39 /etc/pam.d/system-auth -> system-auth-esc-sssd
```

The following table lists the commands to enable IDM authentication for each ESC service.

Table 3: Configuring IDM for ESC Services

ESC Service/Component Command	Command
ESCManager	`# sudo escadm escmanager set --auth PAM:system-auth-esc-sssd`
ETSI	`# sudo escadm etsi set --pam_service system-auth-esc-sssd`
ConfD	`# sudo escadm confd set --auth PAM:system-auth-esc-sssd`

Configuring Cisco Elastic Services Controller as the Identity Policy and Audit Client

To Configure the ESC as an Identity Policy and Audit Client (IPA) client, run the following command:

```
ipa-client-install
```

Following is the example to configure ESC as the IPA client:

```
[root@esc-client-500 admin]# ipa-client-install --domain linuxsysadmins.local --server
idmns.linuxsysadmins.local --realm LINUXSYSADMINS.LOCAL
WARNING: ntpd time&date synchronization service will not be configured as
conflicting service (chronyd) is enabled
Use --force-ntpd option to disable it and force configuration of ntpd

Autodiscovery of servers for failover cannot work with this configuration.
If you proceed with the installation, services will be configured to always access the
discovered server for all operations and will not fail over to other servers in case of
failure.
Proceed with fixed values and no DNS discovery? [no]: yes
Client hostname: esc-client-500.linuxsysadmins.local
Realm: LINUXSYSADMINS.LOCAL
DNS Domain: linuxsysadmins.local
IPA Server: idmns.linuxsysadmins.local
BaseDN: dc=linuxsysadmins,dc=local

Continue to configure the system with these values? [no]: yes
Skipping synchronizing time with NTP server.
User authorized to enroll computers: admin
Password for admin@LINUXSYSADMINS.LOCAL:
Successfully retrieved CA cert
    Subject:     CN=Certificate Authority,O=LINUXSYSADMINS.LOCAL
    Issuer:      CN=Certificate Authority,O=LINUXSYSADMINS.LOCAL
    Valid From:  2019-11-12 23:23:32
    Valid Until: 2039-11-12 23:23:32

Enrolled in IPA realm LINUXSYSADMINS.LOCAL
Created /etc/ipa/default.conf
Configured sudoers in /etc/nsswitch.conf
Configured /etc/sssd/sssd.conf
Configured /etc/krb5.conf for IPA realm LINUXSYSADMINS.LOCAL
trying https://idmns.linuxsysadmins.local/ipa/json
[try 1]: Forwarding 'schema' to json server 'https://idmns.linuxsysadmins.local/ipa/json'
trying https://idmns.linuxsysadmins.local/ipa/session/json
[try 1]: Forwarding 'ping' to json server
'https://idmns.linuxsysadmins.local/ipa/session/json'
[try 1]: Forwarding 'ca_is_enabled' to json server
'https://idmns.linuxsysadmins.local/ipa/session/json'
Systemwide CA database updated.
Adding SSH public key from /etc/ssh/ssh_host_ecdsa_521_key.pub
Adding SSH public key from /etc/ssh/ssh_host_ecdsa_384_key.pub
Adding SSH public key from /etc/ssh/ssh_host_rsa_key.pub
Adding SSH public key from /etc/ssh/ssh_host_ed25519_key.pub
Adding SSH public key from /etc/ssh/ssh_host_ecdsa_key.pub
[try 1]: Forwarding 'host_mod' to json server
'https://idmns.linuxsysadmins.local/ipa/session/json'
Could not update DNS SSHFP records.
SSSD enabled
Configured /etc/openldap/ldap.conf
Configured /etc/ssh/ssh_config
Configured /etc/ssh/sshd_config
Configuring linuxsysadmins.local as NIS domain.
Client configuration complete.
The ipa-client-install command was successful
```

Authenticating REST Requests

ESC REST and ETSI REST APIs use http basic access authentication where the ESC client will have to provide a username and password when making REST requests. The user name and password will be encoded with Base64 in transit, but not encrypted or hashed. HTTPS will be used in conjunction with Basic Authentication to provide the encryption.

This section discusses ESC REST and ETSI REST authentications, how to change the default password of the interfaces, and how to send an authorized requests from the ESC client.

REST Authentication

By default, the REST authentication is enabled. To disable REST authentication, you can pass the argument **--disable-rest-auth** to bootvm. Cisco does not recommend you to use this in a production environment.

ESC also supports https communication over port 8443. ESC will generate a self-signed certificate that the client will need to trust to get the https communication going. By default, REST is enabled as HTTP and restricted to localhost.

ESC can be installed with external access to REST on HTTPS or HTTP enabled with additional bootvm.py arguments: **--enable-https-rest** or **--enable-http-rest**.

It is recommended to use only enabled external REST API when required. When enabled, it is recommended to use **bootvm.py --enable-https-rest --user_rest_pass USERNAME:PASSWORD**.

Note Make sure to pass either **--enable-https-rest** or **--enable-http-etsi-rest** or both the arguments to the bootvm.py script to enable http and https interfaces to the REST API. You must pass **--user_rest_pass** while using either **--enable-https-rest** or **--enable-http-etsi-rest** when REST authentication is not disabled. To enable https or http after ESC VM is booted, use the escadm command specified below.

```
sudo escadm escmanager set --url
http://127.0.0.1:8080/ESCManager,https://0.0.0.0:8443/ESCManager
```

You must change the configuration of the peer instance if ESC is in HA Active/Standby mode.

Enabling ETSI REST Authentication

If the ETSI REST http or https interfaces are enabled, then all requests to an ETSI API must contain authentication data. You can use the **--enable-http-etsi-rest** or **--enable-https-etsi** arguments respectively to enable http and https interfaces to the ESC bootvm.py installation script.

You can enable both interfaces simultaneously, but only the https interface should be enabled in a production environment.

Note To enable http or https after the ESC VM has booted, use the escadm command specified below:

```
sudo escadm etsi enable_http_rest
```

OR

```
sudo escadm etsi enable_https_rest
```

Then restart the ETSI service.

Changing the REST Interface Password

The REST interface has only one default username/password (admin/<default_password>). The password can be updated after the bootup using escadm tool from the ESC VM CLI. You can also update the password through the REST API.

Procedure

Step 1 Log in to ESC VM.

Step 2 To replace the existing password with a new one, use one of the below options:

- Using the escadm tool from the ESC VM CLI, you can generate a random password:

```
[root@test-v44-52 admin]# escadm rest set --help
usage: escadm rest set [-h] [-v] --username USERNAME [--password PASSWORD]

optional arguments:
  -h, --help           show this help message and exit
  -v, --v, --verbose   show verbose output
  --username USERNAME
  --password PASSWORD  new password or use randomly generated password if no
                       password provided
```

- Using the REST API:

```
http://[ESCVM_IP]:8080/ESCManager/v0/authentication/setpassword?userName=admin&password=yourPassword
```

or

```
https://[ESCVM_IP]:8443/ESCManager/v0/authentication/setpassword?userName=admin&password=yourPassword
```

Changing the ETSI REST Interface Password

The ETSI REST interface has only one default username/password (admin/<default_password>). The password can be updated after the bootup using escadm tool from the ESC VM CLI.

Procedure

Step 1 Log in to ESC VM.

Step 2 To set the default ETSI REST username and password, use the following command:

```
sudo escadm etsi set --rest_user username:password
```

or

```
[admin@xyz-esc-4-4-0-59-keep ~]$ escadm etsi set --help
usage: escadm etsi set [-h] [-v] [--startup {0,1,true,false,manual,auto}]
[--rest_user REST_USER] [--pam_service PAM_SERVICE]

optional arguments:
-h, --help show this help message and exit
-v, --v, --verbose show verbose output
--startup {0,1,true,false,manual,auto}
set to false|0|manual to disable etsi at startup.
--rest_user REST_USER
Set the user for rest. Format username:password
--pam_service PAM_SERVICE
Specify a PAM service to use for authentication. This
will override the rest user. To revert to the using
the rest user for authentication, supply an empty
string.
```

Sending an Authorized REST Request

To send an authorized request, the ESC client should send the request with the following header:

```
Authorization: Basic YWRtaW46Y2lzY28xMjM=
```

where *YWRtaW46Y2lzY28xMjM=* is the Base64 encoded string of the default username/password.

Most libraries and web clients have an interface for providing the username/password and the application will encode the username/password and add the HTTP Basic Auth header.

Example using the default credentials:

For HTTP:

http://[ESCVM_IP]:8080/ESCManager/v0/tenants/

For HTTPS:

https://[ESCVM_IP]:8443/ESCManager/v0/tenants/

Sending an Authorized ETSI REST Request

To send an authorized request, the ESC client should send the request with the following header:

```
Authorization: Basic YWRtaW46Y2lzY28xMjM=
```

where *YWRtaW46Y2lzY28xMjM=* is the Base64 encoded string of the default username/password.

Most libraries and web clients have an interface for providing the username/password and the application will encode the username/password and add the HTTP Basic Auth header.

Example using the default credentials:

For HTTP:

http://[ESCVM_IP]:8250/vnflcm/v1/vnf_lcm_op_occs

For HTTPS:

http://[ESCVM_IP]:8251/vnflcm/v1/vnf_lcm_op_occs

Configuring Openstack Credentials

If ESC was deployed without passing VIM credentials, you can set VIM credentials through ESC VIM and through VIM User APIs (REST or Netconf API).

Note ESC will accept the northbound configuration request only if the following conditions are met:

- ESC has VIM or a VIM user configured through APIs(REST/Netconf).
- ESC has VIM or a VIM user configured, and ESC is able to reach the VIM.
- ESC has VIM or a VIM user configured, and ESC is able to authenticate the user.

Configuring using Netconf API

- **Passing VIM credential using Netconf :**

```
<esc_system_config xmlns="http://www.cisco.com/esc/esc">
  <vim_connectors>
    <!--represents a vim-->
    <vim_connector>
      <!--unique id for each vim-->
      <id>my-server-30</id>
      <!--vim type [OPENSTACK|VMWARE_VSPHERE|LIBVIRT|AWS|CSP]-->
      <type>OPENSTACK</type>
      <properties>
        <property>
          <name>os_auth_url</name>
          <value>http://<os_ip:port>/v3</value>
        </property>
        <!-- The project name for openstack authentication and authorization -->
        <property>
          <name>os_project_name</name>
          <value>vimProject</value>
        </property>
        <!-- The project domain name is needed for openstack v3 identity api -->
        <property>
          <name>os_project_domain_name</name>
          <value>default</value>
        </property>
      </properties>
      <users>
        <user>
          <id>admin</id>
          <credentials>
            <properties>
              <property>
                <name>os_password</name>
                <value>********</value>
              </property>
```

```
            <!-- The user domain name is needed for openstack v3 identity api -->
            <property>
              <name>os_user_domain_name</name>
              <value>default</value>
            </property>
          </properties>
        </credentials>
      </user>
    </users>
  </vim_connector>
 </vim_connectors>
</esc_system_config>
```

Note

- From ESC 3.0 onwards, multiple VIM connectors are supported but within one ESC, only one type of VIM are supported. For example, all the Vim Connector(s) has to be for OpenStack only. One ESC VIM cannot have two VIM connector, one points to OpenStack, one points to VMware.
- One VIM is chosen as the default VIM which supports all pre 3.0 config requests and datamodels.
- Deployments can be done on the VIM that is not the default VIM. The deployment to a non default VIM has to have all out-of-band resources (except ephemeral volumes). No other configurations like image, flavor, network, and so on can be done on the VIM that is not the default VIM.
- The default VIM connector will be auto provisioned and does not need to be configured in the following scenarios:
  - If VIM credentials have been passed during ESC boot up.
  - If upgrading from 2.3.x to 3.0.
- The change in the datamodel for Openstack create VIM connector would be handled during upgrade by migration. The 'os_tenant_name' and 'os_project_domain_name' properties would be moved to the VIM Connector properties and 'os_ternant_name' will be renamed to 'os_project_name'.
- For the default VIM Connector, once it is properly authenticated, its properties cannot be updated.
- VIM user can be deleted, recreated, or its properties can be updated at anytime.

- **Updating VIM Connector using Netconf:**

```
<esc_system_config xmlns="http://www.cisco.com/esc/esc">
  <vim_connectors>
    <vim_connector nc:operation="replace">
      <id>example_vim</id>
      <type>OPENSTACK</type>
      <properties>
        <property>
          <name>os_auth_url</name>
          <value>{auth_url}</value>
```

```
        </property>
        <property>
          <name>os_project_name</name>
          <value>vimProject</value>
        </property>
        <!-- The project domain name is only needed for openstack v3 identity api -->
        <property>
          <name>os_project_domain_name</name>
          <value>default</value>
        </property>
        <property>
          <name>os_identity_api_version</name>
          <value>3</value>
        </property>
      </properties>
    </vim_connector>
  </vim_connectors>
</esc_system_config>
```

- Updating VIM user using Netconf:

```
<esc_system_config xmlns="http://www.cisco.com/esc/esc">
  <vim_connectors>
    <vim_connector>
      <id>example_vim</id>
        <users>
          <user nc:operation="replace">
            <id>my_user</id>
            <credentials>
             <properties>
              <property>
                <name>os_password</name>
                <value>********</value>
              </property>
              <!-- The user domain name is only needed for openstack v3 identity api
-->
              <property>
                <name>os_user_domain_name</name>
                <value>default</value>
              </property>
             </properties>
           </credentials>
          </user>
        </users>
    </vim_connector>
  </vim_connectors>
</esc_system_config>
```

- Deleting VIM connector using Netconf:

```
<esc_system_config xmlns="http://www.cisco.com/esc/esc">  <vim_connectors>
    <vim_connector nc:operation="delete">
      <id>example_vim</id>
    </vim_connector>
  </vim_connectors>
</esc_system_config>
```

- Deleting VIM Connector using command:

```
$ esc_nc_cli --user <username> --password <password> delete-vim-connector <vim connector
 id>
```

- Deleting VIM user using command:

```
$ esc_nc_cli --user <username> --password <password> delete-vim-user <vim connector id>
 <vim user id>
```

Configuring using REST API

- Adding VIM using REST:

```
POST /ESCManager/v0/vims/
HEADER: content-type, callback

<?xml version="1.0"?>
<vim_connector xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <id>example_vim</id>
  <type>OPENSTACK</type>
  <properties>
    <property>
      <name>os_auth_url</name>
      <value>{auth_url}</value>
    </property>
    <property>
      <name>os_project_name</name>
      <value>vimProject</value>
    </property>
    <!-- The project domain name is only needed for openstack v3 identity api -->
    <property>
      <name>os_project_domain_name</name>
      <value>default</value>
    </property>
    <property>
      <name>os_identity_api_version</name>
      <value>3</value>
    </property>
  </properties>
</vim_connector>
```

- Adding VIM user using REST:

```
POST /ESCManager/v0/vims/{vim_id}/vim_users
HEADER: content-type, callback

<?xml version="1.0"?>
<user xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <id>my_user</id>
  <credentials>
    <properties>
      <property>
        <name>os_password</name>
        <value>********</value>
      </property>
      <!-- The user domain name is only needed for openstack v3 identity api -->
      <property>
        <name>os_user_domain_name</name>
        <value>default</value>
      </property>
    </properties>
  </credentials>
</user>
```

- Update VIM using REST:

```
PUT /ESCManager/v0/vims/{vim_id}
HEADER: content-type, callback

<?xml version="1.0"?>
<vim_connector xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <!--unique id for each vim-->
  <id>example_vim</id>
  <type>OPENSTACK</type>
  <properties>
    <property>
      <name>os_auth_url</name>
      <value>{auth_url}</value>
    </property>
    <property>
      <name>os_project_name</name>
      <value>vimProject</value>
    </property>
    <!-- The project domain name is only needed for openstack v3 identity api -->
    <property>
      <name>os_project_domain_name</name>
      <value>default</value>
    </property>
    <property>
      <name>os_identity_api_version</name>
      <value>3</value>
    </property>
 </properties>
</vim_connector>
```

- Update VIM user using REST:

```
PUT /ESCManager/v0/vims/{vim_id}/vim_users/{vim_user_id}
HEADER: content-type, callback

<?xml version="1.0"?>
<user xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <id>my_user</id>
  <credentials>
    <properties>
      <property>
        <name>os_password</name>
        <value>********</value>
      </property>
      <!-- The user domain name is only needed for openstack v3 identity api -->
      <property>
        <name>os_user_domain_name</name>
        <value>default</value>
      </property>
    </properties>
  </credentials>
</user>
```

- Delete VIM using REST:

```
DELETE /ESCManager/v0/vims/{vim_id}
```

- Delete VIM user using REST:

```
DELETE /ESCManager/v0/vims/{vim_id}/vim_users/{user_id}
```

- Notification example after each VIM or VIM user configuration is complete:

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
```

```
  <eventTime>2016-10-06T16:24:05.856+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_code>200</status_code>
    <status_message>Created vim connector successfully</status_message>
    <vim_connector_id>my-server-30</vim_connector_id>
    <event>
      <type>CREATE_VIM_CONNECTOR</type>
    </event>
  </escEvent>
</notification>
```

Important Notes:

- In ESC 3.0, you can add multiple VIM Connector for Openstack VIM. Each VIM Connector can have only one VIM User.
- VIM username and password can be updated at anytime. VIM endpoint will not be able to update after a resource is created through ESC.
- After VIM is connected and VIM user is authenticated, VIM can no longer be deleted or updated, only VIM user can be deleted or updated.
- The name of a VIM property or VIM user credentials property is not case sensitive, e.g. OS_AUTH_URL and os_auth_url is the same to ESC.

Enabling Barbican Client in ESC

OpenStack Barbican provides secure storage, provisioning, and management of secret, for example, passwords, encryption keys, and X.509 Certificates.

Enable the Barbican client to manage the secret used to encrypt an OpenStack volume before mounting it on a VM. You can access the OpenStack Barbican APIs through a Python 3 environment. The OpenStack Barbican client python-barbicanclient 5.0.1 is integrated in the ESC.

Enable the virtual environment by using the following command:

```
source /opt/esc_custom_python3_venv/bin/activate
```

.

Staging VPC Volume

During VPC deployment, additional disk permission is required to create a volume. For security purpose, the permission is disabled from ESC 5.6 release, making it unavailable for arbitrary use by LCM scripts. To enable volume creation, temporary permission is granted to the script using the sudo program.

Setting up VPC Volume Staging

Use the following steps to set up VPC Volume Staging.

- Volume creation is performed by the `esc_stage_content_via_volume.sh` script. Edit the `/etc/sudoers.d/50-esc-sudoers` file. Add the following line and replace `/path/to/` with the real path to the script from the root directory.

```
mona-user ALL = (root) NOPASSWD: /path/to/esc_stage_content_via_volume.sh
```

- In the VPC deployment payload, update the deployment policy to use `sudo-esc_stage_content_via_volume.sh`.

```
<policy>
<name>1</name>
<conditions>
<condition>
<name>LCS::PRE_DEPLOY</name>
</condition>
</conditions>
<actions>
<action>
<name>GEN_VPC_ISO</name>
<type>script</type>
<properties>
<property>
<name>script_filename</name>
<value>/path/to/sudo-esc_stage_content_via_volume.sh</value>
</property>
...
</action>
</actions>
</policy>
```

- The `sudo-esc_stage_content_via_volume.sh` script simply calls `esc_stage_content_via_volume.sh` using sudo . Edit the file by setting the `STAGING_FILE` variable, replacing /path/to/ with the real path from the root directory.

```
STAGING_FILE=/path/to/esc_stage_content_via_volume.sh
```

Running MONA in a Root Jail

The ESC MONA service normally runs as the mona-user , sharing the same filesystem as other ESC services. For additional security, it is possible to run MONA in its own filesystem called a root jail. This effectivley sandboxes MONA so that the process (and any LCM scripts called by MONA) cannot access the wider filesystem. Interacting with MONA remains the same. Stopping/starting, health queries, log location and service jar updates are not affected by the root-jail, or the network connectivity.

- When the MONA process starts, it checks for the existence of the jail directory `/opt/cisco/esc/mona-jail`. If it exists, ESC will start (chroot) MONA into this environment.

Installing the MONA Root Jail

Perform the following steps:

- Stop the ESC service using the command sudo service `esc_service stop`
- Untar the MONA jail archive using the command `sudo tar-C/opt/cisco/esc -xvzf mona-jail.tar.gz`. The archive is located in the ESC repository `elastic-services-controller/esc-mona/jail/mona-jail.tar.gz`.
- Start the ESC service using the command sudo `service esc_service start`.

Customising the Root Jail Environment

The jail contains a minimal set of binaries for MONA to operate. Some custom CLM scripts may need additional binaries (plus dependent libraries) added to the jail. If for example you needed the strings command available to MONA scripts, you could use the ldd command to find the dependencies.

```
$ ldd /usr/bin/strings
linux-vdso.so.1 => (0x00007ffd9ef96000)
libbfd-2.27-44.base.el7_9.1.so => /lib64/libbfd-2.27-44.base.el7_9.1.so (0x00007fa950415000)
libdl.so.2 => /lib64/libdl.so.2 (0x00007fa950211000)
libc.so.6 => /lib64/libc.so.6 (0x00007fa94fe43000)
/lib64/ld-linux-x86-64.so.2 (0x00007fa95075e000)
$
```

Copying `/usr/bin/strings into /opt/cisco/esc/mona-jail/bin`, and shared libraries into `/opt/cisco/esc/mona-jail/lib64`. The strings program will then be accessible to MONA.

An easier way to achieve the same is by using JailKit from the Centos RPM repository.

```
sudo jk_cp -j /opt/cisco/esc/mona-jail /usr/bin/strings
```

Reconfiguring ESC Virtual Machine

This section covers the following:

- Reconfiguring Rsyslog
- Reconfiguring NTP
- Reconfiguring DNS
- Reconfiguring Hosts
- Reconfiguring Timezone

Reconfiguring Rsyslog

Rsyslog parameters are optional. If there is a need for customization after booting an ESC VM, you can edit the files in ESC VM (/etc/rsyslog.d/).

Procedure

Step 1 **Editing the Rsyslog file**:

- If you haven't specified the log forwarding configuration at the bootup time, you may create a file under /etc/rsyslog.d/ like /etc/rsyslog.d/log-forwarding.conf.
- If you have specified the log forwarding through installation, you may just need to edit the file. The file could be /etc/rsyslog.d/20-cloud-config.conf. In this file, to forward logs to multiple rsyslog servers, edit the following line:

  ```
  *.* @[server_ip]:port
  ```

Note

- Use '@@' before specifying server ip address (if TCP is the protocol used to forward logs to the rsyslog server).
- Use '@' before specifying server ip address (if UDP is the protocol used to forward logs to the rsyslog server).
- server_ip can either be ipv4/ipv6 address of the rsyslog server.
- '[ ]' around the server_ip is required to separate it from ' :port# ', if an ipv6 server address is specified.

For further information on Rsyslog configuration, see the Red Hat documentation.

Step 2 **Configuring the ESC log file:**Configure which ESC log files you want to forward to the rsyslog server:

a) Navigate to /etc/rsyslog.d/ Create or modify a configuration file, such as **log-esc.conf**. Make a copy of sample log-esc.conf .

b) Specify the following block for every file you want to forward to rsyslog server.

```
$InputFileName /var/log/esc/escmanager.log
$InputFileTag esc-manager:
$InputFileStateFile stat-esc-manager
$InputFileSeverity info
$InputRunFileMonitor
```

For example:

```
$InputFileName /var/log/esc/file1.log
$InputFileTag file1:
$InputFileStateFile stat-file1
$InputFileSeverity info
$InputRunFileMonitor

$InputFileName /var/log/esc/file2.log
$InputFileTag file2:
$InputFileStateFile stat-file2
$InputFileSeverity info
$InputRunFileMonitor
```

Step 3 Restart the rsyslog service

```
# service rsyslog restart
```

Step 4 Configure the server side to receive forwarded logs.

a) On a designated server, go to /etc/rsyslog.conf, and uncomment the lines shown below, depending on if you want to listen to logs from clients based on TCP or UDP:

```
#$ModLoad imudp
#$UDPServerRun 514
```

b) Exit the file. Run this command as the last step.

```
sudo service rsyslog restart
```

Now, the server is listening for logs on port 514, using TCP/UDP.

Reconfiguring NTP

Procedure

Step 1 Open the NTP configuration file /etc/ntp.conf in a text editor such as vi, or create a new one if it does not already exist:

```
# vi /etc/ntp.conf
```

Step 2 Add or edit the list of public NTP servers. If you don't specify the NTP server through the installation, the file should contain the following default lines, but feel free to change or expand these according to your needs:

```
server 0.rhel.pool.ntp.org iburst
server 1.rhel.pool.ntp.org iburst
server 2.rhel.pool.ntp.org iburst
server 3.rhel.pool.ntp.org iburst
server <your_ntp_server_ip> iburst
```

The iburst directive at the end of each line speeds up the initial synchronization.

Step 3 Once you have the list of servers complete, in the same file, set the proper permissions, giving the unrestricted access to localhost only. Make sure those lines are there in your configure file.

```
restrict default kod nomodify notrap nopeer noquery
restrict -6 default kod nomodify notrap nopeer noquery
restrict 127.0.0.1
restrict -6 ::1
```

Step 4 Save all changes, exit the editor, and restart the NTP daemon:

```
# service ntpd restart
```

Step 5 Make sure that ntpd is started at boot time:

```
# chkconfig ntpd on
```

Reconfiguring DNS

Procedure

Step 1 The /etc/resolv.conf file contains the configuration for the DNS client (resolver). It typically looks something like this:

```
search domain.com
nameserver 8.8.4.4
```

This results in a `/etc/resolv.conf`:

```
Created by cloud-init on instance boot automatically, do not edit.
;
#Generated by esc-cloud
domain cisco.com
```

```
search cisco.com
nameserver 8.8.4.4
```

Note It is recommended that you ignore the `do not edit` message, if you want to modify the file.

Step 2 You may modify the IP address of the "nameserver" item or add new nameserver records.

```
search domain.com
nameserver <your_first_dns_ip>
nameserver <your_second_dns_ip>
```

Step 3 Restart Network Service.

```
service network restart
```

Reconfiguring Hosts

The /etc/hosts file allows you to add, edit, or remove hosts . This file contains IP addresses and their corresponding hostnames. If your network contains computers whose IP addresses are not listed in DNS, it is recommended that you add them to the /etc/hosts file.

Procedure

Step 1 Add the IP addresses that are not listed in DNS to the /etc/hosts file.

Step 2 Restart your network for the changes to take effect.

```
service network restart
```

Reconfiguring Timezone

For ESC VM, in /etc the file "localtime" is a link to or copy of a file containing information about your time zone. Access your zone information files from /usr/share/zoneinfo. To change the time zone, find your country, your city or a city in the same time zone from zone information files in /usr/share/zoneinfo and link it to the localtime in the /etc file.

```
$ ln \-sf /usr/share/zoneinfo/America/Los_Angeles /etc/localtime
```

Verifying ESC Configurations and Other Post-Install Operations

This section covers various post-install checks and operations using the escadm tool.

Verifying Existing ESC Configurations

You can use `escadm dump` command for displaying current ESC configurations in yaml format. The output will show the various services in ESC.

```
$ sudo escadm dump

resources:
  confd:
    init_aaa_users:
    - name: admin
      passwd:
    option: start-phase0
  esc_service:
    group:
    - confd
    - mona
    - vimmanager
    - pgsql
    - escmanager
    - portal
    - monitor
    - snmp
    type: group
  escmanager: {}
  mona: {}
  monitor: {}
  pgsql: {}
  portal: {}
  snmp:
    run_forever: true
  vimmanager: {}
```

Verifying VIM configurations

You can use `escadm vim show` command to verify the vim settings are correctly populated:

```
$ sudo escadm vim show

[
    {
        "status": "CONNECTION_SUCCESSFUL",
        "status_message": "Successfully connected to VIM",
        "type": "OPENSTACK",
        "id": "default_openstack_vim",
        "properties": {
            "property": [
                {
                    "name": "os_auth_url",
                    "value": "http://172.16.103.143:35357/v3"
                }
            ]
        }
    }
]
```

Troubleshooting ESC Services Startup Issues

Problem: Issues encountered while verifying ESC services status at the installation time using `sudo escadm status`.

Causes: Some services take time to start or have trouble starting.

Solution:

1. Identify the issues using one of the following method:

- Check the log /var/log/esc/escadm.log

```
$ cat /var/log/esc/escadm.log
2017-06-01 20:35:02,925: escadm.py(2565): INFO: promote drbd to primary...
2017-06-01 20:35:02,934: escadm.py(2605): INFO: Waiting for at least one drbd to be
 UptoDate...
2017-06-01 20:35:02,942: escadm.py(2616): INFO: Waiting for peer drbd node to be
demoted...
2017-06-01 20:35:14,008: escadm.py(2423): INFO: mount: /dev/drbd1
/opt/cisco/esc/esc_database
2017-06-01 20:35:14,017: escadm.py(1755): INFO: Starting filesystem service: [OK]
2017-06-01 20:35:15,039: escadm.py(1755): INFO: Starting vimmanager service: [OK]
2017-06-01 20:35:16,116: escadm.py(1755): INFO: Starting monitor service: [OK]
2017-06-01 20:35:17,163: escadm.py(1755): INFO: Starting mona service: [OK]
2017-06-01 20:35:18,440: escadm.py(1755): INFO: Starting snmp service: [OK]
2017-06-01 20:35:21,397: escadm.py(1770): INFO: Starting confd service:[FAILED]
2017-06-01 20:35:28,304: escadm.py(1755): INFO: Starting pgsql service: [OK]
2017-06-01 20:35:29,331: escadm.py(1755): INFO: Starting escmanager service: [OK]
2017-06-01 20:35:30,354: escadm.py(1755): INFO: Starting portal service: [OK]
2017-06-01 20:35:31,523: escadm.py(1755): INFO: Starting esc_service service: [OK]
```

- Add '-v' to escadm status to show the verbose output of the ESC services.

```
$ sudo escadm status --v
0 ESC status=0 ESC HA Active Healthy
pgsql (pgid 61397) is running
vimmanager (pgid 61138) is running
monitor (pgid 61162) is running
mona (pgid 61190) is running
drbd is active
snmp (pgid 61541) is running
filesystem (pgid 0) is running
<<service>> is dead
keepalived (pgid 60838) is running
portal (pgid 61524) is running
confd (pgid 61263) is running
escmanager (pgid 61491) is running
```

2. Confirm the status of the identified services that has issues and manually start these services.

```
$ sudo escadm <<service>> status// If the status is stopped or dead, manually start the
 services using the next command.

$ sudo escadm <<service>> start --v
```

Logging in to the ESC Portal

Note

- The ESC portal is enabled by default. You must ensure that the ESC portal is not disabled during installation. For more information on enabling or disabling the ESC portal, see Installing Cisco Elastic Services Controller Using the QCOW Image.
- When you log in to the ESC portal for the first time you are prompted to change the default password.

To log in to the ESC portal, do the following:

Before you begin

- Register an instance of ESC. For more information on registering the ESC instance see, Installing Cisco Elastic Services Controller Using the QCOW Image
- Ensure that you have the username and password.

Procedure

Step 1 Using your web browser, enter the IP address of ESC and port 443.

Example:

For example, if the IP address of ESC is 192.0.2.254, enter:

https://192.0.2.254: 443 [login via https]

A Security Alert message is displayed.

Step 2 Click **Yes** to accept the security certificate. The Login page is displayed.

Step 3 Enter the username and password and click **Login** .

If you are logging in for the first time, the login page reappears, prompting you to change your password.

Step 4 Enter the old password in the Old Password field, then enter a new password in the New Password and Confirm Password fields.

Step 5 Click **Update Password** or press **Enter**.

Note

- If the UI becomes unresponsive, restart the UI by executing the **sudo escadm portal restart** from the ESC shell prompt.
- ESC Portal only supports one user.
- Currently, a pre-installed self-signed certificate supports HTTPS. The user must confirm the self-signed certificate before proceeding with the ESC Portal.
- In HTTPS communication mode, if the URL protocol type returned by OpenStack is not HTTPS, the access to the VNF Console may be disabled. For security reasons, while running in HTTPS more non-secure communication will be rejected.