



# Troubleshooting Cisco Elastic Services Controller Installation

---

- [Cisco Elastic Services Controller Installation Troubleshooting Overview](#), on page 1
- [OpenStack Credentials from Cisco Elastic Services Controller Installation Do Not Work](#), on page 1
- [Certificate Verification Failed During Cisco Elastic Services Controller Installation](#), on page 3
- [Bootvm.py Script Failing with Error](#), on page 3

## Cisco Elastic Services Controller Installation Troubleshooting Overview

ESC uses a python-based script called `bootvm.py` for its installation in OpenStack and Libvirt (KVM) environments. Refer to the [Cisco Elastic Services Controller Install and Upgrade Guide](#) to understand all the arguments of the `bootvm.py` script. It is important to use a specific `bootvm.py` that was released along with the ESC image.

`bootvm.py` has the dependency on Python and OpenStack client for ESC OpenStack installation. Ensure that the environment where you plan to execute `bootvm.py` has the python and OpenStack client installed.

## OpenStack Credentials from Cisco Elastic Services Controller Installation Do Not Work

### Problem Statement:

You might encounter some errors while executing `bootvm.py` for ESC installation; one such common error is:

- OpenStack Credentials Errors

### Description:

You may get a long python stack trace information after running `bootvm.py` and you must refer couple of lines at the bottom of the error messages. For example:

```
Unauthorized: The request you have made requires authentication. (HTTP 401) (Request-ID: req-e93d90b0-aced-4b88-b4ca-bcc3d88e8bc0)
```

The request you have made requires authentication. (HTTP 401) (Request-ID: req-e93d90b0-aced-4b88-b4ca-bcc3d88e8bc0) -- Booting up ESC VM has failed.

### Solution:

In such scenarios, you must check your OpenStack credentials information either in your `bootvm.py` arguments, or in your global environment (if you have not specified those in the `bootvm.py` arguments).

Following is an example to check OpenStack credential parameters through global environment:

```
$ env | grep OS_
OS_USER_DOMAIN_NAME=default
OS_IMAGE_API_VERSION=2
OS_PROJECT_NAME=admin
OS_IDENTITY_API_VERSION=3
OS_PASSWORD=cisco123
OS_AUTH_TYPE=password
OS_AUTH_URL=http://10.85.103.145:35357/v3
OS_USERNAME=admin
OS_TENANT_NAME=admin
OS_PROJECT_DOMAIN_NAME=default
```

Similar to other OpenStack clients (OpenStack, Nova, Neutron, etc.), `bootvm.py` is used to install ESC on OpenStack. You can pass OpenStack credentials to ESC installer through the following arguments of `bootvm.py`:

```
--os_auth_url
--os_username
--os_password
--os_tenant_name
--os_project_name
--os_user_domain_name
--os_project_domain_name
--os_identity_api_version

--bs_os_auth_url
--bs_os_username
--bs_os_password
--bs_os_tenant_name
--bs_os_project_name
--bs_os_user_domain_name
--bs_os_project_domain_name
--bs_os_identity_api_version
```

The bootstrap arguments starting with `bs_` is only used for ESC installation on OpenStack, and the arguments starting with `os_` is used for ESC to perform the VNF lifecycle management (as default VIM connector in ESC 3.x).

If you do not specify those arguments, ESC uses the same OpenStack credentials from global environment variables of Linux for both ESC installation and VNF lifecycle management. Similar to the OpenStack client, you can create an `openrc` file and source the file to add global environment variables.

For OpenStack V2 API, you need the following items exported to your global environment variables:

```
OS_PASSWORD
OS_AUTH_URL
OS_USERNAME
OS_TENANT_NAME
```

For OpenStack V3 API, you must set `OS_IDENTITY_API_VERSION=3` to use OpenStack V3 API. You need the following items exported to your global environment variables:

```
OS_USER_DOMAIN_NAME
OS_PROJECT_DOMAIN_NAME
OS_PROJECT_NAME
OS_TENANT_NAME
```

```
OS_PASSWORD
OS_AUTH_URL
OS_USERNAME
OS_IDENTITY_API_VERSION
```

# Certificate Verification Failed During Cisco Elastic Services Controller Installation

## Problem Statement:

If your OpenStack is configured with self-signed certificate but you don't provide the ca\_cert file for ESC installation, you may hit the following errors:

```
SSLERROR: SSL exception connecting to https://10.85.103.49:35357/v3: [SSL:
CERTIFICATE_VERIFY_FAILED] certificate verify failed (_ssl.c:590)
SSL exception connecting to https://10.85.103.49:35357/v3: [SSL: CERTIFICATE_VERIFY_FAILED]
certificate verify failed (_ssl.c:590) -- Booting up ESC VM has failed.
```

## Solution:

Bootvm.py does not provide the arguments to be passed in a command line for ESC Installation for a specific CA certification. If your OpenStack endpoint is configured with https (check OS\_AUTH\_URL) and self-signed certificate, you must set the CA certificate file through the global environment by exporting the following two environment variables:

```
export OS_CACERT=<path_to_ca_cert_file>
export REQUESTS_CA_BUNDLE=<path_to_ca_cert_file>
```



---

**Note** The previous approach specifies the CA certificate for ESC installation and not for VNF lifecycle management.

---

If you want to pass the CA certificate for VNF lifecycle management, specify the following arguments in ESC's bootvm.py commands:

```
--cert_file <path_to_ca_cert_file>
```

## Bootvm.py Script Failing with Error

### Problem Statement:

While executing the bootvm.py script to create ESC VM, you might encounter the following error, and the bootvm.py script exits abnormally:

```
bootvm script fails with error "object of type 'NoneType' has no len()"
```

### Description:

The bootvm.py script fails to authenticate the Open Stack credentials and the connectivity details, because the details are not specified or partially specified.

### Solution:

Ensure that you have an openrc file to source with up-to-date values.

For example:

```
export OS_USERNAME=admin
export OS_PASSWORD=<HIDDEN>
export OS_REGION_NAME=RegionOne
export OS_AUTH_URL=http://172.29.91.77:5000/v3
export OS_PROJECT_NAME=admin
export OS_USER_DOMAIN_NAME=Default
export OS_PROJECT_DOMAIN_NAME=Default
export OS_IDENTITY_API_VERSION=3
```

Source the openrc file before running bootvm.py once again.

You can also pass the values directly on the bootvm.py using `os_<variable_name>`, that is

```
--os_auth_url=http://172.29.91.77:5000/v3.
```

If you still encounter the same error, run the bootvm.py script with the debug option, and redirect the output to a file. To perform this action, add the following command-line argument.

```
--loglevel DEBUG --log /tmp/esc-install.log
```

For any technical support attach the resulting esc-install.log file.