



Installing Cisco Elastic Services Controller on Amazon Web Services

This chapter describes how to install Cisco Elastic Services Controller on AWS and includes the following sections:

- [Prerequisites, on page 1](#)
- [Installing the Elastic Services Controller Instance in AWS, on page 2](#)

Prerequisites

Following are the prerequisites that you must complete before you start installing the ESC instance in AWS.



Note If the ESC AMI images are shared with your AWS account, you can ignore these prerequisites and directly use the AMI image for ESC installation.

Procedure

- Step 1** Configure AWS CLI . You can use pip to install AWS CLI. For more details, refer to the [AWS documentation](#).
- Step 2** Configure the credentials for AWS CLI based on your account information.
- Step 3** Create an Amazon S3 Bucket. Use this for bucket for uploading ESC image.

Note You must have a role named vmimport that allows importing VM and you must attach an IAM policy to the role. For more information, refer to the [documentation](#) on the creation of S3 bucket in AWS.

- Step 4** Extract the vmdk file from ESC ova file.

```
$ tar xvf ESC-<latest image file>.ova ESC-<latest image file>-disk1.vmdk
```

Installing the Elastic Services Controller Instance in AWS

Once you have completed the tasks specified in the prerequisites section, you can use the procedure below to deploy and launch ESC instance in AWS.

Procedure

Step 1 Upload and register ESC image.

- a) Upload the vmdk image to the S3 bucket.

```
aws s3 cp <esc-vmdk-file> s3://<S3 bucket name>/
```

- b) Register the image.

```
aws ec2 import-image --description "<esc-vmdk-file>" --disk-containers
file://containers.json
```

Step 2 Create user data.

- a) Create a user for ESC VM. Without a user, you would not be able to access the VM. It is recommended to configure 'admin' user with sudo access and ssh key.
b) Create the esc-config.yaml in user-data using write_files command.

Each instance can have up to 15 interfaces, depending on the type of instances.

Note If you want to use two interfaces, ensure that you create the two network interfaces before hand. These interfaces on different subnets must belong to the same availability zone. Add the interface details in the 'Configure Instance Details' tab when launching the instance from AWS console.

- c) Enable esc_service and start it.

Following is an example of a complete user data:

```
#cloud-config
# It is recommended to disable password authentication for ssh when ESC runs in public cloud
such as AWS.
ssh_pwauth: False
users:
  - name: admin
    # Put admin in 'esc-user' group, otherwise some scripts of ESC might fail when running
    as admin.
    groups: esc-user
    gecos: User created by cloud-init
    # This is an example of the hashed password for 'admin'.
    passwd:
$6$rounds=656000$pswsUsR7Iz9NIFA4$7E1sEGV8rhDieNDhc8241YwL3cQ8Rsgp9Nds.OZBe9rG/DE56YwK0kDZoB.DsjATrj9pcBnAe.rSQpWl12r0N/

    # The public key for admin user. Replace it with your public key to login.
    ssh-authorized-keys:
      - ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQcQGLe4EVVI/rQy4e4jZUEnc5PvYItc39x5fz9rRggZzpwYzKXSj+UnWQMgvkIai+
M/5vTPiEYISVz9PmIKayZaLr/2GILPmPNEgyzvJD5v77vV3Ag7eHfLXLYbu7ausYqFKEFbNgSTGC1Pwhoz2geY4zND9hS3eM1NzNSIpb03ftzanQoqtWSx2aRc8IM/
piy6NcBzJ3JeH4rOk9bQ+QxRAYm3b0lq/qRfuoxmrsgd68xAlXeDwyGumETHXN9MDEcQMIW054fiPQgkqKbZwztH2EEEnE9/B6rZCRBUUvdoQhQt2L/
hbCZN1k+oqQ53rlG/BjT09CGfYbgoHq2v
    # false allows you to sudo with the password.
```

```

lock-passwd: false
homedir: /home/admin
# sudo settings
sudo: ALL=(ALL) ALL
write_files:
- path: /etc/cloud/cloud.cfg.d/sys-cfg.yaml
  content: |
    network:
      version: 1
      config:
        # You must define the name server when you use the static IP address.
        - type: nameserver
          address:
            - 172.31.0.2
        # Define physical network interface
        - type: physical
          name: eth0
          subnets:
            # Define the static IP address
            - type: static
              address: 172.31.5.66
              netmask: 255.255.240.0
            # Define the routes
            routes:
              - gateway: 172.31.0.1
                # 0.0.0.0 means the default gateway
                network: 0.0.0.0
                netmask: 0.0.0.0
# ESC service config file
- path: /opt/cisco/esc/esc-config/esc-cfg.yaml
  content: |
    confd:
      # AAA users for ConfD
      init_aaa_users:
      # Public key for ConfD user 'admin'
      - key:
c3NoLXJzYSBBQUFBQjNOemFDMXl1jMkVBQUBREFRQUJBQUFCQVFDDeFkwMzByaEMzSXlWekF2bStISVlmMmpkdm
RUZndTTEpCRjVPTjZoUEgvK2FBTKkzb0NCSmJndjhPdJrtVXUvYmlCYmsys240QW52Ni9ROE1YWGducnZST241MlJuODN2ejRCWTAw
Tlh2SzZrT2YrUnZkSDFtNjhscVlrWU9uZVErNETOak5tQXRwV0huT0xCZE1mZ2pzTmFlS1F1QVJUMEtDS2VBS3k4aUVqSUZpZDhWZ3
NiSlA0aDnPtZdjctkza0E1ZGFQb0xiNWRKRvp3ZWl5WS9ENGp6ZnJUeDVKWFFFuMy80SDdaQVZPaWcyNzBGUn1GVkZHNFl1VXNYcDk1d3
QveHdpc0RUREVCYTYdyjkxQzdXamtaNy9rYkRlRW9VSU9OZEExqdEdvbU84c2JRuuJoZHBVTTZlNXJkeUl2VzQ3YTZYOFa5N2lBR3JrQ09
qMwVHNkYgeG1hb3hpbnlAWE1BT1hJT1ktTS1SRVhXCg==
      # Note: 'admin' is the only user supported and you cannot change the name here.
      name: admin
      # Hashed password for admin user.
      passwd:
$6$rounds=656000$4hZhtniblo4/b0m$FD3./1H3jcPlWAENvIFlu70i5wkrH9DIasDwtKl.p70UFZ1falzD907utLlNcKXwuchNhxIOrvYagkBFq6AWh.

      # No specific settings for esc service. Leave it empty.
      esc_service: {}
    runcmd:
      - [ cloud-init-per, once, escservicestart, sh, -c, "chkconfig esc_service on && service
esc_service start" ]

```

Following is an example to define two interfaces in user data:

```

- path: /etc/cloud/cloud.cfg.d/sys-cfg.yaml
  content: |
    network:
      version: 1
      config:
        - type: physical
          name: eth0

```

```

subnets:

- type: static
  address: 172.31.5.66
  netmask: 255.255.240.0
  # Define the routes
  routes:
  - gateway: 172.31.0.1
    # 0.0.0.0 means the default gateway
    network: 0.0.0.0
    netmask: 0.0.0.0

- type: physical
  name: eth1

subnets:

- type: static
  address: 172.31.51.220
  netmask: 255.255.240.0

```

Step 3 Launch ESC VM in AWS

Launch ESC VM using one of the following method:

- **From Portal:**

- Go to EC2 Management Console, IMAGES/AMIs. Select the image you imported and click **Launch**.
- Choose an instance type. Choose t2.xlarge as the instance type.
- Configure the Instance Details. Add details such as User Data, Storage, Tag name, and so on. While using two interfaces, create these network interfaces and them here.
- Configure a security group. Enable ssh only.
- Click **Launch**.

- **From Command Line:** Choose the image, subnet, security group and use the following command to instantiate ESC VM.

```

aws ec2 run-instances --subnet-id <subnet id> --image-id <image id> --security-group-ids
<security group id> --count 1
--instance-type <instance> --key-name <key name> --user-data <user data file location>
--associate-public-ip-address

```

Note ESC does not support HA Active/Standby installation on AWS.

What to do next

After you launch the ESC VM, check the status of the ESC service using the `$ sudo escadm status` command.