How To: EC2 - Launch an EC2 Instance

Created by Cameron Magee, last modified by Paul Kishter on May 17, 2024

⚠ **Regarding backups:** While eVo provides a variety of mechanisms for you to back up your EC2 instances, we do not automatically back them up. We recommend you set up snapshots, AMIs, or AWS Backup to preserve the configuration of important systems. For information on AWS Backup, see How To: Backup

⚠ **Regarding subnet selection:** Be aware that a project in eVo has "public" subnets (has "public" in the name) and "private" subnets (may have "private" in the name, or names just like "us-east-subnet-1"). The intent was for "public" to be for services shared across eVo, but those distinctions have morphed over time. However, at this time, instances in the private subnets have a more direct network path and will experience less network latency. We recommend you put instances in the private subnets whenever possible.

⚠ **Regarding login options:** There are at least three ways you can log into an EC2 instance

1. Username/password login, which is configured and controlled by eVo automation
2. SSH keys (the key you specified before spinning up the instance)
3. Session Manager - see How To: EC2 - Connect to EC2 instances with Session Manager

Overview

There are many AMIs available to be launched in EC2. The AMI details are below. Only approved AMIs that have been secured and hardened by eVo are permitted for use in the environment. AMIs from the Amazon Marketplace are restricted.

Once an image is launched and provisioning is complete, you should be able to use your Active Directory username and password to login.

Note: RHEL7/CentOS7 will be removed 6/30/24 as they are EOL on that date.

Ubuntu 18 was removed 10/31/23 as it was EOL.

AMI Name	OS	Approximate Time Until AD Login	Date Updated
eVo_AMI_AmazonLinux2	Amazon Linux 2	3 minutes	Updated Daily
eVo_AMI_CentOS7	CentOS 7	5 minutes	Updated Daily
eVo_AMI_EKS_Amazon2	EKS Amazon Linux 2	3 minutes	Updated Daily
eVo_AMI_Ubuntu20	Ubuntu 20.04 LTS	5 minutes	Updated Daily

eVo_AMI_DeepLearning_Ubuntu	DeepLearning Ubuntu	7 minutes	Updated Daily
eVo_AMI_DeepLearning_Amazon	DeepLearning Amazon	4 minutes	Updated Daily
eVo_AMI_Windows2016	Windows Server 2016	10 minutes	Updated Daily
eVo_AMI_Windows2019	Windows Server 2019	12 minutes	Updated Daily
eVo_AMI_RHEL7	Red Hat Enterprise Linux 7	5 minutes	Updated Daily
eVo_AMI_RHEL8	Red Hat Enterprise Linux 8	5 minutes	Updated Daily

RHEL8 INFO

Patches for the operating system are enabled for these instance. There are are few enabled repositories enabled already. You'll see errors like the below

Updating Subscription Management repositories.
Unable to read consumer identity

This system is not registered with an entitlement server. You can use subscription-manager to register.

NOTE: this should not prevent patching of the OS

To enable the Red Hat CodeReady Linux Builder for RHEL 8 please run

```
sudo dnf config-manager --set-enabled codeready-builder-for-rhel-8-rhui-rpms
```

To see what is enabled repository wise you can run this

```
sudo yum repolist
```

```
repo id                                  repo name
ansible-2-for-rhel-8-rhui-rpms           Red Hat Ansible Engine 2 for RHEL 8 (RPMs) from RHUI
codeready-builder-for-rhel-8-rhui-rpms   Red Hat CodeReady Linux Builder for RHEL 8 x86_64 (RPMs) from
RHUI
epel                                     Extra Packages for Enterprise Linux 8 - x86_64
rhel-8-appstream-rhui-rpms               Red Hat Enterprise Linux 8 for x86_64 - AppStream from RHUI (RPMs)
rhel-8-baseos-rhui-rpms                  Red Hat Enterprise Linux 8 for x86_64 - BaseOS from RHUI (RPMs)
rhui-client-config-server-8              RHUI Client Configuration Server 8
saltstack                                SaltStack latest Release Channel for RHEL/CentOS 8
```

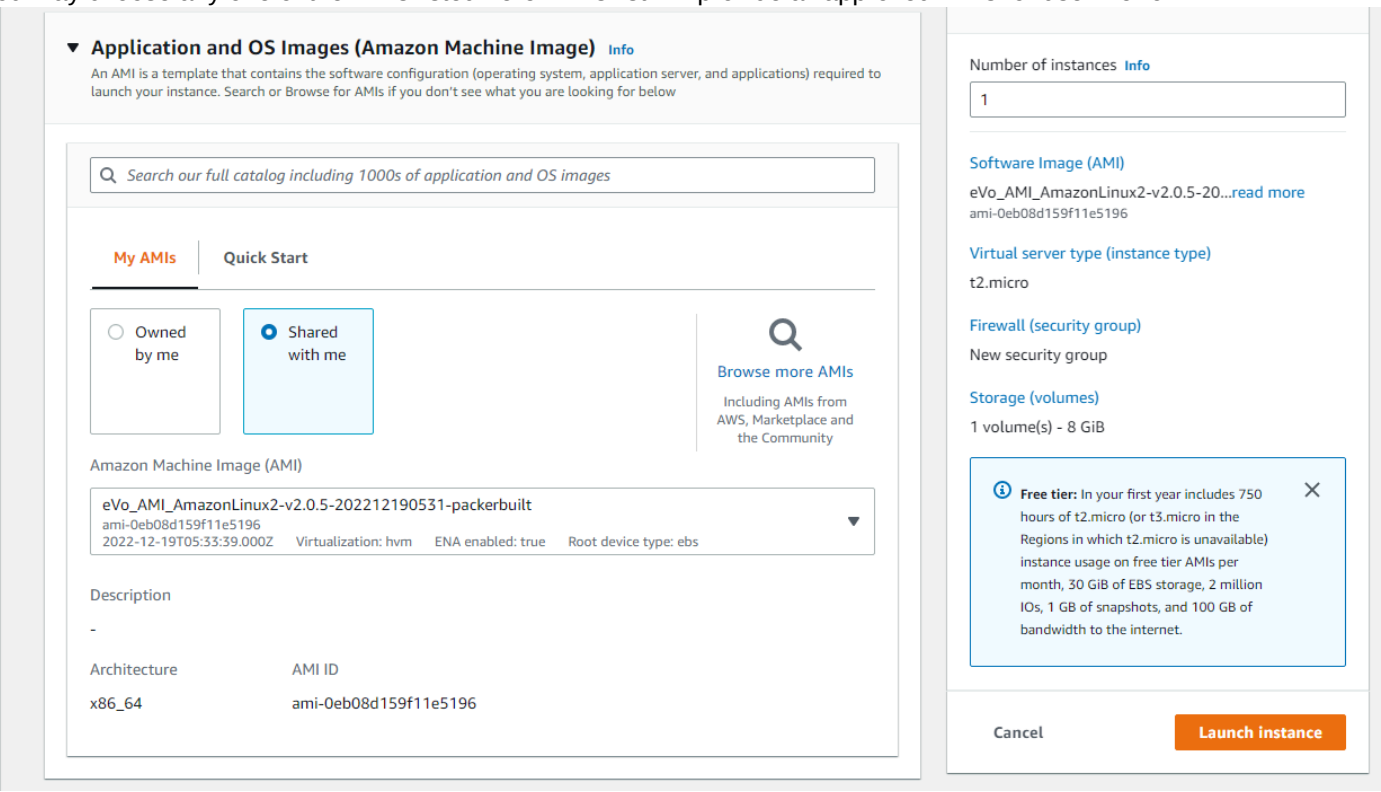
Further info:

https://aws.amazon.com/partners/redhat/faqs/

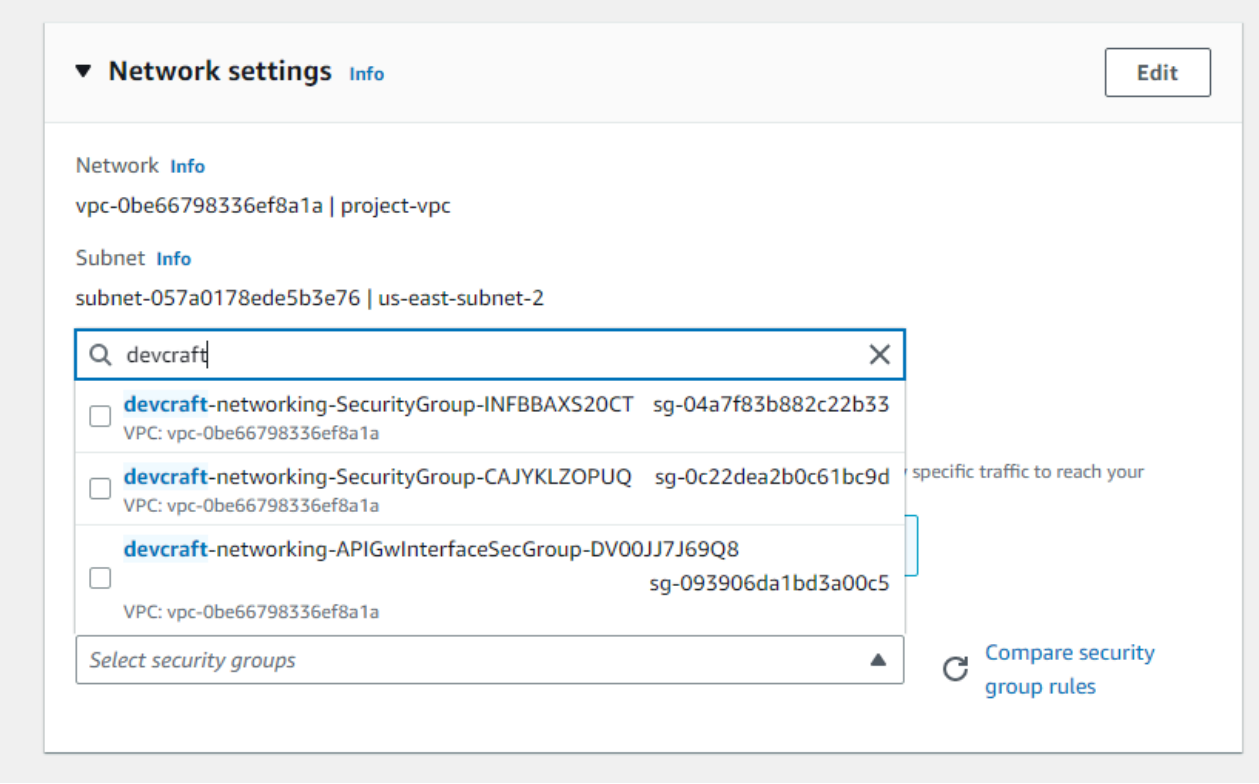
https://www.redhat.com/en/blog/introduction-appstreams-and-modules-red-hat-enterprise-linux

Step-by-step Guide for spinning up EC2 instances

1. **Log into Kion** to access your project's AWS Console
2. Once in the AWS Console, search or click on the **EC2** service.
3. Make sure you are in the **US East (N. Virginia)** region. If you are in the Oregon or Ohio regions (see menu at the top right), change to **N. Virginia**.
4. On the EC2 Dashboard, select **Launch Instance.**
5. On the screen, '**Name and tags'** give your instance a descriptive name for identification in the EC2 dashboard and add additional tags if you like.
6. On the screen, **'Application and OS Images (Amazon Machine Image)'**, select **My AMIs.**
7. Select '**Shared with me'**
8. Under **'Amazon Machine Image (AMI)'**, click on the down arrow to the right to see a list of all EVO AMI's that you can use.
9. You may choose any one of the AMIs listed here. This list will provide all approved AMIs for use in eVo.



10. On the screen, **Instance Type**, select the appropriate instance type. The list of allowed instance types is here: AWS Policies and Allowed Services
11. When prompted to select a keypair, you can select any option. If you choose to "Proceed without a key pair", your AD credentials must be used to login to the instance. Otherwise, you may follow this guide to create a key pair.
12. On the screen, **Network Settings**, You'll see **'Firewall (Security Groups)'** click: **Select an existing security group.**
13. In the search box **'Select security groups'**, type in **'Devcraft'** and select the group: **devcraft-networking-SecurityGroup-XXXXXXXXXXX** (the X will be random letters). Select either of the two you see listed.

a. 

b. **NOTE - avoid using the "default" Security Group, as this is tied to a VPC that no longer exists.**

14. On the screen, **Add Storage**, you can make changes to the storage size and type that you want to attach to your EC2 instance.After the time period listed at the top of this page (about 5 minutes), you should now be able SSH into your instance using the Private IP, and providing your eVo username and password when prompted.
15. On the screen, **Advanced Details,** select **'IAM instance profile'** and then select/search for the "**evoforge-user**" IAM role to attach to your EC2 instance. This will allow programmatic access to preassigned key pairs and access to the AWS Command Line Interface (CLI). You also have the ability to create your own roles - see How To: IAM Roles
16. You may then select and modify all the other options under Advanced Details that you want to customize. Otherwise, leave everything else on the default setting.
17. Review all the settings and ensure all options are correct and then click on **"Launch Instance"** at the bottom of the page.

Related articles

- How To: EC2 - Launch an EC2 Instance
- How To: Backup
- Amazon WorkSpaces
- 01/27/2020 - RDS Database reboots

aws ec2 kb-how-to-article