

aws

Services

Resource Groups

PratikMumbaiSupport

Create file system

Step 1: Configure network access

Step 2: Configure file system settings

Step 3: Configure client access

Step 4: Review and create

Configure network access

An Amazon EFS file system is accessed by EC2 instances running inside one of your VPCs. Instances connect to a file system by using a network interface called a mount target. Each mount target has an IP address, which we assign automatically or you can specify.

VPCvpc-007748679325e3...

Create mount targets

Instances connect to a file system by using mount targets you create. We recommend creating a mount target in each of your VPC's Availability Zones so that EC2 instances across your VPC can access the file system.

Availability Zone	Subnet	IP address	Security groups
<input checked="" type="checkbox"/> ap-south-1a	subnet-0b0c6283ba6504aa7 (default)	Automatic	sg-0a44467f6a5bc43d4 - default
<input checked="" type="checkbox"/> ap-south-1b	subnet-0cf138c8cd8503df (default)	Automatic	sg-0a44467f6a5bc43d4 - default
<input checked="" type="checkbox"/> ap-south-1c	subnet-0392a3630ff128f16 (default)	Automatic	sg-0a44467f6a5bc43d4 - default

CancelNext Step

FeedbackEnglish (US)

© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy PolicyTerms of Use

aws

Services

Resource Groups

PratikMumbaiSupport

Create file system

Step 1: Configure network access

Step 2: Configure file system settings

Step 3: Configure client access

Step 4: Review and create

Configure client access

File system policy

A file system policy is an IAM resource policy that applies to all NFS clients connecting to this file system. You can use the check boxes below to create a simple file system policy. You can also use the JSON editor to create a more advanced policy, such as one that grants permissions to different IAM roles or a different AWS account. [Learn more](#)

Policy settings

{ } JSON

Select a combination of policy statements and set your policy.

☐ Disable root access by default*

☐ Enforce read-only access by default*

☐ Enforce in-transit encryption for all clients

* Identity-based policies can override these default permissions.

Set policy

Access points

You can create access points to provide applications access to your file system. Optionally, you can configure the POSIX identity and root directory for all connections to this access point. If you specify the parent for root directory, EFS will automatically create it with the permissions that you specify once a client connects to the access point. Once you create

```
C:\Windows\system32>helm install stable/jenkins --version 2.1.2 --namespace jenkins
--set master.usePodSecurityContext=True --set master.adminPassword="redhat" --set
persistence.enabled=true --set persistence.storageClass=default --set
persistence.accessMode=ReadWriteOnce --set persistence.size=5Gi
NAME: cautious-giraffe
LAST DEPLOYED: Tue Jul 7 05:21:43 2020
NAMESPACE: jenkins
STATUS: DEPLOYED
```

RESOURCES:

==> v1/ConfigMap

NAME	DATA	AGE
cautious-giraffe-jenkins	2	1s
cautious-giraffe-jenkins-jenkins-jcasc-config	1	
1s cautious-giraffe-jenkins-tests	1	1s

==> v1/Deployment

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
cautious-giraffe-jenkins	0/1	1	0	1s

==> v1/PersistentVolumeClaim

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS	AGE
cautious-giraffe-jenkins	Pending	default	1s			

==> v1/Pod(related)

NAME	READY	STATUS	RESTARTS	AGE
cautious-giraffe-jenkins-846739846-87qdf	0/2	Pending	0	1s

==> v1/Role

NAME	CREATED AT
cautious-giraffe-jenkins-casc-reload	2020-07-06T23:57:21Z
cautious-giraffe-jenkins-schedule-agents	2020-07-06T23:57:21Z

==> v1/RoleBinding

NAME	ROLE	AGE
cautious-giraffe-jenkins-schedule-agents	Role/cautious-giraffe-jenkins-schedule-agents	1s
cautious-giraffe-jenkins-watch-configmaps	Role/cautious-giraffe-jenkins-casc-reload	1s

==> v1/Secret

NAME	TYPE	DATA	AGE
cautious-giraffe-jenkins	Opaque	2	1s

==> v1/Service

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
cautious-giraffe-jenkins	ClusterIP	10.109.108.200	<none>	8080/TCP	1s
cautious-giraffe-jenkins-agent	ClusterIP	10.110.256.189	<none>	50000/TCP	1s

==> v1/ServiceAccount

NAME	SECRETS	AGE
cautious-giraffe-jenkins	1	1s

NOTES:

1. Get your 'admin' user password by running:

```
printf $(kubectl get secret --namespace jenkins cautious-giraffe-jenkins -o  
jsonpath="{.data.jenkins-admin-password}" | base64 --decode);echo
```

2. Get the Jenkins URL to visit by running these commands in the same shell:

```
export POD_NAME=$(kubectl get pods --namespace jenkins -l  
"app.kubernetes.io/component=jenkins-master" -l  
"app.kubernetes.io/instance=cautious-giraffe" -o jsonpath="{.items[0].metadata.name}")  
echo http://127.0.0.1:8080  
kubectl --namespace jenkins port-forward $POD_NAME 8080:8080
```

3. Login with the password from step 1 and the username: admin

4. Use Jenkins Configuration as Code by specifying configScripts in your values.yaml
file, see documentation: <http://configuration-as-code> and examples:
<https://github.com/jenkinsci/configuration-as-code-plugin/tree/master/demos>

For more information on running Jenkins on Kubernetes, visit:

<https://cloud.google.com/solutions/jenkins-on-container-engine>

For more information about Jenkins Configuration as Code, visit:

<https://jenkins.io/projects/jcasc/>

```
C:\Windows\system32>kubectl get pods -n jenkins
```


NAME	READY	STATUS	RESTARTS	AGE
cautious-giraffe-jenkins-856749646-97qdf	0/2	Pending	0	19m



```
C:\Windows\system32>kubectl --namespace jenkins port-forward  
cautious-giraffe-jenkins-856749646-97qdf 8080:8080
```




Welcome to Jenkins!


Sign in


**Jenkins**


 admin  log out


Jenkins >


 New Item


 People


 Build History

 Manage Jenkins

 My Views

 Lockable Resources

 New View

 [add description](#)

Welcome to Jenkins!

[Create a job](#) to start building your software project.

Build Queue

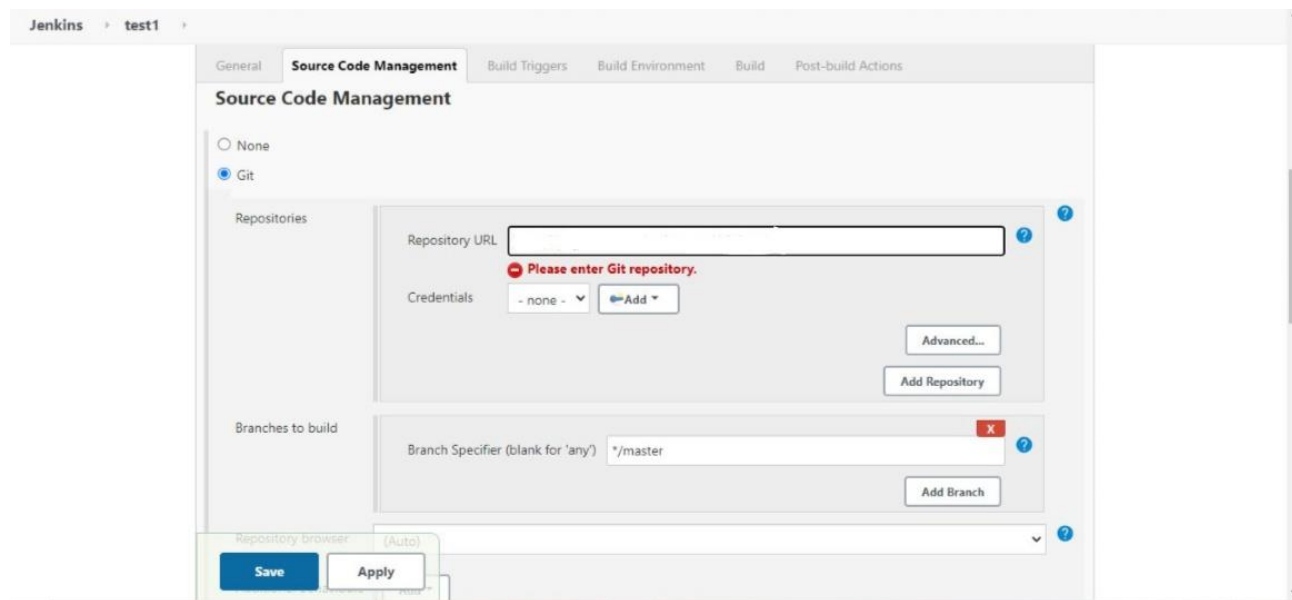
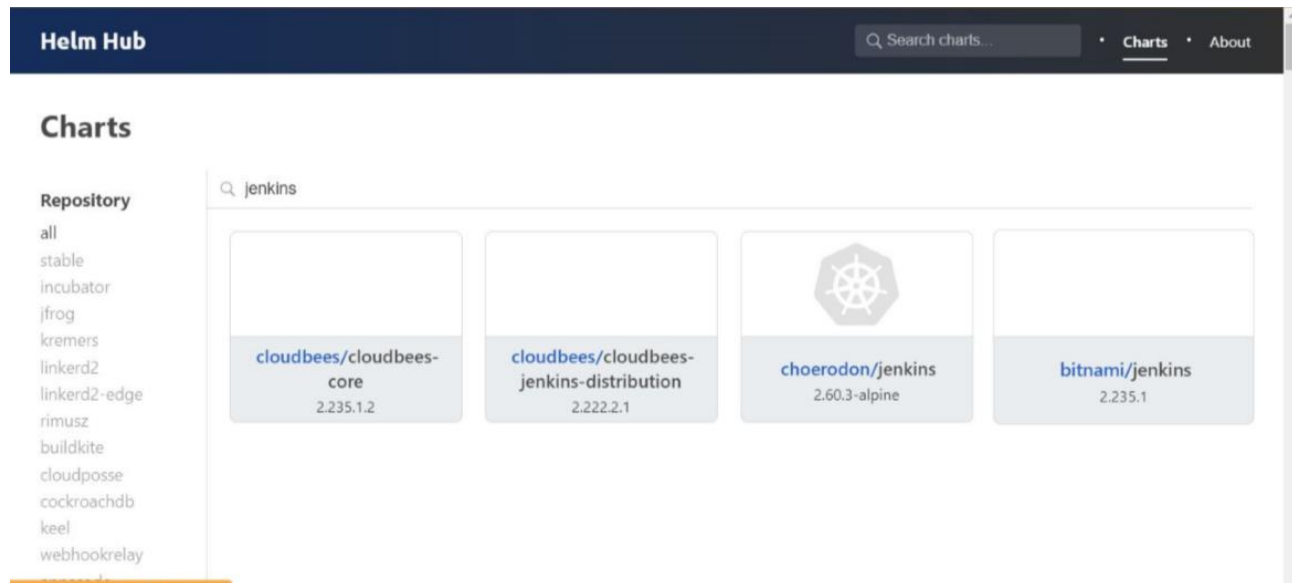
No builds in the queue.

Build Executor Status

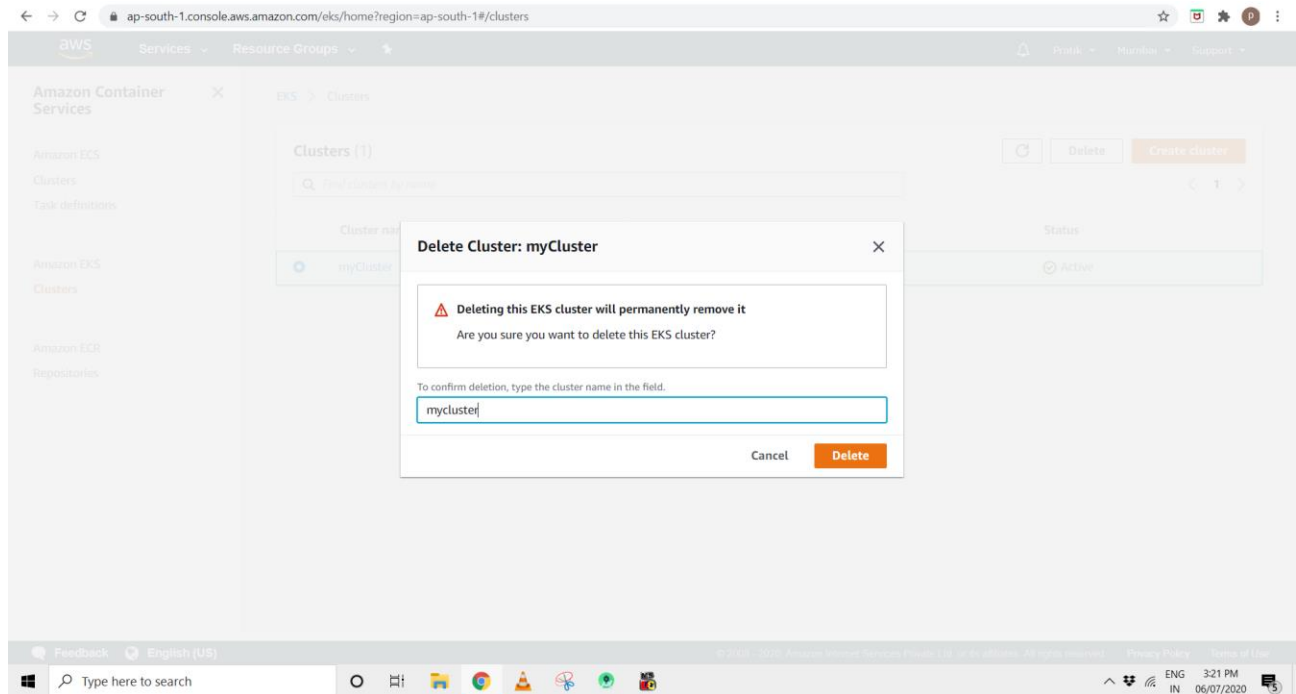
Page generated:
Jul 7, 2020 12:09:01 AM UTC

REST API

Jenkins 2.235.1



```
C:\Windows\system32>kubectl get pods -n jenkins
NAME READY STATUS RESTARTS AGE
default-bw2kw 1/1 Running 0 65s
yucky-panther-jenkins-66bc66cc45-7t2pk 2/2 Running 0 11m
```



Thanks for reading!!!!!!!!!!!!!!