Pre-requisites :

Aws account

Jenkins machine up and running

Aws cli installed on instance

Iam role of s3 full access

S3 bucket

Disaster Recovery: Backups ensure that you have a copy of your Jenkins configuration and data in case of system failures, hardware crashes, or data corruption. If your Jenkins server goes down, you can restore from a backup to get your system up and running again quickly.

Configuration Management: Jenkins backups capture the configuration settings, job configurations, plugin configurations, and other customizations you have made to your Jenkins environment. By taking regular backups, you can preserve your Jenkins setup and easily restore it to the desired state.

Job History and Build Data: Jenkins stores valuable information about past builds, including logs, test results, and build artifacts. Backing up this data ensures that you retain a historical record of your software builds,which can be useful for troubleshooting, auditing, and performance analysis.

Plugin and Tool Preservation: Jenkins relies on various plugins and tools to extend its functionality.Backing up your Jenkins installation ensures that you have copies of these plugins and tools, making it easier to restore the entire ecosystem in case of a catastrophic event.

Smooth Migration or Upgrades: Backups are particularly crucial when migrating or upgrading your Jenkins instance. They allow you to transfer your existing configuration and data to the new environment without losing any critical information or customizations.

Compliance and Security: Some organizations may have compliance requirements that mandate regular backups to ensure data integrity and security. Backing up Jenkins helps meet these requirements and provides an additional layer of protection against data loss or unauthorized access.

0ff217bf9c894f968fad32ea17679aeb

pipeline {

agent any

stages {

stage ("cleaningup workspace") {

steps {

cleanWs()

}

}

stage ("checkout scm code") {

steps {

checkout scmGit(branches: [[name: '\*/main']], extensions: [], userRemoteConfigs: [[url: 'https://github.com/Aj7Ay/complete-prodcution-e2e-pipeline.git']])

}

}

}

}

#!/bin/bash

sudo apt update -y

sudo apt upgrade -y

wget -O - https://packages.adoptium.net/artifactory/api/gpg/key/public | tee /etc/apt/keyrings/adoptium.asc

echo "deb [signed-by=/etc/apt/keyrings/adoptium.asc] https://packages.adoptium.net/artifactory/deb $(awk -F= '/^VERSION\_CODENAME/{print$2}' /etc/os-release) main" | tee /etc/apt/sources.list.d/adoptium.list

sudo apt update -y

sudo apt install temurin-17-jdk -y

/usr/bin/java --version

curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee \

/usr/share/keyrings/jenkins-keyring.asc > /dev/null

echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \

https://pkg.jenkins.io/debian-stable binary/ | sudo tee \

/etc/apt/sources.list.d/jenkins.list > /dev/null

sudo apt-get update -y

sudo apt-get install jenkins –y

sudo systemctl start jenkins

sudo systemctl status jenkins

s3fullaccess777

sudo apt install python3-pip

sudo pip3 install awscli

PROCESS :   
Log into main Jenkins machine

And attach a s3full access role to it

And connect instance using putty or mobaxtreme

Install awscli

sudo apt install python3-pip

sudo pip3 install awscli

And stop Jenkins machine using below command

sudo su

systemctl stop Jenkins

make Jenkins folder as zip by below command

tar –zcvf <file-name> /var/lib/jenkins/

copy file to s3 bucket

aws s3 cp <file-name> s3://<bucket-name>/<file-name>

and delete instance

START A NEW INSTNACE

Launch Jenkins in it

Add iam role to it

Install aws cli as previous

Stop Jenkins machine

Sudo su

And remove Jenkins folder by below command

rm –rf /var/lib/Jenkins

copy file from s3 bucket to instance

aws s3 cp s3://bucketname/filename filename

unzip the file

tar –zxvf <file-name> /

start jenkins

aws

2 sudo apt install python3-pip

3 pip3 install awscli

4 aws

5 sudo systemctl status jenkins

6 systemctl stop jenkins

7 sudo systemctl status jenkins

8 cd /var/lib/jenkins/

9 ls

10 cd jobs

11 ls

12 cd ..

13 cd /home/ubuntu/

14 ls

15 rm -rf /var/lib/jenkins/

16 aws s3 cp s3://s3fullaccess777/jenkins-backup.tar.gz jenkins-backup.tar.gz (s3 to instance )

Aws s3 cp Jenkins-backup.tar.gz s3://bucketname/filename (instance to s3)

17 ls

18 cd /var/lib/

19 ls

20 cd /home/ubuntu/

21 ls

22 tar -zxvf jenkins-backup.tar.gz -C /

23 systemctl start jenkins

24 systemctl status jenkins

25 history