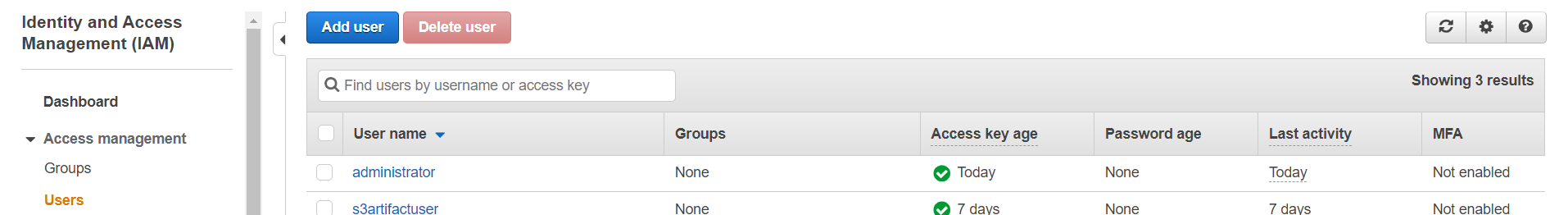
**Automated Daily Backups of Jenkins Master to Amazon S3 Bucket**

Pre-requisites:

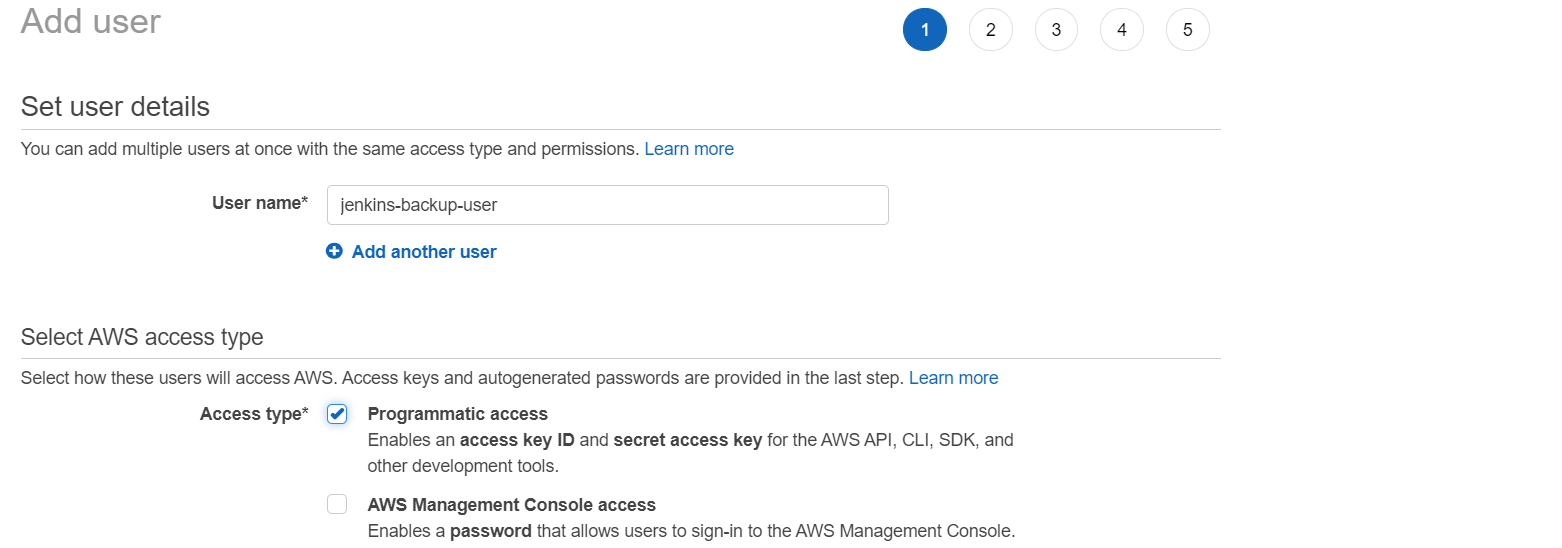
* Amazon-Linux-Instance
* Install Jenkins
* IAM User with AdministratorAccess Policy

IAM User with AdministratorAccess Policy

Goto IAM service 🡪 Users

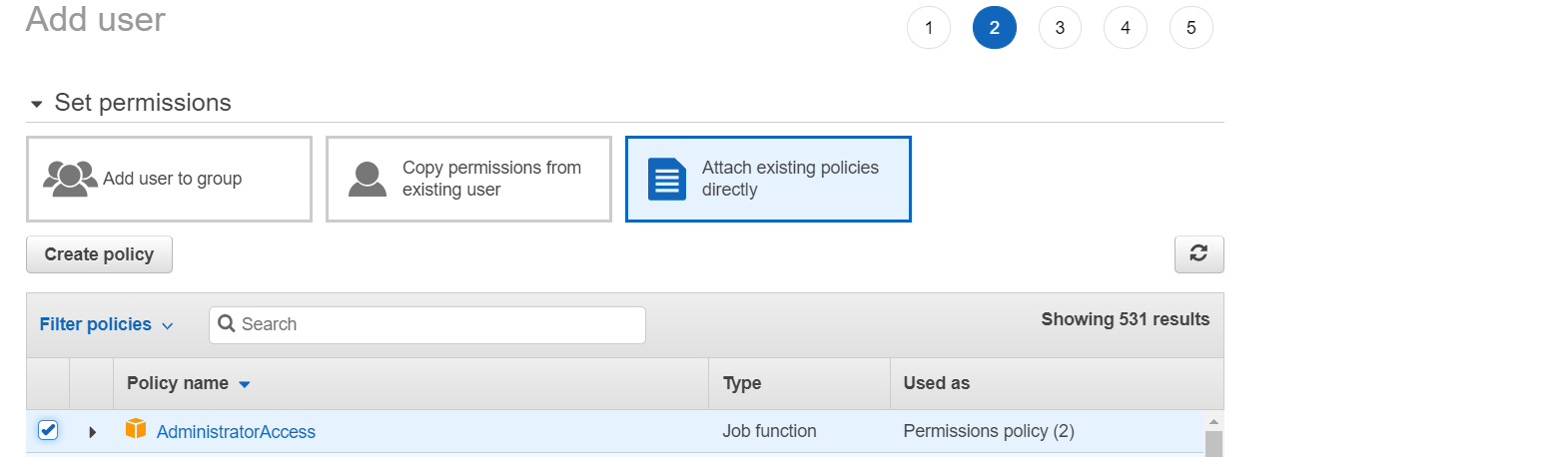


Click on Add User



Give name for user and select programmatic access

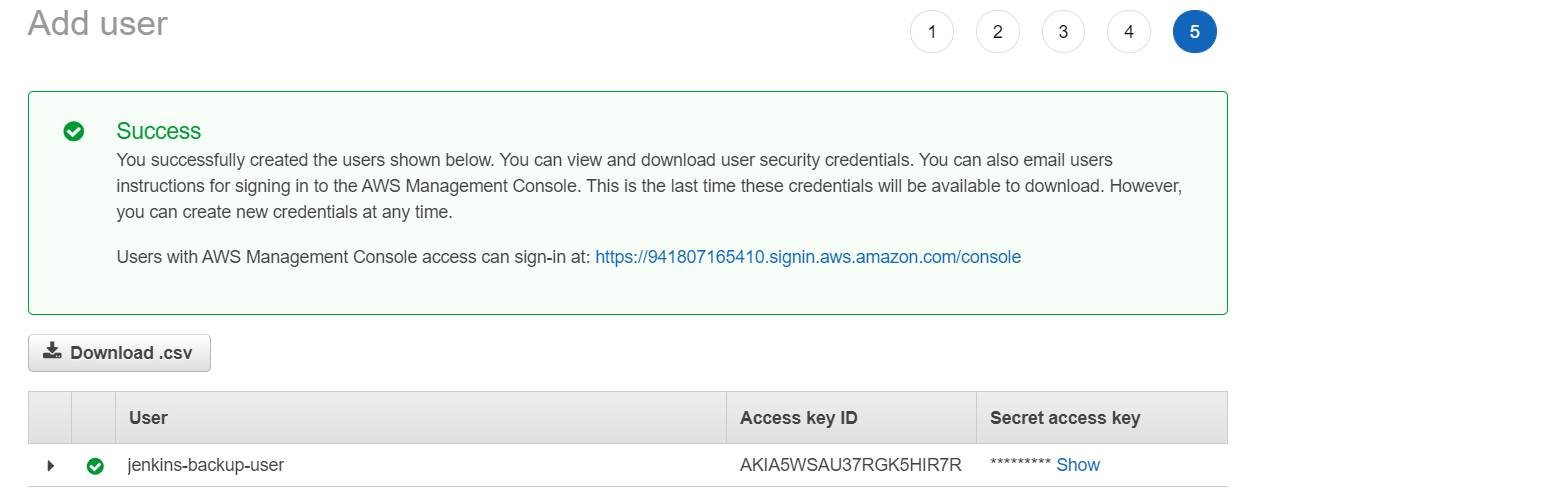
Click on Next: Permissions



Click on Attach existing policies directly and select AdministratorAccess policy

Click on Next

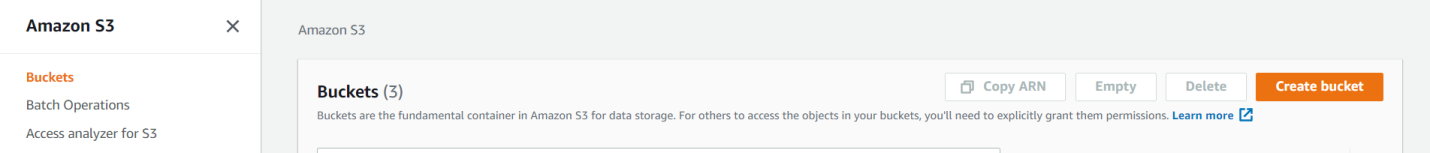
Click on Create User



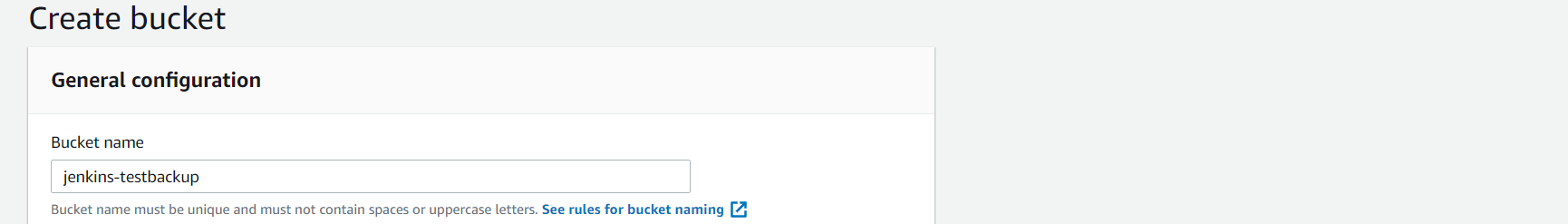
Copy Access key ID and Secret access key

Create S3 bucket to store backup data:

Goto S3 service 🡪 Create bucket



* Click on Create bucket

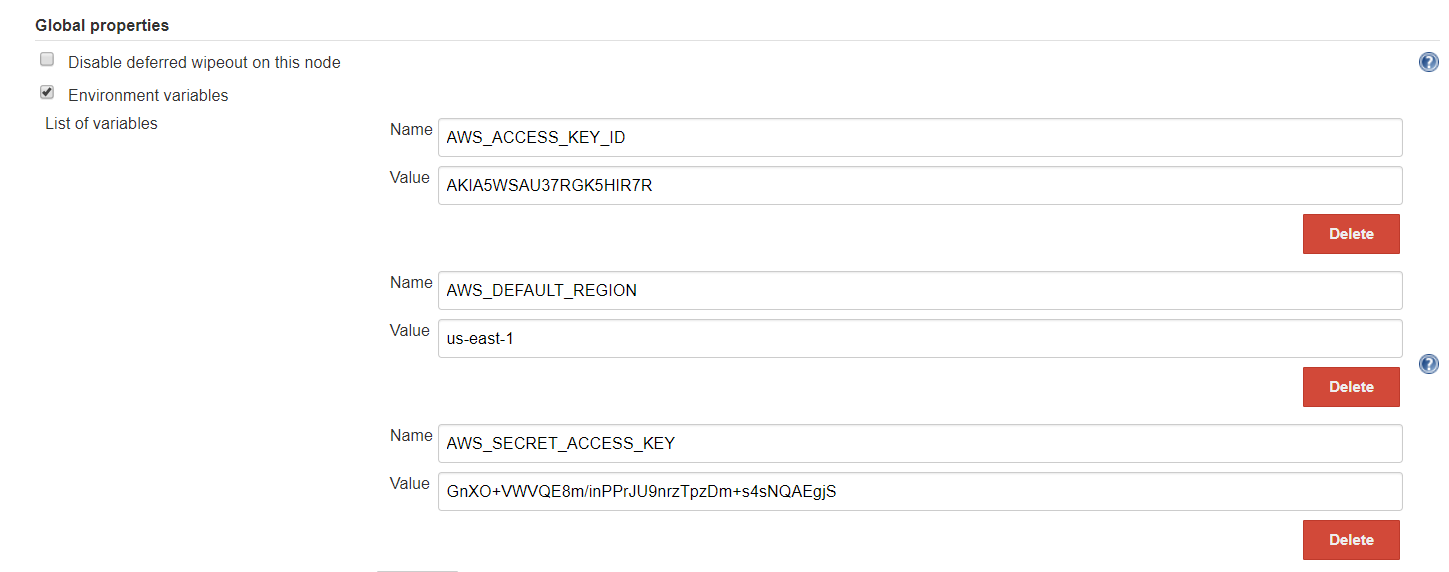


Keep Name for the bucket and Click on create bucket

Configure Access key and Secret access key as environment variables with Jenkins:

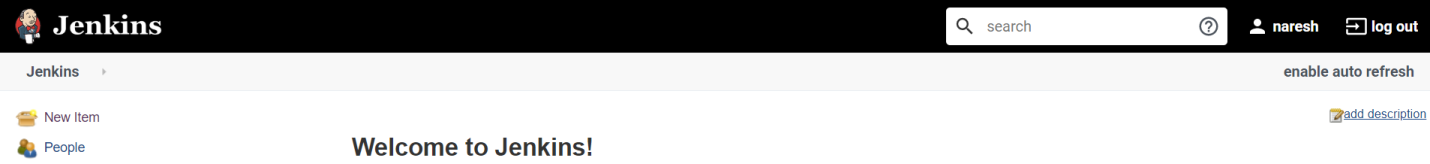
Goto Jenkins dashboard 🡪 Configure system

With in the Global properties keep Environment variables as shown below

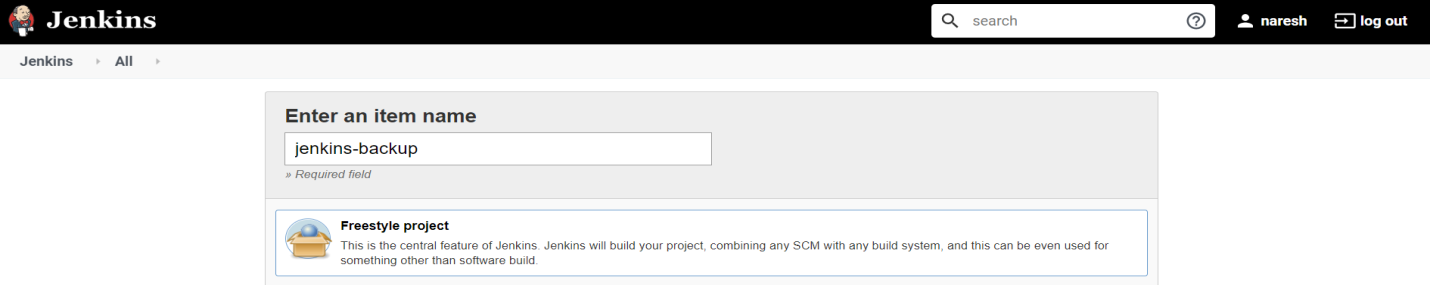


Click on Save

Create a free style job to backup Jenkins into S3 bucket:



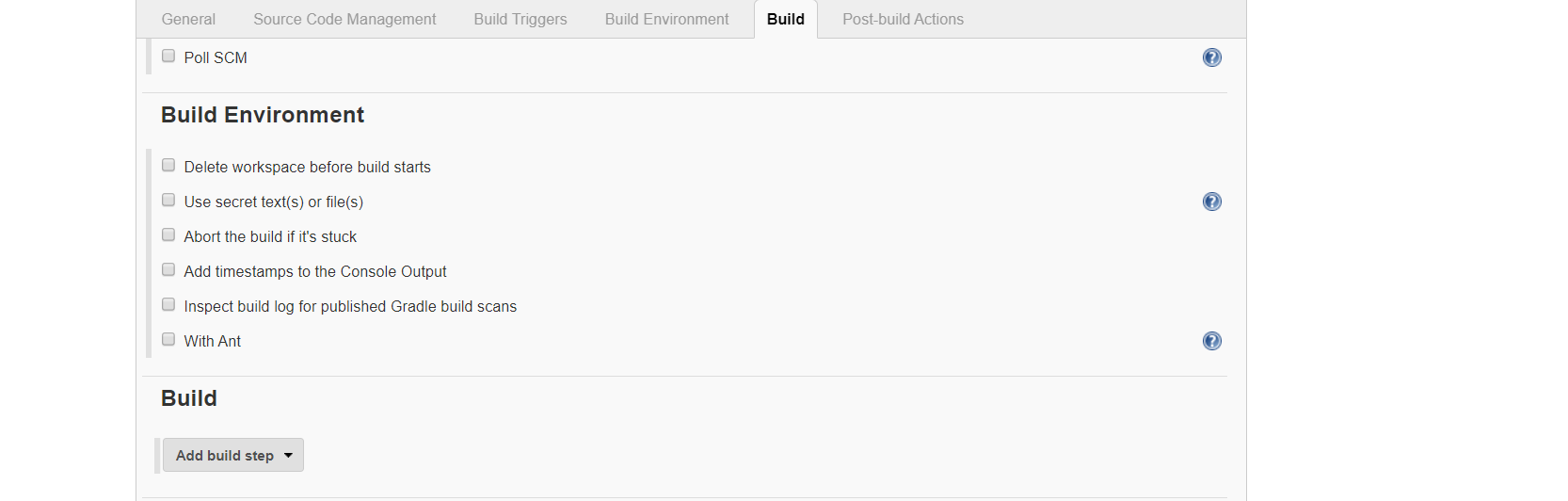
Click on New item



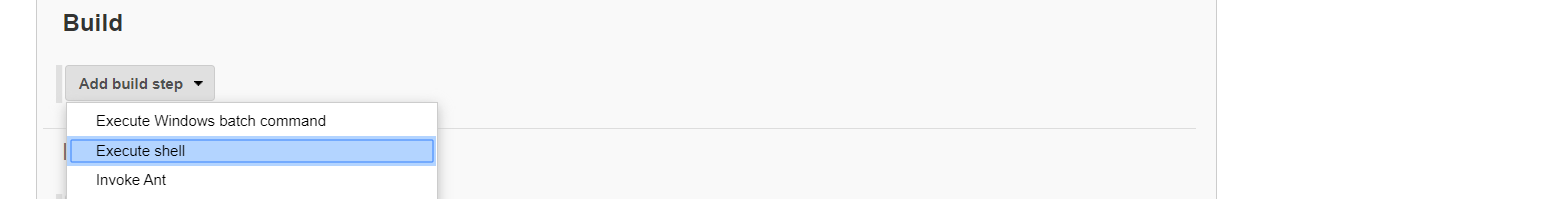
Give a name for Jenkins job and select Freestyle project

Click on OK

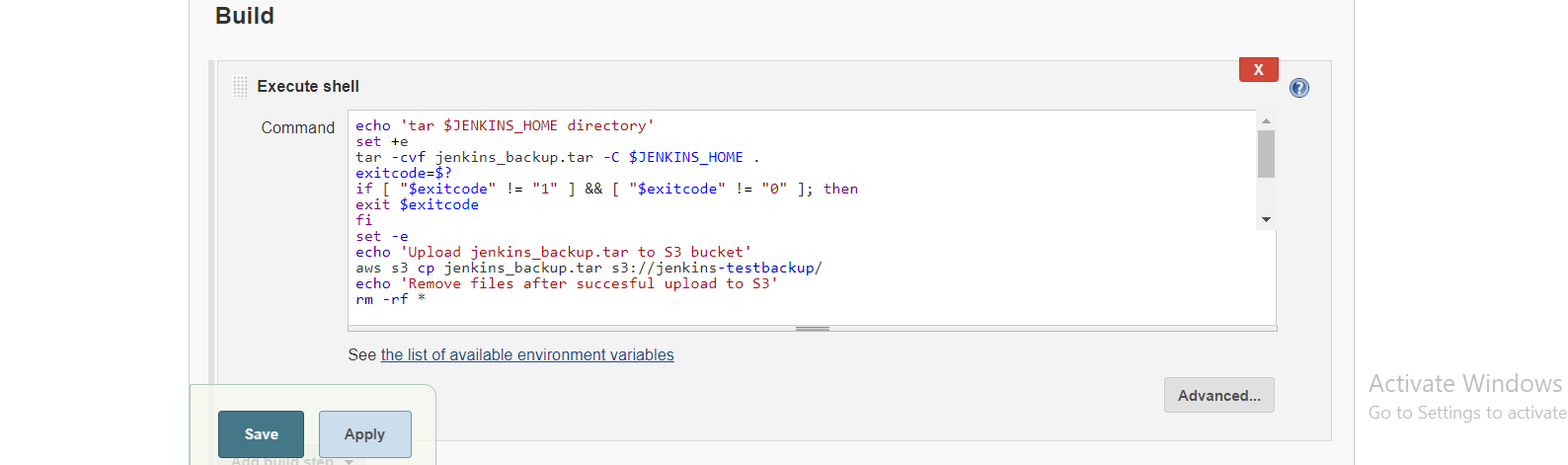
Goto Build Section



Click on Add build step

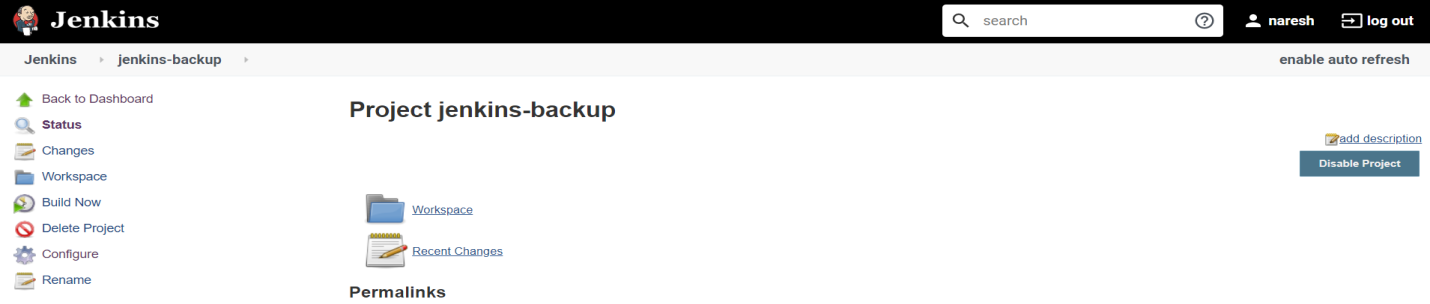


Select Execute shell

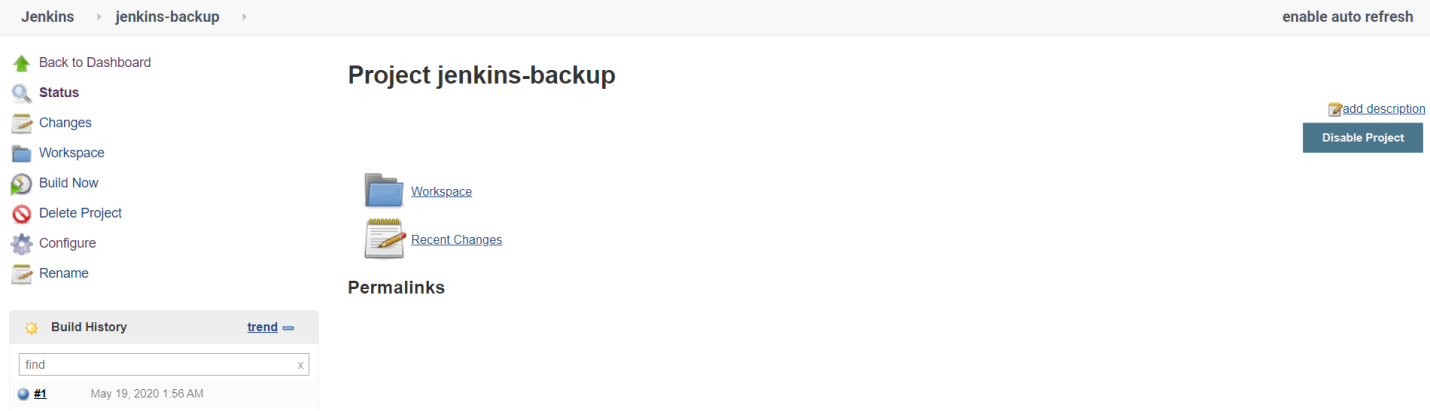


Keep data as shown in above figure

Click on save



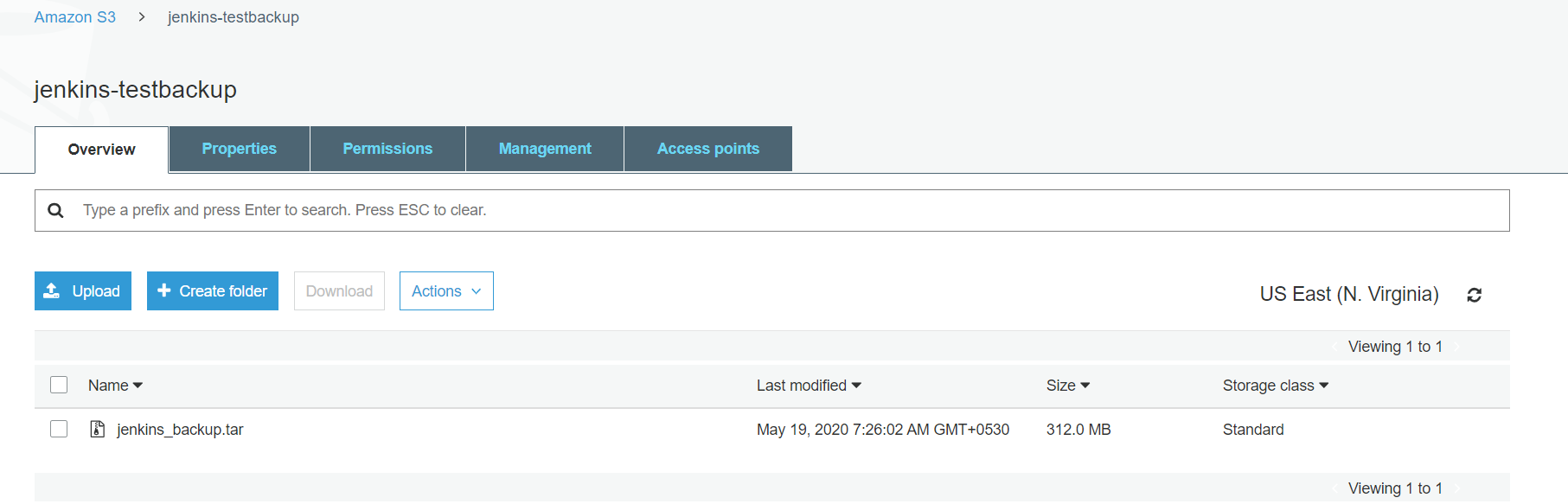
Click on Build



Here we can see build History shown as success

Now check S3 bucket whether the Jenkins backup file saved within our bucket on not

Goto Amazon S3 🡪 Jenkins-testbackup



Data within the Execute shell:

echo 'tar $JENKINS\_HOME directory'

set +e

tar -cvf jenkins\_backup.tar -C $JENKINS\_HOME .

exitcode=$?

if [ "$exitcode" != "1" ] && [ "$exitcode" != "0" ]; then

exit $exitcode

fi

set -e

echo 'Upload jenkins\_backup.tar to S3 bucket'

aws s3 cp jenkins\_backup.tar s3://jenkins-testbackup/

echo 'Remove files after succesful upload to S3'

rm -rf \*