

NAME:HARISH AEYYA

[EMAIL:harishaeyya@gmail.com](mailto:harishaeyya@gmail.com)

MOBILE:8184891562

BATCH:119(7AM)

## **Create a s3 bucket upload a files using terraform**

**Terraform:** Terraform is an Iaas software tool that provides a consistent command line interface workflow to manage hundreds of cloud services.


**S3:** Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. Customers of all sizes and industries can use Amazon S3 to store and protect any amount of data for a range of use cases, such as data lakes, websites, mobile applications, backup and restore, archive, enterprise applications, IoT devices, and big data analytics.

### **process:**

- create a iam user with administration access.

terraform [Info](#)[Delete](#)

## Summary

ARN  
 `arn:aws:iam::905418143553:user/terraform`Console access  
 Enabled without MFAAccess key 1  
[Create access key](#)Created  
April 01, 2024, 22:15 (UTC+05:30)Last console sign-in  
 Never[Permissions](#)[Groups](#)[Tags \(1\)](#)[Security credentials](#)[Access Advisor](#)

## Permissions policies (2)


Permissions are defined by policies attached to the user directly or through groups.

[Remove](#)[Add permissions](#) ▼

Filter by Type

All types ▼

< 1 > 

<input type="checkbox"/>	Policy name <a href="#">?</a>	Type	Attached via <a href="#">?</a>
<input type="checkbox"/>	 <a href="#">AdministratorAccess</a>	AWS managed - job function	Directly

- now create access key.


Retrieve access keys [Info](#)

## Access key

If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key

Secret access key

 `AKIA5FTZAO5AYYN7WQXV` \*\*\*\*\* [Show](#)

## Access key best practices

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the [best practices for managing AWS access keys](#).[Download .csv file](#)[Done](#)

- Lets launch an EC2 instance.

**Instance summary for i-0ab884e0c0bd5e315 (today)**
[Info](#)

Connect

Instance state ▼

Actions ▼

Updated 23 minutes ago

<div>Instance ID</div> <div>  i-0ab884e0c0bd5e315 (today)         </div> <div>IPv6 address</div> <div>–</div> <div>Hostname type</div> <div>IP name: ip-172-31-37-53.ap-southeast-2.compute.internal</div> <div>Answer private resource DNS name</div> <div>IPv4 (A)</div> <div>Auto-assigned IP address</div> <div>  13.55.16.160 [Public IP]         </div> <div>IAM Role</div> <div>–</div> <div>IMDSv2</div> <div>Required</div>	<div>Public IPv4 address</div> <div>  13.55.16.160 <a href="#">open address</a> </div> <div>Instance state</div> <div>  Running         </div> <div>Private IP DNS name (IPv4 only)</div> <div>  ip-172-31-37-53.ap-southeast-2.compute.internal         </div> <div>Instance type</div> <div>t2.micro</div> <div>VPC ID</div> <div>  vpc-08e7af620390d0b41         </div> <div>Subnet ID</div> <div>  subnet-0ce2f1822b5f1aec9         </div>	<div>Private IPv4 addresses</div> <div>  172.31.37.53         </div> <div>Public IPv4 DNS</div> <div>  ec2-13-55-16-160.ap-southeast-2.compute.amazonaws.com <a href="#">open address</a> </div> <div>Elastic IP addresses</div> <div>–</div> <div>AWS Compute Optimizer finding</div> <div>  Opt-in to AWS Compute Optimizer for recommendations.             <a href="#">Learn more</a> </div> <div>Auto Scaling Group name</div> <div>–</div>
--	--	--

- now update application and install awscli.

```

ubuntu@ip-172-31-37-53:~$ sudo -i
root@ip-172-31-37-53:~# apt update -y
Hit:1 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:3 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 https://apt.releases.hashicorp.com jammy InRelease [12.9 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:6 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1519 kB]
Get:7 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [293 kB]
Get:8 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1644 kB]
Get:9 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1060 kB]
Get:10 https://apt.releases.hashicorp.com jammy/main amd64 Packages [125 kB]
Get:11 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1303 kB]
Get:12 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [852 kB]
Fetched 7037 kB in 3s (2418 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
41 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-172-31-37-53:~# apt install awscli -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
awscli is already the newest version (1.22.34-1).
0 upgraded, 0 newly installed, 0 to remove and 41 not upgraded.
root@ip-172-31-37-53:~# apt-get install terraform

```

- now configure your access key id and security key to the resources.

```

root@ip-172-31-37-53:~# aws configure
AWS Access Key ID [*****WQXV]: AKIA5FTZA05A2ANXWTET
AWS Secret Access Key [*****Hhd/]: oc29XXOq9Doils/JP4Zd+VgDs0r3oFWyljx9JE0w
Default region name [ap-southeast-2]: ap-southeast-2
Default output format [table]: table
root@ip-172-31-37-53:~# wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor

```

- install the terraform hasicorp on ubuntu.

```

root@ip-172-31-37-53:~# wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
sudo apt update && sudo apt install terraform
--2024-04-03 08:02:31-- https://apt.releases.hashicorp.com/gpg
Resolving apt.releases.hashicorp.com (apt.releases.hashicorp.com)... 18.67.93.117, 18.67.93.22, 18.67.93.76, ...
Connecting to apt.releases.hashicorp.com (apt.releases.hashicorp.com)|18.67.93.117|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3980 (3.9K) [binary/octet-stream]
Saving to: 'STDOUT'

-
>] 3.89K --.-KB/s in 0s

100%[=====]

2024-04-03 08:02:32 (130 MB/s) - written to stdout [3980/3980]

File '/usr/share/keyrings/hashicorp-archive-keyring.gpg' exists. Overwrite? (y/N) y
deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com jammy main
Hit:1 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://ap-southeast-2.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 https://apt.releases.hashicorp.com jammy InRelease
Hit:5 http://security.ubuntu.com/ubuntu jammy-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
41 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
terraform is already the newest version (1.7.5-1).
0 upgraded, 0 newly installed, 0 to remove and 41 not upgraded.

```

- create a directory and changed to root directory.

```

No VM guests are running outdated hypervisor (qemu) binaries
root@ip-172-31-37-53:~# mkdir terraform
root@ip-172-31-37-53:~# cd terraform
root@ip-172-31-37-53:~/terraform# ls
root@ip-172-31-37-53:~/terraform#

```

- create a file(f1) in directory

```

root@ip-172-31-37-53:~/terraform# ls
f1 provider.tf resource.tf terraformblock.tf

```

- create terraformblock.tf

```
root@ip-172-31-37-53:~/terraform# cat terraformblock.tf
terraform {
  required_providers {
    aws = {
      source = "hashicorp/aws"
      version = "5.43.0"
    }
  }
}
```

- create provider.tf

```
root@ip-172-31-37-53:~/terraform# cat provider.tf
provider "aws" {
  region = "ap-southeast-1"
  profile = "default"
}
```

- create resource.tf

```
root@ip-172-31-37-53:~/terraform# cat resource.tf
resource "aws_s3_bucket" "mybucket" {
  bucket = "hari-rakhi"
  tags = {
    Name = "Mybucket"
  }
}

resource "aws_s3_object" "obj" {
  bucket = aws_s3_bucket.mybucket.id
  key    = "f1"
  source = "f1"
}
```

- lets following the terraform commands
- terraform init

```
root@ip-172-31-37-53:~/terraform# terraform init

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.43.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
```

- terraform validate

```
root@ip-172-31-37-53:~/terraform# terraform validate
Success! The configuration is valid.

root@ip-172-31-37-53:~/terraform#
```

- terraform plan
- terraform apply

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.

Only 'yes' will be accepted to approve.

Enter a value: yes

aws\_s3\_bucket.mybucket: Creating...

aws\_s3\_bucket.mybucket: Creation complete after 4s [id=hari-rakhi]

aws\_s3\_object.obj: Creating...

aws\_s3\_object.obj: Creation complete after 0s [id=f1]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.

root@ip-172-31-37-53:~/terraform#

The image displays two screenshots of the AWS Management Console interface, specifically the Amazon S3 service.

**Top Screenshot:** Shows the 'Amazon S3' console page. The left sidebar contains navigation links for Buckets, Access Grants, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Block Public Access settings, Storage Lens, Dashboards, Storage Lens groups, and AWS Organizations settings. The main content area shows the 'Account snapshot' and 'General purpose buckets' section. A table lists the bucket 'hari-rakhi' in the 'Asia Pacific (Sydney) ap-southeast-2' region, created on 'April 11, 2024, 12:24:10 (UTC+05:30)'. The 'Create bucket' button is highlighted in orange.

**Bottom Screenshot:** Shows the 'hari-rakhi' bucket details page. The left sidebar is the same as the top screenshot. The main content area shows the 'Objects (1)' section. A table lists the object 'f1' with a size of '29.0 B' and a storage class of 'Standard'. The 'Upload' button is highlighted in orange.