

```
Ryans-iMac:Downloads accloudguru$ ssh ec2-user@35.153.184.58 -i MyNewKeyPair.pem
Last login: Thu May  3 13:41:17 2018 from 90.152.124.231

  __|  __|_  )
  _| (      /   Amazon Linux AMI
 ---|\---|---|

https://aws.amazon.com/amazon-linux-ami/2018.03-release-notes/
17 package(s) needed for security, out of 19 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-36-54 ~]$ sudo su
```

@public_ip
Logging into amazon ami

```
[root@ip-172-31-36-54 ec2-user]# aws s3 ls
Unable to locate credentials. You can configure credentials by running "
aws configure".
[root@ip-172-31-36-54 ec2-user]# aws configure
AWS Access Key ID [None]: AKIAI7BPILJB33K7H36Q
AWS Secret Access Key [None]: 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
Default region name [None]:
Default output format [None]:
[root@ip-172-31-36-54 ec2-user]# clea
```

IAM

```
[root@ip-172-31-36-54 ec2-user]# aws s3 ls
[root@ip-172-31-36-54 ec2-user]# aws s3 mb s3://accloudguru1234-rk
make_bucket: accloudguru1234-rk
[root@ip-172-31-36-54 ec2-user]# aws s3 ls
2018-06-05 09:09:28 accloudguru1234-rk
[root@ip-172-31-36-54 ec2-user]# echo "hello cloud gurus" > hello.txt
[root@ip-172-31-36-54 ec2-user]# ls
hello.txt
[root@ip-172-31-36-54 ec2-user]# aws s3 cp hello.txt s3://accloudguru1234-rk
upload failed: ./hello.txt to s3://accloudguru1234-rk/hello.txt An error
occurred (NoSuchBucket) when calling the PutObject operation: The specif
ied bucket does not exist
[root@ip-172-31-36-54 ec2-user]# aws s3 cp hello.txt s3://accloudguru1234-rk
upload: ./hello.txt to s3://accloudguru1234-rk/hello.txt
[root@ip-172-31-36-54 ec2-user]# aws s3 ls s3://accloudguru1234-rk
2018-06-05 09:10:35      18 hello.txt
```

List buckets

Make bucket

Create a file and copy to
s3 bucket created

List contents of buckets

Under IAM → user section → regenerating keys

Access Key ID can be seen again but secret key can be seen only once.

Make inactive
Delete the user – user will not have access to the access key.
Create access key to regenerate access and secret key (seen only once)

Throws invalid access key id because for the existing user, the access key and secret key was regenerated and doesn't match.

This will allow you to add the new access key and secret key.

Search IAM

Dashboard
Groups
Users
Roles
Policies
Identity providers
Account settings
Credential report
Encryption keys

Permissions Groups (1) Security credentials Access Advisor

Sign-in credentials

Console password Disabled [Manage password](#)
Console login link N/A
Last login None
Assigned MFA device No [Manage](#)
Signing certificates None [Manage](#)

Access keys

Use access keys to make secure REST or HTTP Query protocol requests to AWS service APIs. For your protection, you should never share your secret keys with anyone. As a best practice, we recommend frequent key rotation. [Learn more](#)

Create access key

Access key ID	Created	Last used	Status
AKIA7BPILJB33K7H36Q	2018-06-05 10:07 UTC+0100	N/A	Active Make inactive Delete

SSH keys for AWS CodeCommit

Use SSH public keys to authenticate access to AWS CodeCommit repositories. [Learn more](#)

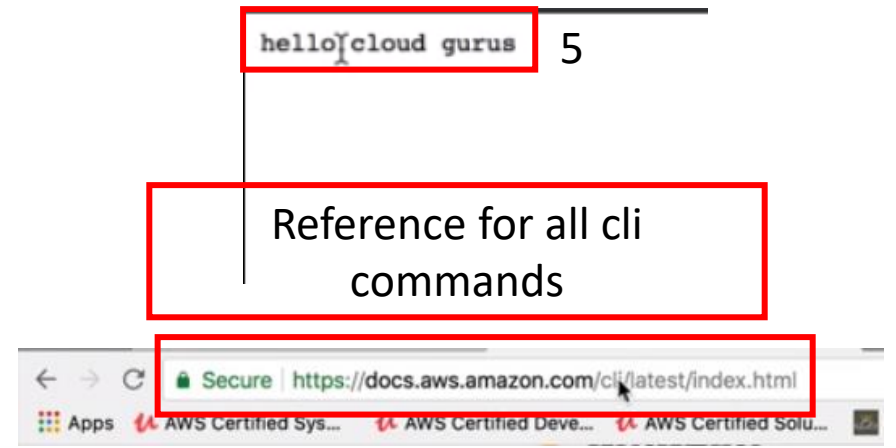
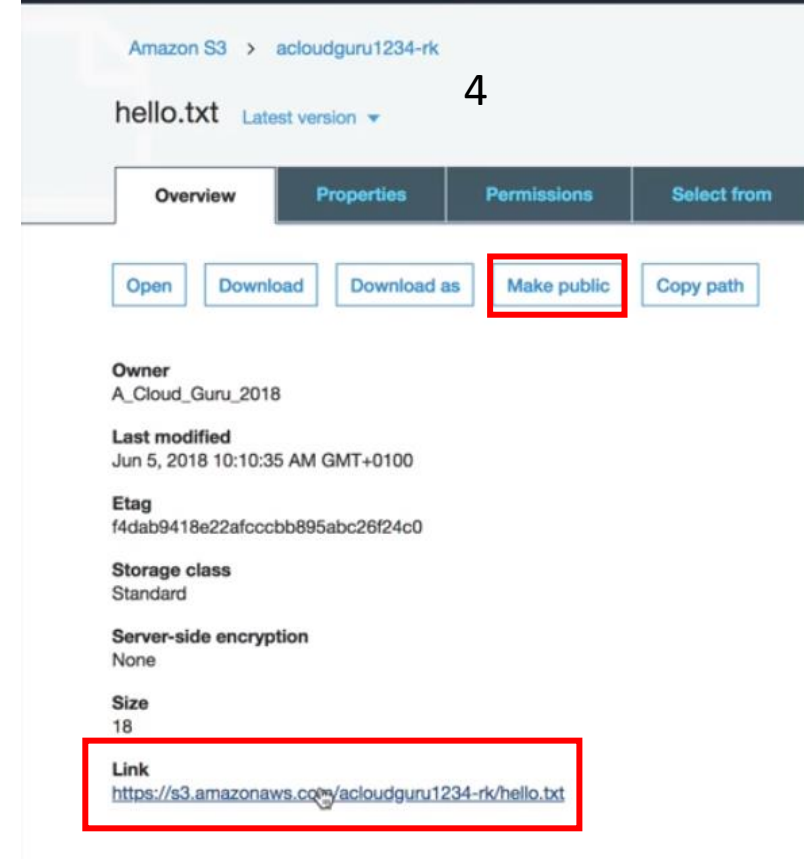
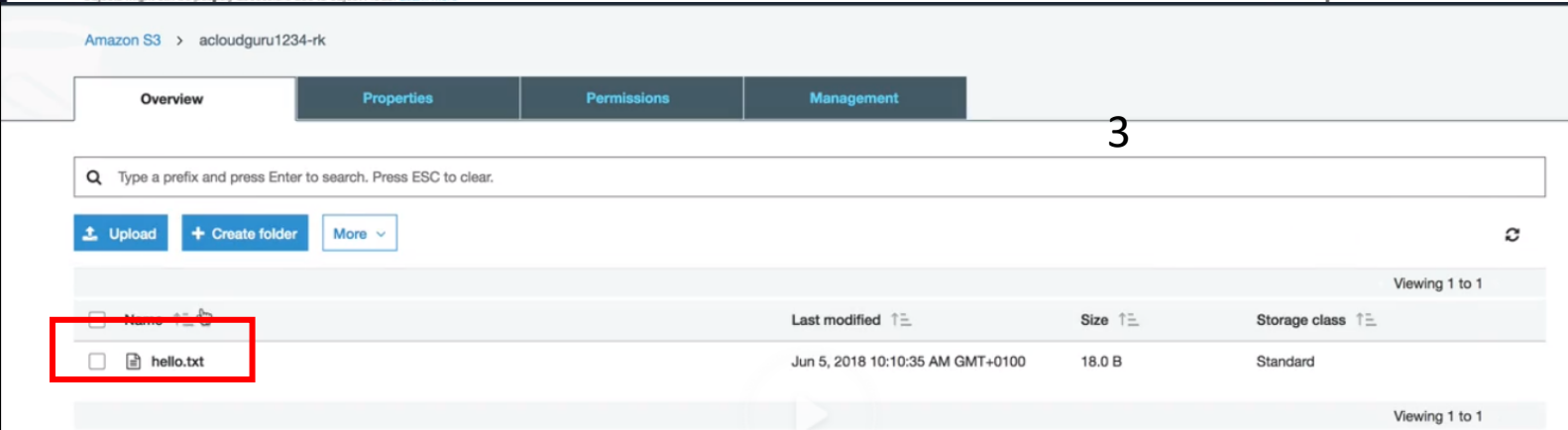
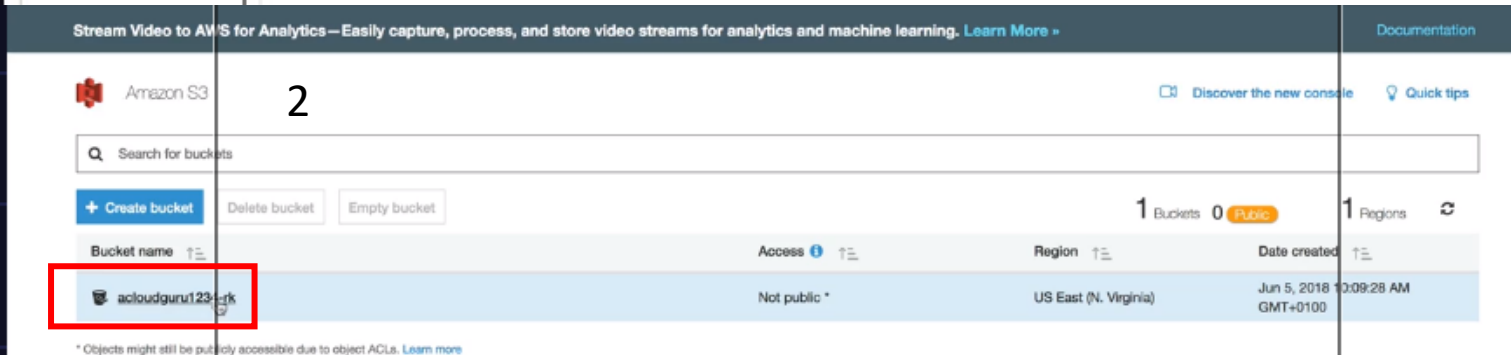
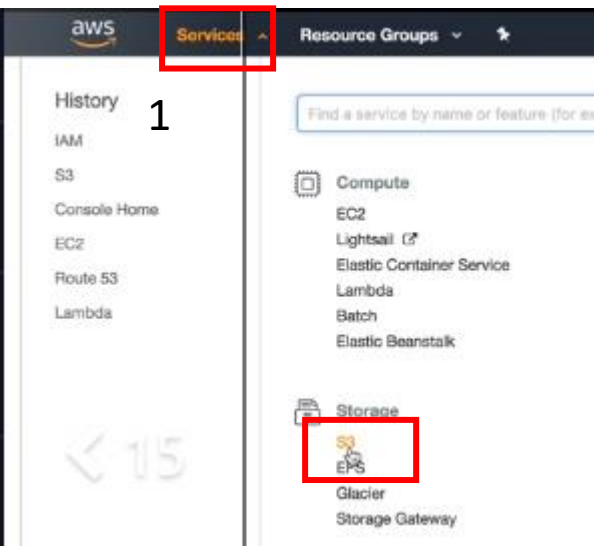
Upload SSH public key

SSH key ID	Uploaded	Status
No results		

HTTPS Git credentials for AWS CodeCommit

Generate a user name and password you can use to authenticate HTTPS connections to AWS CodeCommit repositories. You can generate and store up to 2 sets of credentials. [Learn more](#)

```
[root@ip-172-31-36-54 ec2-user]# aws s3 ls
An error occurred (InvalidAccessKeyId) when calling the ListBuckets operation: The AWS Access Key Id you provided does not exist in our records.
[root@ip-172-31-36-54 ec2-user]# aws configure
AWS Access Key ID [*****H36Q]:
```



Least Privilege - Always give your users the minimum amount of access required.

Create Groups - Assign your users to groups. Your users will automatically inherit the permissions of the group. The groups permissions are assigned using policy documents.

Secret Access Key - You will see this only once. If you do not save it, you can delete the Key Pair (Access Key ID and Secret Access Key) and regenerate it. You will need to run **aws configure** again.

Do not use just one access key - Do not create just one access key and share that with all your developers. If someone leaves the company on bad terms, then you will need to delete the key and create a new one and every developer would then need to update their keys. Instead create one key pair per developer.

You can use the CLI on your PC - You can install the CLI on your Mac, Linux or Windows PC. I personally use S3 to store all my files up in the cloud.