

AWS CLI



“All service interfaces, without exception, must be designed from the ground up to be externalizable.” – Attributed to Jeff Bezos

**Why does this matter?**



# Everything in AWS Uses the API

## Console

Path of Least  
Resistance

## CLI

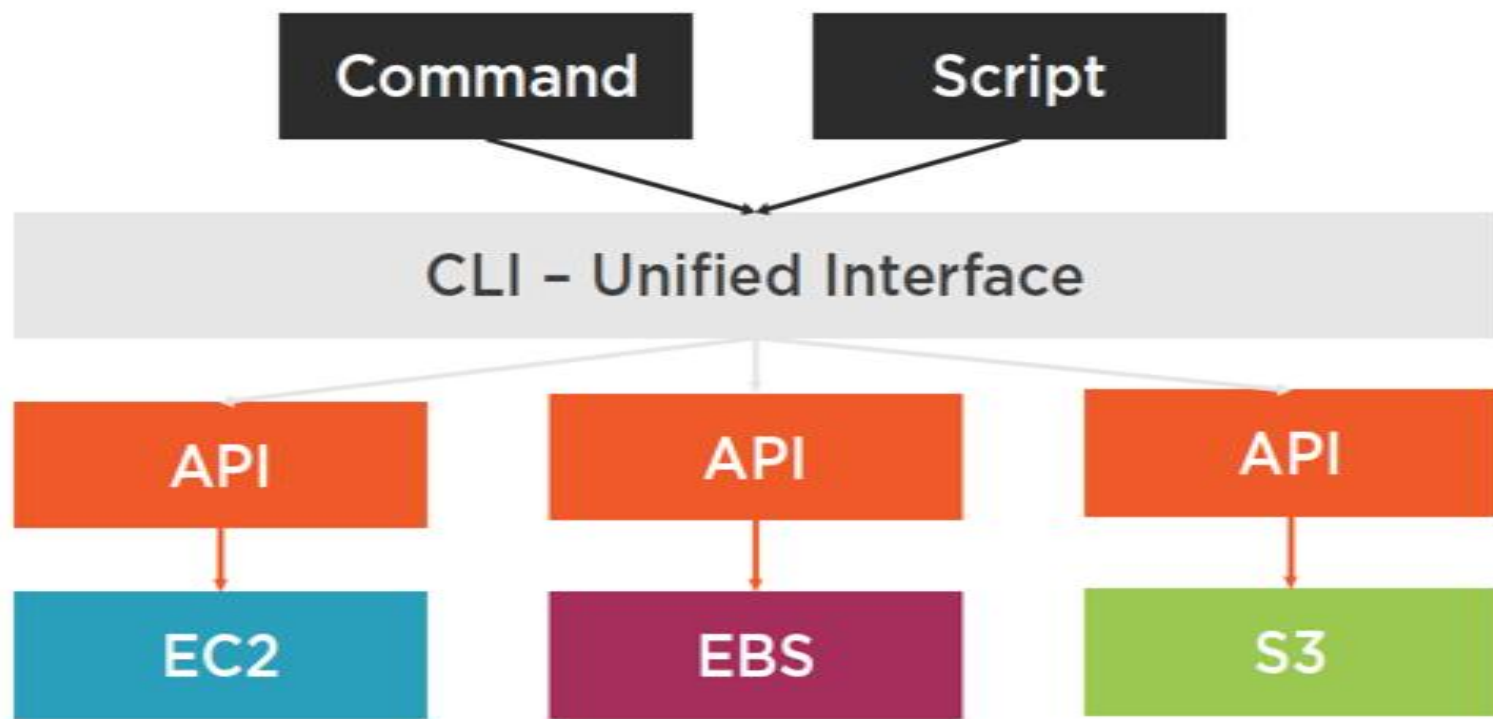
Operations and  
Automation

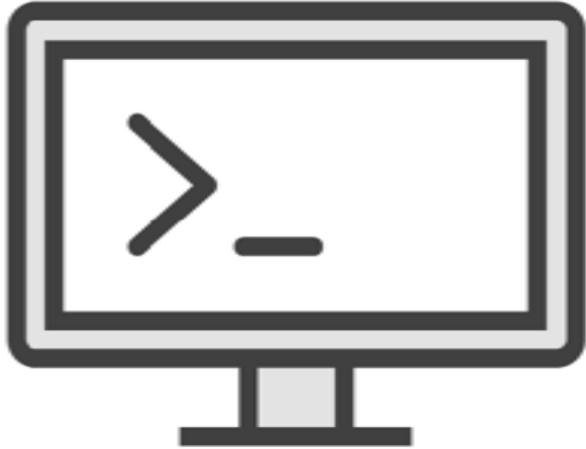
## SDK

Development and  
Automation

API

# What Is the AWS CLI?





CLI available for:

Windows

MacOS

Linux

Demo



Install CLI on Linux

PIP

Standalone

Configure CLI

Basics

Extras

AWS Completer

AWS Shell

# INSTALLATION STEPS (UBUNTU)

- 1.Ensure python is installed (check using “whereis python” or “**python --version**”)
- 2.Install pip using “**apt-get -y install python-pip**”
- 3.Install awscli using pip by executing following command “**pip install awscli**”
- 4.AWS CLI will be installed with this and check by executing “**aws help**”

# CONFIGURING AWS CLI

1. Configure aws by using `aws configure` and enter the required details
2. To enable command line completion execute following command `complete -C '/usr/local/aws/completer' aws`
3. Install `aws-shell` by executing `pip install aws-shell` and execute `a`

```
aws> s3 ls
cognito-sync
ds
es
s3
dms
ecs
efs
kms
rds
sdb
ses
sms
sns
sqs
ssm
sts
```



# SECURING CLI

## Order of Preference for Credentials

### Command line options

### Environment variables

- AWS\_ACCESS\_KEY\_ID
- AWS\_SECRET\_ACCESS\_KEY

### Credentials file

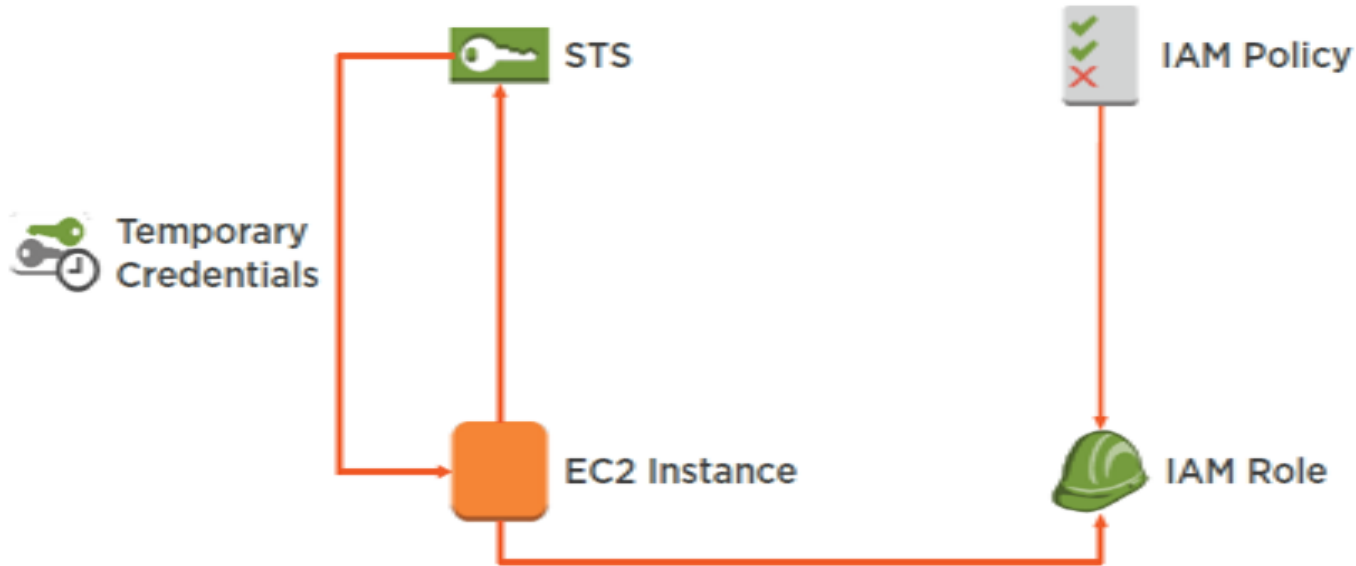
- ~/.aws/credentials
- C:\Users\USERNAME\.aws\credentials

### CLI config file

- ~/.aws/config
- C:\Users\USERNAME\.aws\config

### Instance profile credentials

# Instance Profile Credentials Explained



Demo



## Credential Order of Preference

- Show ways to supply credentials to CLI

# CREDENTIAL ORDER OF PREFERENCE

1. Create an IAM Role for EC2 for s3 read-only access and launch ec2 instance with IAM Role
2. Execute command `aws ec2 describe-instances --region us-west-2` and you will observe `UnauthorizedOperation`
3. Set Environment Variables with `export AWS_ACCESS_KEY_ID=<accesskeyid>` and `export AWS_SECRET_ACCESS_KEY=<secret_key>` of IAM user who has access to ec2
4. Now execute the same command as in step2 and you should be able to see the results as Environment variables are

# CREDENTIAL ORDER OF PREFERENCE (CONTD)

5. Now unset environment variables by “unset AWS\_ACCESS\_KEY\_ID” and “unset AWS\_SECRET\_ACCESS\_KEY” and execute the command in step 2 and you will get unauthorized error.

6. Now execute “aws configure” and provide IAM User access key and secret access key. Execute the CLI Command in step 2 you should see the results

7.

# COPYING AND SYNCING OBJECTS BETWEEN S3 BUCKETS

## Module Intro



**Strategic goal: Simplicity of data migration**

**Scenario: Frequent data transfer between S3 buckets**

**Demo: S3 copy and sync between buckets**



# AWS S3 Dashboard

## Easy Operations

Bucket creation

Single object upload

Single object download

## Difficult Operations

Copy of multiple objects

Synchronization between buckets

Bulk permissions changes

# Why Use S3?

Durability

Ease of access

Sticky glue that binds data-driven workloads together

# Data Transfer Between S3 Buckets

## Scenario 1

Upload existing data

## Scenario 2

Copy between buckets  
same account

## Scenario 3

Copy between buckets  
different account

Demo



## S3 copy and sync between buckets

- Part 1
  - Upload and/or sync data to S3
- Part 2
  - Copy between buckets in same account
  - Copy between buckets in different accounts



# IMPORTANT LINKS

<http://docs.amazonaws.cn/cli/latest/reference/glacier/index.html#cli-aws-glacier>

<http://docs.amazonaws.cn/cli/latest/reference/s3/index.html>

<http://docs.amazonaws.cn/cli/latest/reference/s3api/index.html>

<https://www.linkedin.com/pulse/how-scan-millions-files-aws-s3-jishnu-kinwar>