



# CAPITAL MARKETS APPLICATION SUPPORT ASSIGNMENT

Nadee Ranasinghe

## Contents

IAM User .....	2
EC2 Instance.....	3
S3 Bucket.....	4
Script .....	5
Supplementary requirements .....	7

# IAM User

The screenshot shows the 'Set user details' page in the AWS IAM console. The user name is 'Nadeee'. The access type is 'AWS Management Console access'. The console password is a custom password. The 'Require password reset' checkbox is checked.

**Set user details**

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name\*

[Add another user](#)

**Select AWS access type**

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Access type\* ☐ **Programmatic access**  
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

☒ **AWS Management Console access**  
Enables a **password** that allows users to sign-in to the AWS Management Console.

Console password\* ☐ Autogenerated password  
☒ Custom password

☐ Show password

Require password reset ☒ User must create a new password at next sign-in

\* Required

[Cancel](#) [Next: Permissions](#)

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

AmazonS3FullAccess policy is attached to the newly created group.

The screenshot shows the 'Set permissions' page in the AWS IAM console. A 'Create group' dialog is open, showing a list of policies. The 'AmazonS3FullAccess' policy is selected.

**Set permissions**

**Create group**

Create a group and select the policies to be attached to the group. Using groups is a best-practice way to manage users' permissions by job functions, AWS service access, or your custom permissions. [Learn more](#)

Group name

[Create policy](#) [Refresh](#)

Filter policies  Showing 6 results

	Policy name	Type	Used as	Description
<input type="checkbox"/>	AmazonDMSRedshiftS3Role	AWS managed	None	Provides access to manage S3 settings for Redshift endpoints for DMS.
<input checked="" type="checkbox"/>	AmazonS3FullAccess	AWS managed	None	Provides full access to all buckets via the AWS Management Console.
<input type="checkbox"/>	AmazonS3OutpostsFullAccess	AWS managed	None	Provides full access to Amazon S3 on Outposts via the AWS Management Console.
<input type="checkbox"/>	AmazonS3OutpostsReadOnlyAccess	AWS managed	None	Provides read only access to Amazon S3 on Outposts via the AWS Management Console.
<input type="checkbox"/>	AmazonS3OutpostsReadOnlyAccess	AWS managed	None	Provides read only access to Amazon S3 on Outposts via the AWS Management Console.

[Cancel](#) [Create group](#)

[Cancel](#) [Previous](#) [Next: Tags](#)

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

# EC2 Instance

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

Cancel and Exit

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"

Search by Systems Manager parameter

1 to 39 of 39 AMIs

Quick Start

My AMIs

AWS Marketplace

Community AMIs

Free tier only

Amazon Linux

Free tier eligible

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-08e0ca9924195beba (64-bit x86) / ami-0437d5de8f3d3d52 (64-bit Arm)

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is approaching end of life on December 31, 2020 and has been removed from this wizard.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)

64-bit (Arm)

Red Hat

Free tier eligible

Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-0a9d27a9f45c0efc (64-bit x86) / ami-0816d75a127c17a49 (64-bit Arm)

Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)

64-bit (Arm)

SUSE Linux

Free tier eligible

SUSE Linux Enterprise Server 15 SP2 (HVM), SSD Volume Type - ami-0b3ac3edf2397475 (64-bit x86) / ami-0ab71076ab9b53b0d (64-bit Arm)

SUSE Linux Enterprise Server 15 Service Pack 2 (HVM), EBS General Purpose (SSD) Volume Type. Amazon EC2 AMI Tools preinstalled. Apache 2.2, MySQL 5.5, PHP 5.3, and Ruby 1.8.7 available.

Select

64-bit (x86)

64-bit (Arm)

Feedback

English (US)

© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Privacy Policy

Terms of Use

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

Cancel and Exit

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance families Current generation Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	t2	t2.micro	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes
<input type="checkbox"/>	t2	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
<input type="checkbox"/>	t3	t3.nano	2	0.5	EBS only	Yes	Up to 10 Gbps	Yes

Cancel

Previous

Review and Launch

Next: Configure Instance Details

Feedback

English (US)

© 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Privacy Policy

Terms of Use

New security group is created with following inbound and outbound rules.

**Step 6: Configure Security Group**  
A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☐ Create a new security group ☒ Select an existing security group

Security Group ID	Name	Description	Actions
<input checked="" type="checkbox"/> sg-0c36f455535a7a283	application-support-security-group	Task	<a href="#">Copy to new</a>
<input type="checkbox"/> sg-f372058b	default	default VPC security group	<a href="#">Copy to new</a>

Inbound rules for sg-0c36f455535a7a283 (Selected security groups: sg-0c36f455535a7a283)

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	0.0.0.0/0	Allow HTTP traffic...
HTTP	TCP	80	:::0	Allow HTTP traffic...
SSH	TCP	22	0.0.0.0/0	SSH allowed from a...

[Cancel](#) [Previous](#) [Review and Launch](#)

## S3 Bucket

A S3 bucket was created to log the timestamps.

**Create bucket**  
Buckets are containers for data stored in S3. [Learn more](#)

**General configuration**

Bucket name  
  
Bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

AWS Region

Copy settings from existing bucket - optional  
Only the bucket settings in the following configuration are copied.  
[Choose bucket](#)

**Block Public Access settings for bucket**  
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

☐ Block all public access

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

## Script

```
#!/bin/bash
#shell script
status=$(echo $?)
server_health=$(curl -o /dev/null -s -w '%{http_code}'
http://localhost:80)
pid=$(ps -ef | grep httpd | grep -v grep | head -n 1)

ssh -t -o StrictHostKeyChecking=No $server_IP ;

##### check apache web server installed or not

if which httpd &>/dev/null
then
    echo "httpd is installed on this host"
    http_ver=$(httpd -v | awk -F '[' '/' '{ print $4 }')
    echo "The version of httpd is: $http_ver"
else
    echo "httpd is not installed"
fi

#### add content to document root

echo "Hello Welcome to Capital Market Website !!!" >
/var/www/html/index.html

echo ""

#####check service status

systemctl status httpd &> /dev/null && echo "http service is up & running"
|| sudo systemctl start httpd &> /dev/null

##### Process ID of httpd process

echo ""
echo "httpd is running on the server $pid"

####Loading web server content using curl

if curl -I "http://localhost:80" 2>&1 | grep -w "200" ; then
    echo "web server health check pass $server_health" #### display health
with the status code
    curl -o server-status-$(date +%T) http://localhost:80 2> /dev/null

#### get web server content in to file with the timestamps
echo ""
aws s3 cp $filename s3://mybucket/ #### copy content to s3
```

```

else
    echo "web server is down"
    echo "web server health check failed" | mail -s "health check"
username@example.com ### email for app team
fi

#### please add IAM role for EC2 instance with S3 full access####

##### logs archive #####
logpath=/var/log/httpd/
tempdir=/tmp/web-content/
find $logpath/* -type -f -name "*.log" -mtime 1 -exec cp {} $tempdir/ \;
&& /dev/null
tar -cvzf log_backup -c $tempdir . > /dev/null
aws s3 cp log_backup $bucketname/web-content-$(date +%F).tar &>/dev/null
&& rm -rf web-content.tar || echo "Upload failed" | mail -s "logs upload
failed" $emailid
rm -rf $tempdir

```

```

Last login: Tue Feb 16 11:46:36 2021 from 43.252.14.177

[ec2-user@ip-172-31-38-101 ~]$ sudo bash script1.sh
usage: ssh [-i246AaCfGgKkMmNnqstTvXxy] [-b bind_address] [-c cipher_spec]
[-D [bind_address]:port] [-E log_file] [-e escape_char]
[-F configfile] [-I pkcs11] [-i identity_file]
[-J [user@]host[:port]] [-L address] [-l login_name] [-m mac_spec]
[-O ctl_cmd] [-o option] [-p port] [-Q query_option] [-R address]
[-S ctl_path] [-w host:port] [-x local_tun[:remote_tun]]
[user@hostname] [command]

httpd is installed on this host
The version of httpd is: 2.4.46

http service is up & running

httpd is running on the server root      2854      1 0 Feb15 ?      00:00:04 /usr/sbin/httpd -DFOREGROUND
HTTP/1.1 200 OK
web server health check pass 200

Note: AWS CLI version 2, the latest major version of the AWS CLI, is now stable and recommended for general use. For more information, see the AWS CLI version
2 installation instructions at: https://docs.aws.amazon.com/cli/latest/userguide/install-cliv2.html

usage: aws [options] <command> [<subcommand> [<subcommand> ...] [parameters]]
To see help text, you can run:

    aws help
    aws <command> help
    aws <command> <subcommand> help
aws: error: too few arguments
tar: Removing leading '/' from member names
tar: /tmp/web-content: Cannot stat: No such file or directory
tar: Exiting with failure status due to previous errors
script1.sh: line 43: mail: command not found
[ec2-user@ip-172-31-38-101 ~]$

```

Checking the status of the web server and displaying the content with timestamps.

Hello Welcome to Capital Market Website!!!

## Supplementary requirements

Below is a cloud formation template that can be used to automate this. But this needs to be further developed.

```
Parameters:
  NameOfService:
    Description: "The name of the service this stack is to be used for."
    Type: String
Parameters:
  myKeyPair:
    Description: Amazon EC2 Key Pair
    Type: "AWS::EC2::KeyPair::KeyName"
Resources:
  S3Bucket:
    Type: 'AWS::S3::Bucket'
    DeletionPolicy: Retain
    Properties:
      AccessControl: PublicRead
      BucketName: web-server-2021
Resources:
  EC2Instance:
    Type: AWS::EC2::Instance
    Metadata:
      AWS::CloudFormation::Init:
        config:
          packages:
            yum:
              httpd: []
          services:
            sysvinit:
              httpd:
                enabled: true
                ensureRunning: true
```



```
Properties:
  InstanceType: t2.micro
  ImageId: ami-7a11e213
  SecurityGroupIds:
    - !Ref MySecurityGroup
MySecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Open Ports 22 and 80
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: '22'
        ToPort: '22'
        CidrIp: 0.0.0.0/0
      - IpProtocol: tcp
        FromPort: '80'
        ToPort: '80'
        CidrIp: 0.0.0.0/0
Outputs:
  servername:
    Description: The Public DNS for the EC2 Instance
    Value: !Sub 'http://${EC2Instance.PublicDnsName}'
```