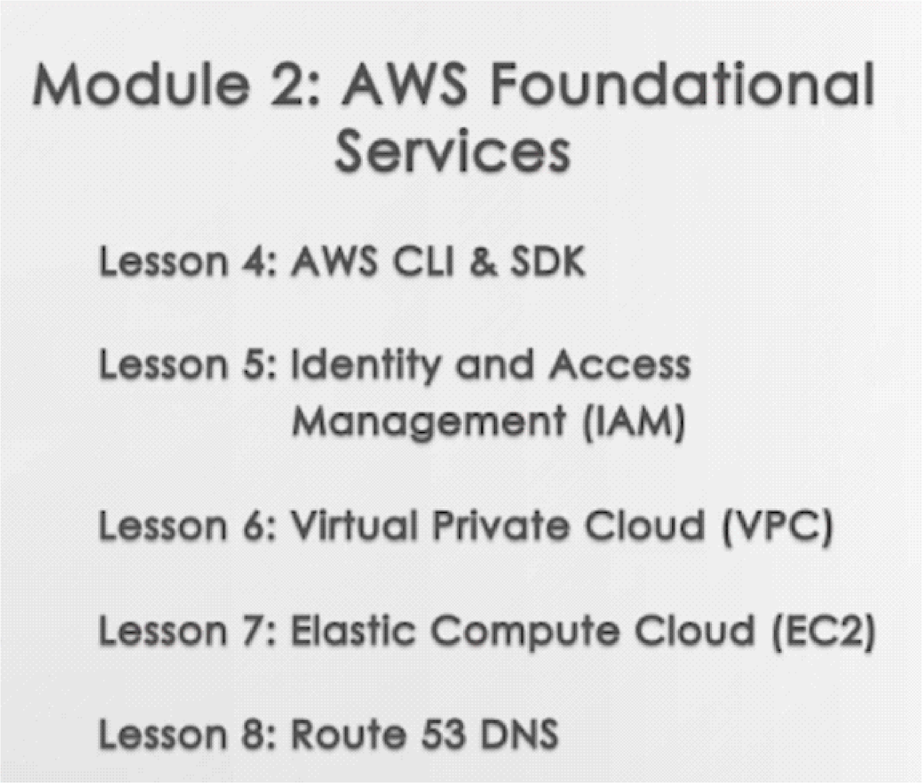
Module Objectives

Sunday, January 9, 2022 5:32 AM



Module 2: AWS Foundational Services

Lesson 4: AWS CLI & SDK

Lesson 5: Identity and Access Management (IAM)

Lesson 6: Virtual Private Cloud (VPC)

Lesson 7: Elastic Compute Cloud (EC2)

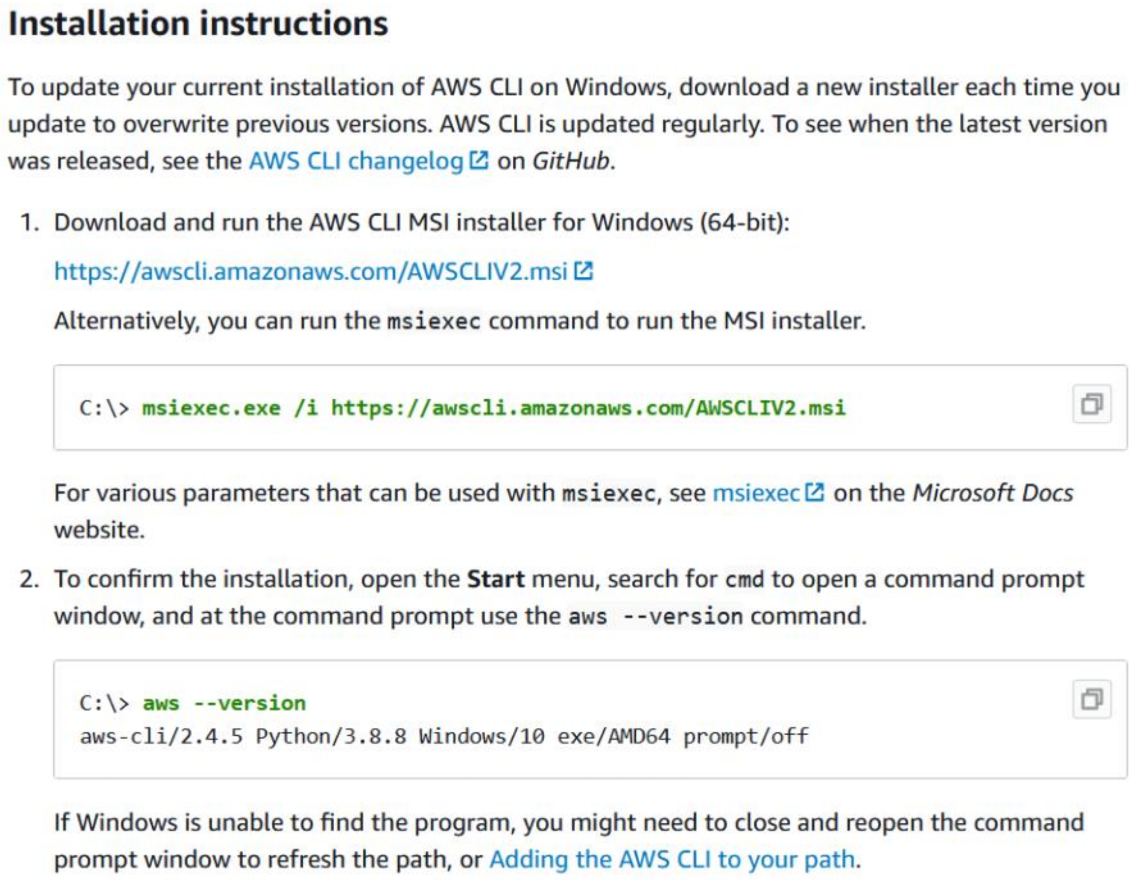
Lesson 8: Route 53 DNS

Screen clipping taken: 1/9/2022 5:32 AM

Installation - CLI

Sunday, January 9, 2022 2:00 AM

CLI Installation: https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-welcome.html



Installation instructions

To update your current installation of AWS CLI on Windows, download a new installer each time you update to overwrite previous versions. AWS CLI is updated regularly. To see when the latest version was released, see the AWS CLI changelog on *GitHub*.

1. Download and run the AWS CLI MSI installer for Windows (64-bit):

   https://awscli.amazonaws.com/AWSCLIV2.msi

   Alternatively, you can run the `msiexec` command to run the MSI installer.

   ```
   C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi
   ```

   For various parameters that can be used with `msiexec`, see msiexec on the *Microsoft Docs* website.

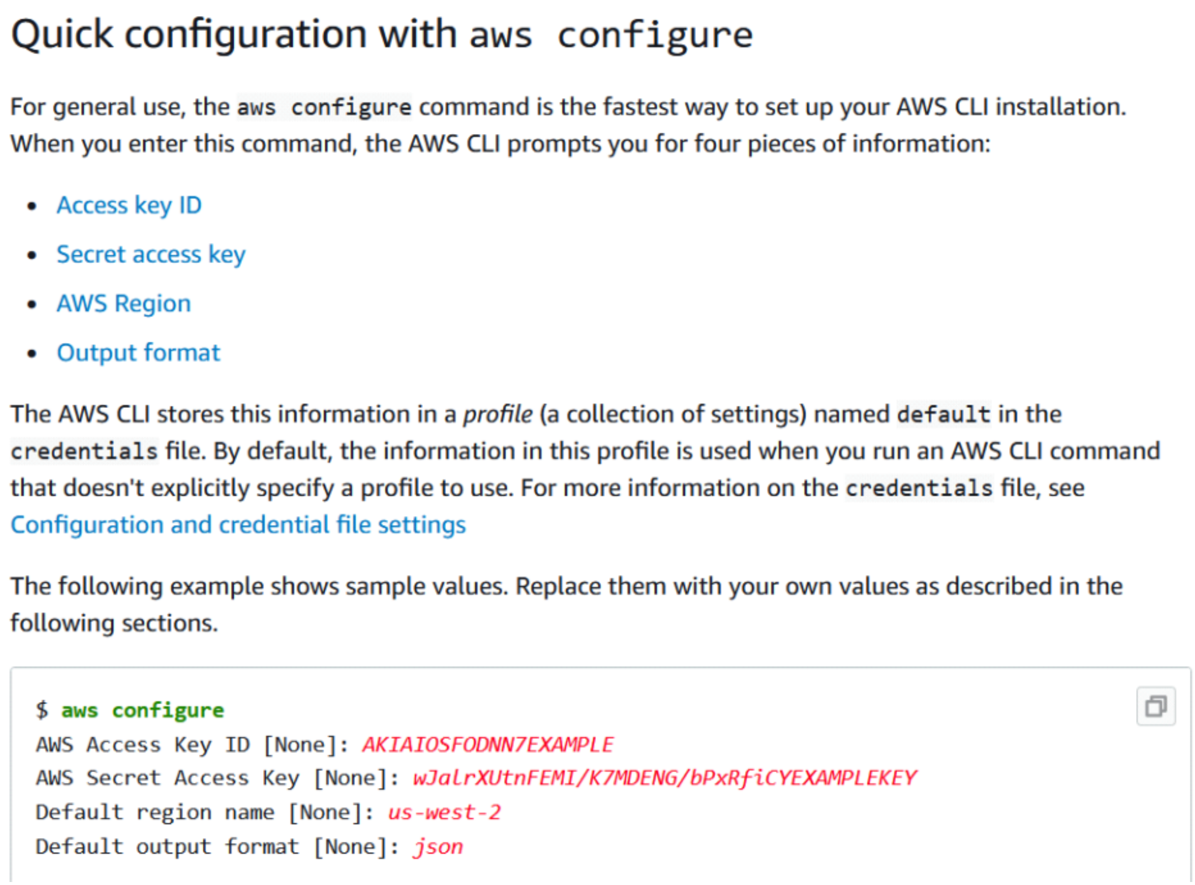
2. To confirm the installation, open the **Start** menu, search for `cmd` to open a command prompt window, and at the command prompt use the `aws --version` command.

   ```
   C:\> aws --version
   aws-cli/2.4.5 Python/3.8.8 Windows/10 exe/AMD64 prompt/off
   ```

   If Windows is unable to find the program, you might need to close and reopen the command prompt window to refresh the path, or Adding the AWS CLI to your path.

Screen clipping taken: 1/9/2022 2:02 AM

CLI Configuration:



Quick configuration with `aws configure`

For general use, the `aws configure` command is the fastest way to set up your AWS CLI installation. When you enter this command, the AWS CLI prompts you for four pieces of information:

- Access key ID
- Secret access key
- AWS Region
- Output format

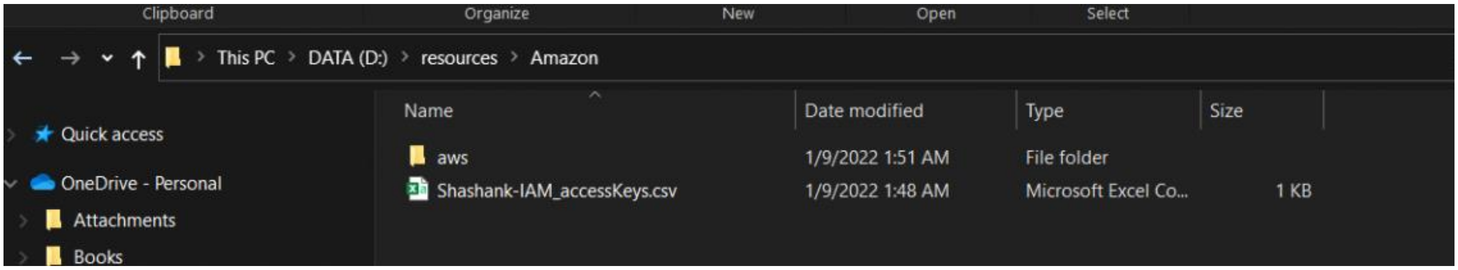
The AWS CLI stores this information in a *profile* (a collection of settings) named `default` in the `credentials` file. By default, the information in this profile is used when you run an AWS CLI command that doesn't explicitly specify a profile to use. For more information on the `credentials` file, see Configuration and credential file settings

The following example shows sample values. Replace them with your own values as described in the following sections.

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

Screen clipping taken: 1/9/2022 2:01 AM

Repository & Credentials: *Secret Access key will be showed only once*

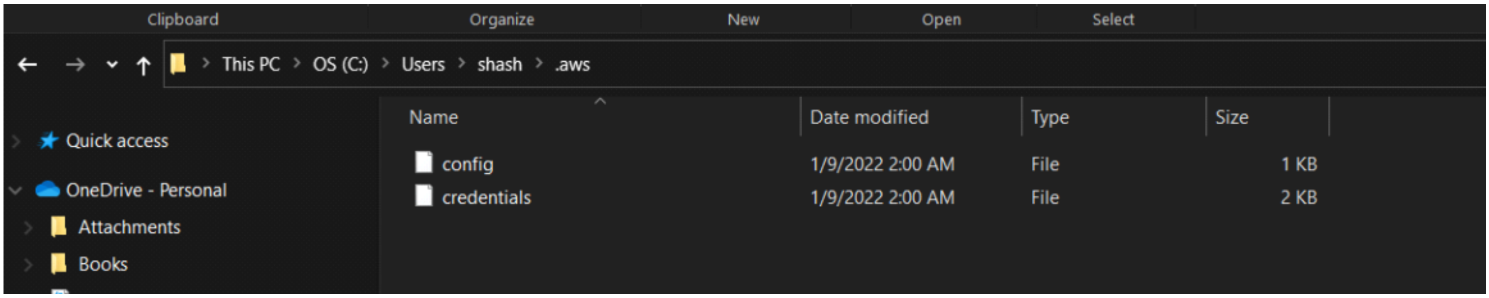


This PC › DATA (D:) › resources › Amazon

Name	Date modified	Type	Size
aws	1/9/2022 1:51 AM	File folder	
Shashank-IAM_accessKeys.csv	1/9/2022 1:48 AM	Microsoft Excel Co...	1 KB

Screen clipping taken: 1/9/2022 2:04 AM

Credentials & Config file location:

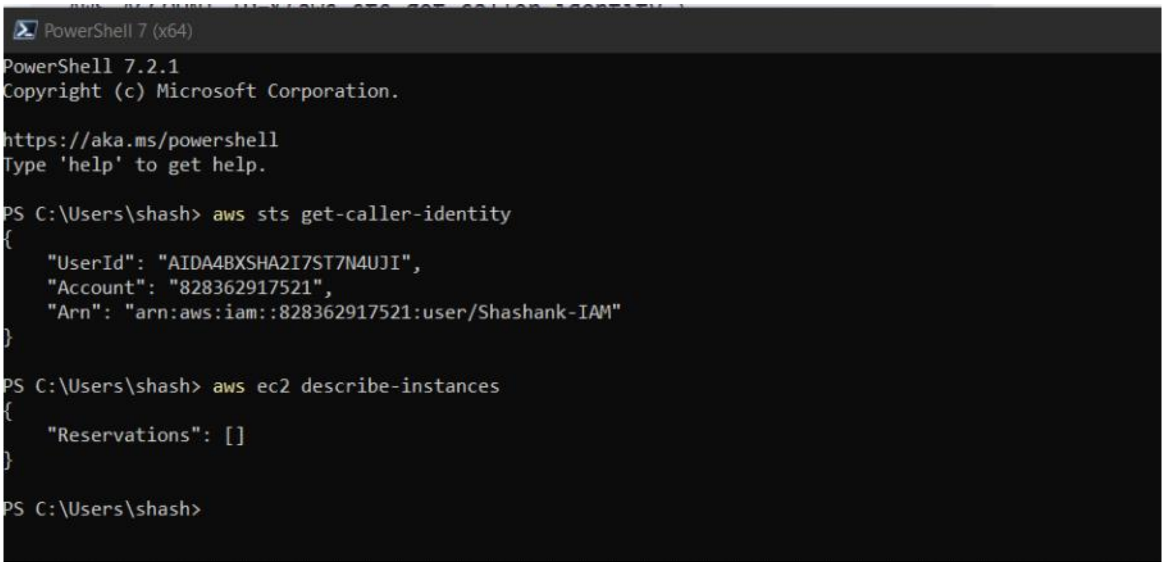


This PC › OS (C:) › Users › shash › .aws

Name	Date modified	Type	Size
config	1/9/2022 2:00 AM	File	1 KB
credentials	1/9/2022 2:00 AM	File	2 KB

Screen clipping taken: 1/9/2022 1:21 PM

Verify CLI: *Use AWS help for the services available; press space*



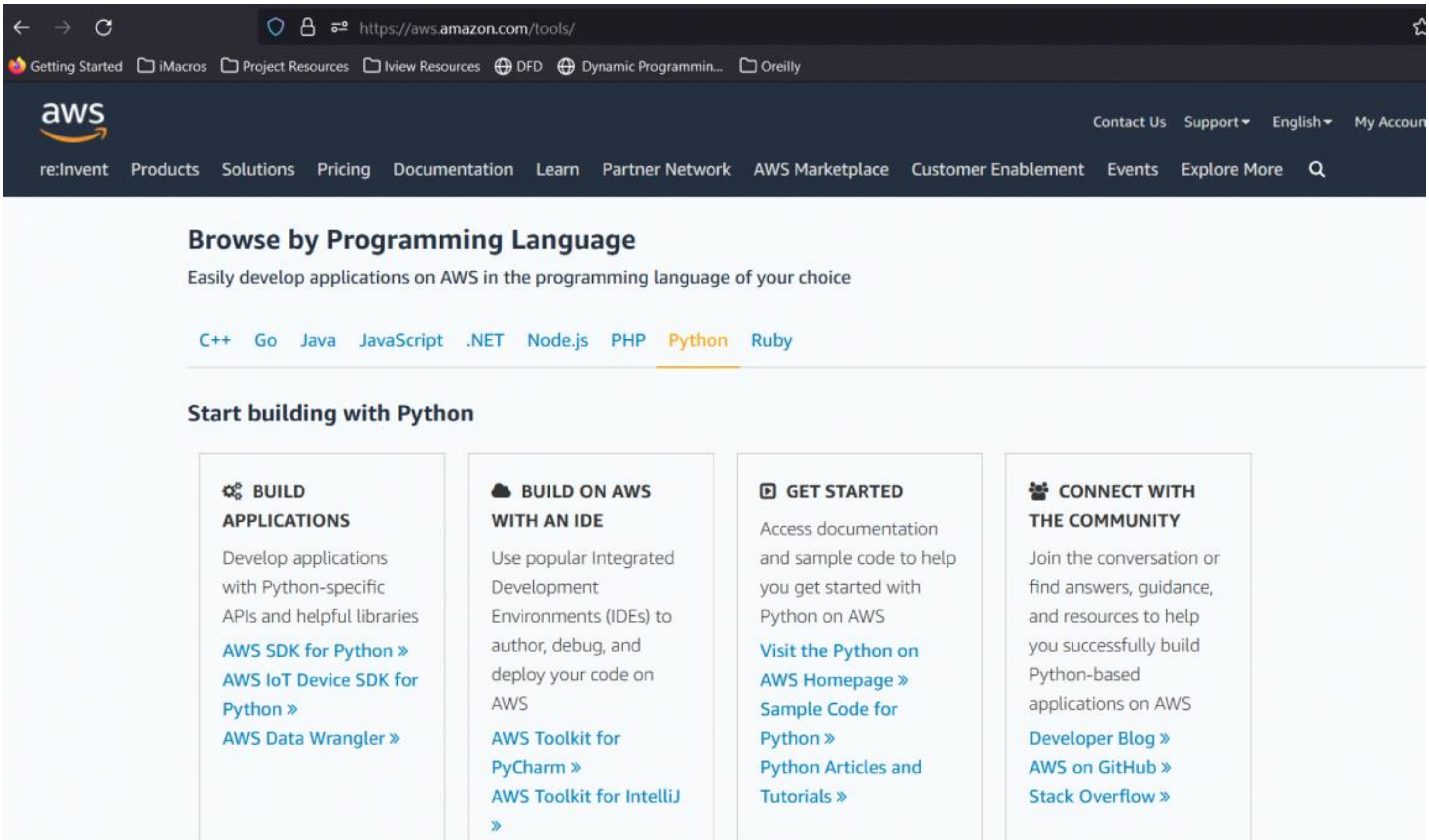
```
PowerShell 7.2.1
Copyright (c) Microsoft Corporation.

https://aka.ms/powershell
Type 'help' to get help.

PS C:\Users\shash> aws sts get-caller-identity
{
    "UserId": "AIDA4BXSHA2I7ST7N4UJI",
    "Account": "828362917521",
    "Arn": "arn:aws:iam::828362917521:user/Shashank-IAM"
}

PS C:\Users\shash> aws ec2 describe-instances
{
    "Reservations": []
}

PS C:\Users\shash>
```

Screen clipping taken: 1/9/2022 2:40 AM

SDK

Sunday, January 9, 2022 2:28 PM

AWS SDKs: https://aws.amazon.com/tools/



Browse by Programming Language

Easily develop applications on AWS in the programming language of your choice

C++ Go Java JavaScript .NET Node.js PHP Python Ruby

Start building with Python

BUILD APPLICATIONS

Develop applications with Python-specific APIs and helpful libraries

AWS SDK for Python »
AWS IoT Device SDK for Python »
AWS Data Wrangler »

BUILD ON AWS WITH AN IDE

Use popular Integrated Development Environments (IDEs) to author, debug, and deploy your code on AWS

AWS Toolkit for PyCharm »
AWS Toolkit for IntelliJ »

GET STARTED

Access documentation and sample code to help you get started with Python on AWS

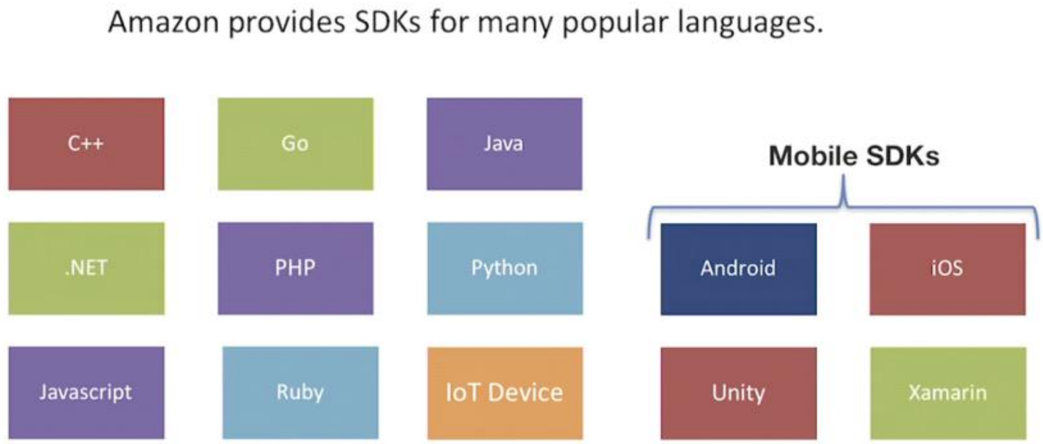
Visit the Python on AWS Homepage »
Sample Code for Python »
Python Articles and Tutorials »

CONNECT WITH THE COMMUNITY

Join the conversation or find answers, guidance, and resources to help you successfully build Python-based applications on AWS

Developer Blog »
AWS on GitHub »
Stack Overflow »

Screen clipping taken: 1/9/2022 2:30 PM



Amazon provides SDKs for many popular languages.

C++ Go Java

.NET PHP Python

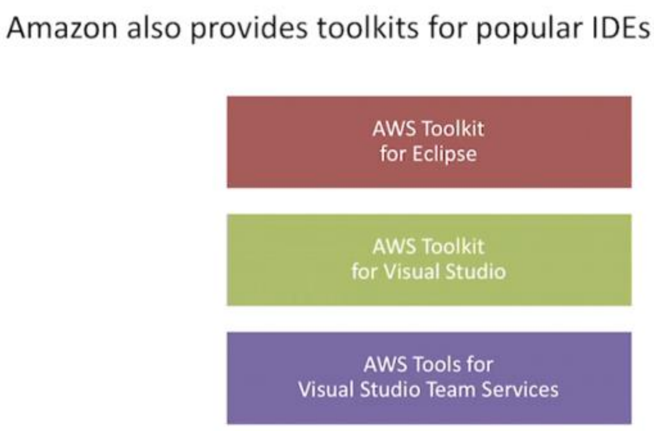
Javascript Ruby IoT Device

Mobile SDKs

Android iOS

Unity Xamarin

Screen clipping taken: 1/9/2022 1:47 PM



Amazon also provides toolkits for popular IDEs

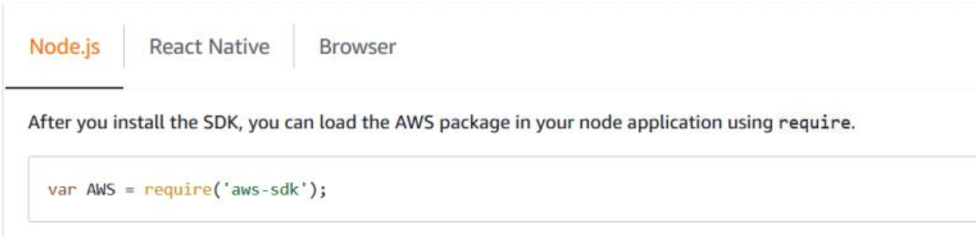
AWS Toolkit for Eclipse
AWS Toolkit for Visual Studio
AWS Tools for Visual Studio Team Services

Screen clipping taken: 1/9/2022 1:48 PM

key points with respect to the SDKs. Each one is developed independently. And as such, not all SDKs are going to offer access to the same services.

From <https://learning.oreilly.com/learning-paths/learning-path-aws/9780135944899/9780134855158-ACDA_02_04_02/>

Node.js: https://docs.aws.amazon.com/sdk-for-javascript/v2/developer-guide/installing-jssdk.html

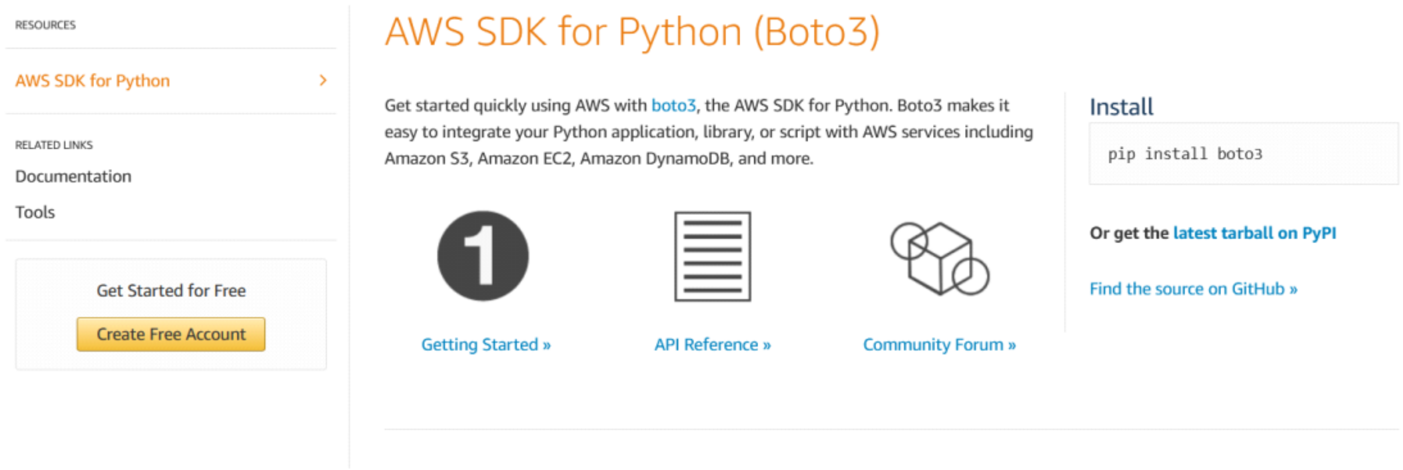


After you install the SDK, you can load the AWS package in your node application using `require`.

```
var AWS = require('aws-sdk');
```

Screen clipping taken: 1/9/2022 2:37 PM

Python: https://aws.amazon.com/developer/language/python/



AWS SDK for Python (Boto3)

Get started quickly using AWS with boto3, the AWS SDK for Python. Boto3 makes it easy to integrate your Python application, library, or script with AWS services including Amazon S3, Amazon EC2, Amazon DynamoDB, and more.

Install

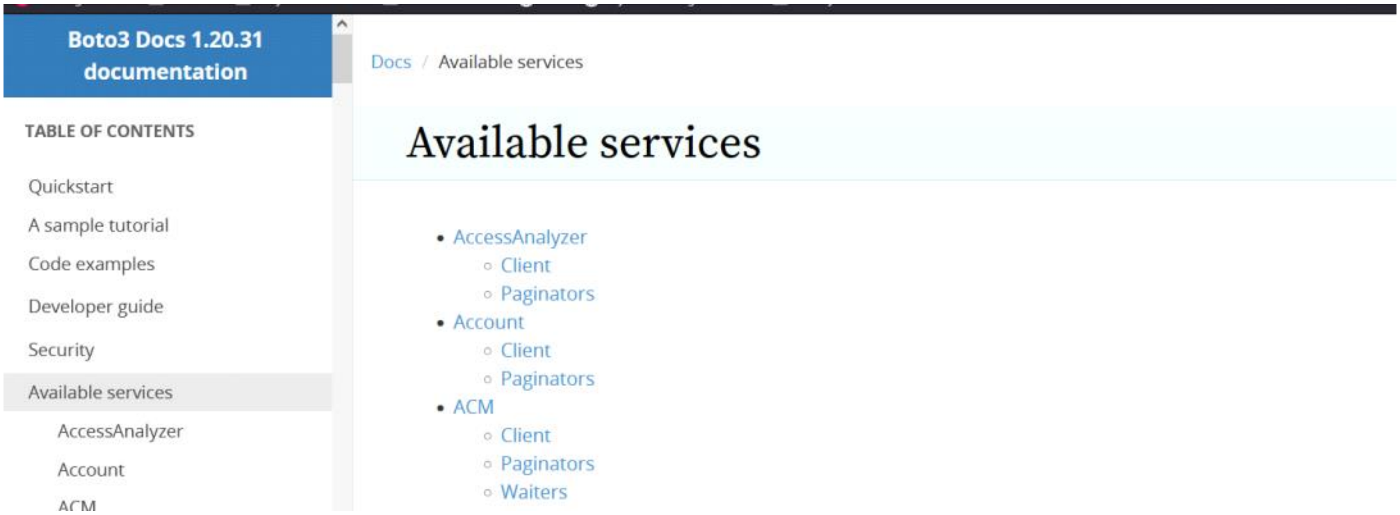
```
pip install boto3
```

Or get the latest tarball on PyPI

Find the source on GitHub »

Screen clipping taken: 1/9/2022 2:39 PM

API References:
https://boto3.amazonaws.com/v1/documentation/api/latest/reference/services/index.html



Boto3 Docs 1.20.31 documentation

TABLE OF CONTENTS

Quickstart

A sample tutorial

Code examples

Developer guide

Security

Available services

AccessAnalyzer

Account

ACM

Docs / Available services

Available services

- AccessAnalyzer
    - Client
    - Paginators
- Account
    - Client
    - Paginators
- ACM
    - Client
    - Paginators
    - Waiters

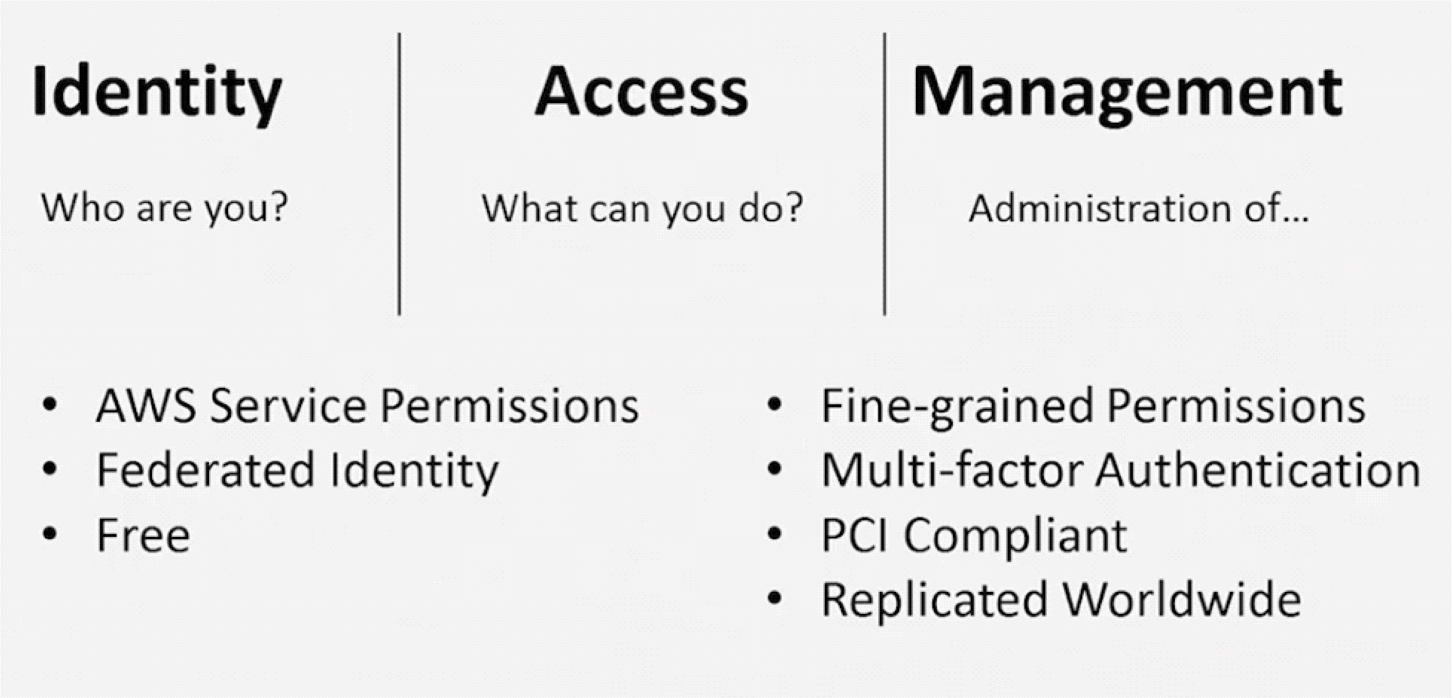
Screen clipping taken: 1/9/2022 2:40 PM

IAM Overview

Sunday, January 9, 2022 3:32 PM

IAM Overview:

Identity and Access Management. A core component of every AWS service is security. Not just restricting and permitting access at the network level, we also have to address authentication. The IAM service provides the means to create user accounts that can be used for either interactive or programmatic access to AWS services



Identity | **Access** | **Management**

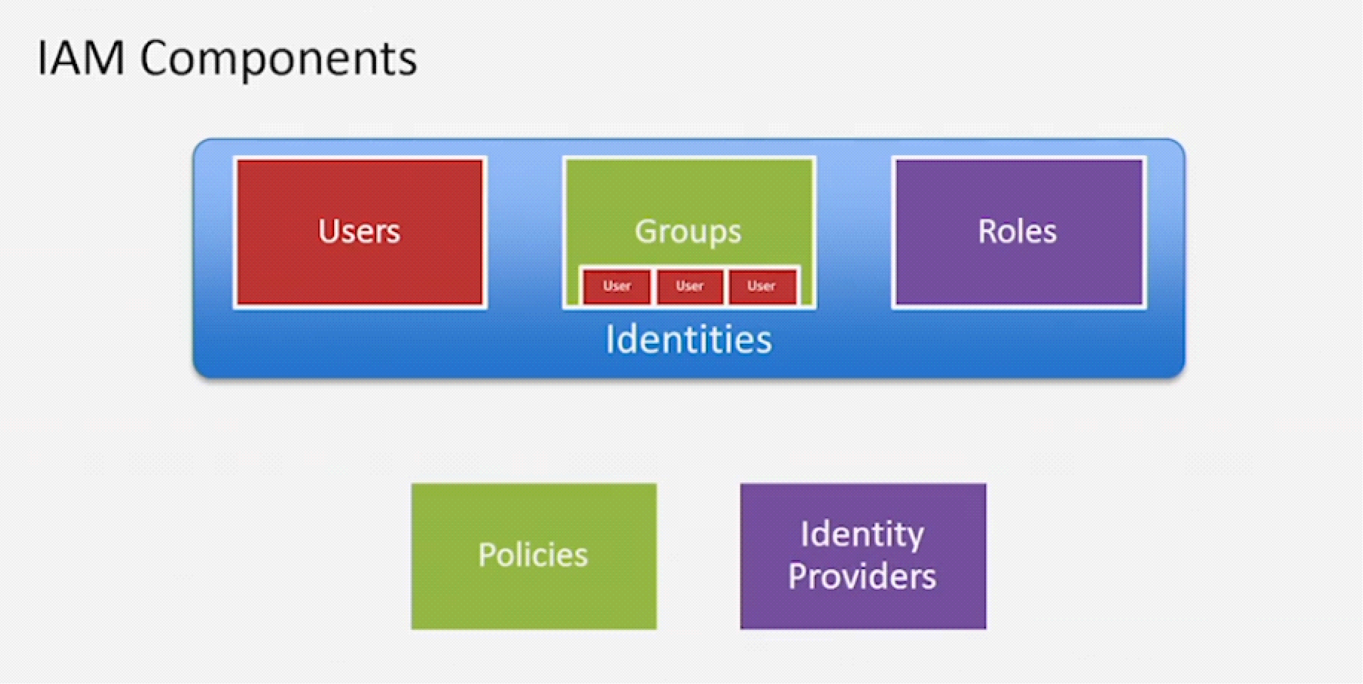
Who are you? | What can you do? | Administration of...

- AWS Service Permissions
- Federated Identity
- Free

- Fine-grained Permissions
- Multi-factor Authentication
- PCI Compliant
- Replicated Worldwide

Screen clipping taken: 1/9/2022 3:41 PM

IAM Components:



IAM Components

Users

Groups

User User User

Roles

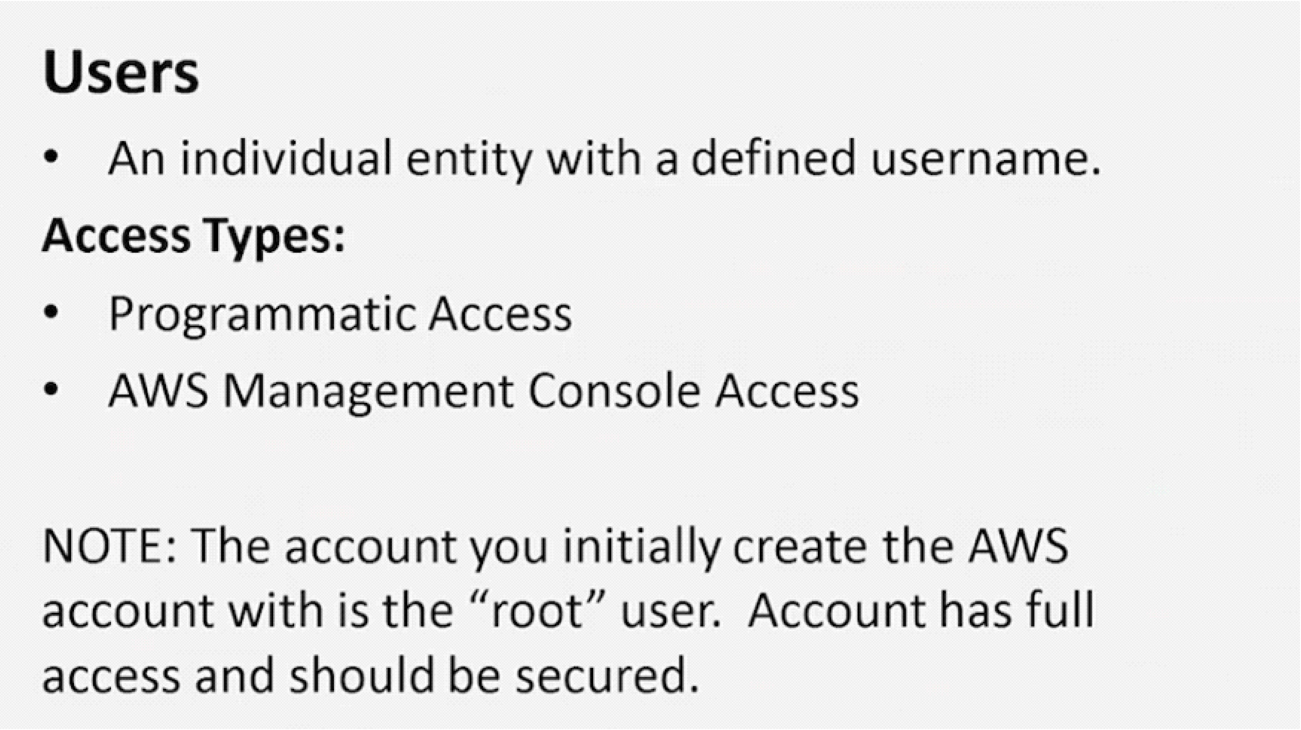
Identities

Policies

Identity Providers

Screen clipping taken: 1/9/2022 11:32 PM

User: An Individual entity with a defined username. Better is to not use root user



Users

- An individual entity with a defined username.

Access Types:

- Programmatic Access
- AWS Management Console Access

NOTE: The account you initially create the AWS account with is the “root” user. Account has full access and should be secured.

Screen clipping taken: 1/9/2022 11:33 PM

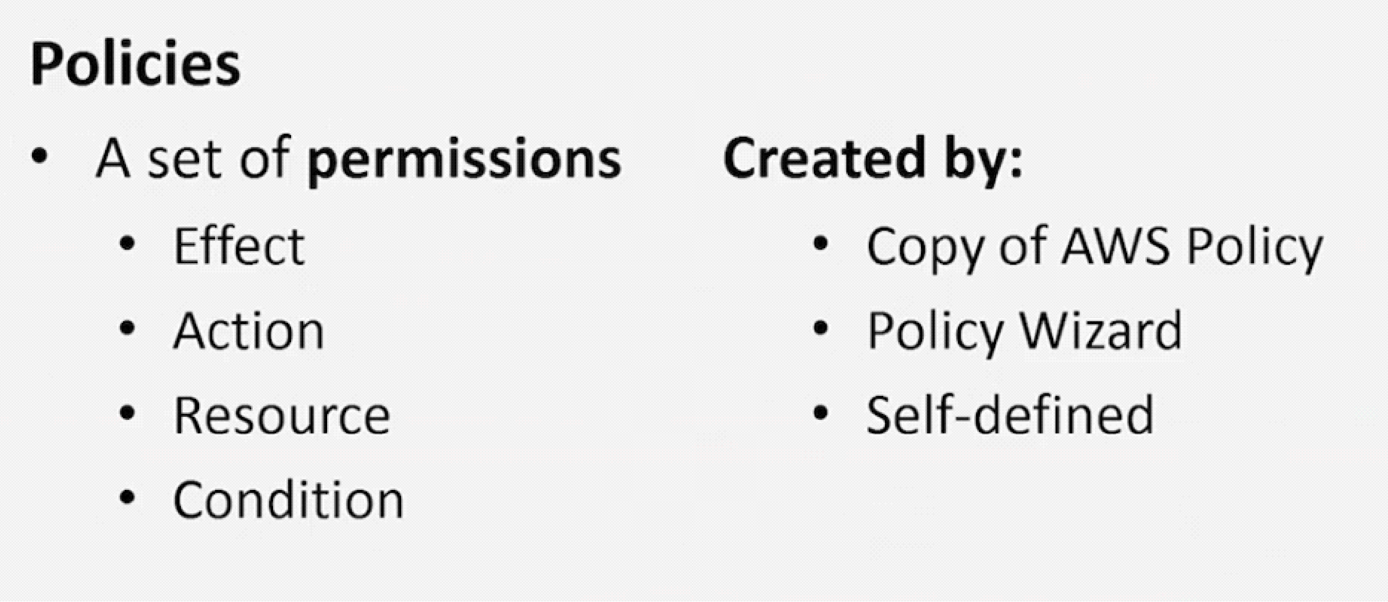
Access type:

- Programmatic Access: Secrets
- AWS Management Console

Policy: A set of permissions

- Effect
- Action

- Resource
- Condition



Policies

- A set of **permissions**
  - Effect
  - Action
  - Resource
  - Condition

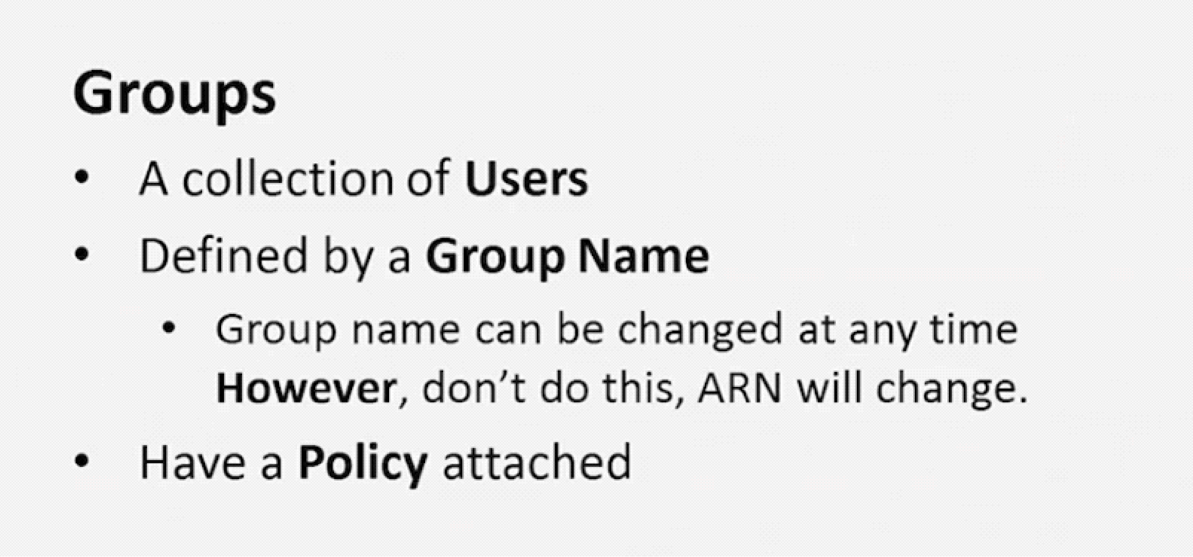
Created by:

- Copy of AWS Policy
- Policy Wizard
- Self-defined

Screen clipping taken: 1/9/2022 11:39 PM

Groups:

- Collation of users.
- Defined unique group name, Once defined do not change(it will change the ARN).
- Have a policy Attached to it. A user in a group inherits policies of a group and can override its own policy, be different than a group.

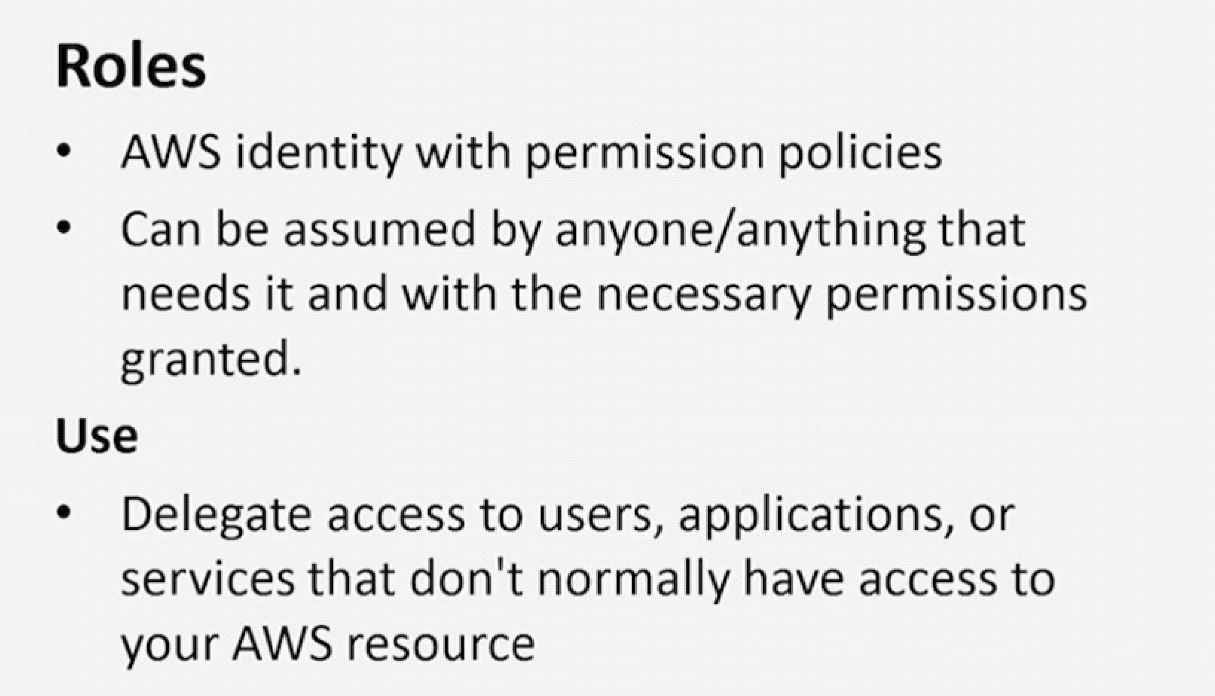


Groups

- A collection of **Users**
- Defined by a **Group Name**
  - Group name can be changed at any time **However**, don’t do this, ARN will change.
- Have a **Policy** attached

Screen clipping taken: 1/9/2022 11:42 PM

Roles: An identity associated with permission policies, used to delegate access to users, application or services
Which normally don’t have access to AWS services



Roles

- AWS identity with permission policies
- Can be assumed by anyone/anything that needs it and with the necessary permissions granted.

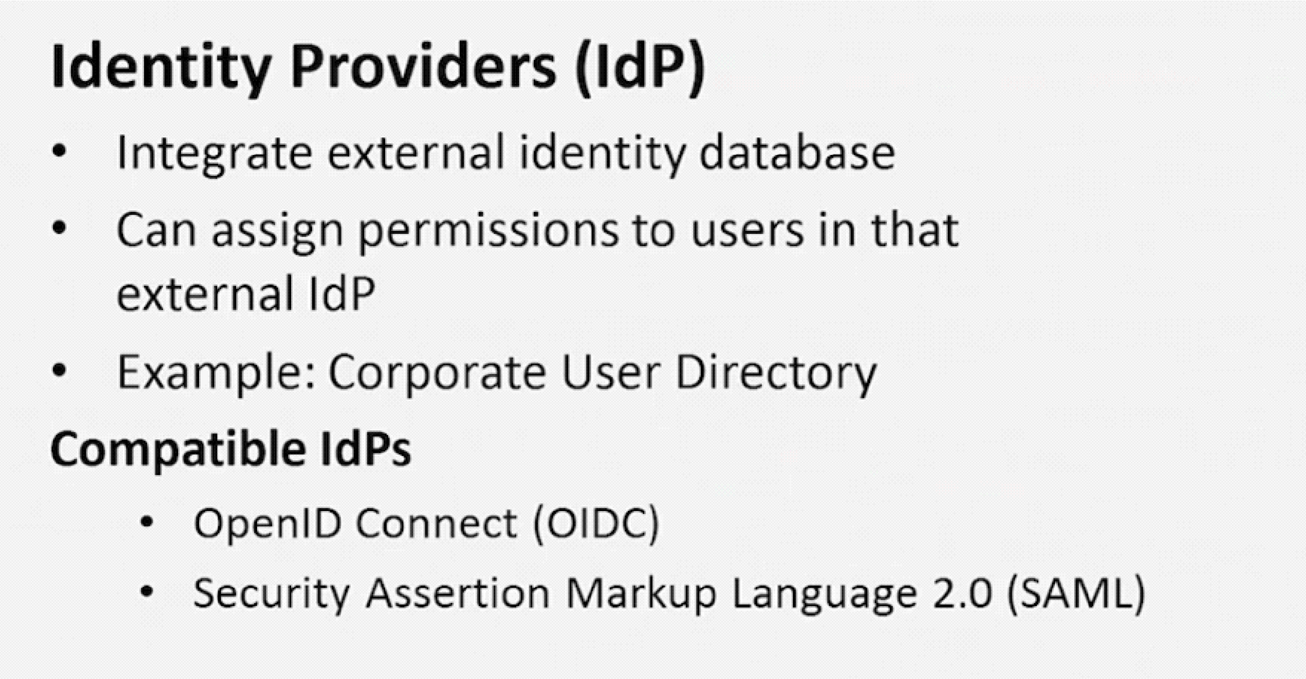
Use

- Delegate access to users, applications, or services that don't normally have access to your AWS resource

Screen clipping taken: 1/9/2022 11:45 PM

Identity Providers:

- Integrates external identity database
- Can assign permissions to users in that external IdP



Identity Providers (IdP)

- Integrate external identity database
- Can assign permissions to users in that external IdP
- Example: Corporate User Directory

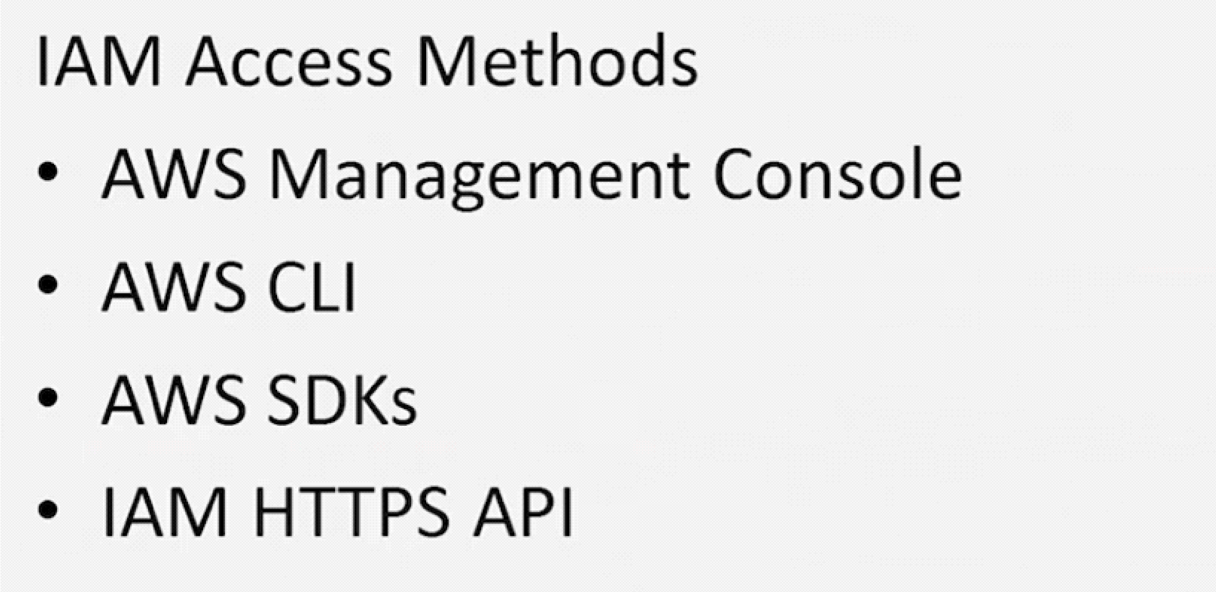
Compatible IdPs

- OpenID Connect (OIDC)
- Security Assertion Markup Language 2.0 (SAML)

Screen clipping taken: 1/10/2022 12:00 AM

IAM Access Methods:

- AWS Management Console
- AWS CLI
- AWS SDKs
- IAM HTTPS API



IAM Access Methods

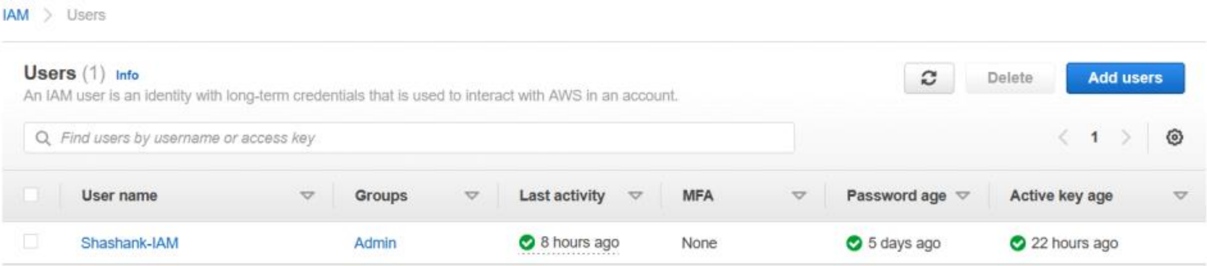
- AWS Management Console
- AWS CLI
- AWS SDKs
- IAM HTTPS API

Screen clipping taken: 1/10/2022 12:06 AM

IAM Components Console

Monday, January 10, 2022 12:09 AM

Create User:

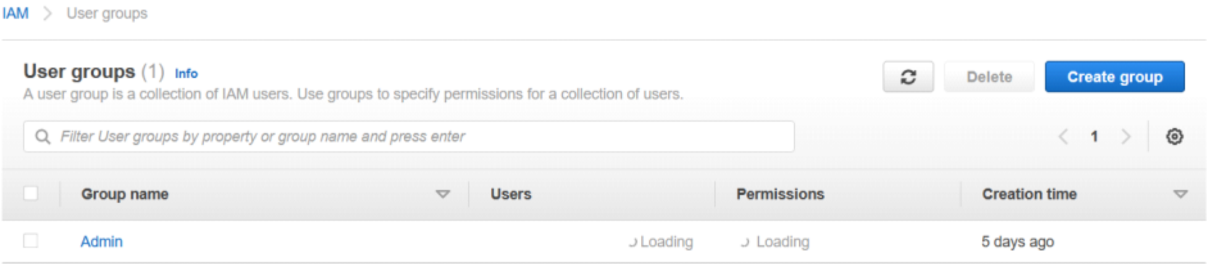


An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

User name	Groups	Last activity	MFA	Password age	Active key age
Shashank-IAM	Admin	✓ 8 hours ago	None	✓ 5 days ago	✓ 22 hours ago

Screen clipping taken: 1/10/2022 12:10 AM

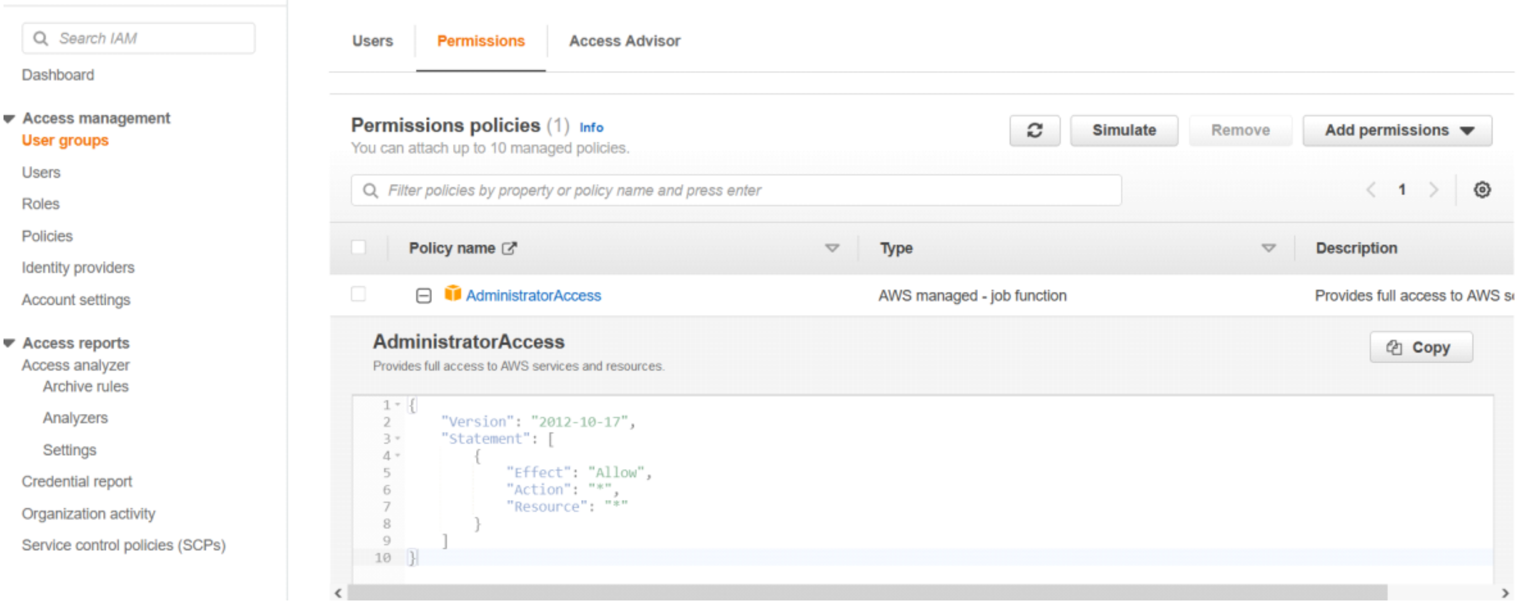
Create User Group:


IAM > User groups
User groups (1) Info
Delete
Create group
Filter User groups by property or group name and press enter
1

A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.

Group name	Users	Permissions	Creation time
Admin	Loading	Loading	5 days ago

Screen clipping taken: 1/10/2022 12:11 AM



You can attach up to 10 managed policies.

Policy name	Type	Description
AdministratorAccess	AWS managed - job function	Provides full access to AWS s

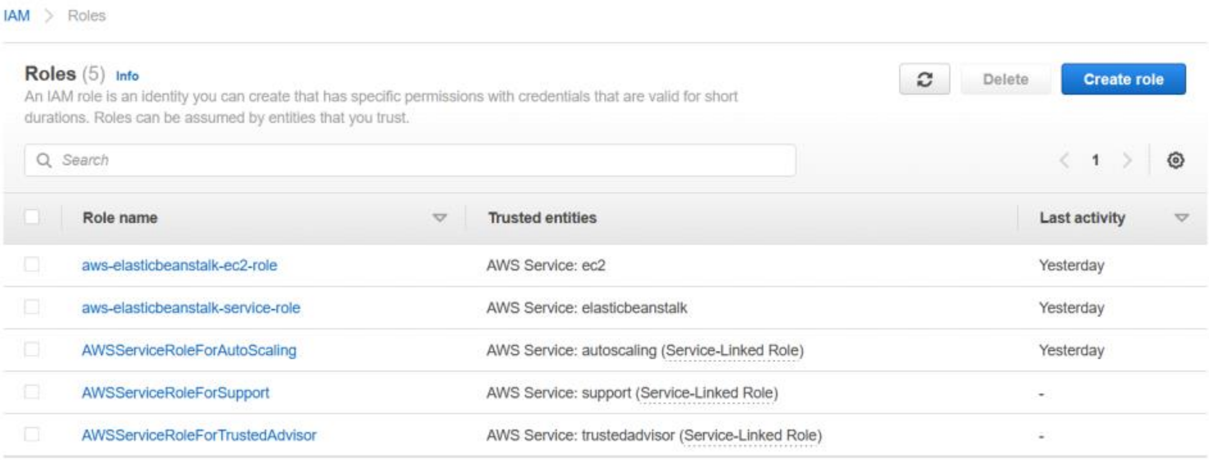
AdministratorAccess

Provides full access to AWS services and resources.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Screen clipping taken: 1/10/2022 12:26 AM

Create Roles:



IAM > Roles

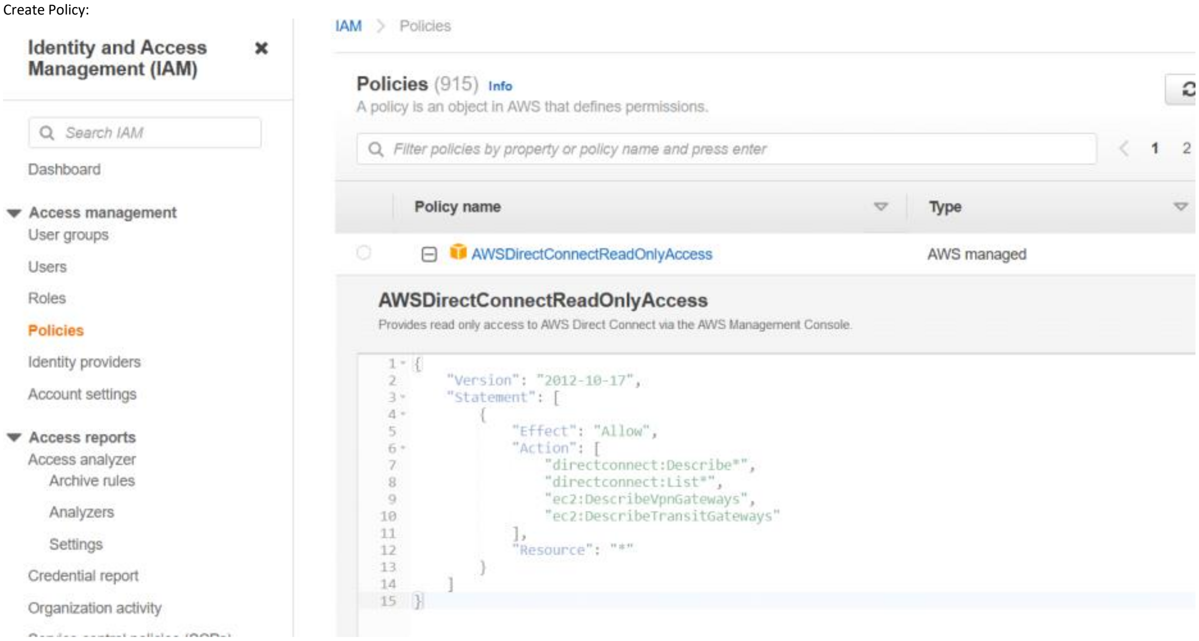
Roles (5) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name	Trusted entities	Last activity
aws-elasticbeanstalk-ec2-role	AWS Service: ec2	Yesterday
aws-elasticbeanstalk-service-role	AWS Service: elasticbeanstalk	Yesterday
AWSServiceRoleForAutoScaling	AWS Service: autoscaling (Service-Linked Role)	Yesterday
AWSServiceRoleForSupport	AWS Service: support (Service-Linked Role)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service-Linked Role)	-

Screen clipping taken: 1/10/2022 12:11 AM

Create Policy:



Policies (915) Info

A policy is an object in AWS that defines permissions.

Policy name	Type
AWSDirectConnectReadOnlyAccess	AWS managed

AWSDirectConnectReadOnlyAccess

Provides read only access to AWS Direct Connect via the AWS Management Console.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "directconnect:Describe*",
                "directconnect:List*",
                "ec2:DescribeVpnGateways",
                "ec2:DescribeTransitGateways"
            ],
            "Resource": "*"
        }
    ]
}
```

Screen clipping taken: 1/10/2022 12:12 AM

Version: Version of the IAM API
Policies: Don't use "*", it means allows all

IAM Components CLI

Monday, January 10, 2022 1:08 AM

1. Create User:

```
   PS C:\Users\shash> aws iam create-user --user-name shashank-exmp
   {
      "User": {
         "Path": "/",
         "UserName": "shashank-exmp",
         "UserId": "AIDA4BXSHA2I7JGVU3IO3",
         "Arn": "arn:aws:iam::828362917521:user/shashank-exmp",
         "CreateDate": "2022-01-09T19:05:06+00:00"
      }
   }
```

2. Create Login profile and password:

```
   PS C:\Users\shash> aws iam create-login-profile --user-name shashank-exmp --password
Password123 --password-reset-required
   {
      "LoginProfile": {
         "UserName": "shashank-exmp",
         "CreateDate": "2022-01-09T19:11:27+00:00",
         "PasswordResetRequired": true
      }
   }
```

3. Get Arn: --query and --output

   Query: James Path queries are case sensitive, won't throw an error but will return null,
   - Strings are mentioned in backtick,
   - Query in an apostrophe
   - '|' grep to iterate over query output, case sensitive

   Output: text

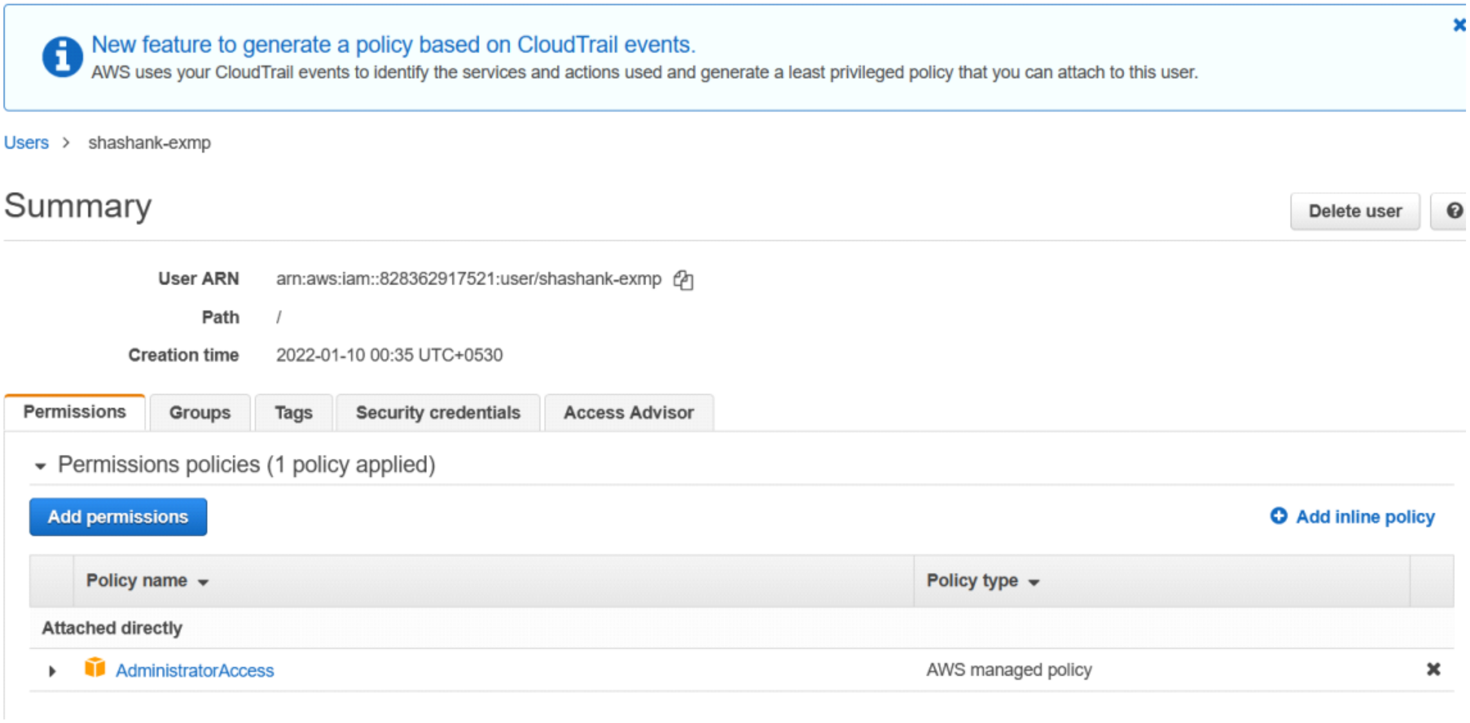
```
   PS C:\Users\shash> aws iam list-policies --query 'Policies[?contains(PolicyName `AdministratorAccess`)
== `true`] | [*].[PolicyName, Arn]' --output text
   AdministratorAccess    arn:aws:iam::aws:policy/AdministratorAccess
   AdministratorAccess-Amplify    arn:aws:iam::aws:policy/AdministratorAccess-Amplify
   AdministratorAccess-AWSElasticBeanstalk arn:aws:iam::aws:policy/AdministratorAccess-
AWSElasticBeanstalk
   AWSAuditManagerAdministratorAccess
arn:aws:iam::aws:policy/AWSAuditManagerAdministratorAccess
```

4. Attach policy to user: string is in an apostrophe

```
   PS C:\Users\shash> aws iam attach-user-policy --user-name shashank-exmp --policy-arn
'arn:aws:iam::aws:policy/AdministratorAccess'
```

   No Output: means no errors

   Verified in Console; Policy Attached



New feature to generate a policy based on CloudTrail events.

AWS uses your CloudTrail events to identify the services and actions used and generate a least privileged policy that you can attach to this user.

Users > shashank-exmp

Summary

User ARN	arn:aws:iam::828362917521:user/shashank-exmp
Path	/
Creation time	2022-01-10 00:35 UTC+0530

Permissions policies (1 policy applied)

Policy name	Policy type
Attached directly	
AdministratorAccess	AWS managed policy

Screen clipping taken: 1/10/2022 1:09 AM

5. Create Group:

```
  PS C:\Users\shash> aws iam create-group --group-name secondary
  {
      "Group": {
          "Path": "/",
          "GroupName": "secondary",
          "GroupId": "AGPA4BXSHA2IVV7U6PVC2",
          "Arn": "arn:aws:iam::828362917521:group/secondary",
          "CreateDate": "2022-01-09T19:58:51+00:00"
      }
  }
```

6. Attach Group Policies:

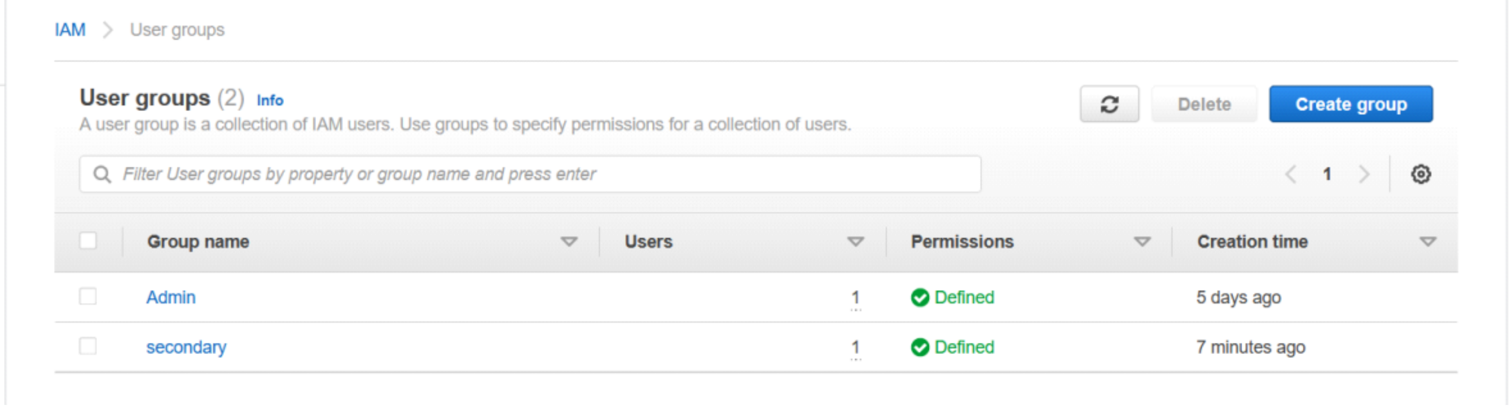
```
PS C:\Users\shash>
aws iam attach-group-policy --group-name secondary --policy-arn
arn:aws:iam::aws:policy/AdministratorAccess'
```

No Output: means no errors
Verified in Console; Policy Attached

7. Add User to group:

```
PS C:\Users\shash> aws iam add-user-to-group --group-name secondary --user-name shashank-exmp
```

No Output: means no errors
Verified in Console; User Attached


IAM > User groups
Info
Delete
Create group
Filter User groups by property or group name and press enter
1

User groups (2)

A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.

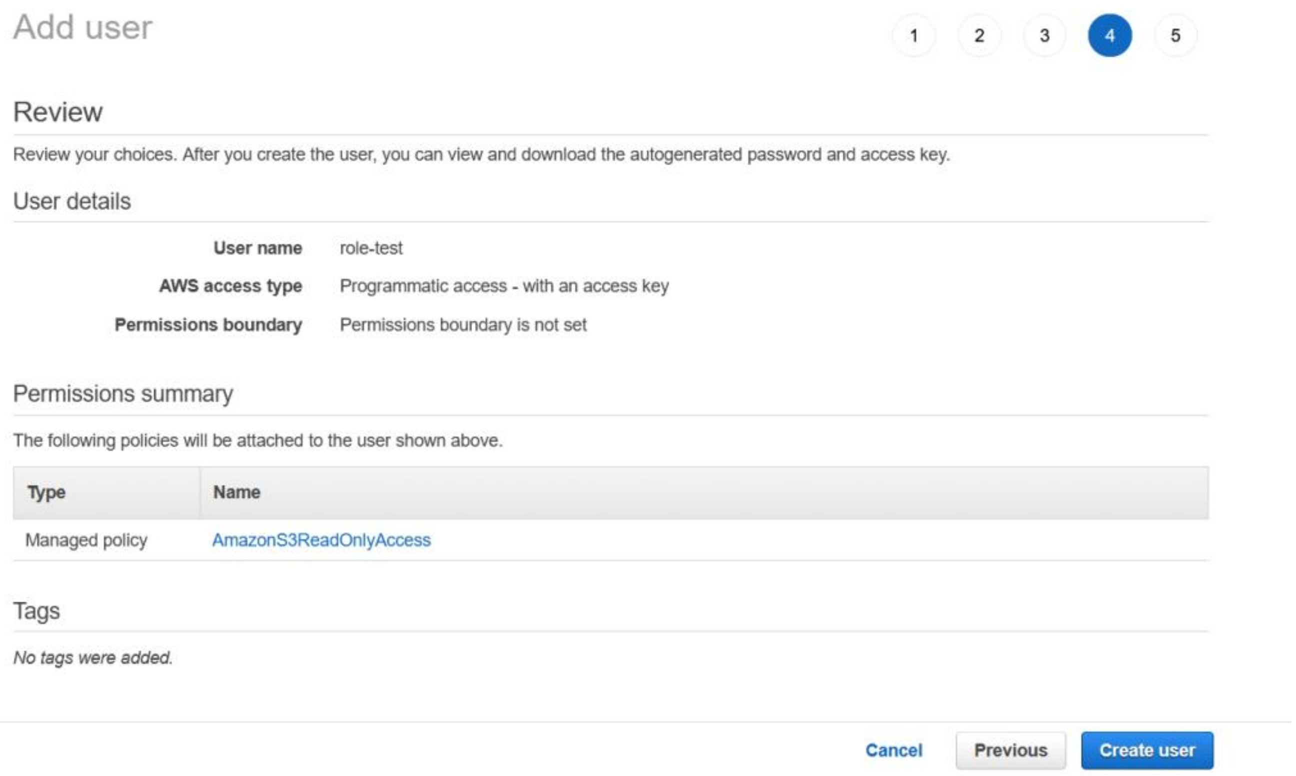
□	Group name	Users	Permissions	Creation time
□	Admin	1	✓ Defined	5 days ago
□	secondary	1	✓ Defined	7 minutes ago

Screen clipping taken: 1/10/2022 1:37 AM

Node APP Test IAM

Monday, January 10, 2022 2:26 AM

Create new User with S3 read only:



Add user

Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name	role-test
AWS access type	Programmatic access - with an access key
Permissions boundary	Permissions boundary is not set

Permissions summary

The following policies will be attached to the user shown above.

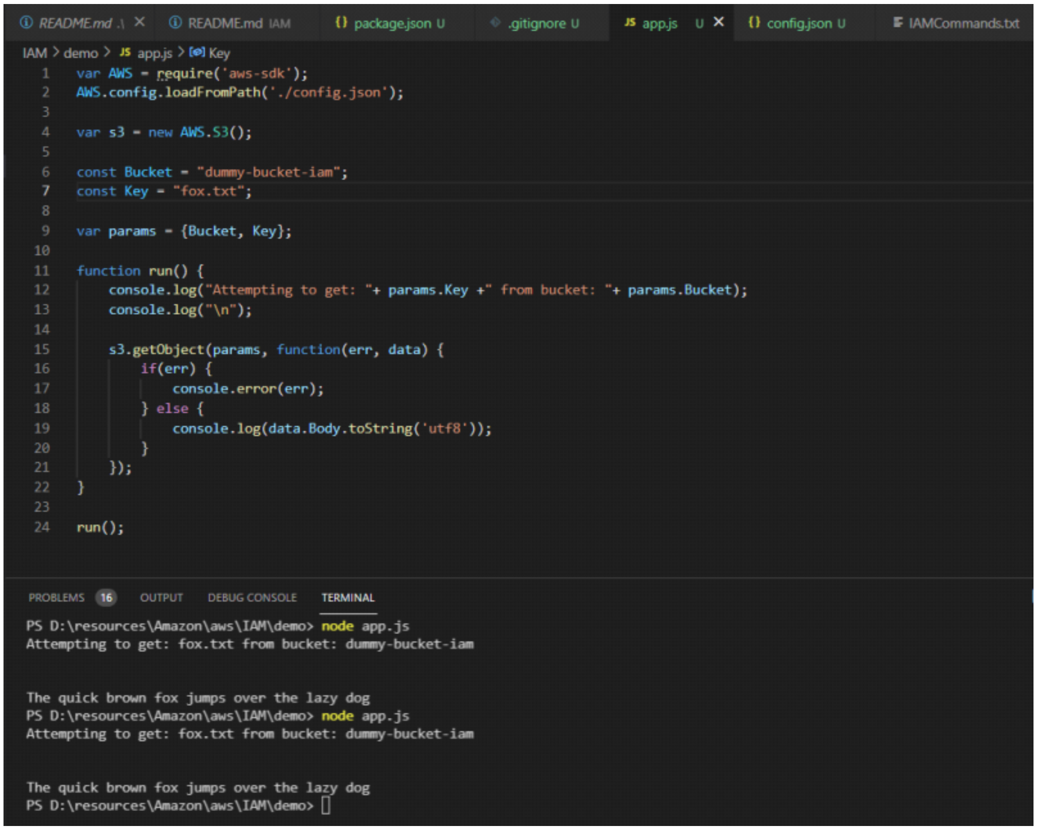
Type	Name
Managed policy	AmazonS3ReadOnlyAccess

Tags

No tags were added.

Screen clipping taken: 1/10/2022 2:27 AM

Able to access:

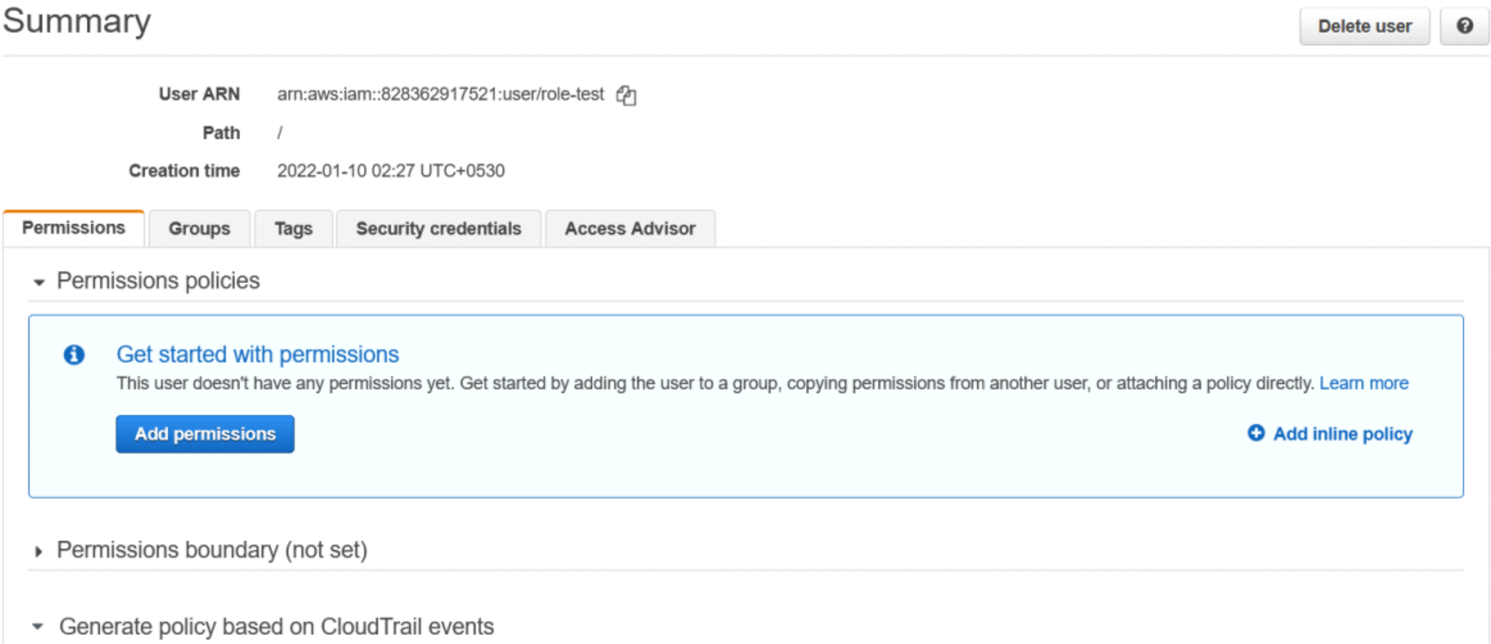


```javascript
var AWS = require('aws-sdk');
AWS.config.loadFromPath('./config.json');

var s3 = new AWS.S3();

const Bucket = "dummy-bucket-iam";
const Key = "fox.txt";

var params = {Bucket, Key};

function run() {
    console.log("Attempting to get: "+ params.Key +" from bucket: "+ params.Bucket);
    console.log("\n");

    s3.getObject(params, function(err, data) {
        if(err) {
            console.error(err);
        } else {
            console.log(data.Body.toString('utf8'));
        }
    });
}

run();
```

```
PS D:\resources\Amazon\aws\IAM\demo> node app.js
Attempting to get: fox.txt from bucket: dummy-bucket-iam


The quick brown fox jumps over the lazy dog
PS D:\resources\Amazon\aws\IAM\demo> node app.js
Attempting to get: fox.txt from bucket: dummy-bucket-iam


The quick brown fox jumps over the lazy dog
PS D:\resources\Amazon\aws\IAM\demo> 
```

Screen clipping taken: 1/10/2022 2:29 AM

Removing Policy:



arn:aws:iam::828362917521:user/role-test

/

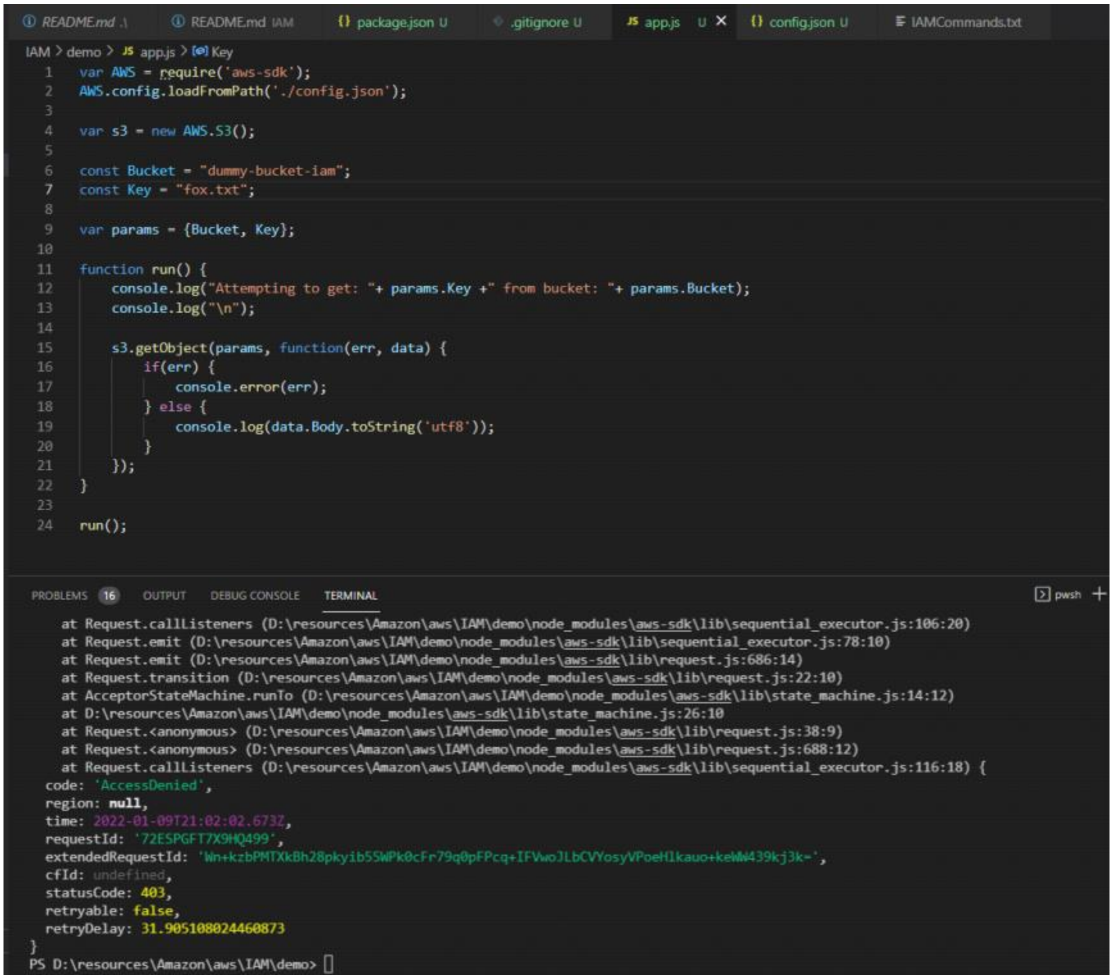
2022-01-10 02:27 UTC+0530

Get started with permissions

This user doesn't have any permissions yet. Get started by adding the user to a group, copying permissions from another user, or attaching a policy directly. Learn more

Screen clipping taken: 1/10/2022 2:31 AM

Access denied:



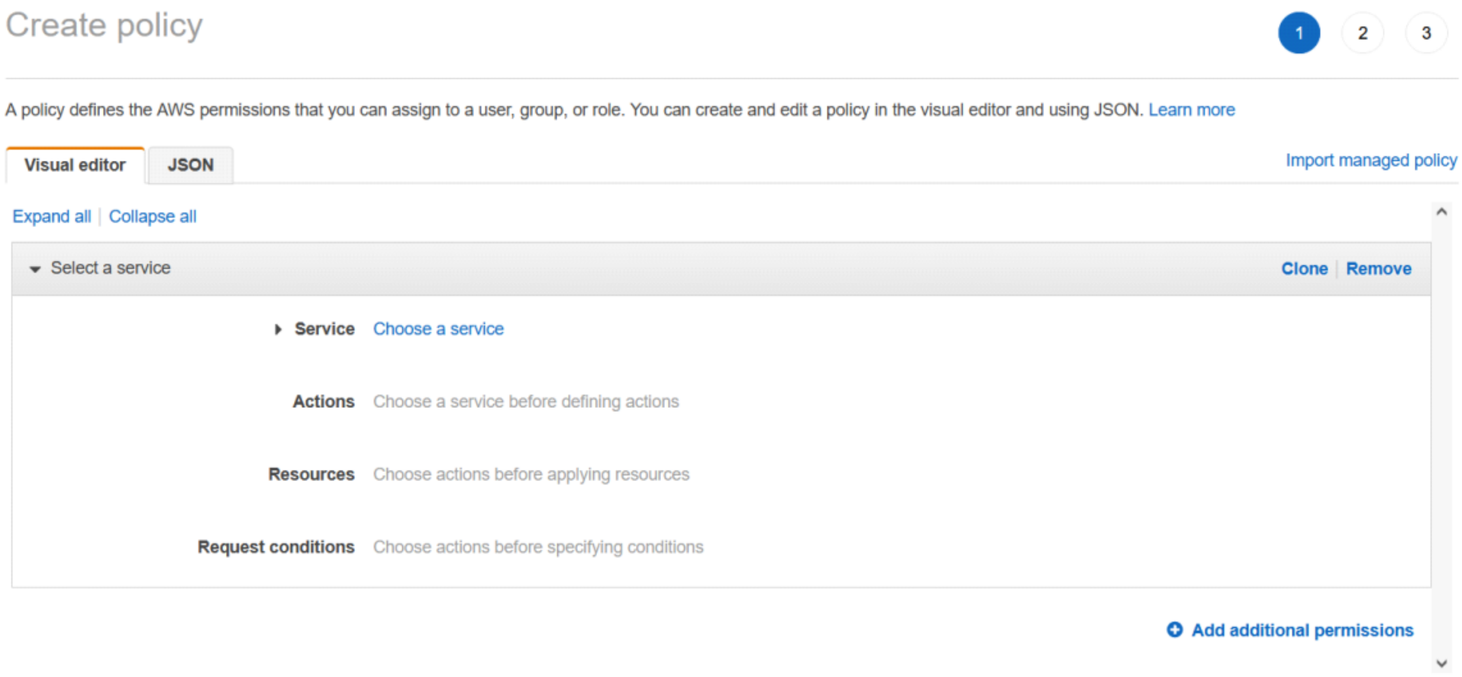
```
var AWS = require('aws-sdk');
AWS.config.loadFromPath('./config.json');

var s3 = new AWS.S3();

const Bucket = "dummy-bucket-iam";
const Key = "fox.txt";

var params = {Bucket, Key};

function run() {
    console.log("Attempting to get: "+ params.Key +" from bucket: "+ params.Bucket);
    console.log("\n");

    s3.getObject(params, function(err, data) {
        if(err) {
            console.error(err);
        } else {
            console.log(data.Body.toString('utf8'));
        }
    });
}

run();
```

```
    at Request.callListeners (D:\resources\Amazon\aws\IAM\demo\node_modules\aws-sdk\lib\sequential_executor.js:106:20)
    at Request.emit (D:\resources\Amazon\aws\IAM\demo\node_modules\aws-sdk\lib\sequential_executor.js:78:10)
    at Request.emit (D:\resources\Amazon\aws\IAM\demo\node_modules\aws-sdk\lib\request.js:686:14)
    at Request.transition (D:\resources\Amazon\aws\IAM\demo\node_modules\aws-sdk\lib\request.js:22:10)
    at AcceptorStateMachine.runTo (D:\resources\Amazon\aws\IAM\demo\node_modules\aws-sdk\lib\state_machine.js:14:12)
    at D:\resources\Amazon\aws\IAM\demo\node_modules\aws-sdk\lib\state_machine.js:26:10
    at Request.<anonymous> (D:\resources\Amazon\aws\IAM\demo\node_modules\aws-sdk\lib\request.js:38:9)
    at Request.<anonymous> (D:\resources\Amazon\aws\IAM\demo\node_modules\aws-sdk\lib\request.js:688:12)
    at Request.callListeners (D:\resources\Amazon\aws\IAM\demo\node_modules\aws-sdk\lib\sequential_executor.js:116:18) {
  code: 'AccessDenied',
  region: null,
  time: 2022-01-09T21:02:02.673Z,
  requestId: '72ESPGFT7X9HQ499',
  extendedRequestId: 'Wn+kzbPMTXkBh28pkyib55WPk0cFr79q0pFPcq+IFVwoJLbCVYosyVPoeHlkauo+keWW439kj3k=',
  cfId: undefined,
  statusCode: 403,
  retryable: false,
  retryDelay: 31.905108024460873
}
PS D:\resources\Amazon\aws\IAM\demo>
```

Screen clipping taken: 1/10/2022 2:32 AM

Create Custom Policy

Monday, January 10, 2022 2:50 AM



Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. Learn more

Select a service

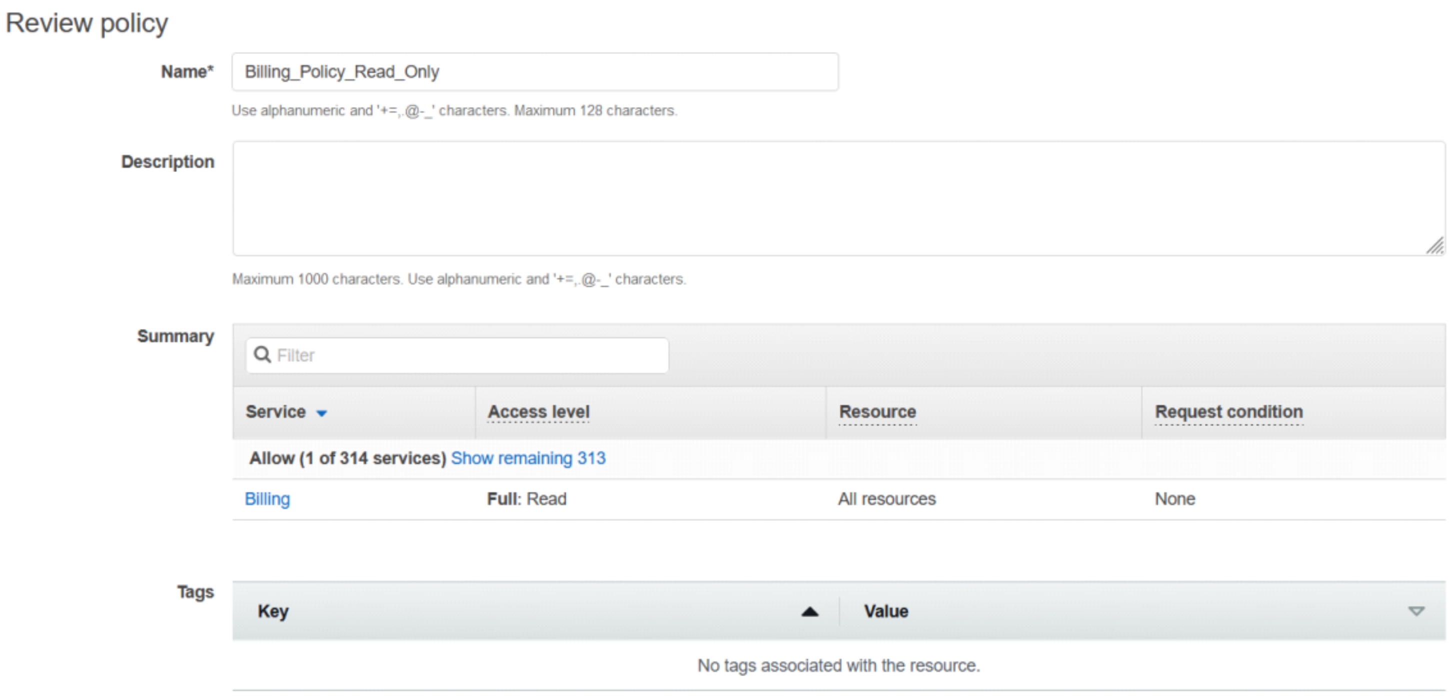
Service Choose a service

Actions Choose a service before defining actions

Resources Choose actions before applying resources

Request conditions Choose actions before specifying conditions

Screen clipping taken: 1/10/2022 2:51 AM



Review policy

Name* Billing_Policy_Read_Only

Use alphanumeric and '+=,.@-_' characters. Maximum 128 characters.

Description

Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.

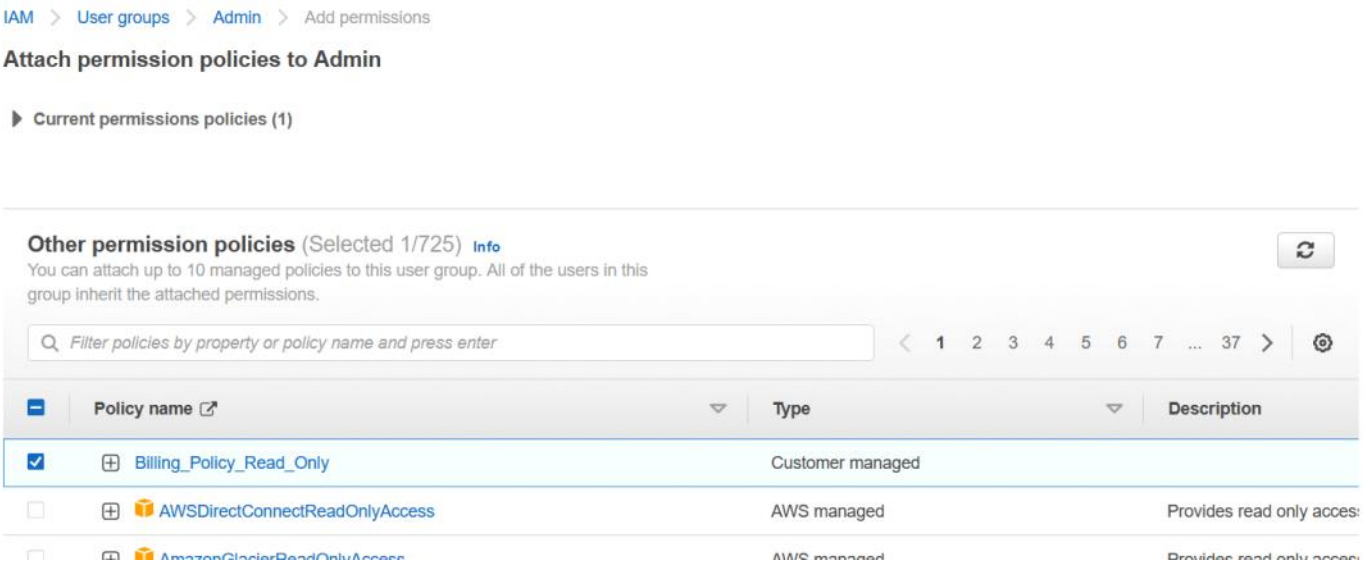
Summary

Service	Access level	Resource	Request condition
Allow (1 of 314 services) Show remaining 313			
Billing	**Full:** Read	All resources	None

Tags

Key	Value
No tags associated with the resource.	

Screen clipping taken: 1/10/2022 2:53 AM



IAM > User groups > Admin > Add permissions

Attach permission policies to Admin

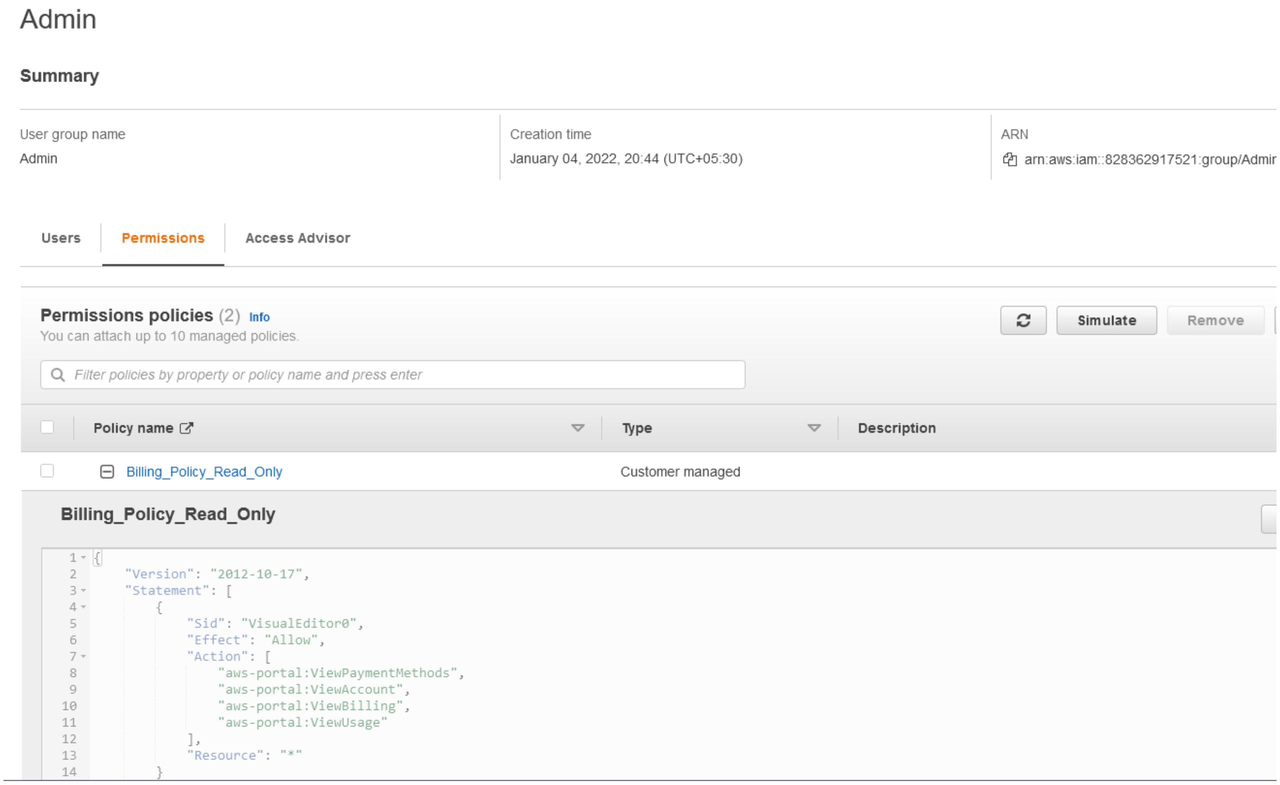
▶ Current permissions policies (1)

Other permission policies (Selected 1/725) Info

You can attach up to 10 managed policies to this user group. All of the users in this group inherit the attached permissions.

☐	Policy name	Type	Description
☑	⊞ Billing_Policy_Read_Only	Customer managed	
☐	⊞ AWSDirectConnectReadOnlyAccess	AWS managed	Provides read only acces
☐	⊞ AmazonGlacierReadOnlyAccess	AWS managed	Provides read only acces

Screen clipping taken: 1/10/2022 2:54 AM



Admin

Summary

User group name	Creation time	ARN
Admin	January 04, 2022, 20:44 (UTC+05:30)	arn:aws:iam::828362917521:group/Admir

Permissions policies (2) Info

You can attach up to 10 managed policies.

☐	Policy name	Type	Description
☐	⊟ Billing_Policy_Read_Only	Customer managed	

Billing_Policy_Read_Only

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "aws-portal:ViewPaymentMethods",
                "aws-portal:ViewAccount",
                "aws-portal:ViewBilling",
                "aws-portal:ViewUsage"
            ],
            "Resource": "*"
        }
```

Screen clipping taken: 1/10/2022 2:55 AM