

Q1. What is the use of SSO. Show its implementation.

Ans- Single sign-on (SSO) is an authentication scheme that allows a user to log in with a single ID and password to any of several related, yet dependent, software systems.

True single sign-on allows the user to log in once and access services without re-entering authentication factors.

Steps -

- 1) Navigate to the AWS SSO console, and choose AWS accounts from the navigation pane.
- 2) Choose users, start typing to search for users, and then choose search connected directory.
- 3) Choose create new permission set.
- 4) We can use an existing job function policy to create a permission set.
- 5) Choose security Audit Job function policy and choose create.
- 6) Choose a permission set to indicate what level of access you want to grant.
- 7) Then choose finish.

Nirmal Giri

Section - K

Roll no - L918515

## AWS Accounts

You can configure which users and groups in your connected directory have SSO access to AWS accounts in your AWS organization. You manage permission sets to control the users level of access to these AWS accounts. [Learn more](#)

AWS organization

Permission sets

Select one or more AWS accounts in your AWS organization to provide SSO access to users and groups. If you have organized your accounts under organizational units (OUs), you can choose an OU to make account selection easier. [Learn more](#)

Assign users

to 3 accounts. Deselect

Find AWS account by ID, name, or email

	AWS account	Permission sets
• All accounts	✓ <b>Stage Account</b> #000417004716 arthur@wallbarnes.com	None
• Root	✓ <b>Test Account</b> #000837737527 arthur@wallbarnes.com	None
• MarketingBU	✓ <b>Production Account</b> #000000000000 arthur@wallbarnes.com	None

# Assign Users



## Select users or groups

You can search for the users and groups in your connected directory to assign SSO access. Type a user or group name to search in your connected directory. You can also specify an Active Directory domain (optional). You can add more than one user or group to your selection. [Learn more](#)

Groups

Users

anandcorp.com

Search connected directory

Found 4 matching users

- ✓ jadams
- jmadison
- jpolk
- jtyler

Selection

anandcorp.com/jadams [Remove](#)

Cancel

Next: Permission sets

# Create new permission set

## How do you want to create your permission set?

- ☒ **Use an existing job function policy**  
Use job function policies to apply predefined AWS managed policies to a permission set. The policies are based on common job functions in the IT industry. [Learn more](#)
- ☐ **Create a custom permission set**  
Use custom policies to select up to 10 AWS managed policies. You can also define a new policy document that best meets your needs. [Learn more](#)

## Select job function policy

<b>AdministratorAccess</b>
Provides full access to AWS services and resources.
<b>Billing</b>
Grants permissions for billing and cost management. This includes viewing account usage and viewing and modifying budgets and payment methods.
<b>DataScientist</b>
Grants permissions to AWS data analytics services.

## Assign Users



### Select permission sets

Permission sets define the level of access that users and groups have to an AWS account. Permission sets are stored in AWS SSO and appear in the AWS account as IAM roles. You can assign more than one permission set to a user. To ensure least privilege access to AWS accounts, users with multiple permission sets on an AWS account must pick a specific permission set when accessing the account and then return to the user portal to pick a different set when necessary. [Learn more](#)

Create new permission set



Permission set	Description	Provisioned status	Created on
----------------	-------------	--------------------	------------

You have not yet created any permission sets.

**Billing**

Grants permissions for billing and cost management. This includes viewing account usage and viewing and modifying budgets and payment methods.

**DataScientist**

Grants permissions to AWS data analytics services.

**DatabaseAdministrator**

Grants full access permissions to AWS services and actions required to set up and configure AWS database services.

**NetworkAdministrator**

Grants full access permissions to AWS services and actions required to set up and configure AWS network resources.

**PowerUserAccess**

Provides full access to AWS services and resources, but does not allow management of Users and groups.

**SecurityAudit**

The security audit template grants access to read security configuration metadata. It is useful for software that audits the configuration of an AWS account.

**SupportUser**

This policy grants permissions to troubleshoot and resolve issues in an AWS account. This policy also enables the user to contact AWS support to create and manage cases.

**SystemAdministrator**

Cancel

Create

## Complete

We have successfully configured your AWS accounts. Your users can access these AWS accounts with the permissions you assigned.

[Proceed to AWS accounts](#)

Account	Status
<b>Stage Account</b> info@.com	Complete <a href="#">Show details</a>
<b>Test Account</b> test@.com	Complete <a href="#">Show details</a>
<b>Production Account</b> prod@.com	Complete <a href="#">Show details</a>



Users and groups

Permission sets

## Assign Users

### Select permission sets

Permission sets define the level of access that users and groups have to an AWS account. Permission sets are stored in AWS SSO and appear in the AWS account as IAM roles. You can assign more than one permission set to a user. To ensure least privilege access to AWS accounts, users with multiple permission sets on an AWS account must pick a specific permission set when accessing the account and then return to the user portal to pick a different set when necessary. [Learn more](#)

Create new permission set



✓	Permission set	Description	Provisioned status	Created on
✓	SecurityAudit		Not provisioned	12/5/2017

Cancel

Previous

Finish



Search



AWS Management Console (3)

- 650 ( Account)
- 903 ( Account)
- 68C (Production Account)**  
SecurityAudit

Q.2. Upload files on Github from Desktop using Git Commands.

Steps - ~~Step~~ 1 - Move your file to the cloned repository

Step 2 → open gitbash

Step 3 → Go to current directory where you want the cloned directory to be added.

Step 4 → Add the file and stage it for commit.

\$ git add 'filename'

Step 5 → Commit the file to your local repository.

\$ git commit -m 'filename'

Step 6 → Push the changes to ~~the~~ github

\$ git push origin your branch name

```

91938@LAPTOP-C07539E8 MINGW64 ~
$ cd /c/Users/91938/OneDrive/Documents/ss/sample_template

91938@LAPTOP-C07539E8 MINGW64 ~/OneDrive/Documents/ss/sample_template (master)
$ git add df
fatal: pathspec 'df' did not match any files

91938@LAPTOP-C07539E8 MINGW64 ~/OneDrive/Documents/ss/sample_template (master)
$ git add /c/Users/91938/OneDrive/Documents/ss/df.PNG
fatal: C:/Users/91938/OneDrive/Documents/ss/df.PNG: 'C:/Users/91938/OneDrive/Documents/ss/df.PNG' is outside repository at 'C:/Users/91938/OneDrive/Documents/ss/sample_template'

91938@LAPTOP-C07539E8 MINGW64 ~/OneDrive/Documents/ss/sample_template (master)
$ git add /c/Users/91938/OneDrive/Documents/ss/sample_template/df.PNG

91938@LAPTOP-C07539E8 MINGW64 ~/OneDrive/Documents/ss/sample_template (master)
$ git commit -m /c/Users/91938/OneDrive/Documents/ss/sample_template/df.PNG
[master bcca24e] C:/Users/91938/OneDrive/Documents/ss/sample_template/df.PNG
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 df.PNG

91938@LAPTOP-C07539E8 MINGW64 ~/OneDrive/Documents/ss/sample_template (master)
$ git push origin https://github.com/NirmalGiri/sample_template.git
fatal: invalid refsPEC 'https://github.com/NirmalGiri/sample_template.git'

91938@LAPTOP-C07539E8 MINGW64 ~/OneDrive/Documents/ss/sample_template (master)
$ git push origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 24.53 KiB | 12.27 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/NirmalGiri/sample_template.git
eec56e7..bcca24e master -> master

91938@LAPTOP-C07539E8 MINGW64 ~/OneDrive/Documents/ss/sample_template (master)
$ |

```

← → ↻ 🔒 github.com/NirmalGiri/sample\_template ☆ ⚙️ 👤 : Reading list

📱 Apps 📧 Gmail 📺 YouTube 🗺️ Maps 📰 News

🐙 Search or jump to... Pull requests Issues Marketplace Explore 🔔 + 👤

🐙 NirmalGiri / sample\_template Public template 🔍 Unwatch 1 ⭐ Star 0 🍴 Fork 0

<> Code ⌚ Issues 🔄 Pull requests ⚙️ Actions 📁 Projects 📖 Wiki 🛡️ Security 📊 Insights ⚙️ Settings

🔗 master 1 branch 0 tags Go to file Add file Code Use this template

👤 NirmalGiri	C:/Users/91938/OneDrive/Documents/ss/sample_template/df.PNG	• bcca24e 3 minutes ago	🕒 2 commits
📄 README.md	Initial commit		1 hour ago
📄 df.PNG	C:/Users/91938/OneDrive/Documents/ss/sample_template/df.PNG		3 minutes ago

README.md

# sample\_template

About ⚙️

No description, website, or topics provided.

📖 Readme

Releases

No releases published  
[Create a new release](#)

Packages

No packages published  
[Publish your first package](#)

Environments 1

🔗 github-pages Active