



CLOUD COMPUTING LAB

BSE (V-B)

LAB 13

Submitted By: Urwa Zahra

Roll No: 2023-BSE-068

Submitted To: Sir Muhammad Shoaib

Task 0 Lab Setup (Codespace & GH CLI)

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cd Lab11
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068/Lab11 (main) $ gh codespace list
```

| NAME | DISPLAY NAME | REPOSITORY | BRANCH | STATE | CREATED AT |
|----------------------------------|-------------------|---------------------------|--------|-----------|---------------------|
| studious-guide-r45rj7gx6699f56wg | studious guide | Urwa012/cc-urwazahra-2... | main* | Shutdown | about 2 days ago |
| turbo-space-yodel-q759wvqx4gj... | turbo space yodel | Urwa012/cc-urwazahra-2... | main | Shutdown | about 1 day ago |
| bookish-goldfish-pj5gx7r6w6pg... | bookish goldfish | Urwa012/cc-urwazahra-2... | main | Available | about 3 minutes ago |

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068/Lab11 (main) $ gh codespace ssh -c bookish-goldfish-pj5gx7r6w6pgc795w
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-1030-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $
```

Task 1 — Create IAM Group and Output Details

```
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ mkdir -p ~/Lab13
@Urwa012 →/workspaces/cc-urwazahra-2023-BSE-068 (main) $ cd ~/Lab13
@Urwa012 →~/Lab13 $ touch main.tf
```

```
provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

resource "aws_iam_group" "developers" {
  name = "developers"
  path = "/groups/"
}

output "group_details" {
  value = {
    group_name = aws_iam_group.developers.name
    group_arn  = aws_iam_group.developers.arn
    unique_id  = aws_iam_group.developers.unique_id
  }
}
```

```
• @Urwa012 →~/Lab13 $ terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v6.27.0
```

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see any changes that are required for your infrastructure. All Terraform commands should now work.

If you ever set or change modules or backend configuration for Terraform, rerun this command to reinitialize your working directory. If you forget, other commands will detect it and remind you to do so if necessary.

```
• @Urwa012 →~/Lab13 $ terraform apply -auto-approve
```

commands will detect it and remind you to do so if necessary.

```
• @Urwa012 →~/Lab13 $ terraform apply -auto-approve
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# aws_iam_group.developers will be created
+ resource "aws_iam_group" "developers" {
  + arn      = (known after apply)
  + id       = (known after apply)
  + name     = "developers"
  + path     = "/groups/"
  + unique_id = (known after apply)
}
```

Plan: 1 to add, 0 to change, 0 to destroy.

Changes to Outputs:

```
+ group_details = {
  + group_arn = (known after apply)
  + group_name = "developers"
  + unique_id = (known after apply)
}
```

aws_iam_group.developers: Creating...

aws_iam_group.developers: Creation complete after 1s [id=developers]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.


Outputs:

```
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
```

```

}
@Urwa012 → ~/Lab13 $ terraform output
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
@Urwa012 → ~/Lab13 $

```

| <input type="checkbox"/> | Group name | ▲ | Users | |
|--------------------------|----------------------------|---|-------|---|
| <input type="checkbox"/> | admin | | | 1 |
| <input type="checkbox"/> | developers | | |  C |

Task 2 — Create IAM User with Group Membership

```

provider "aws" {
  shared_config_files      = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

resource "aws_iam_group" "developers" {
  name = "developers"
  path = "/groups/"
}

output "group_details" {
  value = {
    group_name = aws_iam_group.developers.name
    group_arn  = aws_iam_group.developers.arn
    unique_id  = aws_iam_group.developers.unique_id
  }
}

resource "aws_iam_user" "lb" {
  name = "loadbalancer"
  path = "/users/"
  force_destroy = true
  tags = {
    DisplayName = "Load Balancer"
  }
}

resource "aws_iam_user_group_membership" "lb_membership" {
  user = aws_iam_user.lb.name
  groups = [
    aws_iam_group.developers.name
  ]
}

output "user_details" {
  value = {
    user_name = aws_iam_user.lb.name
    user_arn  = aws_iam_user.lb.arn
    unique_id = aws_iam_user.lb.unique_id
  }
}

```

```
@Urwa012 →~/Lab13 $ terraform apply -auto-approve
```

```
+ tags_all = {
+   "DisplayName" = "Load Balancer"
+ }
+ unique_id = (known after apply)
}

# aws_iam_user_group_membership.lb_membership will be created
+ resource "aws_iam_user_group_membership" "lb_membership" {
+   groups = [
+     "developers",
+   ]
+   id     = (known after apply)
+   user   = "loadbalancer"
+ }
}
```

Plan: 2 to add, 0 to change, 0 to destroy.

Changes to Outputs:

```
+ user_details = {
+   unique_id = (known after apply)
+   user_arn  = (known after apply)
+   user_name = "loadbalancer"
+ }
```

aws_iam_user.lb: Creating...

aws_iam_user.lb: Creation complete after 1s [id=loadbalancer]

aws_iam_user_group_membership.lb_membership: Creating...

aws_iam_user_group_membership.lb_membership: Creation complete after 1s [id=

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.

Outputs:

```
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
user_details = {
  "unique_id" = "AIDAWOW3ND6CPYXAZ2SBZ"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}
```

```
}
● @Urwa012 →~/Lab13 $ terraform output
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
user_details = {
  "unique_id" = "AIDAWOW3ND6CPYXAZ2SBZ"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}
}
```

loadbalancer [Info](#)

Summary

ARN

 arn:aws:iam::443915509636:user/users/loadbala

Console access

Disabled

Access key 1

[Create access key](#)

Created

January 06, 2026, 02:11 (UTC+05:00)

developers [Info](#)

[Delete](#)

Summary

[Edit](#)


User group name

developers

Creation time

January 06, 2026, 02:07 (UTC+05:00)

ARN

 arn:aws:iam::443915509636:group/groups/developers

[Users \(1\)](#)

[Permissions](#)

[Access Advisor](#)

Users in this group (1)

Task 3 — Attach Policies to IAM Group

```

    name = "developers"
    path = "/groups/"
}

output "group_details" {
  value = {
    group_name = aws_iam_group.developers.name
    group_arn  = aws_iam_group.developers.arn
    unique_id  = aws_iam_group.developers.unique_id
  }
}

resource "aws_iam_user" "lb" {
  name = "loadbalancer"
  path = "/users/"
  force_destroy = true
  tags = {
    DisplayName = "Load Balancer"
  }
}

resource "aws_iam_user_group_membership" "lb_membership" {
  user = aws_iam_user.lb.name
  groups = [
    aws_iam_group.developers.name
  ]
}

output "user_details" {
  value = {
    user_name = aws_iam_user.lb.name
    user_arn  = aws_iam_user.lb.arn
    unique_id = aws_iam_user.lb.unique_id
  }
}

resource "aws_iam_group_policy_attachment" "developer_ec2_fullaccess" {
  group = aws_iam_group.developers.name
  policy_arn = "arn:aws:iam::aws:policy/AmazonEC2FullAccess"
}

resource "aws_iam_group_policy_attachment" "change_password" {
  group = aws_iam_group.developers.name
  policy_arn = "arn:aws:iam::aws:policy/IAMUserChangePassword"
}

```

```
@Urwa012 →~/Lab13 $ terraform apply -auto-approve
```

```
aws_iam_user.lb: Refreshing state... [id=loadbalancer]
```

```
aws_iam_user_group_membership.lb_membership: Refreshing state... [id=terraform-20260105211133401200000001]
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# aws_iam_group_policy_attachment.change_password will be created
+ resource "aws_iam_group_policy_attachment" "change_password" {
  + group      = "developers"
  + id         = (known after apply)
  + policy_arn = "arn:aws:iam::aws:policy/IAMUserChangePassword"
}

# aws_iam_group_policy_attachment.developer_ec2_fullaccess will be created
+ resource "aws_iam_group_policy_attachment" "developer_ec2_fullaccess" {
  + group      = "developers"
  + id         = (known after apply)
  + policy_arn = "arn:aws:iam::aws:policy/AmazonEC2FullAccess"
}
```

Plan: 2 to add, 0 to change, 0 to destroy.

aws_iam_group_policy_attachment.change_password: Creating...

aws_iam_group_policy_attachment.developer_ec2_fullaccess: Creating...

aws_iam_group_policy_attachment.change_password: Creation complete after 1s [id=developers-20260105211538735400000001]

aws_iam_group_policy_attachment.developer_ec2_fullaccess: Creation complete after 1s [id=developers-20260105211538751600000002]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.

Outputs:

```
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
user_details = {
  "unique_id" = "AIDAWOW3ND6CPYXAZ2SBZ"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}
```

developers [Info](#)


[Delet](#)

Summary

[Edit](#)

User group name
developers

Creation time
January 06, 2026, 02:07 (UTC+05:00)




ARN
 arn:aws:iam::443915509636:group/groups/developers

[Users \(1\)](#)[Permissions](#)[Access Advisor](#)

Permissions policies (2) [Info](#)

[Simulate](#)[Remove](#)[Add permissions](#)

You can attach up to 10 managed policies.

| Search | | Filter by Type | | | |
|--------------------------|---|----------------|-------------|---|-------------------|
| <input type="text"/> | | All types | | < 1 >  | |
| <input type="checkbox"/> | Policy name ↗ | ▲ | Type | ▼ | Attached entities |
| <input type="checkbox"/> |  AmazonEC2FullAccess | | AWS managed | | <u>1</u> |
| <input type="checkbox"/> |  IAMUserChangePassword | | AWS managed | | <u>3</u> |

Task 4 — Create Login Profile for IAM User

```
variable "iam_password" {  
  description = "Temporary password for the IAM user"  
  type        = string  
  sensitive   = true  
  default     = "1dontKnow"  
}
```

```
#!/usr/bin/env bash  
set -euo pipefail  
  
USERNAME="$1"  
PASSWORD="$2"  
  
# Check if login profile already exists  
if aws iam get-login-profile --user-name "$USERNAME" >/dev/null 2>&1; then  
  echo "Login profile already exists for $USERNAME. Skipping."  
else  
  echo "Creating login profile for $USERNAME"  
  aws iam create-login-profile \  
    --user-name "$USERNAME" \  
    --password "$PASSWORD" \  
    --passwd-reset-required  
fi  
~
```

- @Urwa012 → ~/Lab13 \$ vim create-login-profile.sh
- @Urwa012 → ~/Lab13 \$ chmod +x create-login-profile.sh
- @Urwa012 → ~/Lab13 \$

```
resource "aws_iam_user" "lb" {  
  name = "loadbalancer"  
  path = "/users/"  
  force_destroy = true  
  tags = {  
    DisplayName = "Load Balancer"  
  }  
}  
  
resource "null_resource" "create_login_profile" {  
  triggers = {  
    password_hash = sha256(var.iam_password)  
    user          = aws_iam_user.lb.name  
  }  
  
  depends_on = [aws_iam_user.lb]  
  
  provisioner "local-exec" {  
    command = "${path.module}/create-login-profile.sh ${aws_iam_user.lb.name} '${var.iam_password}'"  
  }  
}
```

commands will detect it and remind you to do so if necessary.

```
@Urwa012 ~ / Lab13 $ terraform apply -auto-approve -var="iam_password=MySecurePass123!"
aws_iam_group.developers: Refreshing state... [id=developers]
aws_iam_user.lb: Refreshing state... [id=loadbalancer]
aws_iam_group_policy_attachment.change_password: Refreshing state... [id=developers-202601052115387354000000001]
aws_iam_user_group_membership.lb_membership: Refreshing state... [id=terraform-202601052111334012000000001]
aws_iam_group_policy_attachment.developer_ec2_fullaccess: Refreshing state... [id=developers-202601052115387516000000002]
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

- + create

Terraform will perform the following actions:

```
# null_resource.create_login_profile will be created
+ resource "null_resource" "create_login_profile" {
  + id          = (known after apply)
  + triggers = {
    + "password_hash" = (sensitive value)
    + "user"          = "loadbalancer"
  }
}
```

Plan: 1 to add, 0 to change, 0 to destroy.

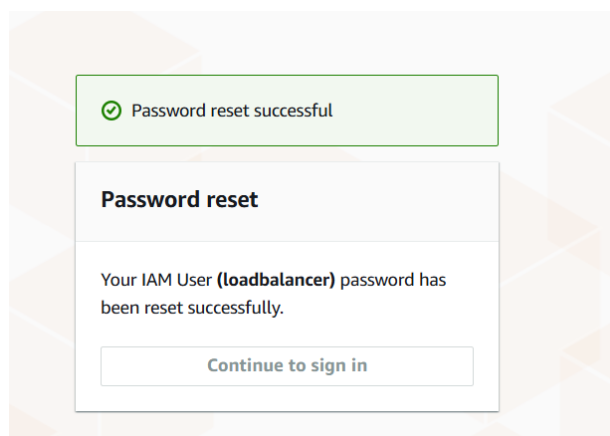
```
null_resource.create_login_profile: Creating...
null_resource.create_login_profile: Provisioning with 'local-exec'...
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile (local-exec): (output suppressed due to sensitive value in config)
null_resource.create_login_profile: Creation complete after 4s [id=3996121946308949874]
```

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

Outputs:

```
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAWOW3ND6CDXUOKYFRS"
}
user_details = {
  "unique_id" = "AIDAOW3ND6CPYXAZ2SBZ"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}
```

```
@Urwa012 ~ / Lab13 $ aws iam get-login-profile --user-name loadbalancer
{
  "LoginProfile": {
    "UserName": "loadbalancer",
    "CreateDate": "2026-01-05T21:21:06+00:00",
    "PasswordResetRequired": true
  }
}
```



Task 5 — Generate Access Keys for IAM User

```
}

resource "aws_iam_access_key" "lb_access_key" {
  user = aws_iam_user.lb.name
}

output "access_key_id" {
  value = aws_iam_access_key.lb_access_key.id
}

output "access_key_secret" {
  value     = aws_iam_access_key.lb_access_key.secret
  sensitive = true
}

-- INSERT --
```

```
Changes to Outputs:
+ access_key_id      = (known after apply)
+ access_key_secret  = (sensitive value)
aws_iam_access_key.lb_access_key: Creating...
aws_iam_access_key.lb_access_key: Creation complete after 1s [id=AKIAW0H3ND6CEA4YPS26]
```

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

Outputs:

```
access_key_id = "AKIAW0H3ND6CEA4YPS26"
access_key_secret = <sensitive>
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AGPAW0H3ND6CEA4YPS26"
}
user_details = {
  "unique_id" = "AGPAW0H3ND6CEA4YPS26"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}
@Urwa012 →~/Lab13 $
```

```

}
• @Urwa012 →~/Lab13 $ terraform output
access_key_id = "AKIAW5W.....26"
access_key_secret = <sensitive>
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "AC.....S"
}
user_details = {
  "unique_id" = "AT.....SBZ"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}
}
• @Urwa012 →~/Lab13 $ cat terraform.tfstate | grep -A 10 "access_key_secret"
  "access_key_secret": {
    "value": "1.....8F1KJURguo33eI219y1+9s4YuAqt4",
    "type": "string",
    "sensitive": true
  },
  "group_details": {
    "value": {
      "group_arn": "arn:aws:iam::443915509636:group/groups/developers",
      "group_name": "developers",
      "unique_id": "AC.....YFRS"
    },
    "sensitive": true
  },
}
• @Urwa012 →~/Lab13 $

```

loadbalancer [Info](#)

[Delete](#)

Summary

ARN

[arn:aws:iam::443915509636:user/users/loadbalancer](#)

Console access

[Enabled without MFA](#)

Access key 1

AKIAW5W.....26 - Active
 Ⓛ Never used. Created today.

Created

January 06, 2026, 02:11 (UTC+05:00)

Last console sign-in

[Today](#)

Access key 2

[Create access key](#)

[Permissions](#)
[Groups \(1\)](#)
[Tags \(1\)](#)
[Security credentials](#)
[Last Accessed](#)

Console sign-in

[Manage console access](#)

Console sign-in link

<https://443915509636.signin.aws.amazon.com/console>

Console password

Updated 8 minutes ago (2026-01-06 02:28 GMT+5)

Last console sign-in

2026-01-06 02:28 GMT+5

Create bucket [Info](#)

Buckets are containers for data stored in S3.

General configuration

AWS Region

Canada (Central) ca-central-1

Bucket name [Info](#)

myapp-s3-bucket-demonstration

Bucket names must be 3 to 63 characters and unique within the global namespace. Bucket names must also begin and end with a letter or number. Valid characters are a-z, 0-9, periods (.), and hyphens (-). [Learn more](#)

Copy settings from existing bucket - optional

Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Format: s3://bucket/prefix

Object Ownership [Info](#)

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

Object Ownership

☒ ACLs disabled (recommended)

All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

☐ ACLs enabled

Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership

Bucket owner enforced

Block Public Access settings for this 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 contents is blocked, you must 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 you 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 case.

☒ Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☒ Block public access to buckets and objects granted through new access control lists (ACLs)

S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing resources using ACLs.

☒ Block public access to buckets and objects granted through any access control lists (ACLs)

S3 will ignore all ACLs that grant public access to buckets and objects.

☒ Block public access to buckets and objects granted through new public bucket or access point policies

S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

☒ Block public and cross-account access to buckets and objects through any public bucket or access point policies

S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. You can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning

☐ Disable

☒ Enable

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. You can easily recover from both unintended user actions and application failures. [Learn more](#)

Bucket Versioning

☐ Disable

☒ Enable

```

terraform {
  backend "s3" {
    bucket = "myapp-s3-bucket-demonstration"
    key    = "myapp/terraform.tfstate"
    region = "me-central-1"
    encrypt = true
    use_lockfile = true
  }
}

provider "aws" {
  shared_config_files = ["~/.aws/config"]
  shared_credentials_files = ["~/.aws/credentials"]
}

resource "aws_iam_group" "developers" {
  name = "developers"
  path = "/groups/"
}

output "group_details" {
  value = {

```

```

@Urwa012 → ~/Lab13 $ terraform init -migrate-state
Initializing the backend...
Do you want to copy existing state to the new backend?
  Pre-existing state was found while migrating the previous "local" backend to the
  newly configured "s3" backend. No existing state was found in the newly
  configured "s3" backend. Do you want to copy this state to the new "s3"
  backend? Enter "yes" to copy and "no" to start with an empty state.

Enter a value: yes

Successfully configured the backend "s3"! Terraform will automatically
use this backend unless the backend configuration changes.
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/null from the dependency lock file
- Using previously-installed hashicorp/aws v6.27.0
- Using previously-installed hashicorp/null v3.2.4

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
@Urwa012 → ~/Lab13 $
spaces: bookish goldfish 0 main 0 0 2

```

```

@Urwa012 →~/Lab13 $ terraform apply -auto-approve -var="iam_password=MySecurePass123!"
aws_iam_group.developers: Refreshing state... [id=developers]
aws_iam_user.lb: Refreshing state... [id=loadbalancer]
aws_iam_group_policy_attachment.developer_ec2_fullaccess: Refreshing state... [id=developers-20260105211538751600000002]
aws_iam_group_policy_attachment.change_password: Refreshing state... [id=developers-20260105211538735400000001]
aws_iam_access_key.lb_access_key: Refreshing state... [id=AKIAW0W3ND6CEA4YPS26]
null_resource.create_login_profile: Refreshing state... [id=3996121946308949874]
aws_iam_user_group_membership.lb_membership: Refreshing state... [id=terraform-20260105211334012000000001]

```

No changes. Your infrastructure matches the configuration.

Terraform has compared your real infrastructure against your configuration and found no differences, so no changes are needed.

Apply complete! Resources: 0 added, 0 changed, 0 destroyed.

Outputs:

```

access_key_id = "AKIAW0W3ND6CEA4YPS26"
access_key_secret = <ensitive>
group_details = {
  "group_arn" = "arn:aws:iam::443915509636:group/groups/developers"
  "group_name" = "developers"
  "unique_id" = "A443915509636:group/groups/developers"
}
user_details = {
  "unique_id" = "A443915509636:user/users/loadbalancer"
  "user_arn" = "arn:aws:iam::443915509636:user/users/loadbalancer"
  "user_name" = "loadbalancer"
}

```

```

@Urwa012 →~/Lab13 $

```

> [myapp-s3-bucket-demonstration](#) > myapp/

<

myapp/

Objects

Properties

Objects (1)

Copy S3 URI

Copy URL

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For more information, see [Amazon S3 inventory](#).

Find objects by prefix

| <input type="checkbox"/> | Name | Type | Last modified | Size | Storage class |
|--------------------------|-----------------------------------|---------|---------------------------------------|--------|---------------|
| <input type="checkbox"/> | terraform.tfstate | tfstate | January 6, 2026, 02:46:05 (UTC+05:00) | 6.7 KB | Standard |

```

@Urwa012 →~/Lab13 $ ls -la terraform.tfstate*
-rw-r--r-- 1 codespace codespace 0 Jan 5 21:46 terraform.tfstate
-rw-r--r-- 1 codespace codespace 6882 Jan 5 21:46 terraform.tfstate.backup
@Urwa012 →~/Lab13 $ terraform destroy -auto-approve

```

```
@Urwa012 → ~/Lab13 $ terraform destroy -auto-approve
```

```
    } -> null
  - unique_id      = "AIDAWOW3ND6CPYXAZ2SBZ" -> null
    # (1 unchanged attribute hidden)
}

# aws_iam_user_group_membership.lb_membership will be destroyed
- resource "aws_iam_user_group_membership" "lb_membership" {
  - groups = [
    - "developers",
  ] -> null
  - id      = "terraform-20260105211133401200000001" -> null
  - user    = "loadbalancer" -> null
}

# null_resource.create_login_profile will be destroyed
- resource "null_resource" "create_login_profile" {
  - id      = "3996121946308949874" -> null
  - triggers = {
    - "password_hash" = (sensitive value)
    - "user"          = "loadbalancer"
  } -> null
}
```

Plan: 0 to add, 0 to change, 7 to destroy.

Changes to Outputs:

```
- access_key_id      = "AKIAWOW3ND6CEA4YPS26" -> null
- access_key_secret = (sensitive value) -> null
- group_details      = {
  - group_arn = "arn:aws:iam::443915509636:group/groups/developers"
  - group_name = "developers"
  - unique_id  = "AGPAWOW3ND6CDXUOKYFRS"
} -> null
- user_details      = {
  - unique_id = "AIDAWOW3ND6CPYXAZ2SBZ"
  - user_arn  = "arn:aws:iam::443915509636:user/users/loadbalancer"
  - user_name = "loadbalancer"
} -> null
```

null_resource.create_login_profile: Destroying... [id=3996121946308949874]

null_resource.create_login_profile: Destruction complete after 0s

aws_iam_group_policy_attachment.developer_ec2_fullaccess: Destroying... [id=developers-20260105211

aws_iam_access_key.lb_access_key: Destroying... [id=AKIAWOW3ND6CEA4YPS26]

aws_iam_group_policy_attachment.change_password: Destroying... [id=developers-20260105211538735400

aws_iam_user_group_membership.lb_membership: Destroying... [id=terraform-20260105211133401200000000

aws_iam_user_group_membership.lb_membership: Destruction complete after 1s

aws_iam_group_policy_attachment.change_password: Destruction complete after 1s

aws_iam_access_key.lb_access_key: Destruction complete after 1s

aws_iam_user.lb: Destroying... [id=loadbalancer]

aws_iam_group_policy_attachment.developer_ec2_fullaccess: Destruction complete after 1s

aws_iam_group.developers: Destroying... [id=developers]

aws_iam_group.developers: Destruction complete after 0s

aws_iam_user.lb: Destruction complete after 2s

Destroy complete! Resources: 7 destroyed.

```
@Urwa012 → ~/Lab13 $
```

PropertiesPermissionsVersions

Versions (1)


Download

Open ↗

Delete

Actions ▼

< 1 >

| <input type="checkbox"/> | Version ID | Type | Last modified | Size | Storage class |
|--------------------------|---|---------|--------------------|---------|---------------|
| <input type="checkbox"/> |  null (Current version) | tfstate | January 6, 2026... | 181.0 B | Standard |

Task 7 — Create Multiple Users from CSV File

PROBLEMSOUTPUTDEBUG CONSOLETERMINALPORTS

```
locals {  
  users = csvdecode(file("users.csv"))  
}
```

PROBLEMSOUTPUTDEBUG CONSOLETERMINAL

```
user_name  
Michael  
Dwight  
Jim  
Pam  
Ryan  
Andy  
Robert  
Stanley  
Kevin  
Angela  
Oscar  
Phyllis  
Toby  
Kelly  
Darryl  
Creed  
Meredith  
Erin  
Gabe  
Jan  
David  
Holly  
Charles  
Jo  
Clark  
Peter  
~  
~  
~  
~  
~
```

```
    CreatedBy    = "Terraform"
  }
}

# Add all users to developers group
resource "aws_iam_user_group_membership" "users_membership" {
  for_each = aws_iam_user.users

  user = each.value.name
  groups = [
    aws_iam_group.developers.name
  ]
}

# Create login profiles for all users
resource "null_resource" "create_login_profiles" {
  for_each = aws_iam_user.users

  triggers = {
    password_hash = sha256(var.iam_password)
    user          = each.value.name
  }

  depends_on = [aws_iam_user.users]

  provisioner "local-exec" {
    command = "${path.module}/create-login-profile.sh ${each.value.name} '${var.iam_password}'"
  }
}

# Create access keys for all users
resource "aws_iam_access_key" "users_access_keys" {
  for_each = aws_iam_user.users

  user = each.value.name
}

# Output all user details
output "all_users_details" {
  value = {
    for user_name, user in aws_iam_user.users : user_name => {
      user_arn      = user.arn
      user_unique_id = user.unique_id
      access_key_id = aws_iam_access_key.users_access_keys[user_name].id
    }
  }
}

# Output all access key secrets (sensitive)
output "all_access_key_secrets" {
  value = {
    for user_name, key in aws_iam_access_key.users_access_keys : user_name => key.secret
  }
  sensitive = true
}
```

```
• @Urwa012 →~/Lab13 $ terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/null from the dependency lock file
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/null v3.2.4
- Using previously-installed hashicorp/aws v6.27.0
```

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see any changes that are required for your infrastructure. All Terraform commands should now work.

If you ever set or change modules or backend configuration for Terraform, rerun this command to reinitialize your working directory. If you forget, other commands will detect it and remind you to do so if necessary.

```
@Urwa012 →~/Lab13 $ terraform apply -auto-approve -var="iam_password=MySecurePass123!"
null_resource.create_login_profiles["Darryl"]: Creation complete after 7s [id=8876531924476751998]
```

Apply complete! Resources: 26 added, 0 changed, 26 destroyed.

Outputs:

```
access_key_id = "AKIAWOW3ND6CHOZYNUGE"
access_key_secret = <sensitive>
all_access_key_secrets = <sensitive>
all_users_details = {
  "Andy" = {
    "access_key_id" = "AKIAWOW3ND6CGHJ6QYHK"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Andy"
    "user_unique_id" = "AIDAWOW3ND6CLHNCXNK6V"
  }
  "Angela" = {
    "access_key_id" = "AKIAWOW3ND6CC7GBZWOF"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Angela"
    "user_unique_id" = "AIDAWOW3ND6CP3EW2QDW4"
  }
  "Charles" = {
    "access_key_id" = "AKIAWOW3ND6CQOFF4AJY"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Charles"
    "user_unique_id" = "AIDAWOW3ND6CKMDBSMPLK"
  }
  "Clark" = {
    "access_key_id" = "AKIAWOW3ND6CBSYF6NMA"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Clark"
    "user_unique_id" = "AIDAWOW3ND6CNVF6A3YF4"
  }
  "Creed" = {
    "access_key_id" = "AKIAWOW3ND6CCOF2L6ED"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Creed"
    "user_unique_id" = "AIDAWOW3ND6CDX5J45HZG"
  }
  "Darryl" = {
    "access_key_id" = "AKIAWOW3ND6CBHE66AJG"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Darryl"
    "user_unique_id" = "AIDAWOW3ND6CMFTNMXQ46"
  }
  "David" = {
    "access_key_id" = "AKIAWOW3ND6CNDK6MJVM"
    "user_arn" = "arn:aws:iam::443915509636:user/users/David"
    "user_unique_id" = "AIDAWOW3ND6CAGYVVYVUH"
  }
  "Dwight" = {
    "access_key_id" = "AKIAWOW3ND6CFCVSF5XY"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Dwight"
    "user_unique_id" = "AIDAWOW3ND6CA2CFRUSHV"
  }
  "Erin" = {
    "access_key_id" = "AKIAWOW3ND6CPAM63A4H"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Erin"
    "user_unique_id" = "AIDAWOW3ND6CNUISVRRGP"
  }
}
```

```
● @Urwa012 → ~/Lab13 $ terraform output
access_key_id = "AKIAWOW3ND6CGHJ6QMHK"
access_key_secret = <sensitive>
all_access_key_secrets = <sensitive>
all_users_details = {
  "Andy" = {
    "access_key_id" = "AKIAWOW3ND6CGHJ6QMHK"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Andy"
    "user_unique_id" = "AIDAWOW3ND6CLHNCXNK6V"
  }
  "Angela" = {
    "access_key_id" = "AKIAWOW3ND6CC7GBZWOF"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Angela"
    "user_unique_id" = "AIDAWOW3ND6CP3EW2QDW4"
  }
  "Charles" = {
    "access_key_id" = "AKIAWOW3ND6CQOFF4AJY"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Charles"
    "user_unique_id" = "AIDAWOW3ND6CKWDBSMPLK"
  }
  "Clark" = {
    "access_key_id" = "AKIAWOW3ND6CBSYF6NMA"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Clark"
    "user_unique_id" = "AIDAWOW3ND6CNV6A3YF4"
  }
  "Creed" = {
    "access_key_id" = "AKIAWOW3ND6CCOF2L6ED"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Creed"
    "user_unique_id" = "AIDAWOW3ND6CDX5J45HZG"
  }
  "Darryl" = {
    "access_key_id" = "AKIAWOW3ND6CBHE66AJG"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Darryl"
    "user_unique_id" = "AIDAWOW3ND6CMFTNMXQ46"
  }
  "David" = {
    "access_key_id" = "AKIAWOW3ND6CNDK6MJVM"
    "user_arn" = "arn:aws:iam::443915509636:user/users/David"
    "user_unique_id" = "AIDAWOW3ND6CAGYVVYVUH"
  }
  "Dwight" = {
    "access_key_id" = "AKIAWOW3ND6CFC5VF5XY"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Dwight"
    "user_unique_id" = "AIDAWOW3ND6CA2CFRUSHV"
  }
  "Erin" = {
    "access_key_id" = "AKIAWOW3ND6CPAM63A4H"
    "user_arn" = "arn:aws:iam::443915509636:user/users/Erin"
    "user_unique_id" = "AIDAWOW3ND6CNUI5VRRGP"
  }
  "Gabe" = {

```

| | | | | | | | | | | | | | | | |
|--------------------------|-----------------------------|---------|---|---------------|---|-----------|------------|-------------------------|-----------|---------------|---|---------------|--|---------|---|
| <input type="checkbox"/> | Admin | / | 0 | 4 days ago | - | 4 days | 4 days ago | - | - | - | arnauvenciam:443915509655cuser/Admin | 4 days ago | | Enabled | - |
| <input type="checkbox"/> | Andy | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/Andr... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Angelita | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Ansgarwem | / | 1 | Yesterday | - | 11 days | - | Active - AKUAWOWSHND... | 11 days | Yesterday | arnauvenciam:443915509655cuser/fuser... | 11 days ago | | Enabled | - |
| <input type="checkbox"/> | Ansgarwem2 | / | 0 | - | - | 12 days | - | Active - AKUAWOWSHND... | 11 days | - | arnauvenciam:443915509655cuser/fuser... | 12 days ago | | Enabled | - |
| <input type="checkbox"/> | CC | / | 0 | - | - | - | - | Active - AKUAWOWSHND... | 11 days | - | arnauvenciam:443915509655cuser/CC | 11 days ago | | Enabled | - |
| <input type="checkbox"/> | Charles | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Clark | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Creed | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Darryl | /users/ | 1 | - | - | 2 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | David | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Denzel | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Dora | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Edu | /users/ | 1 | - | - | 2 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Holly | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Jani | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Jani | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Jo | /users/ | 1 | - | - | 2 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Joely | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Karin | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | LailaBer | / | 0 | 4 days ago | - | 4 days | 4 days ago | - | - | - | arnauvenciam:443915509655cuser/Laila... | 4 days ago | | Enabled | - |
| <input type="checkbox"/> | Lindalancer | /users/ | 1 | - | - | 8 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Meredith | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Michael | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Oscar | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Pam | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Peter | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Phyllis | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Roberts | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Ryan | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Stanley | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Toby | /users/ | 1 | - | - | 5 minutes | - | Active - AKUAWOWSHND... | 8 minutes | - | arnauvenciam:443915509655cuser/fuser... | 8 minutes ago | | Enabled | - |
| <input type="checkbox"/> | Tina | / | 0 | 5 minutes ago | - | - | - | Active - AKUAWOWSHND... | 2 days | 6 minutes ago | arnauvenciam:443915509655cuser/Tina | 11 days ago | | Enabled | - |

Users in this group (27)

[Remove](#)

An IAM user is an entity that you create in AWS to represent the person or application that uses it to interact with AWS.

| <input type="checkbox"/> | User name | Groups | Last activity | Creation time |
|--------------------------|------------------------------|--------|---------------|----------------|
| <input type="checkbox"/> | Andy | | None | 10 minutes ago |
| <input type="checkbox"/> | Angela | | None | 10 minutes ago |
| <input type="checkbox"/> | Charles | | None | 10 minutes ago |
| <input type="checkbox"/> | Clark | | None | 10 minutes ago |
| <input type="checkbox"/> | Creed | | None | 10 minutes ago |
| <input type="checkbox"/> | Darryl | | None | 10 minutes ago |
| <input type="checkbox"/> | David | | None | 10 minutes ago |
| <input type="checkbox"/> | Dwight | | None | 10 minutes ago |
| <input type="checkbox"/> | Erin | | None | 10 minutes ago |
| <input type="checkbox"/> | Gabe | | None | 10 minutes ago |
| <input type="checkbox"/> | Holly | | None | 10 minutes ago |
| <input type="checkbox"/> | Jan | | None | 10 minutes ago |
| <input type="checkbox"/> | Jim | | None | 10 minutes ago |
| <input type="checkbox"/> | Jo | | None | 10 minutes ago |
| <input type="checkbox"/> | Kelly | | None | 10 minutes ago |
| <input type="checkbox"/> | Kevin | | None | 10 minutes ago |
| <input type="checkbox"/> | loadbalancer | | None | 10 minutes ago |
| <input type="checkbox"/> | Meredith | | None | 10 minutes ago |
| <input type="checkbox"/> | Michael | | None | 10 minutes ago |
| <input type="checkbox"/> | Oscar | | None | 10 minutes ago |
| <input type="checkbox"/> | Pam | | None | 10 minutes ago |
| <input type="checkbox"/> | Peter | | None | 10 minutes ago |
| <input type="checkbox"/> | Phyllis | | None | 10 minutes ago |
| <input type="checkbox"/> | Robert | | None | 10 minutes ago |
| <input type="checkbox"/> | Ryan | | None | 10 minutes ago |
| <input type="checkbox"/> | Stanley | | None | 10 minutes ago |
| <input type="checkbox"/> | Toby | | None | 10 minutes ago |

Summary

ARN
arn:aws:iam::443915509636:user/users/Michael

Created
January 06, 2026, 02:55 (UTC+05:00)

Console access
Enabled without MFA

Last console sign-in
Never

Access key 1
AKIAWOW3ND6CA3F24SN4 - Active
Never used. Created today.

Access key 2
Create access key

Console sign-in

Console sign-in link
https://443915509636.signin.aws.amazon.com/console

Console password
Updated 5 minutes ago (2026-01-06 03:01 GMT+5)

Last console sign-in
Never

Manage console access

Multi-factor authentication (MFA) (0)

RemoveResyncAssign MFA device

Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. [Learn more](#)

| Type | Identifier | Certifications | Created on |
|--|------------|----------------|------------|
| No MFA devices. Assign an MFA device to improve the security of your AWS environment | | | |

Assign MFA device

Access keys (1)

Create access key

Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. [Learn more](#)

AKIAWOW3ND6CA3F24SN4

Actions

```

    "type": "string"
  },
  "access_key_secret": {
    "value": "pvNw/9vCXfc10Yqm3+Nx/2aqXfGhi/C5CRJ2Cne+",
    "type": "string",
    "sensitive": true
  },
  "all_access_key_secrets": {
    "value": {
      "Andy": "MmrAVcBSjV4fuE1kgBYrFFP/BHN243vidLBhJek1",
      "Angela": "ScBNJHPK5gzM08QWw666h+NyOwXnmminA1yQxVxr",
      "Charles": "gOwLO1Juy1CkuvUvMx1vuSI2w8B5q7tJz1102GGR",
      "Clark": "p5QqbSXX4Y5Te4jqTtks/dd9ms9RqnJvpbats2vj",
      "Creed": "2RdyDS3cS0AtVSd+IF7YcLKwecfDhDA6kRnq1YZe",
      "Darryl": "BfKfF1YBMybU0AP9xdmz4V19ELsrFM2Nye9Wsowk",
      "David": "YyJw8VQW0PwW9f11jAKLzG5qg/GENqt7Rc2y5/o",
      "Dwight": "mykNanxM4h0k0qccGBF9QhTg4/P7FH8CD0e+9ri7",
      "Erin": "CGizONwtrhed01FgowraBnmxbax4W4rWC6p1G5fV",
      "Gabe": "0AdtNUCjWJ2TindF35ruy0LXVsZoOeGuIw4WKduB",
      "Holly": "Kvk13vVw/h2YTLgpjowSb1mV/99x+2oY0vpXBdek",
      "Jan": "Mt48xThodv7XNLEWh93qgHk4yld/a3VY7GnLPP44",
      "Jim": "UE1A69gUWTDbtbwogIBEGczT0pAu7wn3vjHK3UeX",
      "Jo": "sgczcZnd54OWFGQkthn0egeiKLP6QHxIHmkZHqH/",
      "Kelly": "BxfY3TfXDOyJCylqgguc3502o3FpFnX5CyKC6yae",
      "Kevin": "gqA2Uy6y7ZE6n+19d5iQpVsn6l7bbKxkr+aLl0+q",
      "Meredith": "jsfxlpVuiC10U909YZ/qFK4c7BRo/fAUOUKZXg7V",
      "Michael": "M/b+Y3rj0sGUyhckdo2gg3ZTVtd/wBQoThr7sA01",
      "Oscar": "1CjNUxLuF/8K7CYyAyvgjSF7ukWgZjTgQVYUviwC",
      "Pam": "J0xJHSobznbiJTJrTOrx078GH7sjZf9s1qcq/Oxj",
      "Peter": "NfqWq85U1WKjLIYdJtzv/0I10Ncan2gHa2MAj6Jr",
      "Phyllis": "qm/sallrS01dMzgeJKNw8r2TEILBg04A6c7RdACS",
      "Robert": "wnyql1mtNp9vrrTAT7/8f2En4ssFImmborm+WCSy",
      "Ryan": "FVpfTUVcXR9uYRa88djvIW4ntNbRjFz8G7LfGlP1",
      "Stanley": "mopuEFTbnGlbyNOLj6G0x3mkn3amkm900njNWEK0",
      "Toby": "c93hk2Tz0sjghPpruZmNZ5j+ou4w/pgsReUTIJ0m"
    }
  },
  "type": [
    "object",
    {
      "Andy": "string",
      "Angela": "string",
      "Charles": "string",
      "Clark": "string",
      "Creed": "string",
      "Darryl": "string",
      "David": "string",
      "Dwight": "string",
      "Erin": "string",
      "Gabe": "string",
      "Holly": "string",
      "Jan": "string",
      "Jim": "string",
      "Jo": "string",
      "Kelly": "string",
      "Kevin": "string",
      "Meredith": "string",
      "Michael": "string",
      "Oscar": "string",
      "Pam": "string",
      "Peter": "string",
      "Phyllis": "string",
      "Robert": "string",
      "Ryan": "string",
      "Stanley": "string",
      "Toby": "string"
    }
  ],
  "-----": "-----"

```

Cleanup:

```
aws_iam_user.users["Stanley"]: Destruction complete after 5s
aws_iam_user.users["Charles"]: Destroying... [id=Charles]
aws_iam_user.users["Toby"]: Destruction complete after 5s
aws_iam_user.users["Robert"]: Destroying... [id=Robert]
aws_iam_user.users["Andy"]: Destruction complete after 2s
aws_iam_user.users["Kevin"]: Destroying... [id=Kevin]
aws_iam_user.users["Jan"]: Destruction complete after 2s
aws_iam_user.users["Erin"]: Destroying... [id=Erin]
aws_iam_user.users["Creed"]: Destruction complete after 2s
aws_iam_user.users["Phyllis"]: Destroying... [id=Phyllis]
aws_iam_user.users["Peter"]: Destruction complete after 2s
aws_iam_user.users["Jim"]: Destroying... [id=Jim]
aws_iam_user.users["David"]: Destruction complete after 4s
aws_iam_user.users["Gabe"]: Destroying... [id=Gabe]
aws_iam_user.users["Charles"]: Destruction complete after 2s
aws_iam_user.users["Kelly"]: Destroying... [id=Kelly]
aws_iam_user.users["Clark"]: Destruction complete after 4s
aws_iam_user.users["Ryan"]: Destroying... [id=Ryan]
aws_iam_user.users["Robert"]: Destruction complete after 2s
aws_iam_user.users["Darryl"]: Destruction complete after 7s
aws_iam_user.users["Kevin"]: Destruction complete after 2s
aws_iam_user.users["Erin"]: Destruction complete after 2s
aws_iam_user.users["Dwight"]: Destruction complete after 5s
aws_iam_user.users["Jim"]: Destruction complete after 3s
aws_iam_user.users["Kelly"]: Destruction complete after 2s
aws_iam_user.users["Ryan"]: Destruction complete after 3s
aws_iam_user.users["Gabe"]: Destruction complete after 4s
aws_iam_user.users["Phyllis"]: Destruction complete after 7s
```

Destroy complete! Resources: 111 destroyed.

@Urwa012 → ~/Lab13 \$

```
aws_iam_user.users["Stanley"]: Destruction complete after 5s
aws_iam_user.users["Charles"]: Destroying... [id=Charles]
aws_iam_user.users["Toby"]: Destruction complete after 5s
aws_iam_user.users["Robert"]: Destroying... [id=Robert]
aws_iam_user.users["Andy"]: Destruction complete after 2s
aws_iam_user.users["Kevin"]: Destroying... [id=Kevin]
aws_iam_user.users["Jan"]: Destruction complete after 2s
aws_iam_user.users["Erin"]: Destroying... [id=Erin]
aws_iam_user.users["Creed"]: Destruction complete after 2s
aws_iam_user.users["Phyllis"]: Destroying... [id=Phyllis]
aws_iam_user.users["Peter"]: Destruction complete after 2s
aws_iam_user.users["Jim"]: Destroying... [id=Jim]
aws_iam_user.users["David"]: Destruction complete after 4s
aws_iam_user.users["Gabe"]: Destroying... [id=Gabe]
aws_iam_user.users["Charles"]: Destruction complete after 2s
aws_iam_user.users["Kelly"]: Destroying... [id=Kelly]
aws_iam_user.users["Clark"]: Destruction complete after 4s
aws_iam_user.users["Ryan"]: Destroying... [id=Ryan]
aws_iam_user.users["Robert"]: Destruction complete after 2s
aws_iam_user.users["Darryl"]: Destruction complete after 7s
aws_iam_user.users["Kevin"]: Destruction complete after 2s
aws_iam_user.users["Erin"]: Destruction complete after 2s
aws_iam_user.users["Dwight"]: Destruction complete after 5s
aws_iam_user.users["Jim"]: Destruction complete after 3s
aws_iam_user.users["Kelly"]: Destruction complete after 2s
aws_iam_user.users["Ryan"]: Destruction complete after 3s
aws_iam_user.users["Gabe"]: Destruction complete after 4s
aws_iam_user.users["Phyllis"]: Destruction complete after 7s
```

Destroy complete! Resources: 111 destroyed.

@Urwa012 → ~/Lab13 \$

Users (0) [Info](#)

[Delete](#)

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

<

| <input type="checkbox"/> | User name | ▲ | Path | ▼ | Groups | ▼ | Last active |
|--------------------------|-----------|---|------|---|--------|---|-------------|
|--------------------------|-----------|---|------|---|--------|---|-------------|

No resources to display



User groups (0) [Info](#)

[Delete](#)

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

<

| <input type="checkbox"/> | Group name | ▲ | Users | ▼ | Permissions |
|--------------------------|------------|---|-------|---|-------------|
|--------------------------|------------|---|-------|---|-------------|

No resources to display



myapp-s3-bucket-demonstration Info

Objects

Metadata

Properties

Permissions

Metrics

Management

Access Points

Objects (0)

Copy S3 URI

Copy URL

Download

Open ↗

Delete

Actions ▾

Create folder

Upload

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

Show versions

< 1 >

Name

Type

Last modified

Size

Storage class

No objects

You don't have any objects in this bucket.

Upload

```
Destroy complete! Resources: 111 destroyed.
@Urwa012 →~/Lab13 $ ls -la
total 61764
drwxr-xr-x 4 codespace codespace 4096 Jan 5 22:00 .
drwxr-xr-x 1 codespace codespace 4096 Jan 5 22:00 ..
drwxr-xr-x 3 codespace codespace 4096 Jan 5 21:46 .terraform
-rw-r--r-- 1 codespace codespace 2422 Jan 5 21:20 .terraform.lock.hcl
drwxr-xr-x 3 codespace codespace 4096 Jan 5 19:11 aws
-rw-r--r-- 1 codespace codespace 63189840 Jan 5 20:57 awscli2.zip
-rwxr-xr-x 1 codespace codespace 423 Jan 5 21:18 create-login-profile.sh
-rw-r--r-- 1 codespace codespace 50 Jan 5 21:52 locals.tf
-rw-r--r-- 1 codespace codespace 3566 Jan 5 22:00 main.tf
-rw-r--r-- 1 codespace codespace 0 Jan 5 21:46 terraform.tfstate
-rw-r--r-- 1 codespace codespace 6882 Jan 5 21:46 terraform.tfstate.backup
-rw-r--r-- 1 codespace codespace 167 Jan 5 21:53 users.csv
-rw-r--r-- 1 codespace codespace 150 Jan 5 21:18 variables.tf
```