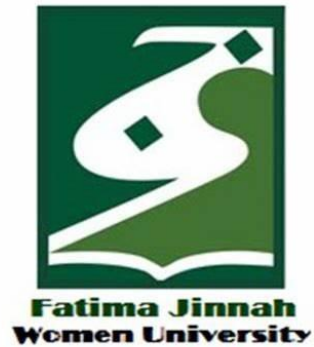


CLOUD COMPUTING



SUBMITTED TO

SIR WAQAS SALEEM & SIR SHOAIB

SUBMITTED BY

HAMNA MAHMOOD

2023-BSE-025

BSE V-A

Lab 10 – GH CLI Codespaces + AWS + Terraform: CLI Automation of VPC/Subnet Creation

Task 1 — GitHub CLI Codespace Setup & Authentication

- task1_gh_install.png

```
gh_2.83.2_windows_amd64 1/2/2026 11:30 PM File folder

C:\Users\ABC\Downloads\gh_2.83.2_windows_amd64\bin>gh --version
gh version 2.83.2 (2025-12-10)
https://github.com/cli/cli/releases/tag/v2.83.2
```

- task1_gh_auth_login.png

```
C:\Users\ABC\Downloads\gh_2.83.2_windows_amd64\bin>gh auth login -s codespace
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Paste an authentication token
Tip: you can generate a Personal Access Token here https://github.com/settings/tokens
The minimum required scopes are 'repo', 'read:org', 'workflow'.
? Paste your authentication token:
- gh config set -h github.com git_protocol https
[X] Configured git protocol
[X] Logged in as hamna-mahmood
```

- task1_codespace_list.png

```
C:\Users\ABC\Downloads\gh_2.83.2_windows_amd64\bin>gh codespace list
NAME          DISPLAY NAME  REPOSITORY                                BRANCH  STATE    CREATED AT
studious-journey-q7x57x9ww649...  studious journey  HamnaMahmood20/CC-Hamna-Mahmo...  main*   Shutdown  about 3 days ago
```

- task1_codespace_ssh_connected.png

```
C:\Users\ABC\Downloads\gh_2.83.2_windows_amd64\bin>gh codespace ssh -c studious-journey-q7x57x9ww649hx57g
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
Last login: Mon Jan  5 09:32:29 2026 from ::1
@hamna-mahmood [X] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $
```

Task 2 – Install AWS CLI, Terraform CLI, Provider Setup

- task2_aws_install_and_version.png

```
@hamna-mahmood [X] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left   Speed
100 60.2M  100 60.2M    0     0  170M      0  0:00:03  0:00:03  0:00:00  170M
@hamna-mahmood [X] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ unzip awscliv2.zip
Archive:  awscliv2.zip
  creating: aws/
  creating: aws/dist/
  creating: aws/dist/linux-x86_64/
  creating: aws/dist/linux-x86_64/bin/
  creating: aws/dist/linux-x86_64/bin/
@hamna-mahmood [X] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ sudo ./aws/install
You can now run: /usr/local/bin/aws --version
@hamna-mahmood [X] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ aws --version
aws-cli/2.32.26 Python/3.13.11 Linux/6.8.0-1030-azure exe/x86_64.ubuntu.24
```

- task2_aws_configure_and_files.png

```
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ aws configure
AWS Access Key ID [None]:
AWS Secret Access Key [None]
Default region name [None]: me-central-1
Default output format [None]: json
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $

@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cat ~/.aws/credentials
[default]
aws_access_key_id =
aws_secret_access_key =
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cat ~/.aws/config
[default]
region = me-central-1
output = json
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $
```

- task2_aws_get_caller_identity.png

```
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ aws sts get-caller-identity
{
  "UserId": "AIDAX4VWZI3PVPP1QABGB",
  "Account": "542622959327",
  "Arn": "arn:aws:iam::542622959327:user/Hamna_Assignment"
}
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $
```

- task2_terraform_install_and_version.png

```
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ wget -O - https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-
archive-keyring.gpg
100%[=====] 3.89K --KB/s in 0s
026-01-06 12:07:02 (820 MB/s) - written to stdout [3980/3980]
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gp
] https://apt.releases.hashicorp.com $(grep -oP '(?<UBUNTU_CODENAME+)' /etc/os-release || lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ sudo apt update
Reading state information... Done
51 packages can be upgraded. Run 'apt list --upgradable' to see them.
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ sudo apt install terraform
Unpacking terraform (1.14.3-1) ...
Setting up terraform (1.14.3-1) ...
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ which terraform
/usr/bin/terraform
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform --version
Terraform v1.14.3
on linux_amd64
```

- task2_provider_file_creation.png

```
@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $
vim main.tf
```

- task2_provider_block.png

```
Command Prompt - gh_codespace ssh -c studios-journey-q7x57x9ww649hx57g

provider "aws"

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

- ```
provider "aws"

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

~
~
~
~
~
~
~
~
~
~
~
~
```

- ```
ghamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding latest version of hashicorp/aws...
- Installing hashicorp/aws v6.27.0...
- Installed hashicorp/aws v6.27.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

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.
```

- ```
@hamna-mahmood /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cat .terraform.lock.hcl
This file is maintained automatically by "terraform init".
Manual edits may be lost in future updates.

provider "registry.terraform.io/hashicorp/aws" {
 version = "6.27.0"
 hashes = [
 "h1:bixp2PSsP5ZGBczGCxcbsDn6lF5QFLUXlNroq9cdab4=",
 "zh:177a24b806c72e8484b5c5cab9c3b2b38e3d770ae6f745a998b4d6619fd0e8129",
 "zh:44c4a85c1f7b868a3306b542e6a56c10bd6c6d5a67bc0c9b8f6a9060cf5f3be7",
 "zh:552652185bc85c8ba1da1de447c454728a5c6839c458b6dcd3ce71c19ccfc",
 "zh:60284b8172d09aee91eae0856f09855eaf040ce3a58d6933602ae17c53f8ed04",
 "zh:6be38d156756ca61fe87be7c3f52cc5d769cd709686700ac4b230f40a6e95b5dbc9",
 "zh:7a409138fae4ef42f3a637e37cb9efedf96459e28a3c764fc4e855e8db9a7485",
 "zh:8070cf5224ed1ed3a3e9a59f7c30ff88bf071c7567165275d477c1738a56c064",
 "zh:894439ef340a9a79f69cd759e27ad11c7826adeca27be1b1ca82b3c9702fa300",
 "zh:89d035eebf08a97c89374ff06040955ddc09f275ecca609d0c9d58d149bef5cf",
 "zh:895b1145d724fc1f38369099e4a5087141885740fd6c0b1dbc492171e73c2e49",
 "zh:9b12af85486a96aed8d7984b0fff811a4b42e3c8d8dad1a3fb4c0b580d04fa425",
 "zh:a80b47ae8d1475201c86bd94a5dcb9dd4da5e8b73102a0820b68b66b76d50fd",
 "zh:d3395be1556210f82199b9166a6b2e677cee9c4b67e96ae63f6c3a98325ad7ab0",
 "zh:db0b869d09657f6f1e4110b56093c5fcd9dbdd97c020db1e577b239c0adcbe",
 "zh:ffc72e680370ae7c21f9bd3082c6317730df805c6797427839a6b6b7e9a26a01",
]
}
```

- task2\_terraform\_dir\_ls.png

```
@hamna-mahmood /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ ls .terraform/
providers
```

### Task3 – VPC Subnet Creation, Initialization, Verification

- task3\_main\_tf\_resource\_add.png

```
Command Prompt - gh codespace ssh -c studios-journey-q7x57x9ww649hx57g
provider "aws" {
 shared_config_files = ["~/.aws/config"]
 shared_credentials_files = ["~/.aws/credentials"]
}
resource "aws_vpc" "development_vpc" {
 cidr_block = "10.0.0.0/16"
}

resource "aws_subnet" "dev_subnet_1" {
 vpc_id = aws_vpc.development_vpc.id
 cidr_block = "10.0.10.0/24"
 availability_zone = "me-central-1a"
}
```

- task3\_terraform\_apply\_vpc\_subnet.png

```
Enter a value: yes

aws_vpc.development_vpc: Creating...
aws_vpc.development_vpc: Creation complete after 1s [id=vpc-0c75646c981fbaa67]
aws_subnet.dev_subnet_1: Creating...
aws_subnet.dev_subnet_1: Creation complete after 1s [id=subnet-0e1f662733444c28b]

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

- task3\_aws\_cli\_verify\_subnet.png

```
@hamna-mahmood /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ aws ec2 describe-subnets --filter "Name=subnet-id,Values=subnet-078f1b79825a5fee0"
{
 "Subnets": [
 {
 "AvailabilityZoneId": "mec1-az2",
 "MapCustomerOwnedIpOnLaunch": false,
 "OwnerId": "542622959327",
 "AssignIpv6AddressOnCreation": false,
 "Ipv6CidrBlockAssociationSet": [],
 "SubnetArn": "arn:aws:ec2:me-central-1:542622959327:subnet/subnet-078f1b79825a5fee0",
 "EnableDns64": false,
 "Ipv6Native": false,
 "PrivateDnsNameOptionsOnLaunch": {
 "HostnameType": "ip-name",
 "EnableResourceNameDnsARecord": false,
 "EnableResourceNameDnsAAAARecord": false
 },
 "BlockPublicAccessStates": {
 "InternetGatewayBlockMode": "off"
 },
 "SubnetId": "subnet-078f1b79825a5fee0",
 "State": "available",
 "VpcId": "vpc-0b412746b28b797e7",
 "CidrBlock": "172.31.16.0/20",
 ...skipping...
 }
]
}
```

- task3\_aws\_cli\_verify\_vpc.png

```

@hamna-mahmood @ /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ aws ec2 describe-vpcs \
> --filters "Name=vpc-id,Values=vpc-0b412746b28b797e7" \
> --region me-central-1
{
 "Vpcs": [
 {
 "OwnerId": "542622959327",
 "InstanceTenancy": "default",
 "CidrBlockAssociationSet": [
 {
 "AssociationId": "vpc-cidr-assoc-0d7f0fa773ce8d726",
 "CidrBlock": "172.31.0.0/16",
 "CidrBlockState": {
 "State": "available"
 }
 }
]
 }
]
}
...skipping...

```

#### Task4 – Data Source, Targeted Destroy, Tags

- task4\_main\_tf\_datasource\_resource\_add.png

```

data "aws_vpc" "existing_vpc" {
 default = true
}

resource "aws_subnet" "dev_subnet_1_existing" {
 vpc_id = data.aws_vpc.existing_vpc.id
 cidr_block = "172.31.48.0/24"
 availability_zone = "me-central-1a"
}

```

- task4\_terraform\_apply\_datasource\_resource.png

```

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

Do you want to perform these actions?
 Terraform will perform the actions described above.
 Only 'yes' will be accepted to approve.

 Enter a value: yes

aws_subnet.dev_subnet_1_existing: Creating...
aws_subnet.dev_subnet_1_existing: Creation complete after 1s [id=subnet-0e3d0deccd5087b1f]

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

```

- task4\_terraform\_destroy\_targeted.png

```

aws_subnet.dev_subnet_1_existing: Destroying... [id=subnet-0e3d0deccd5087b1f]
aws_subnet.dev_subnet_1_existing: Destruction complete after 1s

Warning: Applied changes may be incomplete

The plan was created with the -target option in effect, so some changes requested in the configuration may have been
ignored and the output values may not be fully updated. Run the following command to verify that no other changes are
pending:
 terraform plan

Note that the -target option is not suitable for routine use, and is provided only for exceptional situations such as
recovering from errors or mistakes, or when Terraform specifically suggests to use it as part of an error message.

Destroy complete! Resources: 1 destroyed.

```

- task4\_terraform\_refresh\_state.png

```
@hamna-mahmood [] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform refresh
data.aws_vpc.existing_vpc: Reading...
aws_vpc.development_vpc: Refreshing state... [id=vpc-0c75646c981fbaa67]
data.aws_vpc.existing_vpc: Read complete after 1s [id=vpc-0b412746b28b797e7]
aws_subnet.dev_subnet_1: Refreshing state... [id=subnet-0e1f662733444c28b]
```

- task4\_terraform\_apply\_after\_refresh.png

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

Do you want to perform these actions?
 Terraform will perform the actions described above.
 Only 'yes' will be accepted to approve.

 Enter a value: yes

aws_subnet.dev_subnet_1_existing: Creating...
aws_subnet.dev_subnet_1_existing: Creation complete after 1s [id=subnet-00960f277fd06040a]

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

- task4\_terraform\_destroy\_all.png

```
Plan: 0 to add, 0 to change, 3 to destroy.

Do you really want to destroy all resources?
 Terraform will destroy all your managed infrastructure, as shown above.
 There is no undo. Only 'yes' will be accepted to confirm.

 Enter a value: yes

aws_subnet.dev_subnet_1: Destroying... [id=subnet-0e1f662733444c28b]
aws_subnet.dev_subnet_1_existing: Destroying... [id=subnet-00960f277fd06040a]
aws_subnet.dev_subnet_1_existing: Destruction complete after 0s
aws_subnet.dev_subnet_1: Destruction complete after 0s
aws_vpc.development_vpc: Destroying... [id=vpc-0c75646c981fbaa67]
aws_vpc.development_vpc: Destruction complete after 1s

Destroy complete! Resources: 3 destroyed.
```

- task4\_terraform\_plan\_output.png

```
aws_vpc.development_vpc will be created
+ resource "aws_vpc" "development_vpc" {
+ arn = (known after apply)
+ cidr_block = "10.0.0.0/16"
+ default_network_acl_id = (known after apply)
+ default_route_table_id = (known after apply)
+ default_security_group_id = (known after apply)
+ dhcp_options_id = (known after apply)
+ enable_dns_hostnames = (known after apply)
+ enable_dns_support = true
+ enable_network_address_usage_metrics = (known after apply)
+ id = (known after apply)
+ instance_tenancy = "default"
+ ipv6_association_id = (known after apply)
+ ipv6_cidr_block = (known after apply)
+ ipv6_cidr_block_network_border_group = (known after apply)
+ main_route_table_id = (known after apply)
+ owner_id = (known after apply)
+ region = "me-central-1"
+ tags_all = (known after apply)
+ }

Plan: 3 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
```

- task4\_terraform\_apply\_after\_destroy.png

```
Plan: 3 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
 Terraform will perform the actions described above.
 Only 'yes' will be accepted to approve.

 Enter a value: yes

aws_vpc.development_vpc: Creating...
aws_subnet.dev_subnet_1_existing: Creating...
aws_subnet.dev_subnet_1_existing: Creation complete after 0s [id=subnet-024745bf211a02a21]
aws_vpc.development_vpc: Creation complete after 1s [id=vpc-0511dfe8d2c0fa70f]
aws_subnet.dev_subnet_1: Creating...
aws_subnet.dev_subnet_1: Creation complete after 1s [id=subnet-0adf2d7099862dd16]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
```

- task4\_main\_tf\_tagging.png

```
resource "aws_vpc" "development_vpc" {
 cidr_block = "10.0.0.0/16"
 tags = {
 Name = "development"
 vpc_env = "dev"
 }
}

resource "aws_subnet" "dev_subnet_1" {
 vpc_id = aws_vpc.development_vpc.id
 cidr_block = "10.0.10.0/24"
 availability_zone = "me-central-1a"
 tags = {
 Name = "subnet-1-dev"
 }
}

resource "aws_subnet" "dev_subnet_1_existing" {
 vpc_id = data.aws_vpc.existing_vpc.id
 cidr_block = "172.31.48.0/24"
 availability_zone = "me-central-1a"
 tags = {
 Name = "subnet-1-default"
 }
}
```

- task4\_terraform\_apply\_tagging.png



```

@hamna-mahmood [/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main)] $ terraform refresh
data.aws_vpc.existing_vpc: Reading...
aws_vpc.development_vpc: Refreshing state... [id=vpc-0511dfe8d2c0fa70f]
data.aws_vpc.existing_vpc: Read complete after 1s [id=vpc-0b412746b28b797e7]
aws_subnet.dev_subnet_1_existing: Refreshing state... [id=subnet-024745bf211a02a21]
aws_subnet.dev_subnet_1: Refreshing state... [id=subnet-0adf2d7099862dd16]

@hamna-mahmood [/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main)] $ terraform apply -auto-approve
data.aws_vpc.existing_vpc: Reading...
aws_vpc.development_vpc: Refreshing state... [id=vpc-0511dfe8d2c0fa70f]
data.aws_vpc.existing_vpc: Read complete after 0s [id=vpc-0b412746b28b797e7]
aws_subnet.dev_subnet_1_existing: Refreshing state... [id=subnet-024745bf211a02a21]
aws_subnet.dev_subnet_1: Refreshing state... [id=subnet-0adf2d7099862dd16]

 ~ tags_all = {
 + "Name" = "development"
 + "vpc_env" = "dev"
 }
 # (19 unchanged attributes hidden)
 }

Plan: 0 to add, 3 to change, 0 to destroy.
aws_vpc.development_vpc: Modifying... [id=vpc-0511dfe8d2c0fa70f]
aws_subnet.dev_subnet_1_existing: Modifying... [id=subnet-024745bf211a02a21]
aws_subnet.dev_subnet_1_existing: Modifications complete after 0s [id=subnet-024745bf211a02a21]
aws_vpc.development_vpc: Modifications complete after 1s [id=vpc-0511dfe8d2c0fa70f]
aws_subnet.dev_subnet_1: Modifying... [id=subnet-0adf2d7099862dd16]
aws_subnet.dev_subnet_1: Modifications complete after 0s [id=subnet-0adf2d7099862dd16]

Apply complete! Resources: 0 added, 3 changed, 0 destroyed.

```

- task4\_terraform\_plan\_remove\_tag.png

```

resource "aws_vpc" "development_vpc" {
 cidr_block = "10.0.0.0/16"
 tags = {
 Name = "development"
 }
}

 ~ tags = {
 "Name" = "development"
 - "vpc_env" = "dev" -> null
 }
 ~ tags_all = {
 - "vpc_env" = "dev" -> null
 # (1 unchanged element hidden)
 }
 # (19 unchanged attributes hidden)
 }

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

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take
exactly these actions if you run "terraform apply" now.

```

- task4\_terraform\_apply\_remove\_tag.png

```

~ tags_all = {
 - "vpc_env" = "dev" -> null
 # (1 unchanged element hidden)
 }
 # (19 unchanged attributes hidden)
}

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

Do you want to perform these actions?
 Terraform will perform the actions described above.
 Only 'yes' will be accepted to approve.

 Enter a value: yes

aws_vpc.development_vpc: Modifying... [id=vpc-0511dfe8d2c0fa70f]
aws_vpc.development_vpc: Modifications complete after 1s [id=vpc-0511dfe8d2c0fa70f]

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

```

## Task 5 – State File Inspection and Terraform State Commands

- task5\_terraform\_destroy.png

```

Plan: 0 to add, 0 to change, 3 to destroy.

Do you really want to destroy all resources?
 Terraform will destroy all your managed infrastructure, as shown above.
 There is no undo. Only 'yes' will be accepted to confirm.

 Enter a value: yes

aws_subnet.dev_subnet_1: Destroying... [id=subnet-0adf2d7099862dd16]
aws_subnet.dev_subnet_1_existing: Destroying... [id=subnet-024745bf211a02a21]
aws_subnet.dev_subnet_1: Destruction complete after 1s
aws_subnet.dev_subnet_1_existing: Destruction complete after 1s
aws_vpc.development_vpc: Destroying... [id=vpc-0511dfe8d2c0fa70f]
aws_vpc.development_vpc: Destruction complete after 1s

Destroy complete! Resources: 3 destroyed.

```

- task5\_terraform\_state\_file\_empty.png

```

@hamna-mahmood /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cat terraform.tfstate
{
 "version": 4,
 "terraform_version": "1.14.3",
 "serial": 30,
 "lineage": "44daad15-ff50-9119-441e-f71bf3fee6a5",
 "outputs": {},
 "resources": [],
 "check_results": null
}

```

- task5\_terraform\_state\_backup\_prev.png

```
@hamna-mahmood [] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cat terraform.tfstate.backup
{
 "version": 4,
 "terraform_version": "1.14.3",
 "serial": 25,
 "lineage": "44daad15-ff50-9119-441e-f71bf3fee6a5",
 "outputs": {},
 "resources": [
 {
 "mode": "data",
 "type": "aws_vpc",
 "name": "existing_vpc",
 "provider": "provider[\"registry.terraform.io/hashicorp/aws\"]",
 "instances": [
 {
 "schema_version": 0,
 "attributes": {
 "arn": "arn:aws:ec2:me-central-1:542622959327:vpc/vpc-0b412746b28b797e7",
 "cidr_block": "172.31.0.0/16",
 "cidr_block_associations": [
 {
 "association_id": "vpc-cidr-assoc-0d7f0fa773ce8d726",
 "cidr_block": "172.31.0.0/16",
 "state": "associated"
 }
]
 }
 }
]
 }
]
}
```

- task5\_terraform\_apply\_recreated.png

```
Plan: 3 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
 Terraform will perform the actions described above.
 Only 'yes' will be accepted to approve.

 Enter a value: yes

aws_vpc.development_vpc: Creating...
aws_subnet.dev_subnet_1_existing: Creating...
aws_subnet.dev_subnet_1_existing: Creation complete after 1s [id=subnet-02f7c11ef8f628582]
aws_vpc.development_vpc: Creation complete after 1s [id=vpc-0255d13bb547ce1a6]
aws_subnet.dev_subnet_1: Creating...
aws_subnet.dev_subnet_1: Creation complete after 1s [id=subnet-02c263dc2531503d6]

Apply complete! Resources: 3 added, 0 changed, 0 destroyed.
```

- task5\_terraform\_state\_file\_populated.png

```
@hamna-mahmood [] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cat terraform.tfstate
{
 "version": 4,
 "terraform_version": "1.14.3",
 "serial": 34,
 "lineage": "44daad15-ff50-9119-441e-f71bf3fee6a5",
 "outputs": {},
 "resources": [
 {
 "mode": "data",
 "type": "aws_vpc",
 "name": "existing_vpc",
 "provider": "provider[\"registry.terraform.io/hashicorp/aws\"]",
 "instances": [
 {
 "schema_version": 0,
 "attributes": {
 "arn": "arn:aws:ec2:me-central-1:542622959327:vpc/vpc-0b412746b28b797e7",
 "cidr_block": "172.31.0.0/16",
 "cidr_block_associations": [
 {
 "association_id": "vpc-cidr-assoc-0d7f0fa773ce8d726",
 "cidr_block": "172.31.0.0/16",
 "state": "associated"
 }
],
 "default": true,
 "dhcp_options_id": "dopt-073b0c55c0a106d5a",
 "enable_dns_hostnames": true,
 "enable_dns_support": true,
 "enable_network_address_usage_metrics": false,
 "filter": null,
 "id": "vpc-0b412746b28b797e7",
 "instance_tenancy": "default",
 "ipv6_association_id": "",
 "ipv6_cidr_block": ""
 }
 }
]
 }
]
}
```

- task5\_terraform\_state\_backup\_empty.png

```
@hamna-mahmood [] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cat terraform.tfstate.backup
{
 "version": 4,
 "terraform_version": "1.14.3",
 "serial": 30,
 "lineage": "44daad15-ff50-9119-441e-f71bf3fee6a5",
 "outputs": {},
 "resources": [],
 "check_results": null
}
```

- task5\_terraform\_state\_list.png

```
@hamna-mahmood [] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ terraform state list
data.aws_vpc.existing_vpc
aws_subnet.dev_subnet_1
aws_subnet.dev_subnet_1_existing
aws_vpc.development_vpc
```

- task5\_terraform\_state\_show\_resource.png

```
@hamna-mahmood [/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main)] $ terraform state show aws_subnet.de
v_subnet_1
aws_subnet.dev_subnet_1:
resource "aws_subnet" "dev_subnet_1" {
 arn = "arn:aws:ec2:me-central-1:542622959327:subnet/subnet-02c263dc2531503d6"
 assign_ipv6_address_on_creation = false
 availability_zone = "me-central-1a"
 availability_zone_id = "mec1-az1"
 cidr_block = "10.0.10.0/24"
 customer_owned_ipv4_pool = null
 enable_dns64 = false
 enable_lni_at_device_index = 0
 enable_resource_name_dns_a_record_on_launch = false
 enable_resource_name_dns_aaaa_record_on_launch = false
 id = "subnet-02c263dc2531503d6"
 ipv6_cidr_block = null
 ipv6_cidr_block_association_id = null
 ipv6_native = false
 map_customer_owned_ip_on_launch = false
 map_public_ip_on_launch = false
 outpost_arn = null
 owner_id = "542622959327"
 private_dns_hostname_type_on_launch = "ip-name"
 region = "me-central-1"
 tags = {
 "Name" = "subnet-1-dev"
 }
 tags_all = {
 "Name" = "subnet-1-dev"
 }
 vpc_id = "vpc-0255d13bb547ce1a6"
}
```

## Task 6—Terraform Outputs & Attributes Reporting

- task6\_terraform\_outputs\_basic.png

```
output "dev-vpc-id" {
 value = aws_vpc.development_vpc.id
}
output "dev-subnet-id" {
 value = aws_subnet.dev_subnet_1.id
}
output "dev-vpc-arn" {
 value = aws_vpc.development_vpc.arn
}
output "dev-subnet-arn" {
 value = aws_subnet.dev_subnet_1.arn
}
-- INSERT --
```

- task6\_expanded\_outputs.png

```
Apply complete! Resources: 0 added, 0 changed, 0 destroyed.
Outputs:
dev-subnet-cidr_block = "10.0.10.0/24"
dev-subnet-region = "me-central-1a"
dev-subnet-tags_all = tomap({
 "Name" = "subnet-1-dev"
})
dev-subnet-tags_name = "subnet-1-dev"
dev-vpc-cidr_block = "10.0.0.0/16"
dev-vpc-region = "me-central-1"
dev-vpc-tags_all = tomap({
 "Name" = "development"
})
dev-vpc-tags_name = "development"
@hamna-mahmood [/workspaces/CC-Hamna-Mahmood-25-BSE-VA (main)] $
```

## Cleanup

- cleanup\_destroy\_resources.png

```
Plan: 0 to add, 0 to change, 3 to destroy.

Changes to Outputs:
- dev-subnet-cidr_block = "10.0.10.0/24" -> null
- dev-subnet-region = "me-central-1a" -> null
- dev-subnet-tags_all = {
 - Name = "subnet-1-dev"
} -> null
- dev-subnet-tags_name = "subnet-1-dev" -> null
- dev-vpc-cidr_block = "10.0.0.0/16" -> null
- dev-vpc-region = "me-central-1" -> null
- dev-vpc-tags_all = {
 - Name = "development"
} -> null
- dev-vpc-tags_name = "development" -> null

Do you really want to destroy all resources?
Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

aws_subnet.dev_subnet_1: Destroying... [id=subnet-02c263dc2531503d6]
aws_subnet.dev_subnet_1_existing: Destroying... [id=subnet-02f7c11ef8f628582]
aws_subnet.dev_subnet_1_existing: Destruction complete after 0s
aws_subnet.dev_subnet_1: Destruction complete after 0s
aws_vpc.development_vpc: Destroying... [id=vpc-0255d13bb547ce1a6]
aws_vpc.development_vpc: Destruction complete after 0s

Destroy complete! Resources: 3 destroyed.
```

- cleanup\_state\_files.png

```
@hamna-mahmood [] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cat terraform.tfstate
{
 "version": 4,
 "terraform_version": "1.14.3",
 "serial": 40,
 "lineage": "44daad15-ff50-9119-441e-f71bf3fee6a5",
 "outputs": {},
 "resources": [],
 "check_results": null
}

@hamna-mahmood [] /workspaces/CC-Hamna-Mahmood-25-BSE-VA (main) $ cat terraform.tfstate.backup
{
 "version": 4,
 "terraform_version": "1.14.3",
 "serial": 35,
 "lineage": "44daad15-ff50-9119-441e-f71bf3fee6a5",
 "outputs": {
 "dev-subnet-cidr_block": {
 "value": "10.0.10.0/24",
 "type": "string"
 },
 "dev-subnet-region": {
 "value": "me-central-1a",
 "type": "string"
 },
 "dev-subnet-tags_all": {
 "value": {
 "Name": "subnet-1-dev"
 },
 "type": [
 "map",
 "string"
]
 }
 }
}
```