Cloned the Repo to WSL:

```
root@Aravind:~# git clone https://github.com/Vennilavan12/Trend.git
Cloning into 'Trend'...
remote: Enumerating objects: 77, done.
remote: Counting objects: 100% (2/2), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 77 (delta 0), reused 0 (delta 0), pack-reused 75 (from 1)
Receiving objects: 100% (77/77), 8.58 MiB | 4.73 MiB/s, done.
Resolving deltas: 100% (1/1), done.
root@Aravind:~#
```

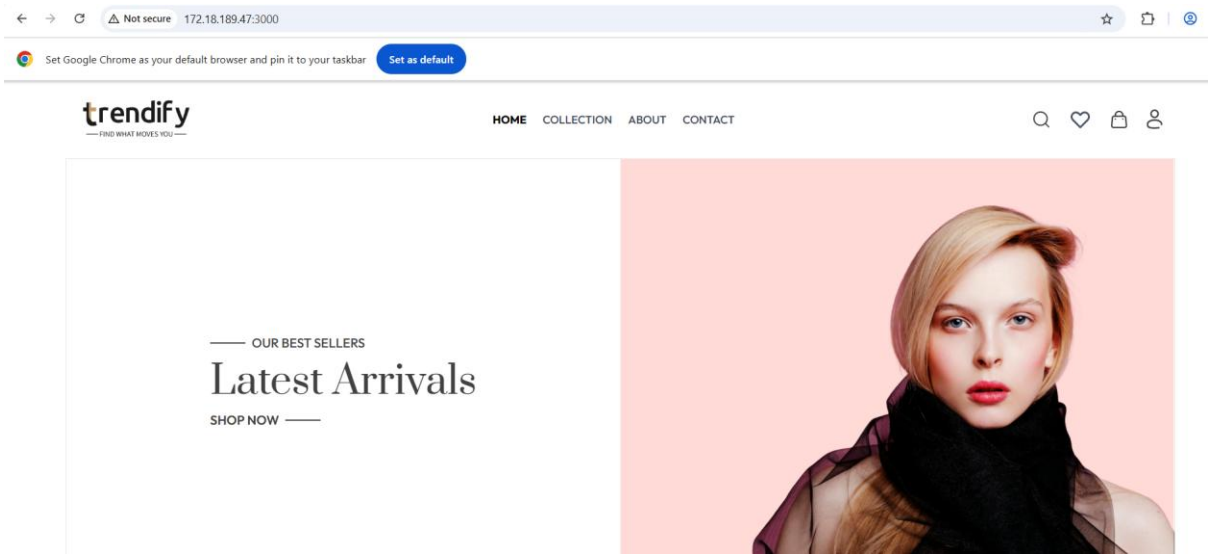
Installed npm and started the service:

```
root@Aravind:~/Trend# apt list npm
Listing... Done
npm/noble,now 9.2.0~ds1-2 all [installed]
root@Aravind:~/Trend#
```

Ran the app and hosted:

```
root@Aravind:~/Trend/dist# ls -rlt
total 12
-rw-r--r-- 1 root root 1497 Feb 17 14:58 vite.svg
-rw-r--r-- 1 root root  415 Feb 17 14:58 index.html
drwxr-xr-x 2 root root 4096 Feb 17 14:58 assets
root@Aravind:~/Trend/dist# npx serve -l 3000
ERROR Cannot copy server address to clipboard: Command failed with exit code 1: /root/.npm/_npx/aab42732f01924e5/node_
modules/clipboardy/fallbacks/windows/clipboard_x86_64.exe --copy
/root/.npm/_npx/aab42732f01924e5/node_modules/clipboardy/fallbacks/windows/clipboard_x86_64.exe: Invalid argument.

   Serving!

   - Local:    http://localhost:3000
   - Network:  http://172.18.189.47:3000

HTTP 2/17/2026 2:59:11 PM 172.18.176.1 GET /
HTTP 2/17/2026 2:59:11 PM 172.18.176.1 Returned 200 in 26 ms
HTTP 2/17/2026 2:59:11 PM 172.18.176.1 GET /assets/index-BjbOuBLy.js
HTTP 2/17/2026 2:59:11 PM 172.18.176.1 GET /assets/index-B52p61-W.css
HTTP 2/17/2026 2:59:11 PM 172.18.176.1 Returned 200 in 6 ms
```

Launched in browser and the app is running:



Created docker file and build docker image:

FROM nginx:alpine

COPY dist/ /usr/share/nginx/html

EXPOSE 80

```
root@Aravind:~/Trend# cat dockerfile
FROM nginx:alpine
COPY dist/ /usr/share/nginx/html
EXPOSE 80
root@Aravind:~/Trend#
```

```
root@Aravind:~/Trend# docker build -t trend-app .
[+] Building 15.1s (7/7) FINISHED                                                                     docker:default
 => [internal] load build definition from dockerfile                                                             0.1s
 => => transferring dockerfile: 98B                                                                              0.0s
 => [internal] load metadata for docker.io/library/nginx:alpine                                                  5.0s
 => [internal] load .dockerignore                                                                                0.1s
 => => transferring context: 2B                                                                                  0.0s
 => [internal] load build context                                                                                0.3s
 => => transferring context: 9.24MB                                                                              0.2s
 => [1/2] FROM docker.io/library/nginx:alpine@sha256:1d13701a5f9f3fb01aaa88cef2344d65b6b5bf6b7d9fa4cf0dca557a8d7702ba  7.7s
 => => resolve docker.io/library/nginx:alpine@sha256:1d13701a5f9f3fb01aaa88cef2344d65b6b5bf6b7d9fa4cf0dca557a8d7702ba  0.1s
 => => sha256:6d397a54a185dd0b6638d1a3934b81daef7a140741e12697377d6279066f7ca7 1.40kB / 1.40kB                   0.8s
 => => sha256:399d0898a94e0084f81499a3e3c29824357118c7ce551648ea1dbab813884661 404B / 404B                       1.1s
 => => sha256:5e7756927bef33a266e1221356d5da8553139cb80bc5b1b3827010811d9ea268 20.24MB / 20.24MB                 4.6s
 => => sha256:955a8478f9aceb66cbf2f579fa3c24e1af278d1fa3ffd3043d6260e21d2f7416 0B / 1.21kB                       9.4s
 => => sha256:6b7b6c7061b76cdb8601e18722d12ae3232f0ddcfa1d2983754abcc6ce0a8a83 954B / 954B                       0.7s
 => => sha256:3e2c181db1b0985ce357c7aaf48ac615f30f392cd15d5c5ba34c4faa1f4f39a2 626B / 626B                       0.8s
 => => sha256:589002ba0eaed121a1dbf42f6648f29e5be55d5c8a6ee0f8eaa0285cc21ac153 3.86MB / 3.86MB                   3.7s
 => => sha256:bca5d04786e112d958f100a66f8257b2aeefc14b64d81e405c3c44acff2fb000 1.86MB / 1.86MB                   2.7s
 => => extracting sha256:589002ba0eaed121a1dbf42f6648f29e5be55d5c8a6ee0f8eaa0285cc21ac153                        0.4s
 => => extracting sha256:bca5d04786e112d958f100a66f8257b2aeefc14b64d81e405c3c44acff2fb000                        0.3s
 => => extracting sha256:3e2c181db1b0985ce357c7aaf48ac615f30f392cd15d5c5ba34c4faa1f4f39a2                        0.1s
 => => extracting sha256:6b7b6c7061b76cdb8601e18722d12ae3232f0ddcfa1d2983754abcc6ce0a8a83                        0.1s
 => => extracting sha256:399d0898a94e0084f81499a3e3c29824357118c7ce551648ea1dbab813884661                        0.1s
 => => extracting sha256:955a8478f9aceb66cbf2f579fa3c24e1af278d1fa3ffd3043d6260e21d2f7416                        0.1s
 => => extracting sha256:6d397a54a185dd0b6638d1a3934b81daef7a140741e12697377d6279066f7ca7                        0.1s
 => => extracting sha256:5e7756927bef33a266e1221356d5da8553139cb80bc5b1b3827010811d9ea268                        1.0s
 => [2/2] COPY dist/ /usr/share/nginx/html                                                                       0.2s
 => exporting to image                                                                                           1.6s
 => => exporting layers                                                                                          1.0s
 => => exporting manifest sha256:62789207fa33a3b93a2ff1dce64bc6512a72e8ea1a3d8a2692643402400251f13               0.1s
 => => exporting config sha256:a334d385777263a1ea4c85cbfc1286719433f1cdcd63a40575b7af89b8075dde                  0.1s
 => => exporting attestation manifest sha256:1d1918d08fa3b2695cff6743b0a87b8aa8dd862114df80c86e0d289d0ce49550    0.1s
 => => exporting manifest list sha256:901d173fe4730dd0c17f015ba390f77fd663b0353e3ef2b33447da9ee03c0457           0.0s
 => => naming to docker.io/library/trend-app:latest                                                              0.0s
 => => unpacking to docker.io/library/trend-app:latest                                                           0.3s
root@Aravind:~/Trend#
```

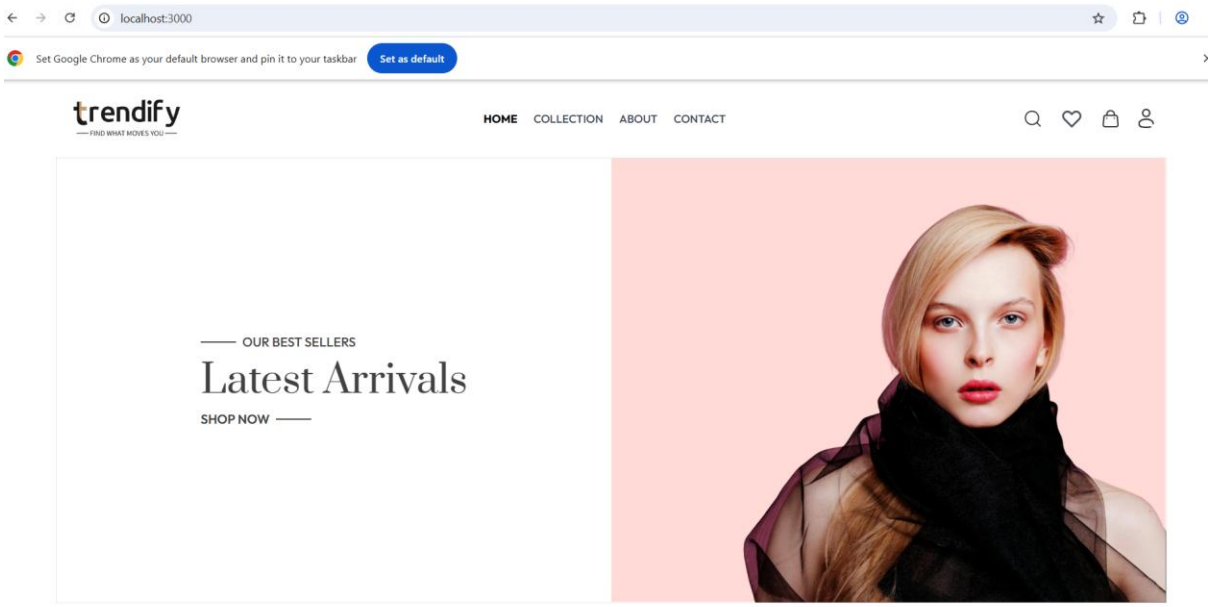
Docker image created:

```
root@Aravind:~/Trend# docker images
REPOSITORY   TAG       IMAGE ID       CREATED         SIZE
trend-app    latest    76fb947d3472   4 seconds ago   71.1MB
nginx        alpine    004f022f4ef5   5 weeks ago     61.9MB
root@Aravind:~/Trend#
```

Ran the docker in WSL and verified in browser:

```
root@Aravind:~/Trend# docker run -d -p 3000:80 --name trend-container trend-app
3bb6d0dcac0e716df09a695bea6856a64356354d5a7d901c675084c8a5d4000b
root@Aravind:~/Trend# docker ps
CONTAINER ID   IMAGE       COMMAND                  CREATED          STATUS          PORTS                                     NAMES
3bb6d0dcac0e   trend-app   "/docker-entrypoint.…"   53 seconds ago   Up 53 seconds   0.0.0.0:3000->80/tcp, [::]:3000->80/tcp   trend-container
root@Aravind:~/Trend# |
```

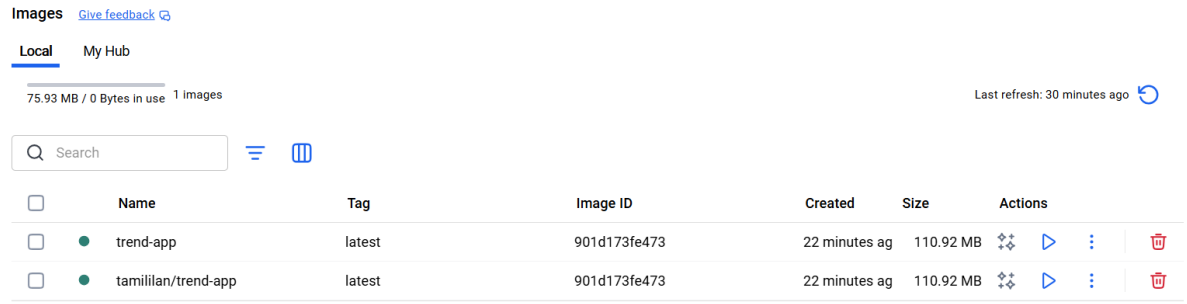
app is running in browser:

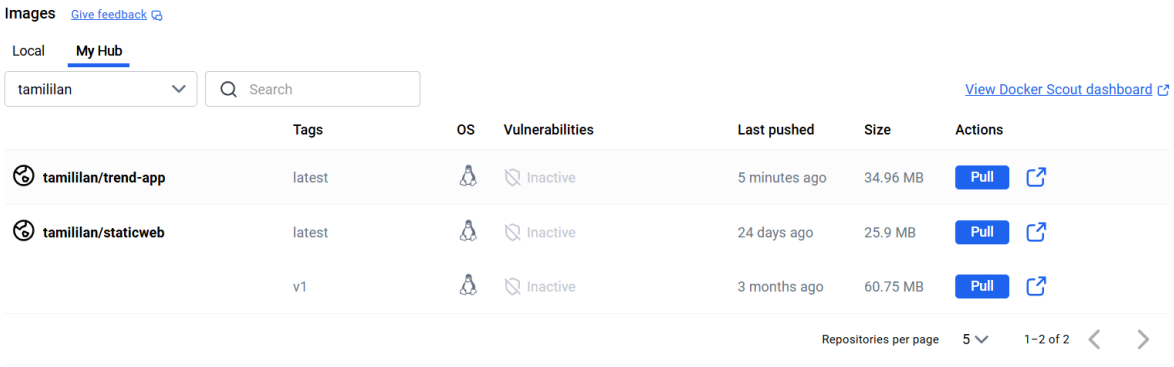


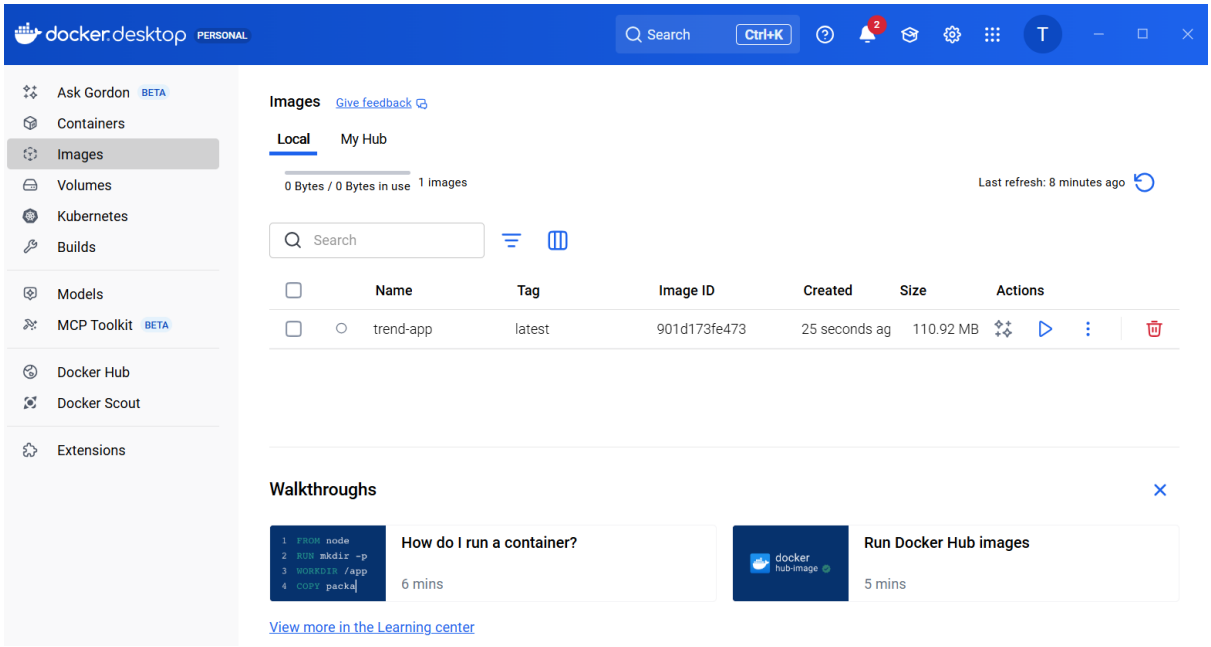
Tagged images with dockerhub username :

```
root@Aravind:~/Trend# docker tag trend-app tamililan/trend-app:latest
root@Aravind:~/Trend# docker images
REPOSITORY            TAG       IMAGE ID       CREATED          SIZE
trend-app             latest    901d173fe473   15 minutes ago   111MB
tamililan/trend-app   latest    901d173fe473   15 minutes ago   111MB
root@Aravind:~/Trend#
```

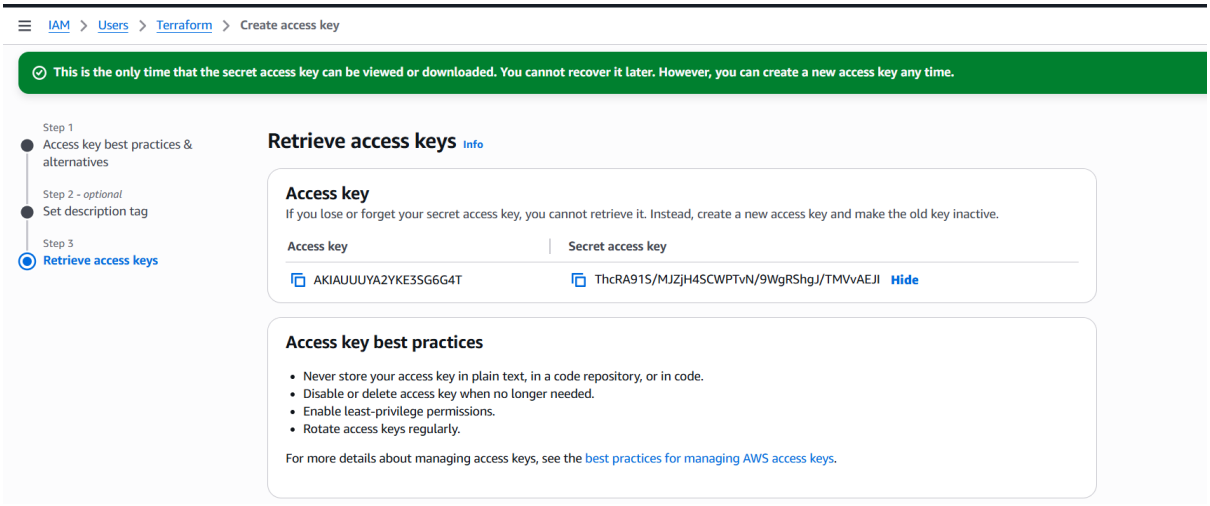
```
Login Succeeded
root@Aravind:~/Trend# docker push tamililan/trend-app:latest
The push refers to repository [docker.io/tamililan/trend-app]
5e7756927bef: Pushed
399d0898a94e: Pushed
bca5d04786e1: Pushed
6b7b6c7061b7: Pushed
589002ba0eae: Pushed
74de98cf0e76: Pushed
36cda81c417b: Pushed
3e2c181db1b0: Pushed
955a8478f9ac: Pushed
6d397a54a185: Pushed
latest: digest: sha256:901d173fe4730dd0c17f015ba390f77fd663b0353e3ef2b33447da9ee03c0457 size: 856
root@Aravind:~/Trend#
```







Created Access key for Terraform authentication to AWS from WSL:




```
root@Aravind:~/Trend# aws configure
AWS Access Key ID [****************LJV4]: AKIAUUUYA2YKE3SG6G4T
AWS Secret Access Key [****************20uE]: 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
Default region name [ap-south-1]:
Default output format [json]: json
root@Aravind:~/Trend#
```

```
root@Aravind:~/Trend# aws sts get-caller-identity
{
    "UserId": "AIDAUUUYA2YKOT6Y25QYI",
    "Account": "319220143636",
    "Arn": "arn:aws:iam::319220143636:user/Terraform"
}
root@Aravind:~/Trend#
```

Created main.tf file

```
    dnf update -y
    dnf install java-17-amazon-corretto -y

    wget -O /etc/yum.repos.d/jenkins.repo \
    https://pkg.jenkins.io/redhat-stable/jenkins.repo

    rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key

    dnf install jenkins -y

    systemctl daemon-reload
    systemctl enable jenkins
    systemctl start jenkins
  EOF
}

# ----------------------------
# EC2 Instance (Amazon Linux 2023)
# ----------------------------
resource "aws_instance" "jenkins_server" {
  ami                         = "ami-0317b0f0a0144b137" # Amazon Linux 2023 (ap-south-1)
  instance_type               = "t3.micro"
  subnet_id                   = aws_subnet.public_subnet.id
  vpc_security_group_ids      = [aws_security_group.jenkins_sg.id]
  iam_instance_profile        = aws_iam_instance_profile.jenkins_profile.name
  associate_public_ip_address = true
  key_name                    = "terraform-key"

  user_data = local.jenkins_user_data

  tags = {
    Name = "Jenkins-Server-AL2023"
  }
}

# ----------------------------
# Outputs
# ----------------------------
output "jenkins_public_ip" {
  value = aws_instance.jenkins_server.public_ip
}

output "jenkins_url" {
  value = "http://${aws_instance.jenkins_server.public_ip}:8080"
}

root@Aravind:~/Trend#
```

Initialized backend :

#terraform init

```
root@Aravind:~/Trend# terraform init
Initializing the backend...
Initializing provider plugins...
- Finding hashicorp/aws versions matching "~> 5.0"...
- Installing hashicorp/aws v5.100.0...
- Installed hashicorp/aws v5.100.0 (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.
root@Aravind:~/Trend#
```

Verified main.tf configuration :

#terraform validate

```
root@Aravind:~/Trend# terraform validate
Success! The configuration is valid.

root@Aravind:~/Trend#
```

Planned the configuration:

#terraform plan

```
root@Aravind:~/Trend# terraform plan

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_instance_profile.jenkins_profile will be created
  + resource "aws_iam_instance_profile" "jenkins_profile" {
      + arn         = (known after apply)
      + create_date = (known after apply)
      + id          = (known after apply)
      + name        = "jenkins-instance-profile"
      + name_prefix = (known after apply)
      + path        = "/"
      + role        = "jenkins-ec2-role"
      + tags_all    = (known after apply)
      + unique_id   = (known after apply)
    }

  # aws_iam_role.jenkins_ec2_role will be created
  + resource "aws_iam_role" "jenkins_ec2_role" {
      + arn                   = (known after apply)
      + assume_role_policy    = jsonencode(
            {
              + Statement = [
                  + {
                      + Action    = "sts:AssumeRole"
                      + Effect    = "Allow"
                      + Principal = {
                          + Service = "ec2.amazonaws.com"
                        }
                    },
                ]
              + Version   = "2012-10-17"
            }
        )
      + create_date           = (known after apply)
      + force_detach_policies = false
      + id                    = (known after apply)
      + managed_policy_arns   = (known after apply)
      + max_session_duration  = 3600
      + name                  = "jenkins-ec2-role"
      + name_prefix           = (known after apply)
      + path                  = "/"
```

```
# aws_iam_role_policy_attachment.admin_attach will be created
+ resource "aws_iam_role_policy_attachment" "admin_attach" {
    + id         = (known after apply)
    + policy_arn = "arn:aws:iam::aws:policy/AdministratorAccess"
    + role       = "jenkins-ec2-role"
  }

# aws_instance.jenkins_server will be created
+ resource "aws_instance" "jenkins_server" {
    + ami                                  = "ami-0317b0f0a0144b137"
    + arn                                  = (known after apply)
    + associate_public_ip_address          = true
    + availability_zone                    = (known after apply)
    + cpu_core_count                       = (known after apply)
    + cpu_threads_per_core                 = (known after apply)
    + disable_api_stop                     = (known after apply)
    + disable_api_termination              = (known after apply)
    + ebs_optimized                        = (known after apply)
    + enable_primary_ipv6                  = (known after apply)
    + get_password_data                    = false
    + host_id                              = (known after apply)
    + host_resource_group_arn              = (known after apply)
    + iam_instance_profile                 = "jenkins-instance-profile"
    + id                                   = (known after apply)
    + instance_initiated_shutdown_behavior = (known after apply)
    + instance_lifecycle                   = (known after apply)
    + instance_state                       = (known after apply)
    + instance_type                        = "t3.micro"
    + ipv6_address_count                   = (known after apply)
    + ipv6_addresses                       = (known after apply)
    + key_name                             = "terraform-key"
    + monitoring                           = (known after apply)
    + outpost_arn                          = (known after apply)
    + password_data                        = (known after apply)
    + placement_group                      = (known after apply)
    + placement_partition_number           = (known after apply)
    + primary_network_interface_id         = (known after apply)
    + private_dns                          = (known after apply)
    + private_ip                           = (known after apply)
    + public_dns                           = (known after apply)
    + public_ip                            = (known after apply)
    + secondary_private_ips                = (known after apply)
    + security_groups                      = (known after apply)
    + source_dest_check                    = true
    + spot_instance_request_id             = (known after apply)
```

```
    + tags                                 = {
        + "Name" = "Jenkins-Server-AL2023"
      }
    + tags_all                             = {
        + "Name" = "Jenkins-Server-AL2023"
      }
    + tenancy                              = (known after apply)
    + user_data                            = "e88a37889d931af8c79d28948ecd6bcbeedc37af"
    + user_data_base64                     = (known after apply)
    + user_data_replace_on_change          = false
    + vpc_security_group_ids               = (known after apply)

    + capacity_reservation_specification (known after apply)

    + cpu_options (known after apply)

    + ebs_block_device (known after apply)

    + enclave_options (known after apply)

    + ephemeral_block_device (known after apply)

    + instance_market_options (known after apply)

    + maintenance_options (known after apply)

    + metadata_options (known after apply)

    + network_interface (known after apply)

    + private_dns_name_options (known after apply)

    + root_block_device (known after apply)
  }

# aws_internet_gateway.igw will be created
+ resource "aws_internet_gateway" "igw" {
    + arn      = (known after apply)
    + id       = (known after apply)
    + owner_id = (known after apply)
    + tags     = {
```

```
# aws_internet_gateway.igw will be created
+ resource "aws_internet_gateway" "igw" {
    + arn      = (known after apply)
    + id       = (known after apply)
    + owner_id = (known after apply)
    + tags     = {
        + "Name" = "jenkins-igw"
      }
    + tags_all = {
        + "Name" = "jenkins-igw"
      }
    + vpc_id   = (known after apply)
  }

# aws_route_table.public_rt will be created
+ resource "aws_route_table" "public_rt" {
    + arn              = (known after apply)
    + id               = (known after apply)
    + owner_id         = (known after apply)
    + propagating_vgws = (known after apply)
    + route            = [
        + {
            + cidr_block                = "0.0.0.0/0"
            + gateway_id                = (known after apply)
              # (11 unchanged attributes hidden)
          },
      ]
    + tags             = {
        + "Name" = "jenkins-public-rt"
      }
    + tags_all         = {
        + "Name" = "jenkins-public-rt"
      }
    + vpc_id           = (known after apply)
  }

# aws_route_table_association.rta will be created
+ resource "aws_route_table_association" "rta" {
    + id             = (known after apply)
    + route_table_id = (known after apply)
    + subnet_id      = (known after apply)
  }
```

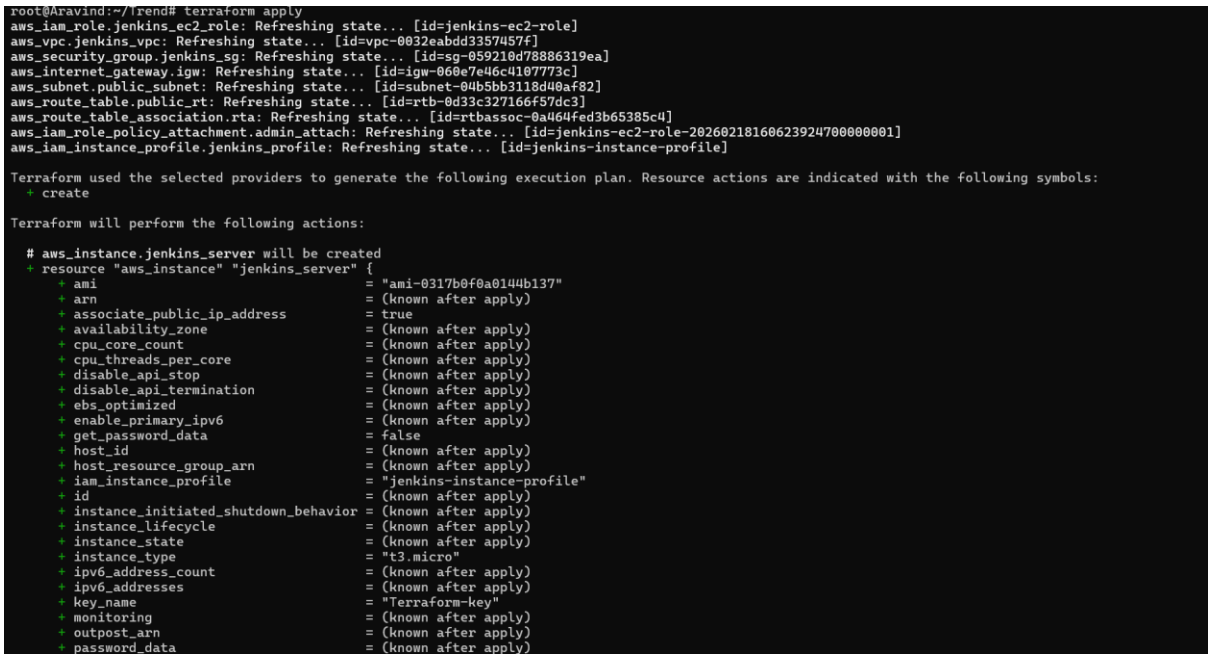
```
# aws_security_group.jenkins_sg will be created
+ resource "aws_security_group" "jenkins_sg" {
    + arn                    = (known after apply)
    + description            = "Managed by Terraform"
    + egress                 = [
        + {
            + cidr_blocks      = [
                + "0.0.0.0/0",
              ]
            + from_port        = 0
            + ipv6_cidr_blocks = []
            + prefix_list_ids  = []
            + protocol         = "-1"
            + security_groups  = []
            + self             = false
            + to_port          = 0
              # (1 unchanged attribute hidden)
          },
      ]
    + id                     = (known after apply)
    + ingress                = [
        + {
            + cidr_blocks      = [
                + "0.0.0.0/0",
              ]
            + description      = "Jenkins UI"
            + from_port        = 8080
            + ipv6_cidr_blocks = []
            + prefix_list_ids  = []
            + protocol         = "tcp"
            + security_groups  = []
            + self             = false
            + to_port          = 8080
          },
        + {
            + cidr_blocks      = [
                + "0.0.0.0/0",
              ]
            + description      = "SSH"
            + from_port        = 22
            + ipv6_cidr_blocks = []
            + prefix_list_ids  = []
            + protocol         = "tcp"
            + security_groups  = []
```

```
  # aws_subnet.public_subnet will be created
  + resource "aws_subnet" "public_subnet" {
      + arn                                            = (known after apply)
      + assign_ipv6_address_on_creation                = false
      + availability_zone                              = "ap-south-1a"
      + availability_zone_id                           = (known after apply)
      + cidr_block                                     = "10.0.1.0/24"
      + enable_dns64                                   = false
      + enable_resource_name_dns_a_record_on_launch    = false
      + enable_resource_name_dns_aaaa_record_on_launch = false
      + id                                             = (known after apply)
      + ipv6_cidr_block_association_id                 = (known after apply)
      + ipv6_native                                    = false
      + map_public_ip_on_launch                        = true
      + owner_id                                       = (known after apply)
      + private_dns_hostname_type_on_launch            = (known after apply)
      + tags                                           = {
          + "Name" = "jenkins-public-subnet"
        }
      + tags_all                                       = {
          + "Name" = "jenkins-public-subnet"
        }
      + vpc_id                                         = (known after apply)
    }

  # aws_vpc.jenkins_vpc will be created
  + resource "aws_vpc" "jenkins_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                 = true
      + 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)
```

```
  # aws_vpc.jenkins_vpc will be created
  + resource "aws_vpc" "jenkins_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                 = true
      + 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)
      + tags                                 = {
          + "Name" = "jenkins-vpc"
        }
      + tags_all                             = {
          + "Name" = "jenkins-vpc"
        }
    }

Plan: 10 to add, 0 to change, 0 to destroy.

Changes to Outputs:
  + jenkins_public_ip = (known after apply)
  + jenkins_url       = (known after apply)

─────────────────────────────────────────────────────────────────────────────

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.
root@Aravind:~/Trend#
```

Creating infrastructure:

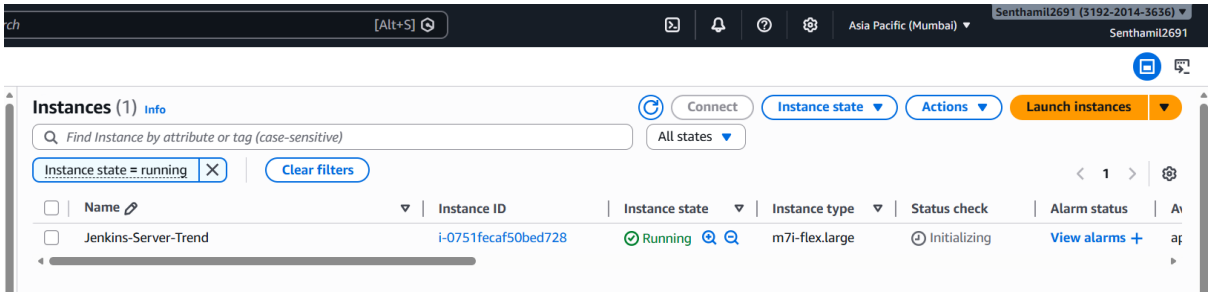
terraform apply

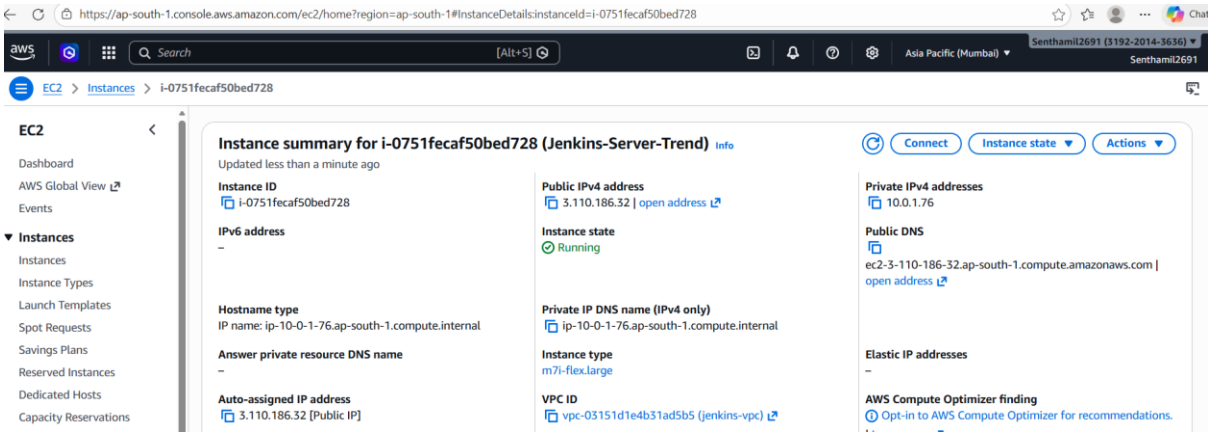


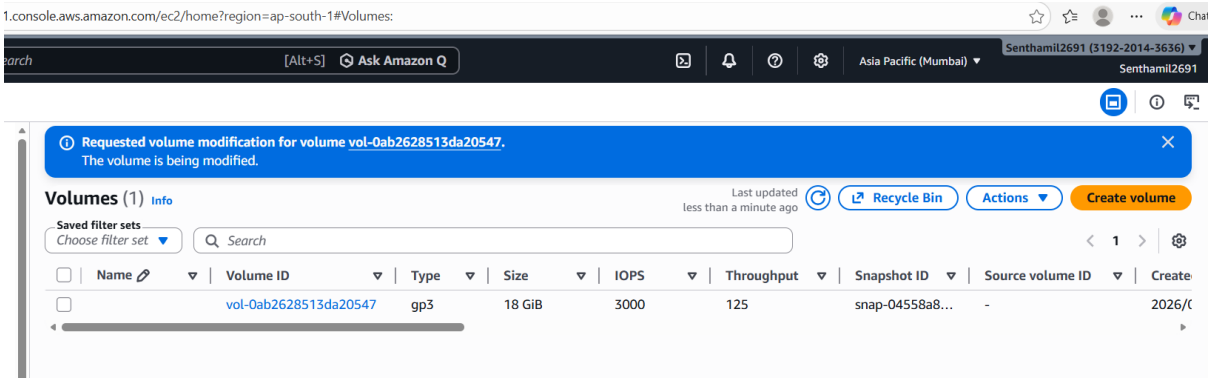
```
root@Aravind:~/Trend# terraform apply
aws_iam_role.jenkins_ec2_role: Refreshing state... [id=jenkins-ec2-role]
aws_vpc.jenkins_vpc: Refreshing state... [id=vpc-0032eabdd3357457f]
aws_security_group.jenkins_sg: Refreshing state... [id=sg-059210d78886319ea]
aws_internet_gateway.igw: Refreshing state... [id=igw-060e7e46c4107773c]
aws_subnet.public_subnet: Refreshing state... [id=subnet-04b5bb3118d40af82]
aws_route_table.public_rt: Refreshing state... [id=rtb-0d33c327166f57dc3]
aws_route_table_association.rta: Refreshing state... [id=rtbassoc-0a464fed3b65385c4]
aws_iam_role_policy_attachment.admin_attach: Refreshing state... [id=jenkins-ec2-role-20260218160623924700000001]
aws_iam_instance_profile.jenkins_profile: Refreshing state... [id=jenkins-instance-profile]

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_instance.jenkins_server will be created
  + resource "aws_instance" "jenkins_server" {
      + ami                                  = "ami-0317b0f0a0144b137"
      + arn                                  = (known after apply)
      + associate_public_ip_address          = true
      + availability_zone                    = (known after apply)
      + cpu_core_count                       = (known after apply)
      + cpu_threads_per_core                 = (known after apply)
      + disable_api_stop                     = (known after apply)
      + disable_api_termination              = (known after apply)
      + ebs_optimized                        = (known after apply)
      + enable_primary_ipv6                  = (known after apply)
      + get_password_data                    = false
      + host_id                              = (known after apply)
      + host_resource_group_arn              = (known after apply)
      + iam_instance_profile                 = "jenkins-instance-profile"
      + id                                   = (known after apply)
      + instance_initiated_shutdown_behavior = (known after apply)
      + instance_lifecycle                   = (known after apply)
      + instance_state                       = (known after apply)
      + instance_type                        = "t3.micro"
      + ipv6_address_count                   = (known after apply)
      + ipv6_addresses                       = (known after apply)
      + key_name                             = "Terraform-key"
      + monitoring                           = (known after apply)
      + outpost_arn                          = (known after apply)
      + password_data                        = (known after apply)
```

```
      + spot_instance_request_id             = (known after apply)
      + subnet_id                            = "subnet-04b5bb3118d40af82"
      + tags                                 = {
          + "Name" = "Jenkins-Server-Trend"
        }
      + tags_all                             = {
          + "Name" = "Jenkins-Server-Trend"
        }
      + tenancy                              = (known after apply)
      + user_data                            = "e88a37889d931af8c79d28948ecd6bcbeedc37af"
      + user_data_base64                     = (known after apply)
      + user_data_replace_on_change          = false
      + vpc_security_group_ids               = [
          + "sg-059210d78886319ea",
        ]

      + capacity_reservation_specification (known after apply)

      + cpu_options (known after apply)

      + ebs_block_device (known after apply)

      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)
    }

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

Changes to Outputs:
  + jenkins_public_ip = (known after apply)
  + jenkins_url       = (known after apply)
```

```
aws_vpc.jenkins_vpc: Creating...
aws_iam_role.jenkins_ec2_role: Creation complete after 3s [id=jenkins-ec2-role]
aws_iam_role_policy_attachment.admin_attach: Creating...
aws_iam_instance_profile.jenkins_profile: Creating...
aws_iam_role_policy_attachment.admin_attach: Creation complete after 2s [id=jenkins-ec2-role-20260221061946788900000001]
aws_vpc.jenkins_vpc: Still creating... [00m10s elapsed]
aws_iam_instance_profile.jenkins_profile: Creation complete after 10s [id=jenkins-instance-profile]
aws_vpc.jenkins_vpc: Creation complete after 13s [id=vpc-03151d1e4b31ad5b5]
aws_internet_gateway.igw: Creating...
aws_subnet.public_subnet: Creating...
aws_security_group.jenkins_sg: Creating...
aws_internet_gateway.igw: Creation complete after 1s [id=igw-03370b0e495b44f6b]
aws_route_table.public_rt: Creating...
aws_route_table.public_rt: Creation complete after 2s [id=rtb-0e8cb76a8b3a958c7]
aws_security_group.jenkins_sg: Creation complete after 4s [id=sg-07ea338efb0e0786e]
aws_subnet.public_subnet: Still creating... [00m10s elapsed]
aws_subnet.public_subnet: Creation complete after 12s [id=subnet-0f0d11ad8b6a3ab5c]
aws_route_table_association.rta: Creating...
aws_instance.jenkins_server: Creating...
aws_route_table_association.rta: Creation complete after 1s [id=rtbassoc-0f2b8074d65f63006]
aws_instance.jenkins_server: Still creating... [00m11s elapsed]
aws_instance.jenkins_server: Creation complete after 14s [id=i-0751fecaf50bed728]

Apply complete! Resources: 10 added, 0 changed, 0 destroyed.

Outputs:

jenkins_public_ip = "3.110.186.32"
jenkins_url = "http://3.110.186.32:8080"
root@Aravind:~/Trend#
```







Increased rootFS size to 18G:

#growpart /dev/nvme0n1 1

#xfs_growfs -d /

```
[root@ip-10-0-1-76 ~]# growpart /dev/nvme0n1 1
CHANGED: partition=1 start=24576 old: size=16752607 end=16777183 new: size=37724127 end=37748703
[root@ip-10-0-1-76 ~]# df -hT
Filesystem       Type      Size  Used Avail Use% Mounted on
devtmpfs         devtmpfs  4.0M     0  4.0M   0% /dev
tmpfs            tmpfs     3.9G     0  3.9G   0% /dev/shm
tmpfs            tmpfs     1.6G  616K  1.6G   1% /run
/dev/nvme0n1p1   xfs       8.0G  2.1G  5.9G  26% /
tmpfs            tmpfs     3.9G  4.8M  3.8G   1% /tmp
/dev/nvme0n1p128 vfat       10M  1.3M  8.7M  13% /boot/efi
tmpfs            tmpfs     779M     0  779M   0% /run/user/1000
[root@ip-10-0-1-76 ~]# xfs_growfs -d /
meta-data=/dev/nvme0n1p1         isize=512    agcount=2, agsize=1047040 blks
         =                       sectsz=4096  attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1    bigtime=1 inobtcount=1 nrext64=0
         =                       exchange=0
data     =                       bsize=4096   blocks=2094075, imaxpct=25
         =                       sunit=128    swidth=128 blks
naming   =version 2              bsize=16384  ascii-ci=0, ftype=1, parent=0
log      =internal log           bsize=4096   blocks=16384, version=2
         =                       sectsz=4096  sunit=4 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
data blocks changed from 2094075 to 4715515
[root@ip-10-0-1-76 ~]# df -hT
Filesystem       Type      Size  Used Avail Use% Mounted on
devtmpfs         devtmpfs  4.0M     0  4.0M   0% /dev
tmpfs            tmpfs     3.9G     0  3.9G   0% /dev/shm
tmpfs            tmpfs     1.6G  616K  1.6G   1% /run
/dev/nvme0n1p1   xfs        18G  2.2G   16G  12% /
tmpfs            tmpfs     3.9G  4.8M  3.8G   1% /tmp
/dev/nvme0n1p128 vfat       10M  1.3M  8.7M  13% /boot/efi
tmpfs            tmpfs     779M     0  779M   0% /run/user/1000
```

Increased swap size to 4G:

swapoff -a

fallocate -l 4G /swapfile

dd if=/dev/zero of=/swapfile bs=1M count=4096

chmod 600 /swapfile

mkswap /swapfile

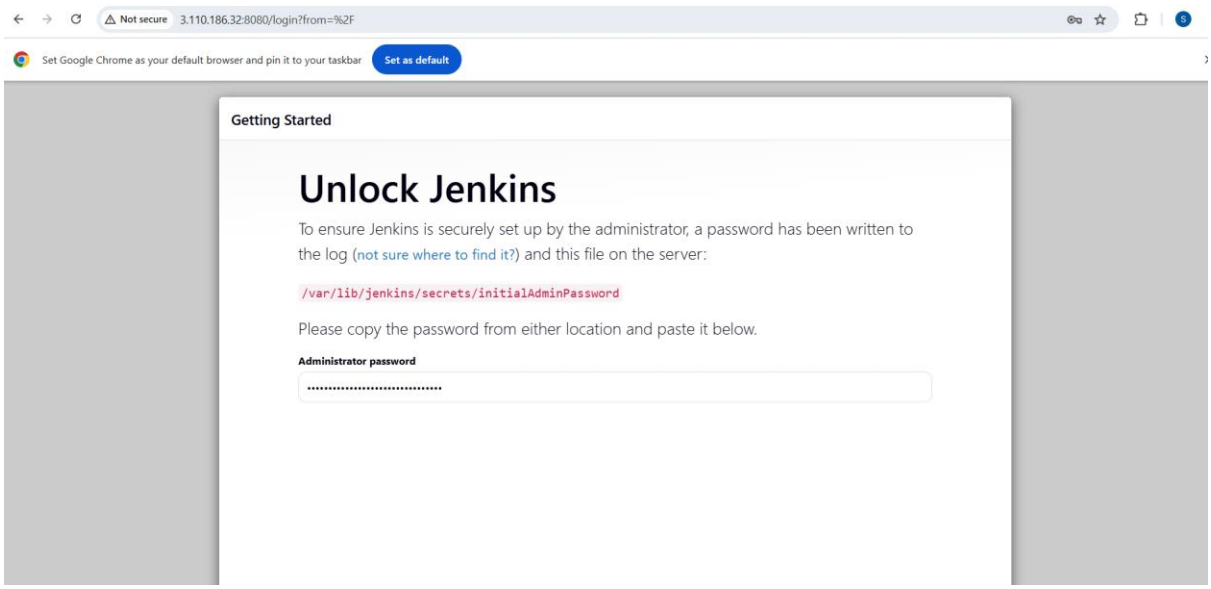
swapon /swapfile

echo '/swapfile none swap sw 0 0' | sudo tee -a /etc/fstab

```
[root@ip-10-0-1-76 ~]# free -h
               total        used        free      shared  buff/cache   available
Mem:           7.6Gi       951Mi       5.8Gi       5.0Mi       855Mi       6.4Gi
Swap:             0B          0B          0B
[root@ip-10-0-1-76 ~]# swapoff -a
[root@ip-10-0-1-76 ~]# fallocate -l 4G /swapfile
[root@ip-10-0-1-76 ~]# dd if=/dev/zero of=/swapfile bs=1M count=4096
4096+0 records in
4096+0 records out
4294967296 bytes (4.3 GB, 4.0 GiB) copied, 22.515 s, 191 MB/s
[root@ip-10-0-1-76 ~]# chmod 600 /swapfile
[root@ip-10-0-1-76 ~]# mkswap /swapfile
Setting up swapspace version 1, size = 4 GiB (4294963200 bytes)
no label, UUID=ff85caab-9c2b-44f3-ab7d-8c1393b4e57f
[root@ip-10-0-1-76 ~]# swapon /swapfile
[root@ip-10-0-1-76 ~]# echo '/swapfile none swap sw 0 0' | sudo tee -a /etc/fstab
/swapfile none swap sw 0 0
[root@ip-10-0-1-76 ~]#
```

http://3.110.186.32:8080

Launched Jenkins from the url :

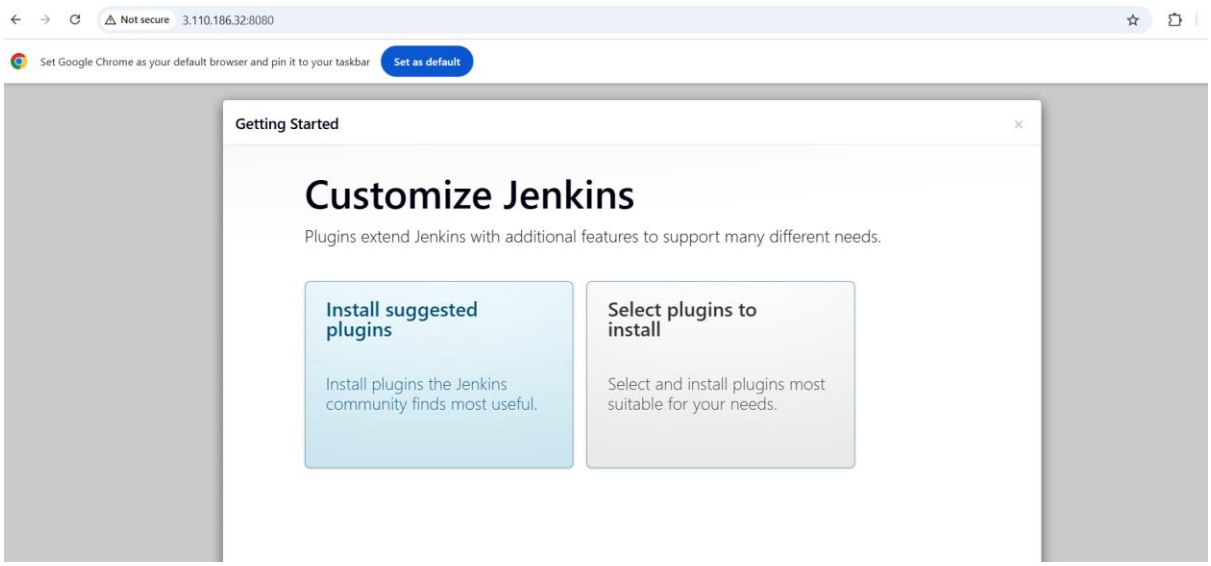


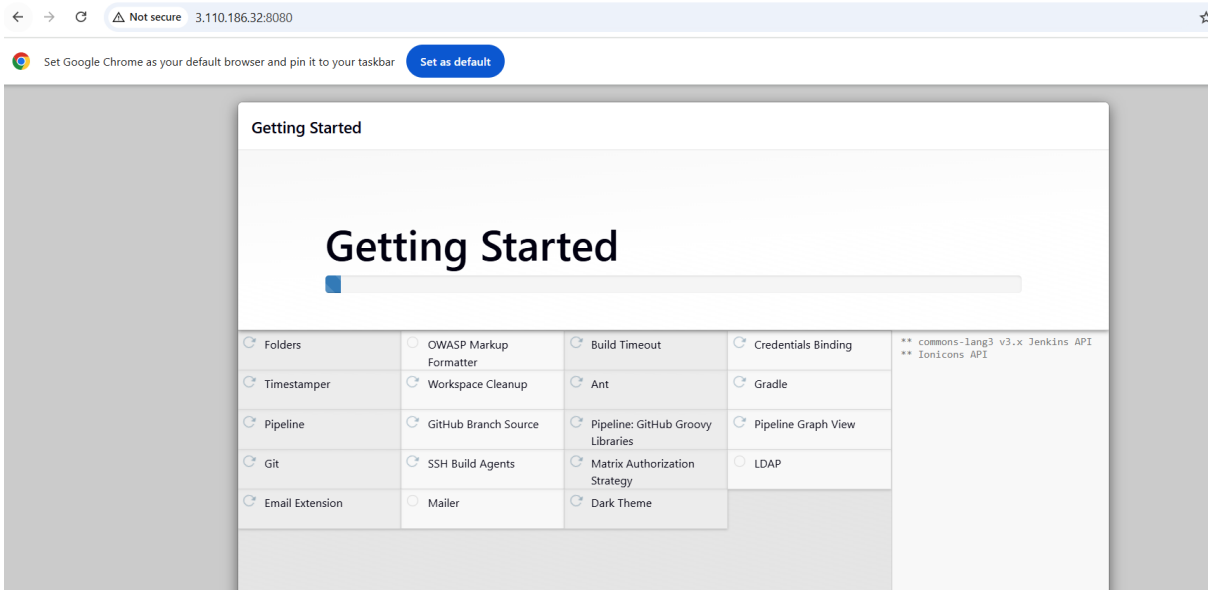
Retrieved Jenkins password from EC2 and launched Jenkins in browser:

[root@ip-10-0-1-76 ~]# cat /var/lib/jenkins/secrets/initialAdminPassword

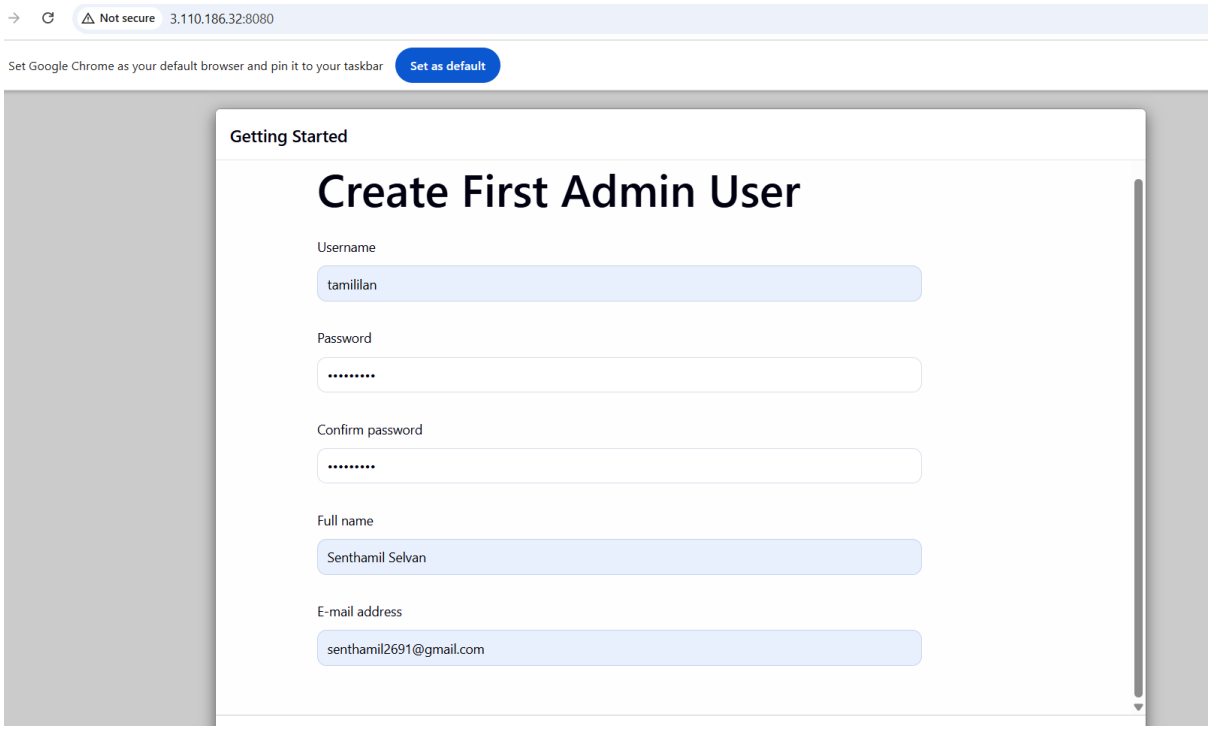
e61a10be40ff43a48ceaac91132d9fa2

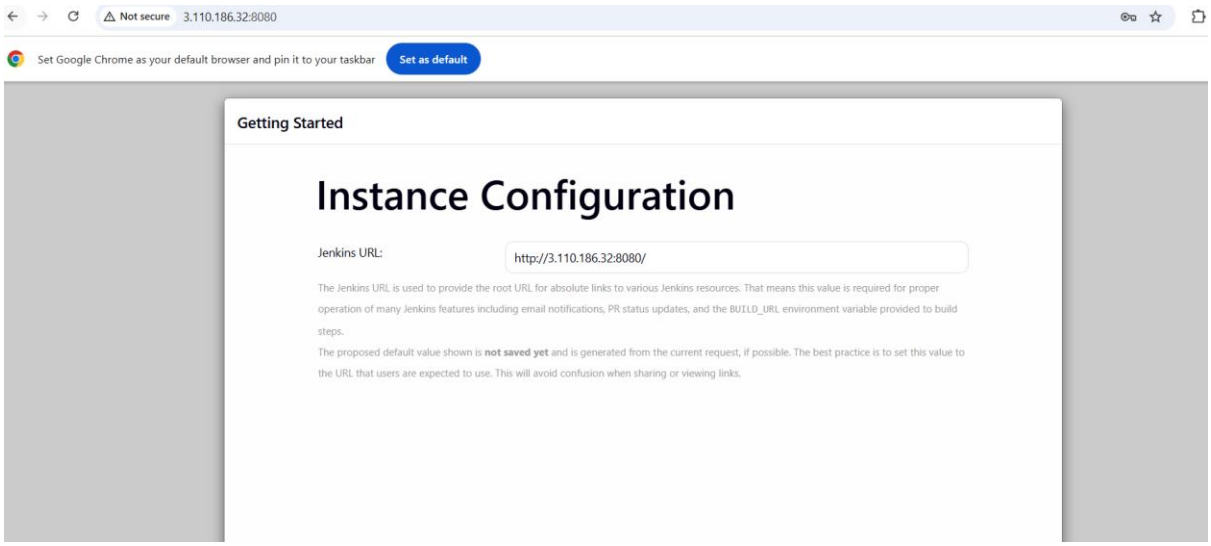
[root@ip-10-0-1-76 ~]#

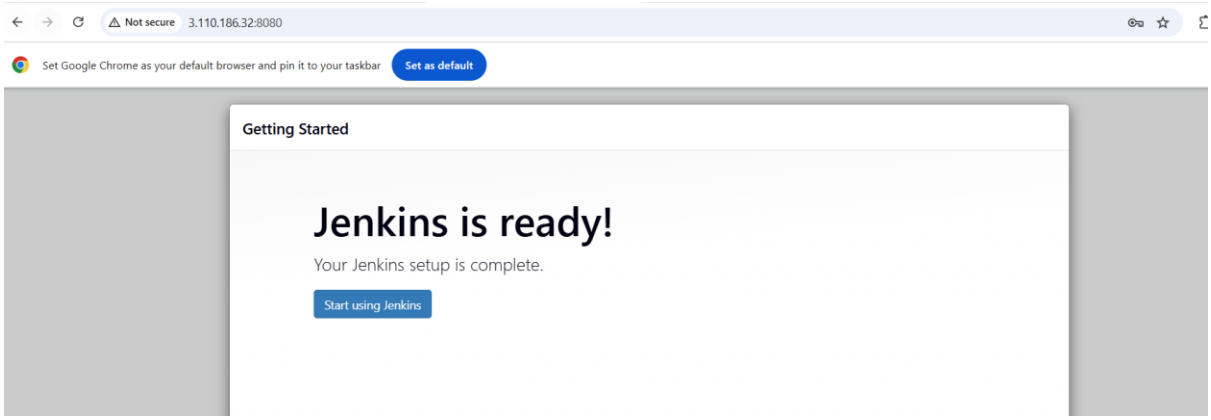




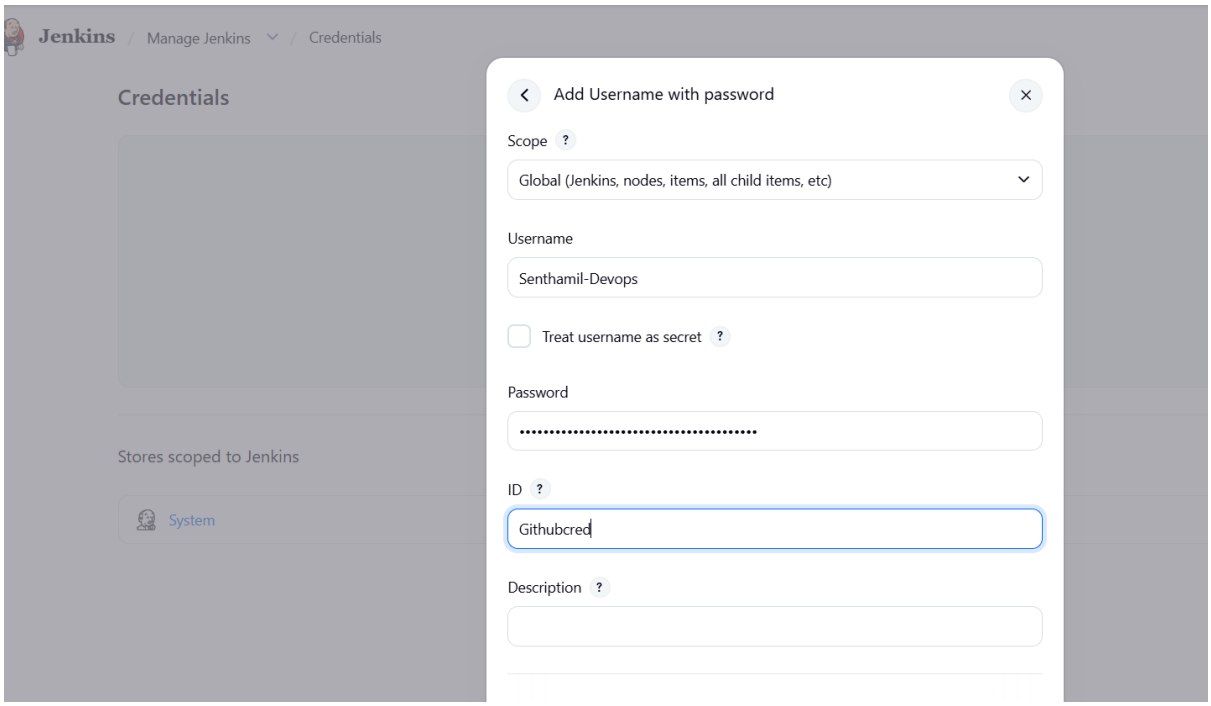
Created user in Jenkins:

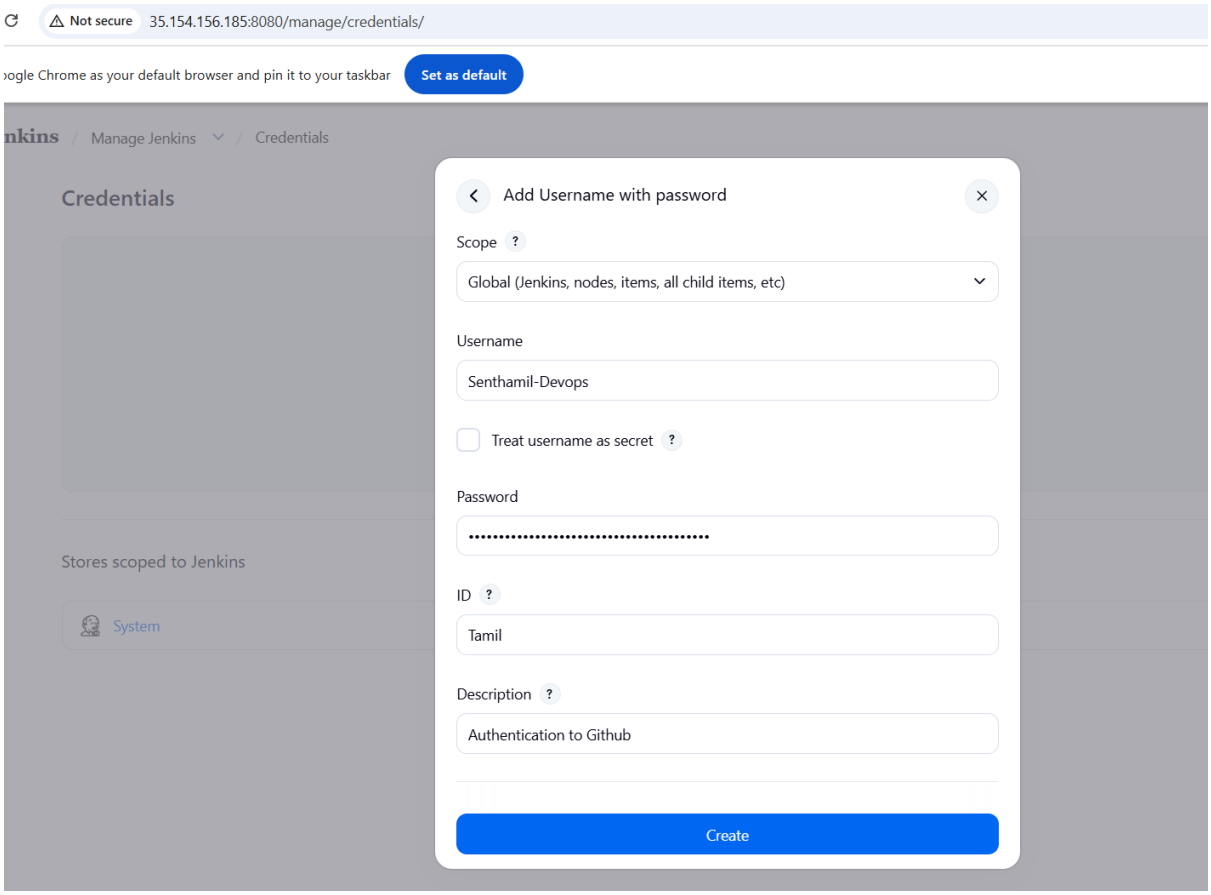

Not secure 3.110.186.32:8080
Set Google Chrome as your default browser and pin it to your taskbar
Set as default
Getting Started
Create First Admin User
Username
tamililan
Password
Confirm password
Full name
Senthamil Selvan
E-mail address
senthamil2691@gmail.com


Not secure 3.110.186.32:8080
Set Google Chrome as your default browser and pin it to your taskbar
Set as default
Getting Started
Instance Configuration
Jenkins URL:
http://3.110.186.32:8080/
The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD_URL environment variable provided to build steps.
The proposed default value shown is not saved yet and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

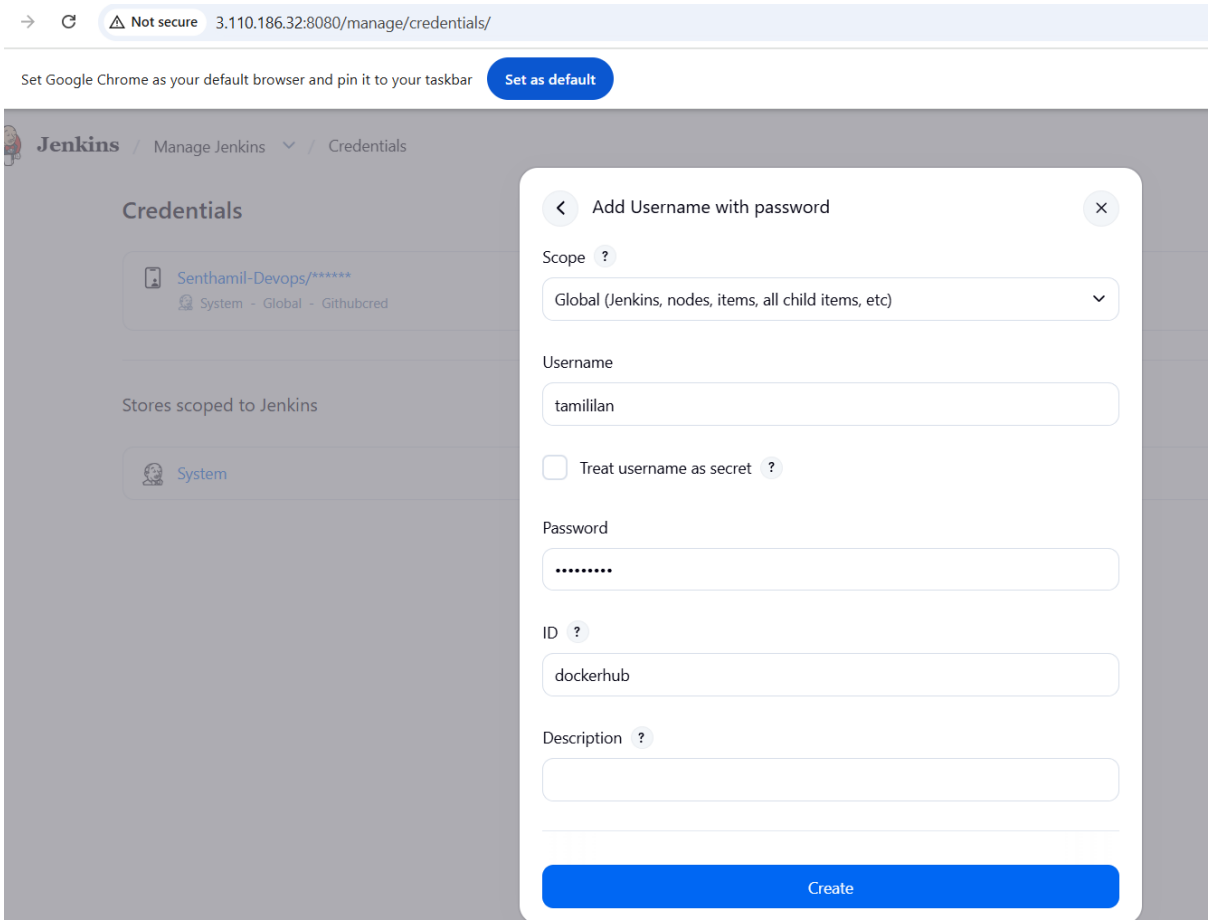


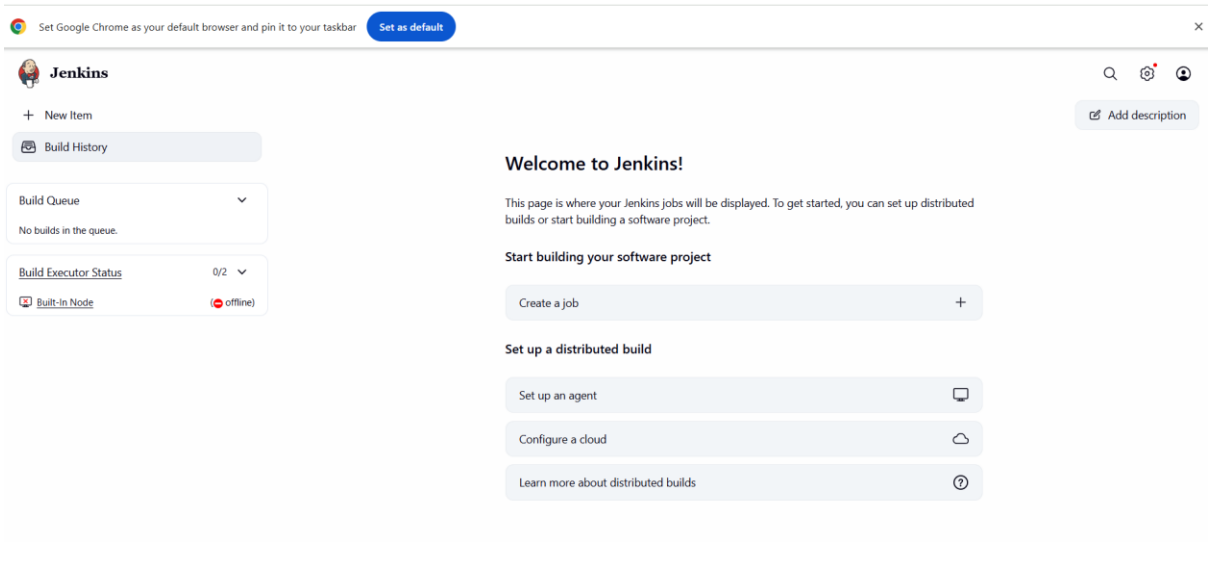
Added git credential to Jenkins:

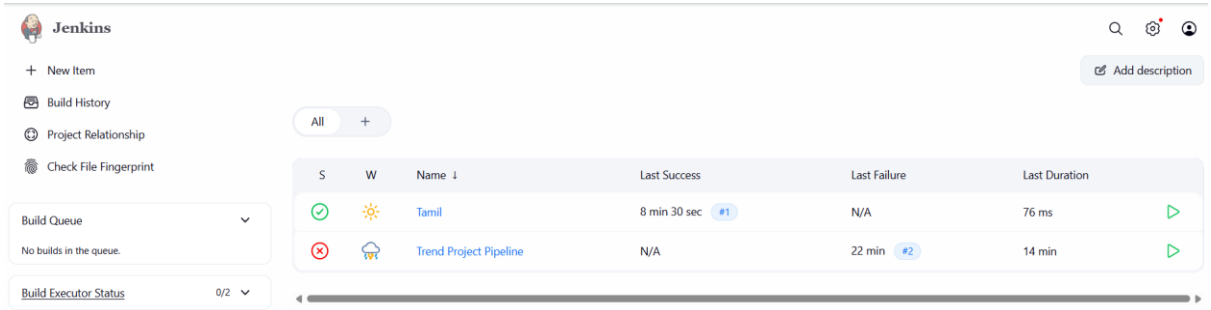


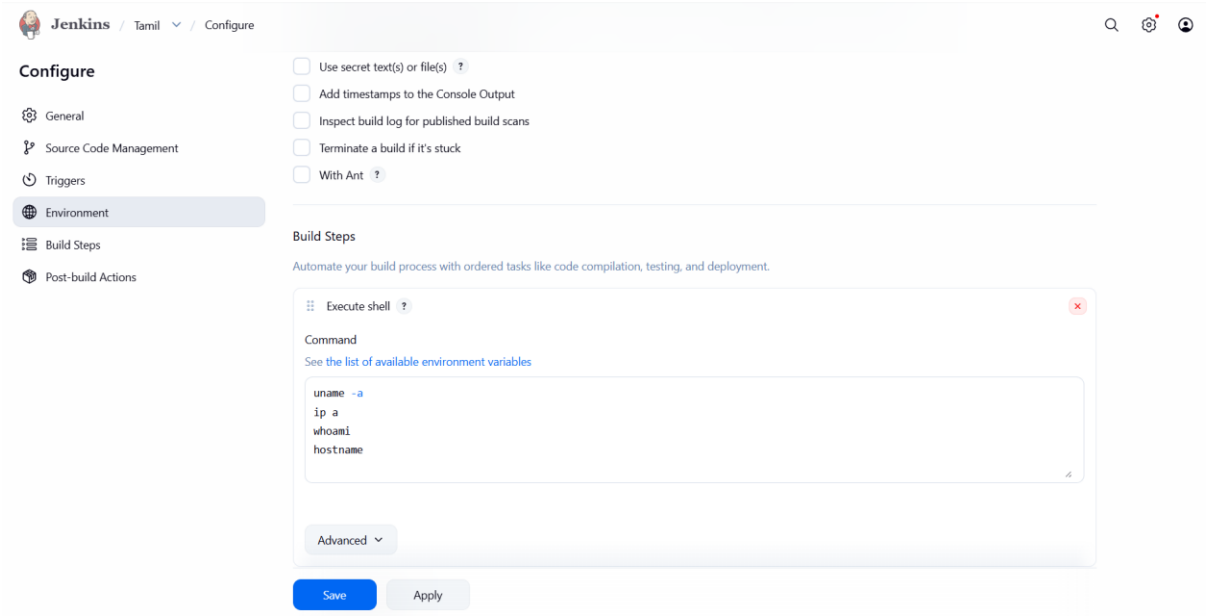


Added dockerhub credential to Jenkins:


Not secure 3.110.186.32:8080/manage/credentials/
Set Google Chrome as your default browser and pin it to your taskbar
Set as default
Jenkins / Manage Jenkins / Credentials
Credentials
Senthamil-Devops/******
System - Global - Githubcred
Stores scoped to Jenkins
System
Add Username with password
Scope
Global (Jenkins, nodes, items, all child items, etc)
Username
tamililan
Treat username as secret
Password
ID
dockerhub
Description
Create


Set Google Chrome as your default browser and pin it to your taskbar
Set as default
Jenkins
New Item
Build History
Build Queue
No builds in the queue.
Build Executor Status 0/2
Built-In Node (offline)
Add description
Welcome to Jenkins!
This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.
Start building your software project
Create a job
Set up a distributed build
Set up an agent
Configure a cloud
Learn more about distributed builds


Jenkins
New Item
Build History
Project Relationship
Check File Fingerprint
Build Queue
No builds in the queue.
Build Executor Status 0/2
Add description
All
S W Name Last Success Last Failure Last Duration
Tamil 8 min 30 sec #1 N/A 76 ms
Trend Project Pipeline N/A 22 min #2 14 min



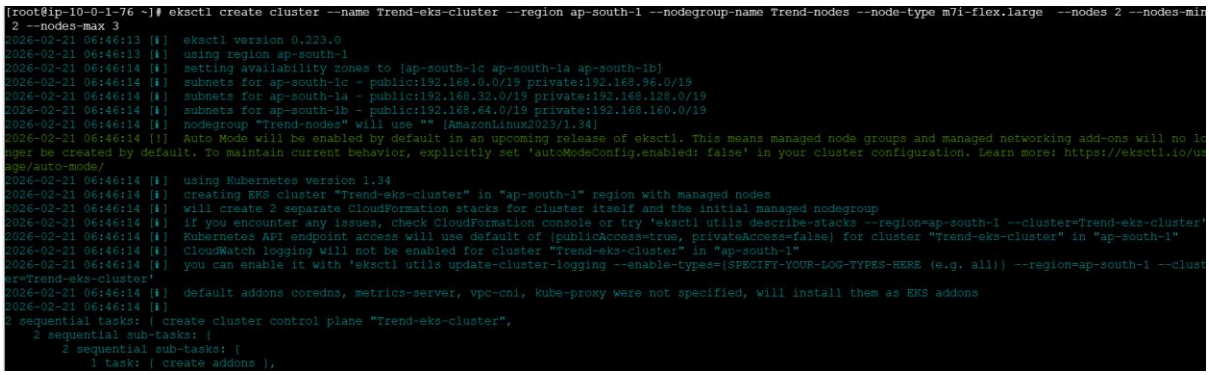
Installed Eksctl and kubectl in EC2:

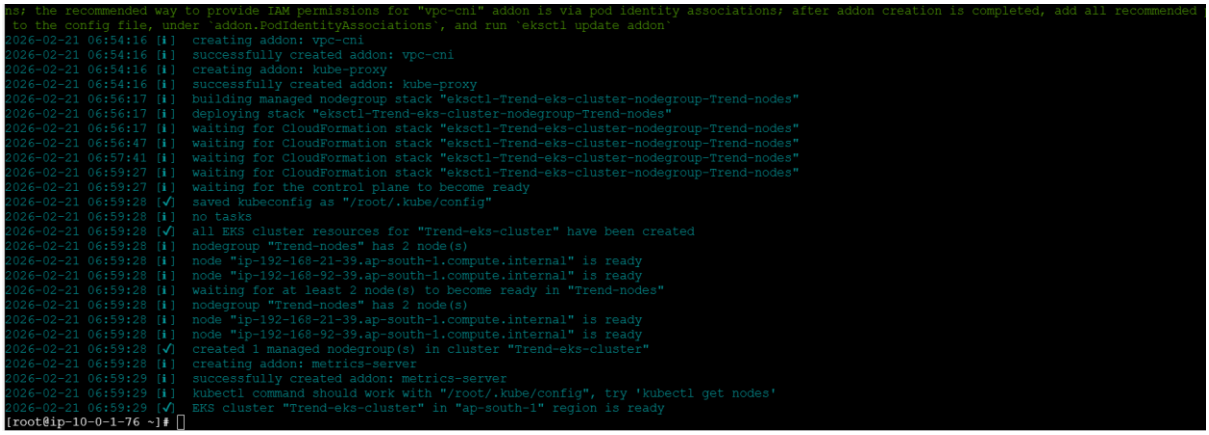
```
[root@ip-10-0-1-76 ~]# curl -LO https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_Linux_amd64.tar.gz
tar -xzf eksctl_Linux_amd64.tar.gz
sudo mv eksctl /usr/local/bin
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100 35.1M  100 35.1M    0     0  38.8M      0 --:--:-- --:--:-- --:--:-- 38.8M
[root@ip-10-0-1-76 ~]# eksctl version
0.223.0
[root@ip-10-0-1-76 ~]# sudo chmod +x /usr/local/bin/eksctl
[root@ip-10-0-1-76 ~]# curl -LO https://dl.k8s.io/release/v1.29.0/bin/linux/amd64/kubectl
chmod +x kubectl
sudo mv kubectl /usr/local/bin/
kubectl version --client
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 47.4M  100 47.4M    0     0   413M      0 --:--:-- --:--:-- --:--:--  412M
error: extra arguments: [--client]
[root@ip-10-0-1-76 ~]# kubectl version --client
Client Version: v1.29.0
Kustomize Version: v5.0.4-0.20230601165947-6ce0bf390ce3
[root@ip-10-0-1-76 ~]#
```

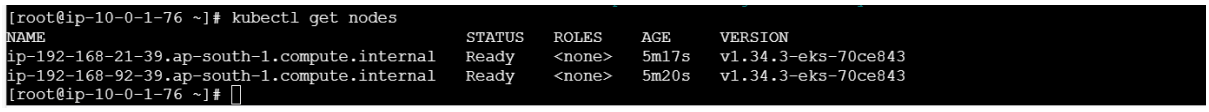
i-0751fecaf50bed728 (Jenkins-Server-Trend)

PublicIPs: 3.110.186.32 PrivateIPs: 10.0.1.76

eksctl create cluster --name Trend-eks-cluster --region ap-south-1 --nodegroup-name Trend-nodes --node-type m7i-flex.large --nodes 2 --nodes-min 2 --nodes-max 3







Created deployment.yaml and service.yaml

```
root@Aravind:~/Trend# cat deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: trend-app
spec:
  replicas: 2
  selector:
    matchLabels:
      app: trend
  template:
    metadata:
      labels:
        app: trend
    spec:
      containers:
      - name: trend
        image: tamililan/trend-app:latest
          ports:
        - containerPort: 80
root@Aravind:~/Trend# cat service.yaml
apiVersion: v1
kind: Service
metadata:
  name: trend-service
spec:
  type: LoadBalancer
  selector:
    app: trend
  ports:
    - port: 80
      targetPort: 80
root@Aravind:~/Trend#
```

Initiated git:

```
root@Aravind:~/Trend# git init
Reinitialized existing Git repository in /root/Trend/.git/
root@Aravind:~/Trend# git add .
root@Aravind:~/Trend# # Terraform
```

Added gitignore file:

```
root@Aravind:~/Trend# ls -la
total 88
drwxr-xr-x  5 root root  4096 Feb 18 17:37 .
drwx------ 13 root root  4096 Feb 18 17:37 ..
drwxr-xr-x  8 root root  4096 Feb 18 17:44 .git
-rw-r--r--  1 root root   235 Feb 18 17:37 .gitignore
drwxr-xr-x  3 root root  4096 Feb 18 15:46 .terraform
-rw-r--r--  1 root root  1407 Feb 18 15:47 .terraform.lock.hcl
-rw-r--r--  1 root root   703 Feb 18 17:35 Jenkinsfile
-rw-r--r--  1 root root   325 Feb 18 14:54 deployment.yaml
drwxr-xr-x  3 root root  4096 Feb 17 14:58 dist
-rw-r--r--  1 root root    61 Feb 18 14:17 dockerfile
-rw-r--r--  1 root root  4219 Feb 18 16:30 main.tf
-rw-r--r--  1 root root   160 Feb 18 14:55 service.yaml
-rw-r--r--  1 root root 17865 Feb 18 16:33 terraform.tfstate
-rw-r--r--  1 root root 12592 Feb 18 16:33 terraform.tfstate.backup
root@Aravind:~/Trend#
```

Commited the code:

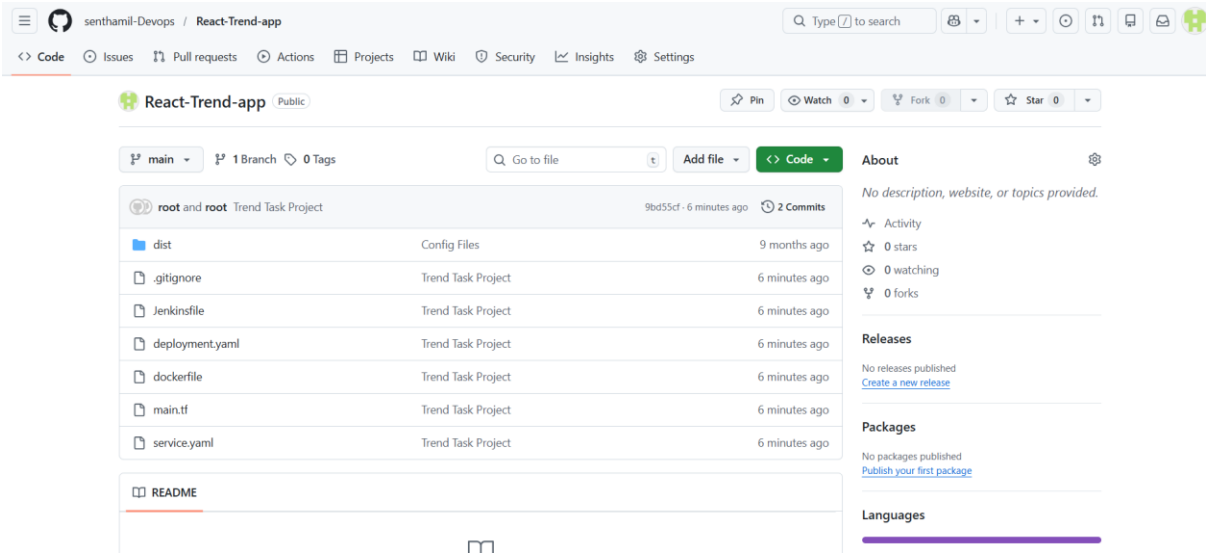
```
root@Aravind:~/Trend# git commit -m "Trend Task Project"
[main 9bd55cf] Trend Task Project
 Committer: root <root@Aravind.localdomain>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

 6 files changed, 287 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 Jenkinsfile
 create mode 100644 deployment.yaml
 create mode 100644 dockerfile
 create mode 100644 main.tf
 create mode 100644 service.yaml
root@Aravind:~/Trend#
```

Added my repo origin:

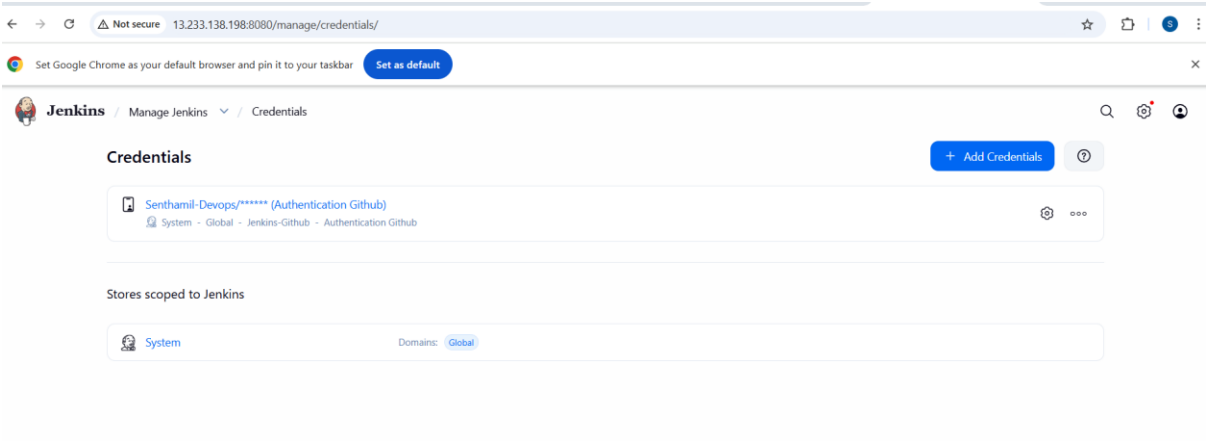
```
root@Aravind:~/Trend# git remote set-url origin https://github.com/senthamil-Devops/React-Trend-app.git
root@Aravind:~/Trend# git remote add origin  https://github.com/senthamil-Devops/React-Trend-app.git
error: remote origin already exists.
root@Aravind:~/Trend# git remote remove origin
root@Aravind:~/Trend# git remote add origin  https://github.com/senthamil-Devops/React-Trend-app.git
root@Aravind:~/Trend#
```

Pushed code to the repository:

```
root@Aravind:~/Trend# git push -u origin main
Username for 'https://github.com': Senthamil-Devops
Password for 'https://Senthamil-Devops@github.com':
Enumerating objects: 85, done.
Counting objects: 100% (85/85), done.
Delta compression using up to 12 threads
Compressing objects: 100% (83/83), done.
Writing objects: 100% (85/85), 8.59 MiB | 3.27 MiB/s, done.
Total 85 (delta 1), reused 77 (delta 1), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/senthamil-Devops/React-Trend-app.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
root@Aravind:~/Trend#
```



Added usercredential to authenticate github from Jenkins:

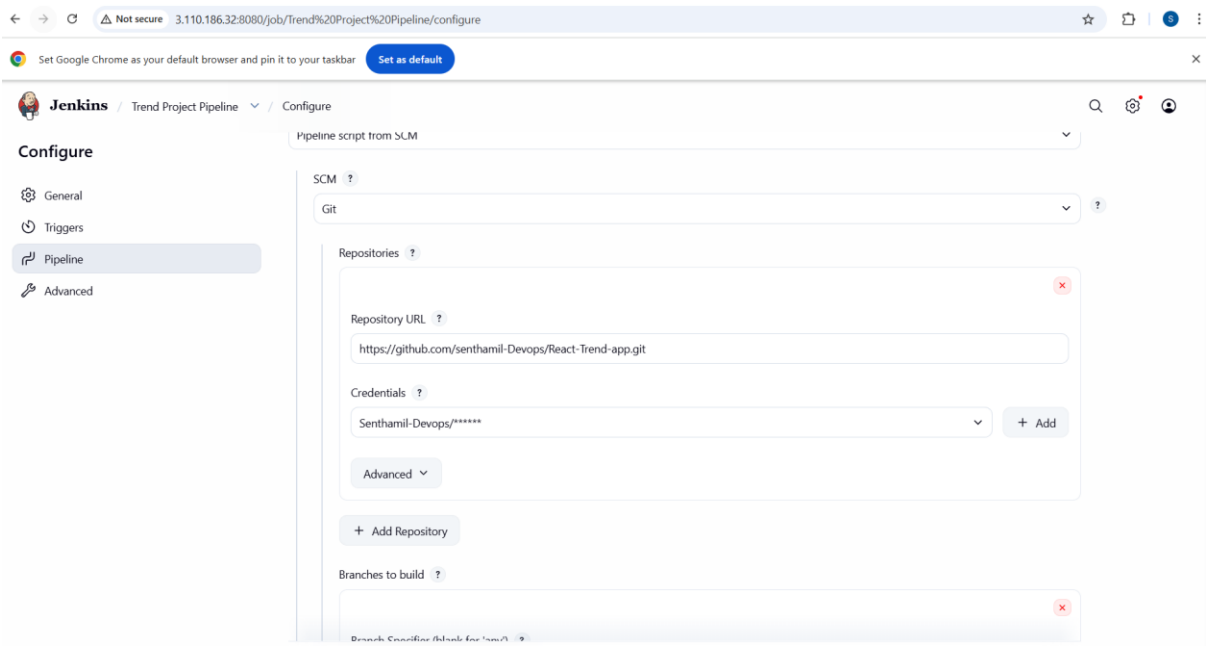


Tested manually Jenkins connection:

git ls-remote https://github.com/senthamil-Devops/React-Trend-app.git

```
[root@ip-10-0-1-76 ~]# git ls-remote https://github.com/biplabbapparaj-tech/Trendify_aws_project-2.git
f863071f4cc86781a899562914c9fbf9afad375a        HEAD
f863071f4cc86781a899562914c9fbf9afad375a        refs/heads/main
[root@ip-10-0-1-76 ~]#
```

Created Pilepline in Jenkins:



Installed nodejs in Jenkins server:

#curl -fsSL https://rpm.nodesource.com/setup_18.x | sudo bash -

#sudo yum install -y nodejs

#node -v

#npm -v

```
[root@ip-10-0-1-76 ~]# curl -fsSL https://rpm.nodesource.com/setup_18.x | sudo bash -
sudo yum install -y nodejs
2026-02-21 07:08:28 -
==========================================================================
                        DEPRECATION WARNING
==========================================================================
Node.js 18.x is no longer actively supported!
You will not receive security or critical stability updates for this version.

You should migrate to a supported version of Node.js as soon as possible.

Please see https://nodesource.com/products/distributions for details about which
version may be appropriate for you.

The NodeSource Node.js distributions site contains
information both about supported versions of Node.js and N|Solid supported Linux
distributions. To learn more about usage, see:
https://nodesource.com/products/distributions

==========================================================================

Continuing in 10 seconds ...
```

```
[root@ip-10-0-1-76 ~]# node -v
npm -v
v18.20.8
10.8.2
[root@ip-10-0-1-76 ~]#
```

Installed docker in Jenkins server:

sudo yum update -y

dnf install -y docker

sudo systemctl enable --now docker

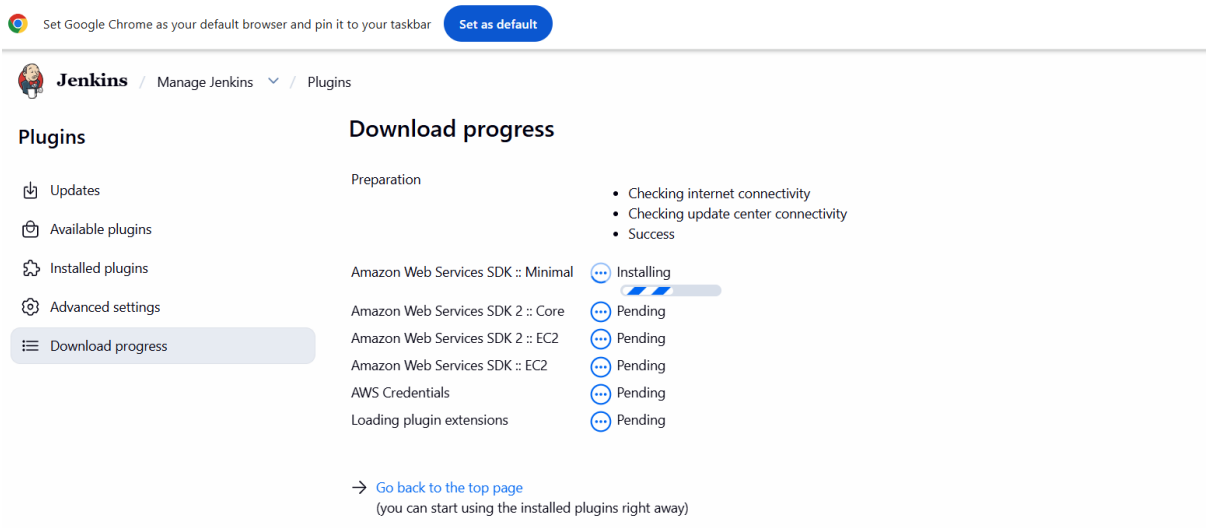
usermod -aG docker Jenkins

usermod -aG docker Jenkins

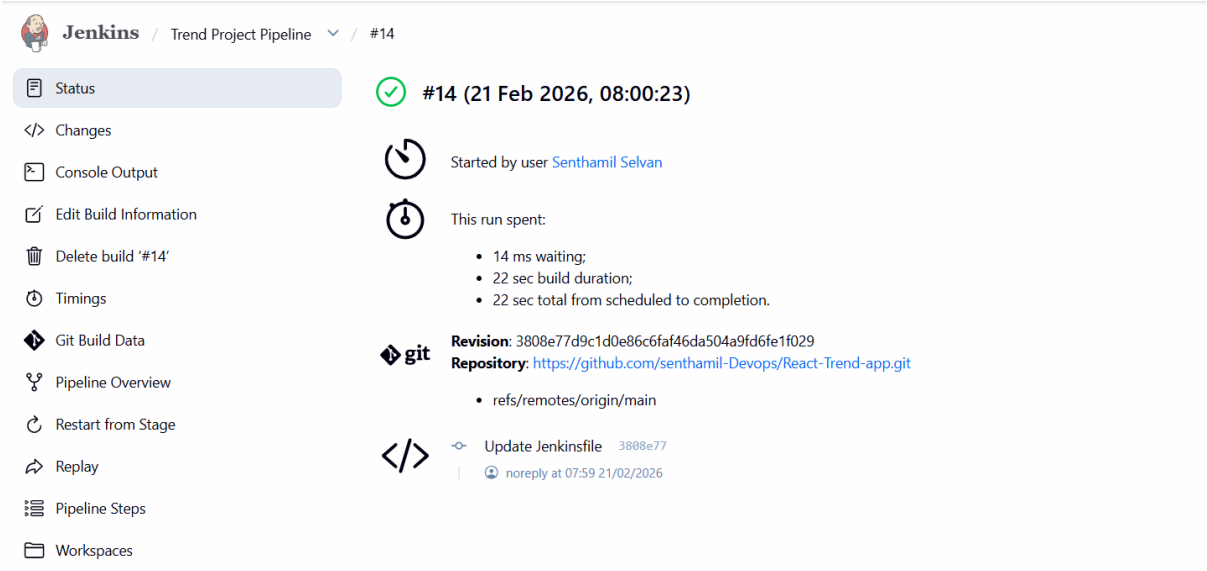
systemctl restart Jenkins

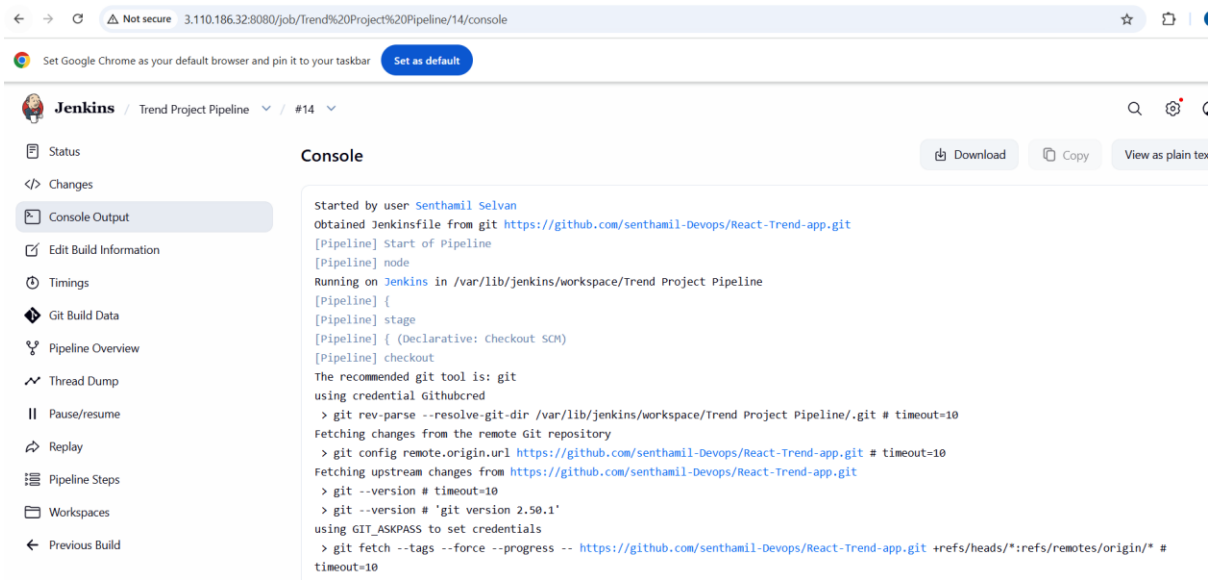
```
[root@ip-10-0-1-76 ~]# docker version
Client:
 Version:           25.0.14
 API version:       1.44
 Go version:        go1.24.11
 Git commit:        0bab007
 Built:             Wed Dec 10 00:00:00 2025
 OS/Arch:           linux/amd64
 Context:           default

Server:
 Engine:
  Version:          25.0.14
  API version:      1.44 (minimum version 1.24)
  Go version:       go1.24.11
  Git commit:       d334795
  Built:            Wed Dec 10 00:00:00 2025
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          2.1.5
  GitCommit:        fcd43222d6b07379a4be9786bda52438f0dd16a1
 runc:
  Version:          1.3.4
  GitCommit:        d6d73eb8c60246978da649ffe75ce5c8bca8f856
 docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0
[root@ip-10-0-1-76 ~]#
```

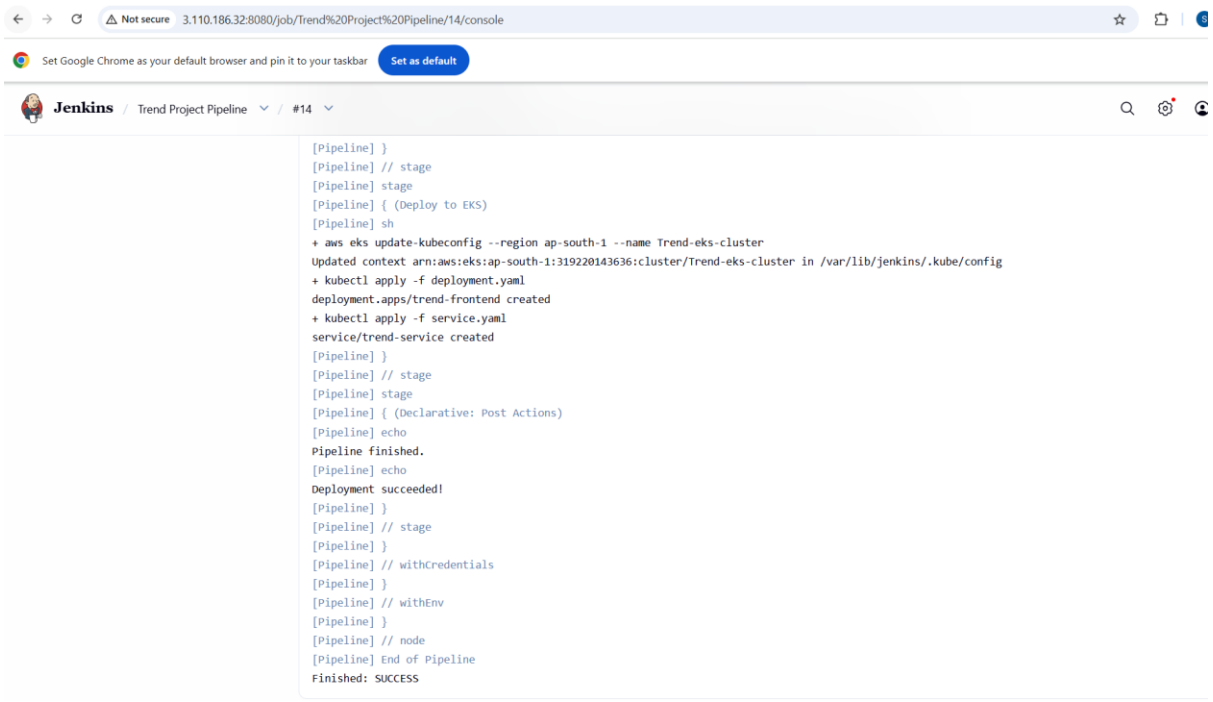
Installing aws cred plugins in Jenkins:



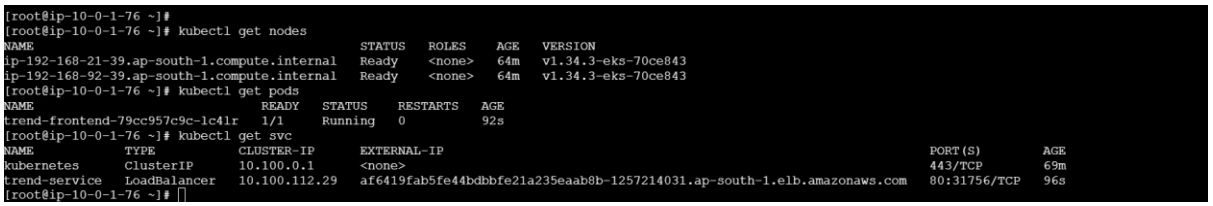
Initiated Build and Build ran successful:







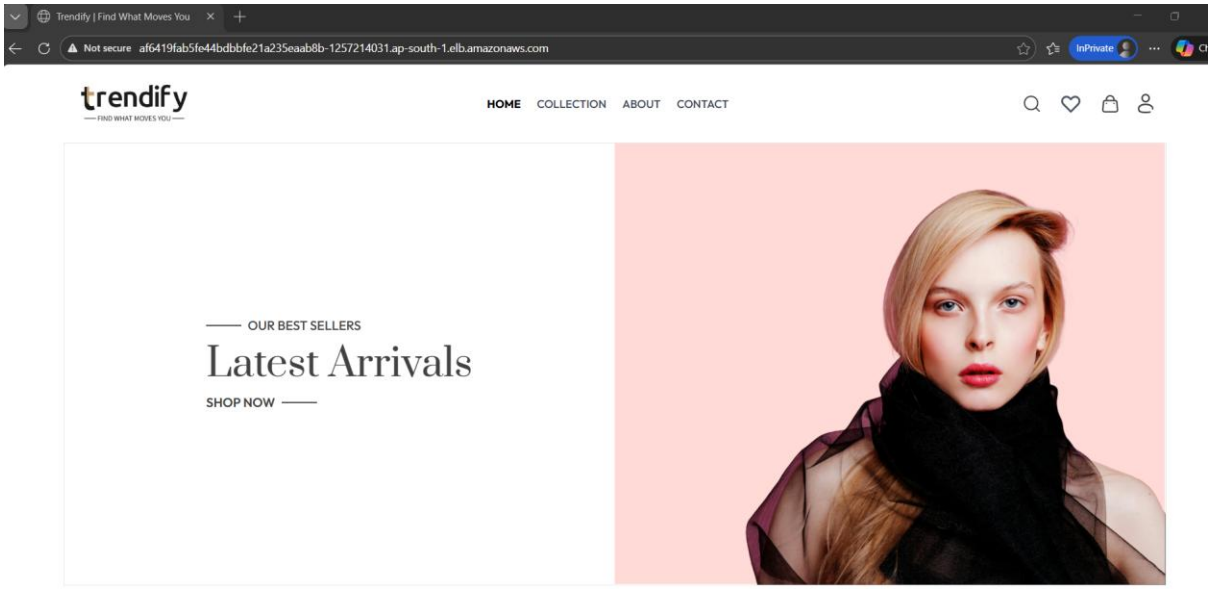
Verified pods and service in EC2:



LoadBalancer ARN:

af6419fab5fe44bdbbfe21a235eaab8b-1257214031.ap-south-1.elb.amazonaws.com

Verified in Browser using loadbalancer ARN:



Installed helm in EC2:

```
[root@ip-10-0-1-76 ~]# curl -fsSL https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3 | bash
Downloading https://get.helm.sh/helm-v3.20.0-linux-amd64.tar.gz
Verifying checksum... Done.
Preparing to install helm into /usr/local/bin
helm installed into /usr/local/bin/helm
[root@ip-10-0-1-76 ~]# helm version
version.BuildInfo{Version:"v3.20.0", GitCommit:"b2e4314fa0f229a1de7b4c981273f61d69ee5a59", GitTreeState:"clean", GoVersion:"go1.25.6"}
[root@ip-10-0-1-76 ~]#
```

Installed helm:

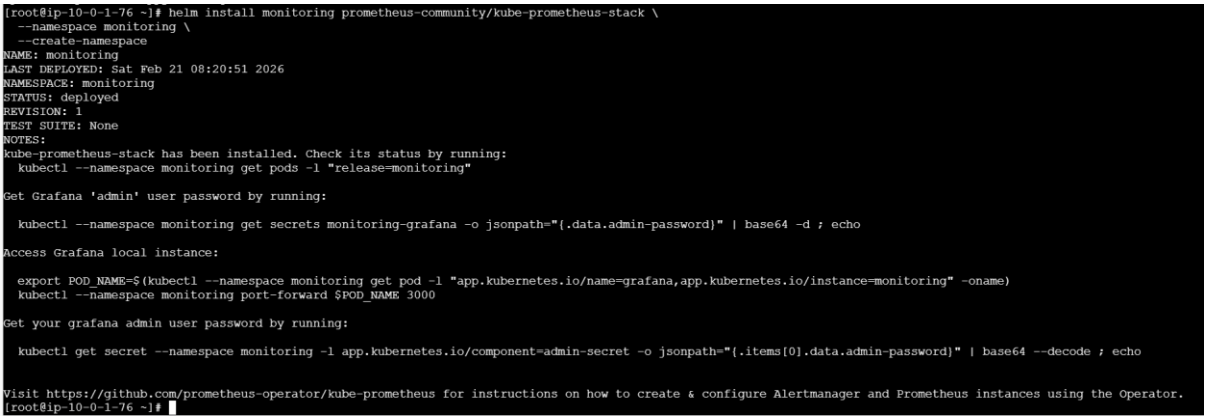
#curl -fsSL https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3 | bash

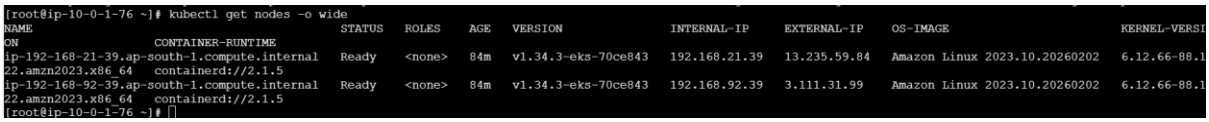
#helm version

helm repo update

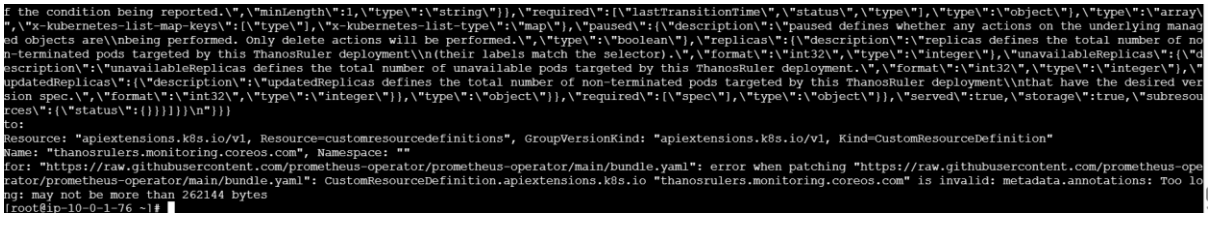
```
[root@ip-10-0-1-76 ~]# curl -fsSL https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3 | bash
Downloading https://get.helm.sh/helm-v3.20.0-linux-amd64.tar.gz
Verifying checksum... Done.
Preparing to install helm into /usr/local/bin
helm installed into /usr/local/bin/helm
[root@ip-10-0-1-76 ~]# helm version
version.BuildInfo{Version:"v3.20.0", GitCommit:"b2e4314fa0f229a1de7b4c981273f61d69ee5a59", GitTreeState:"clean", GoVersion:"go1.25.6"}
[root@ip-10-0-1-76 ~]# helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
"prometheus-community" has been added to your repositories
[root@ip-10-0-1-76 ~]# helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "prometheus-community" chart repository
Update Complete. ⎈Happy Helming!⎈
[root@ip-10-0-1-76 ~]#
```

helm install monitoring prometheus-community/kube-prometheus-stack --namespace monitoring --create-namespace





kubectl apply -f https://raw.githubusercontent.com/prometheus-operator/prometheus-operator/main/bundle.yaml



#kubectl get pods -n monitoring --show-labels

Created a loadbalancer for exposing graffana outside the cluster:

Filename: monitoring-grafana-lb.yaml

```
apiVersion: v1
kind: Service
metadata:
  name: monitoring-grafana-lb
  namespace: monitoring
spec:
  type: LoadBalancer
  selector:
    app.kubernetes.io/name: grafana
  ports:
    - port: 80      # External port
```

targetPort: 3000 # Grafana container port

protocol: TCP

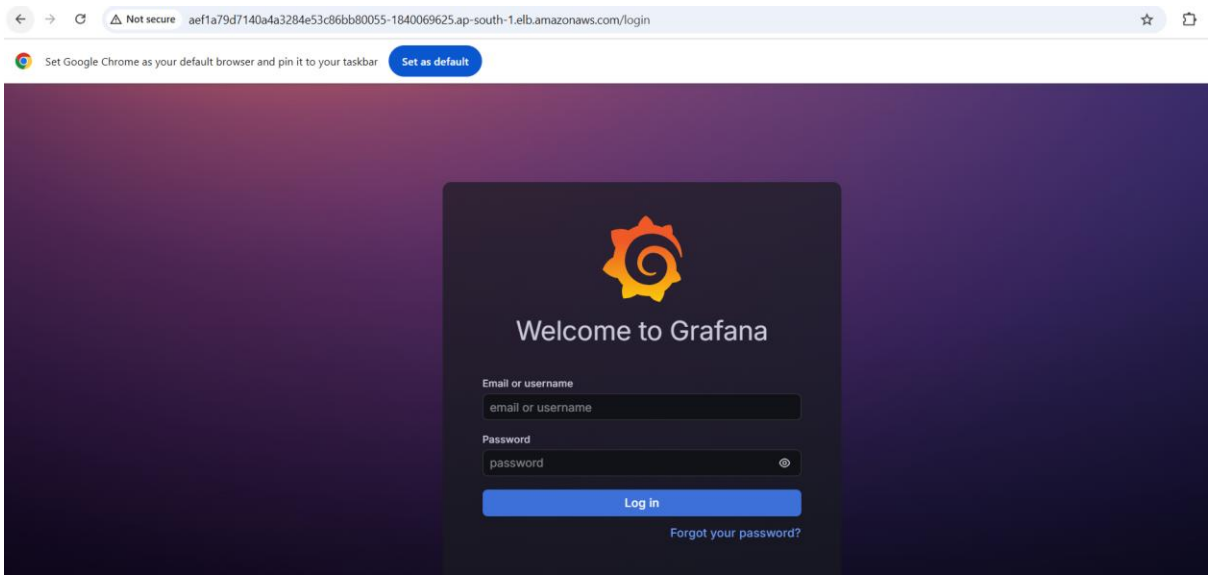
```
[root@ip-10-0-1-76 ~]# cat monitoring-grafana-lb.yaml
apiVersion: v1
kind: Service
metadata:
  name: monitoring-grafana-lb
  namespace: monitoring
spec:
  type: LoadBalancer
  selector:
    app.kubernetes.io/name: grafana
  ports:
    - port: 80        # External port
      targetPort: 3000  # Grafana container port
      protocol: TCP
[root@ip-10-0-1-76 ~]#
```

```
[root@ip-10-0-1-76 ~]# kubectl apply -f monitoring-grafana-lb.yaml
service/monitoring-grafana-lb created
[root@ip-10-0-1-76 ~]# kubectl get svc -n monitoring
NAME                                      TYPE           CLUSTER-IP       EXTERNAL-IP                                                              PORT(S)
        AGE
alertmanager-operated                     ClusterIP      None             <none>                                                                   9093/TCP,9094/TCP,9
094/UDP   20m
monitoring-grafana                        ClusterIP      10.100.221.150   <none>                                                                   80/TCP
        20m
monitoring-grafana-lb                     LoadBalancer   10.100.42.251    aef1a79d7140a4a3284e53c86bb80055-1840069625.ap-south-1.elb.amazonaws.com   80:32291/TCP
        8s
monitoring-kube-prometheus-alertmanager   ClusterIP      10.100.41.114    <none>                                                                   9093/TCP,8080/TCP
        20m
monitoring-kube-prometheus-operator       ClusterIP      10.100.200.223   <none>                                                                   443/TCP
        20m
monitoring-kube-prometheus-prometheus     ClusterIP      10.100.232.95    <none>                                                                   9090/TCP,8080/TCP
        20m
monitoring-kube-state-metrics             ClusterIP      10.100.18.114    <none>                                                                   8080/TCP
        20m
monitoring-prometheus-node-exporter       ClusterIP      10.100.88.89     <none>                                                                   9100/TCP
        20m
prometheus-operated                       ClusterIP      None             <none>                                                                   9090/TCP
        20m
[root@ip-10-0-1-76 ~]# kubectl get svc
NAME                TYPE           CLUSTER-IP      EXTERNAL-IP                                                              PORT(S)        AGE
kubernetes          ClusterIP      10.100.0.1      <none>                                                                   443/TCP        109m
prometheus-operator ClusterIP      None            <none>                                                                   8080/TCP       19m
trend-service       LoadBalancer   10.100.112.29   af6419fab5fe44bdbbfe21a235eaab8b-1257214031.ap-south-1.elb.amazonaws.com   80:31756/TCP   41m
[root@ip-10-0-1-76 ~]#
```

Launched in browser:

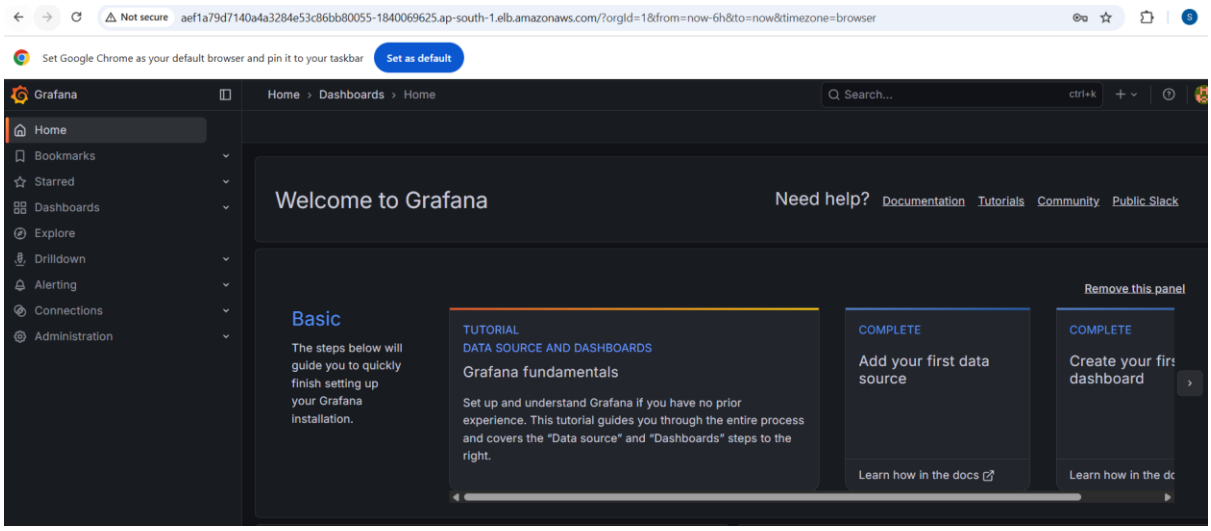
Graffana load Balancer IP:

af6419fab5fe44bdbbfe21a235eaab8b-1257214031.ap-south-1.elb.amazonaws.com



To fetch admin password:

kubectl get secret monitoring-grafana -n monitoring -o jsonpath="{.data.admin-password}" | base64 --decode; echo



Promotheus and Graffana pods running:

```
[root@ip-10-0-1-76 ~]# kubectl top pods -n monitoring
NAME                                                     CPU(cores)   MEMORY(bytes)
monitoring-grafana-784cdfd5db-2tnfh                      5m           250Mi
monitoring-kube-prometheus-operator-5b7fccb6d5-cfwrb     553m         29Mi
monitoring-kube-state-metrics-786449b944-h7p5m           12m          17Mi
monitoring-prometheus-node-exporter-4hz5x                1m           5Mi
monitoring-prometheus-node-exporter-hf8dj                1m           5Mi
[root@ip-10-0-1-76 ~]#
```