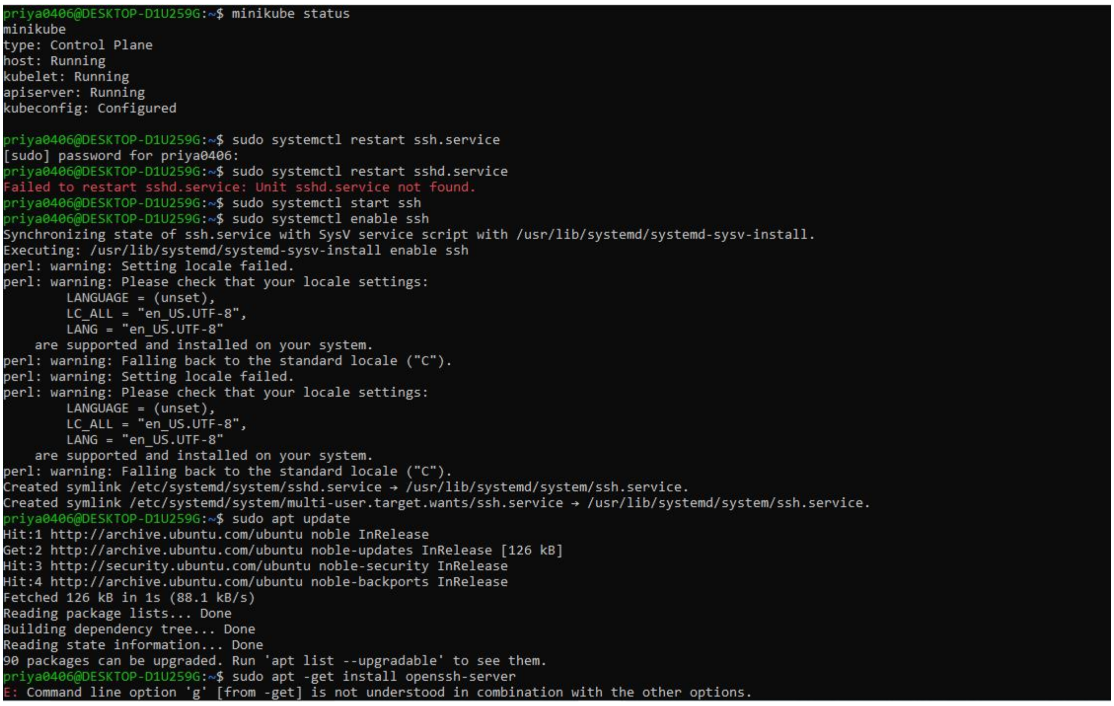
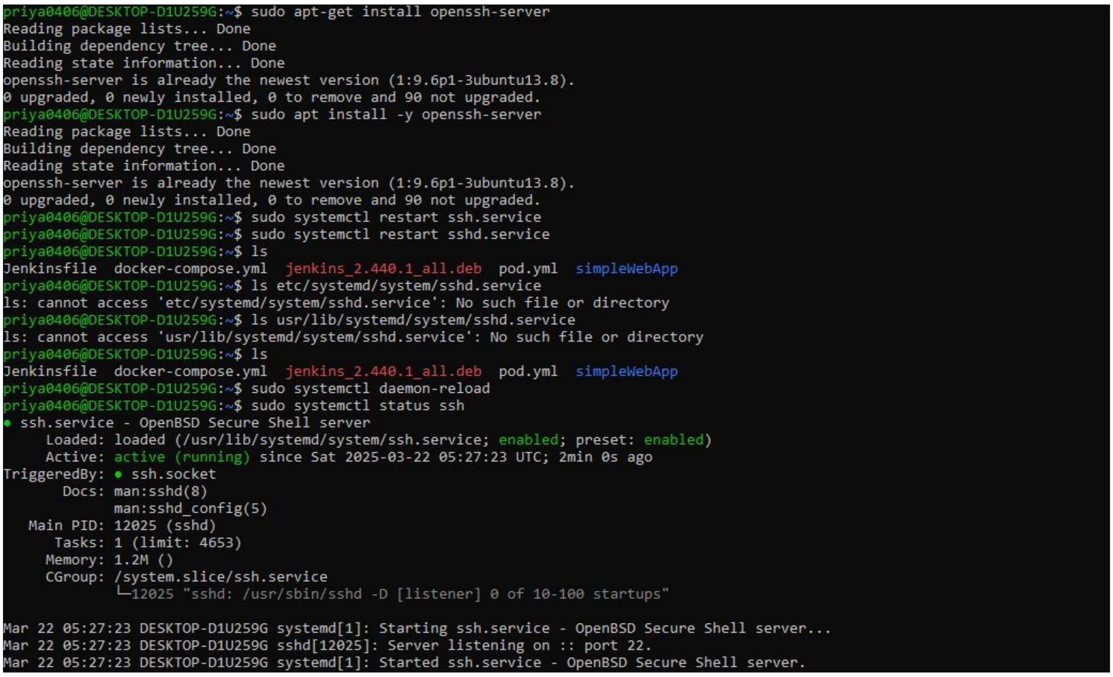
DAY-5

DATE:21.03.2025



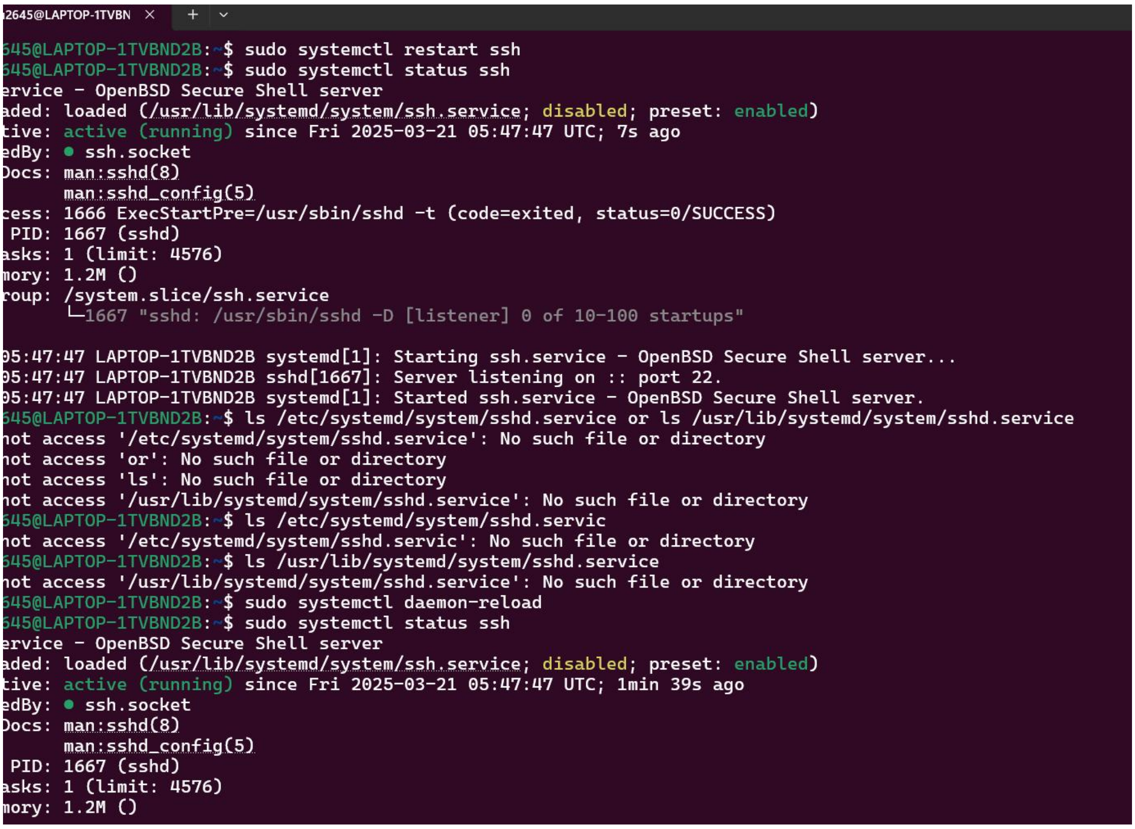
```
priya0406@DESKTOP-D1U259G:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

priya0406@DESKTOP-D1U259G:~$ sudo systemctl restart ssh.service
[sudo] password for priya0406:
priya0406@DESKTOP-D1U259G:~$ sudo systemctl restart sshd.service
Failed to restart sshd.service: Unit sshd.service not found.
priya0406@DESKTOP-D1U259G:~$ sudo systemctl start ssh
priya0406@DESKTOP-D1U259G:~$ sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable ssh
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
        LANGUAGE = (unset),
        LC_ALL = "en_US.UTF-8",
        LANG = "en_US.UTF-8"
    are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").
perl: warning: Setting locale failed.
perl: warning: Please check that your locale settings:
        LANGUAGE = (unset),
        LC_ALL = "en_US.UTF-8",
        LANG = "en_US.UTF-8"
    are supported and installed on your system.
perl: warning: Falling back to the standard locale ("C").
Created symlink /etc/systemd/system/sshd.service → /usr/lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /usr/lib/systemd/system/ssh.service.
priya0406@DESKTOP-D1U259G:~$ sudo apt update
Hit:1 http://archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:3 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:4 http://archive.ubuntu.com/ubuntu noble-backports InRelease
Fetched 126 kB in 1s (88.1 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
90 packages can be upgraded. Run 'apt list --upgradable' to see them.
priya0406@DESKTOP-D1U259G:~$ sudo apt -get install openssh-server
E: Command line option 'g' [from -get] is not understood in combination with the other options.
```



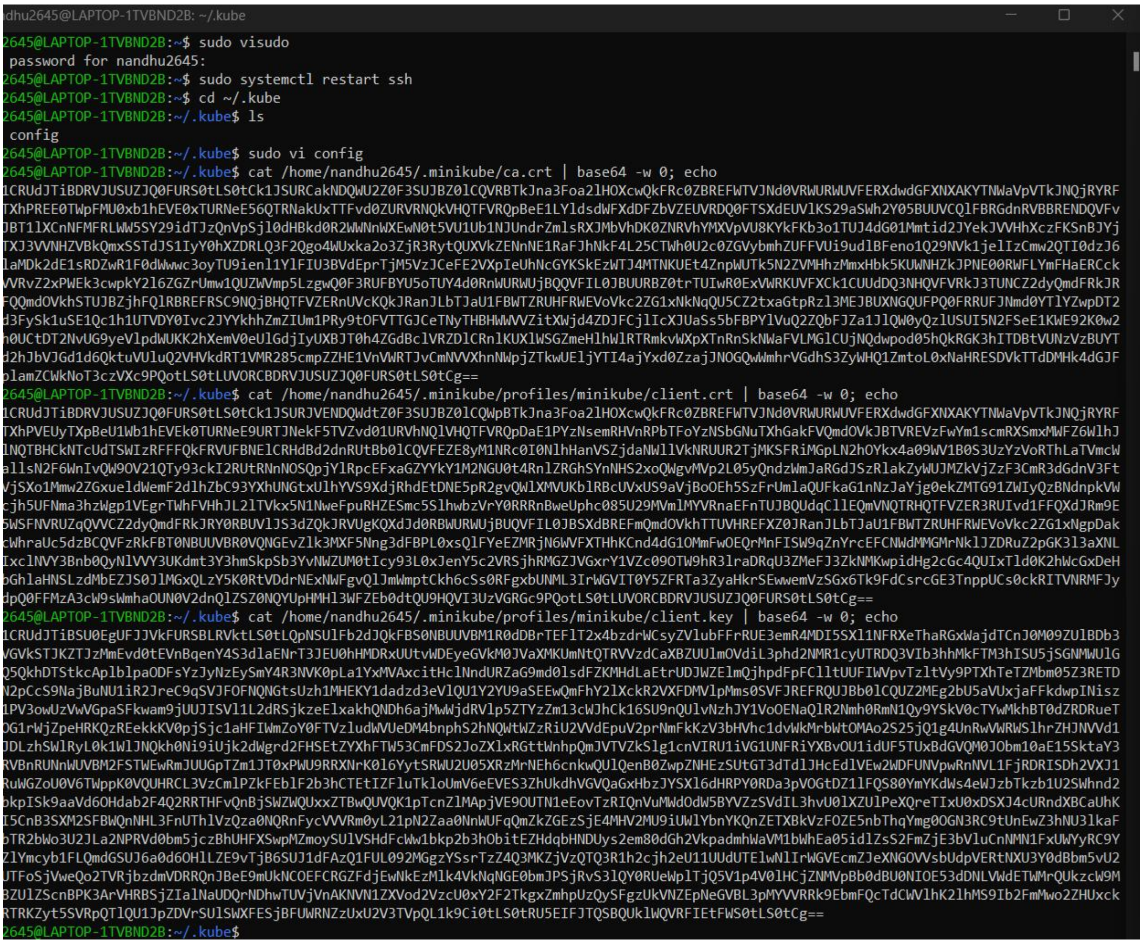
```
priya0406@DESKTOP-D1U259G:~$ sudo apt-get install openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openssh-server is already the newest version (1:9.6p1-3ubuntu13.8).
0 upgraded, 0 newly installed, 0 to remove and 90 not upgraded.
priya0406@DESKTOP-D1U259G:~$ sudo apt install -y openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openssh-server is already the newest version (1:9.6p1-3ubuntu13.8).
0 upgraded, 0 newly installed, 0 to remove and 90 not upgraded.
priya0406@DESKTOP-D1U259G:~$ sudo systemctl restart ssh.service
priya0406@DESKTOP-D1U259G:~$ sudo systemctl restart sshd.service
priya0406@DESKTOP-D1U259G:~$ ls
Jenkinsfile  docker-compose.yml  jenkins_2.440.1_all.deb  pod.yml  simpleWebApp
priya0406@DESKTOP-D1U259G:~$ ls etc/systemd/system/sshd.service
ls: cannot access 'etc/systemd/system/sshd.service': No such file or directory
priya0406@DESKTOP-D1U259G:~$ ls usr/lib/systemd/system/sshd.service
ls: cannot access 'usr/lib/systemd/system/sshd.service': No such file or directory
priya0406@DESKTOP-D1U259G:~$ ls
Jenkinsfile  docker-compose.yml  jenkins_2.440.1_all.deb  pod.yml  simpleWebApp
priya0406@DESKTOP-D1U259G:~$ sudo systemctl daemon-reload
priya0406@DESKTOP-D1U259G:~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: enabled)
     Active: active (running) since Sat 2025-03-22 05:27:23 UTC; 2min 0s ago
TriggeredBy: ● ssh.socket
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 12025 (sshd)
      Tasks: 1 (limit: 4653)
     Memory: 1.2M ()
     CGroup: /system.slice/ssh.service
             └─12025 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Mar 22 05:27:23 DESKTOP-D1U259G systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Mar 22 05:27:23 DESKTOP-D1U259G sshd[12025]: Server listening on :: port 22.
Mar 22 05:27:23 DESKTOP-D1U259G systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
```



```
545@LAPTOP-1TVBND2B:~$ sudo systemctl restart ssh
545@LAPTOP-1TVBND2B:~$ sudo systemctl status ssh
ervice - OpenBSD Secure Shell server
aded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: enabled)
tive: active (running) since Fri 2025-03-21 05:47:47 UTC; 7s ago
edBy: ● ssh.socket
Docs: man:sshd(8)
      man:sshd_config(5)
cess: 1666 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
 PID: 1667 (sshd)
asks: 1 (limit: 4576)
mory: 1.2M ()
roup: /system.slice/ssh.service
      └─1667 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

05:47:47 LAPTOP-1TVBND2B systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
05:47:47 LAPTOP-1TVBND2B sshd[1667]: Server listening on :: port 22.
05:47:47 LAPTOP-1TVBND2B systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
545@LAPTOP-1TVBND2B:~$ ls /etc/systemd/system/sshd.service or ls /usr/lib/systemd/system/sshd.service
not access '/etc/systemd/system/sshd.service': No such file or directory
not access 'or': No such file or directory
not access 'ls': No such file or directory
not access '/usr/lib/systemd/system/sshd.service': No such file or directory
545@LAPTOP-1TVBND2B:~$ ls /etc/systemd/system/sshd.servic
not access '/etc/systemd/system/sshd.servic': No such file or directory
545@LAPTOP-1TVBND2B:~$ ls /usr/lib/systemd/system/sshd.service
not access '/usr/lib/systemd/system/sshd.service': No such file or directory
545@LAPTOP-1TVBND2B:~$ sudo systemctl daemon-reload
545@LAPTOP-1TVBND2B:~$ sudo systemctl status ssh
ervice - OpenBSD Secure Shell server
aded: loaded (/usr/lib/systemd/system/ssh.service; disabled; preset: enabled)
tive: active (running) since Fri 2025-03-21 05:47:47 UTC; 1min 39s ago
edBy: ● ssh.socket
Docs: man:sshd(8)
      man:sshd_config(5)
 PID: 1667 (sshd)
asks: 1 (limit: 4576)
mory: 1.2M ()
```



```
2645@LAPTOP-1TVBND2B:~$ sudo visudo
 password for nandhu2645:
2645@LAPTOP-1TVBND2B:~$ sudo systemctl restart ssh
2645@LAPTOP-1TVBND2B:~$ cd ~/.kube
2645@LAPTOP-1TVBND2B:~/.kube$ ls
 config
2645@LAPTOP-1TVBND2B:~/.kube$ sudo vi config
2645@LAPTOP-1TVBND2B:~/.kube$ cat /home/nandhu2645/.minikube/ca.crt | base64 -w 0; echo
```

```
1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURCakNDQWU2Z0F3SUJBZ0lCQVRBTkJna3Foa2lHOXcwQkFRc0ZBREFWTVJNd0VRWURWUVFERXdwdGFXNXAKYTNWaVpVTkJNQjRYRF
TXhPREE0TWpFMU0xb1hEVE0xTURNeE56QTRNakUxTTFvd0ZURVRNQkVHQTFVRVQpBeE1LYldsdWFXdDFZbVZEUVRDQ0FTSXdEUVlKS29aSWh2Y05BUUVCQlFBRGdnRVBBRENDQVFv
JBT1lXCnNFMFRLWW5SY29idTJzQnVpSjl0dHBkd0R2WWNnWXEwN0t5VU1Ub1NJUndrZmlsRXJMbVhDK0ZNRVhYMXVpVU8KYkFKb3o1TUJ4dG01Mmtid2JYekJVVHhXczFKSnBJYj
TXJ3VVNHZVBkQmxSSTdJS1IyY0hXZDRLQ3F2Qgo4WUxka2o3ZjR3RytQUXVkZENnNE1RaFJhNkF4L25CTWh0U2c0ZGVybmhZUFFVUi9udlBFeno1Q29NVk1jelIzCmw2QTI0dzJ6
laMDk2dE1sRDZwR1F0dWwwc3oyTU9ienl1YlFIU3BVdEprTjM5VzJCeFE2VXpIeUhNcGYKSkEzWTJ4MTNKUEt4ZnpWUTk5N2ZVMHhzMmxHbk5KUWNHZkJPNE00RWFLYmFHaERCck
VVRvZ2xPWEk3cwpkY2l6ZGZrUmw1QUZWVmp5LzgwQ0F3RUFBYU5oTUY4d0RnWURWUjBQQVFIL0JBUURBZ0trTUIwR0ExVWRKUVFXCk1CUUdDQ3NHQVFVRkJ3TUNCZ2dyQmdFRkJR
FQQmdOVkhSTUJBZjhFQlRBREFRSC9NQjBHQTFVZERnUVcKQkJRanJLbTJaU1FBWTZRUHFRWEVoVkc2ZG1xNkNqQU5CZ2txaGtpRzl3MEJBUXNGQUFPQ0FRRUFJNmd0YTlYZwpDT2
d3FySk1uSE1Qc1h1UTVDY0Ivc2JYYkhhZmZIUm1PRy9tOFVTTGJCeTNyTHBHWWVVZitXWjd4ZDJFCjlIcXJUaSs5bFBPYlVuQ2ZQbFJZa1JlQW0yQzlUSUI5N2FSeE1KWE92K0w2
h0UCtDT2NvUG9yeVlpdWUKK2hXemV0eUlGdjIyUXBJT0h4ZGdBclVRZDlCRnlKUXlWSGZmeHlhWlRTRmkvWXpXTnRnSkNWaFVLMGlCUjNQdwpod05hQkRGK3hITDBtVUNzVzBUYT
d2hJbVJGd1d6QktuVUluQ2VHVkdRT1VMR285cmpZZHE1VnVWRTJvCmNVVXhnNWpjZTkwUEljYTI4ajYxd0ZzajJNOGQwWmhrVGdhS3ZyWHQ1ZmtoL0xNaHRESDVkTTdDMHk4dGJF
plamZCWkNoT3czVXc9PQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0tCg==
```

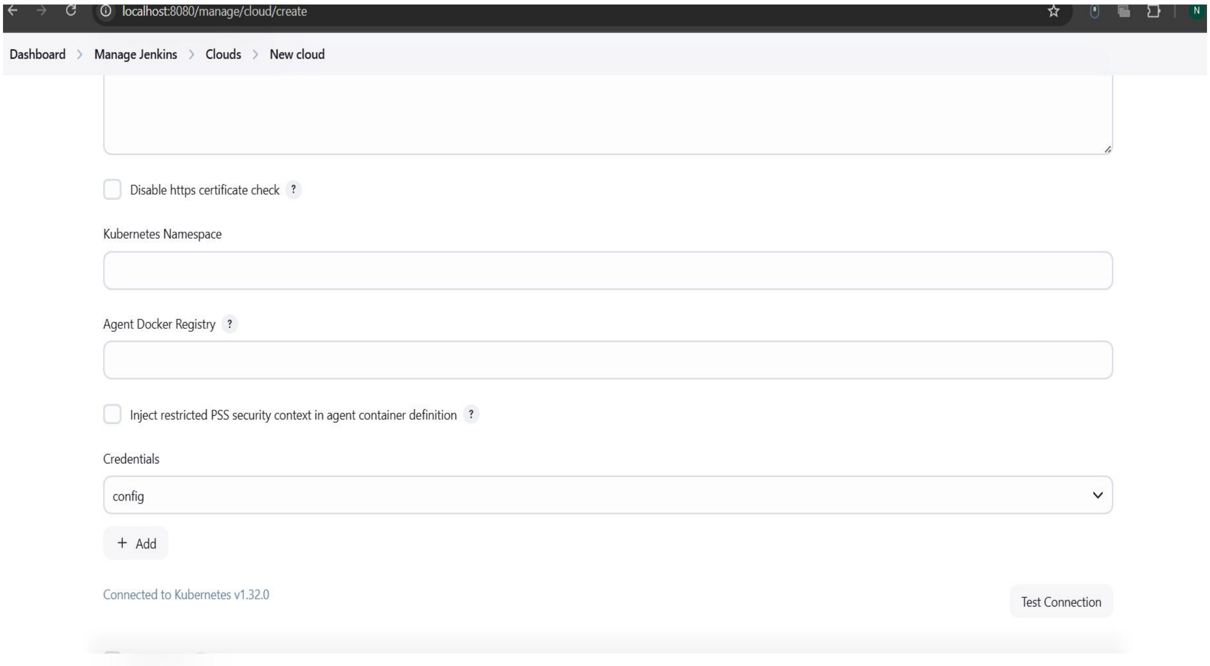
```
2645@LAPTOP-1TVBND2B:~/.kube$ cat /home/nandhu2645/.minikube/profiles/minikube/client.crt | base64 -w 0; echo
```

```
1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURJVENDQWdtZ0F3SUJBZ0lCQWpBTkJna3Foa2lHOXcwQkFRc0ZBREFWTVJNd0VRWURWUVFERXdwdGFXNXAKYTNWaVpVTkJNQjRYRF
TXhPVEUyTXpBeU1Wb1hEVEk0TURNeE9URTJNekF5TVZvd01URVhNQlVHQTFVRVQpDaE1PYzNsemRHVnRPbTFoYzNSbGNuTXhGakFVQmdOVkJBTVREVzFwYm1scmRXSmxMWFZ6WlhJ
lNQTBHCkNTcUdTSWIzRFFFQkFRVUFBNElCRHdBd2dnRUtBb0lCQVFEZE8yM1NRc0I0NlhhanVSZjdaNWllVkNRUUR2TjMKSFRiMGpLN2hOYkx4a09WV1B0S3UzYzVoRThLaTVmcW
allsN2F6WnIvQW9OV21QTy93ckI2RUtRNnNOSQpjYlRpcEFxaGZYYkY1M2NGU0t4RnlZRGhSYnNHS2xoQWgvMVp2L05yQndzWmJaRGdJSzRlakZyWUJMZkVjZzF3CmR3dGdnV3Ft
VjSXo1Mmw2ZGxueldWemF2dlhZbC93YXhUNGtxUlhYVS9XdjRhdEtDNE5pR2gvQWlXMVUKblRBcUVxUS9aVjBoOEh5SzFrUmlaQUFkaG1nNzJaYjg0ekZMTG91ZWIyQzBNdnpkVW
cjh5UFNma3hzWgp1VEgrTWhFVHhJL2lTVkx5N1NweFpuRHZESmc5SlhwbzVrY0RRRnBweUphc085U29MVmlMYVRnaEFnTUJBQUdqClEQmVNQTRHQTFVZER3RUIvd1FFQXdJRm9E
5WSFNVRUZqQVVCZ2dyQmdFRkJRY0RBUVlJS3dZQkJRVUgKQXdJd0RBWURWUjBUQVFIL0JBSXdBREFmQmdOVkhTTUVHREFXZ0JRanJLbTJaU1FBWTZRUHFRWEVoVkc2ZG1xNgpDak
cWhraUc5dzBCQVFzRkFBT0NBUUVBR0VQNGEvZlk3MXF5Nng3dFBPL0xsQlFYeEZMRjN6WVFXTHhKCnd4dG1OMmFwOEQrMnFISW9qZnYrcEFCNWdMMGMrNklJZDRuZ2pGK3l3aXNL
IxclNVY3Bnb0QyNlVVY3UKdmt3Y3hmSkpSb3YvNWZUM0tIcy93L0xJenY5c2VRSjhRMGZJVGxrY1VZc09OTW9hR3lraDRqU3ZMeFJ3ZkNMKwpidHg2cGc4QUIxTld0K2hWcGxDeH
bGhlaHNSLzdMbEZJS0J1MGxQLzY5K0RtVDdrNExNWFgvQlJmWmptCkh6cSs0RFgxbUNML3IrWGVIT0Y5ZFRTa3ZyaHkrSEwwemVzSGx6Tk9FdCsrcGE3TnppUCs0ckRITVNRMFJy
dpQ0FFMzA3cW9sWmhaOUN0V2dnQlZSZ0NQYUpHMHl3WFZEb0dtQU9HQVI3UzVGRGc9PQotLS0tLUVORCBDRVJUSUZJQ0FURS0tLS0tCg==
```

```
2645@LAPTOP-1TVBND2B:~/.kube$ cat /home/nandhu2645/.minikube/profiles/minikube/client.key | base64 -w 0; echo
```

```
1CRUdJTiBSU0EgUFJJVkFURSBLRVktLS0tLQpNSUlFb2dJQkFBS0NBUUVBM1R0dDBrTEFlT2x4bzdrWCsyZVlubFFrRUE3emR4MDI5SXl1NFRXeThaRGxWajdTCnJ0M09ZUlBDb3
VGVkSTJKZTJzMmEvd0tEVnBqenY4S3dlaENrT3JEU0hHMDRxUUtvWDEyeGVkM0JVaXMKUmNtQTRVVzdCaXBZUUlmOVdiL3phd2NMR1cyUTRDQ3VIb3hhMkFTM3hISU5jSGNMWUlG
Q5QkhDTStkcAplblpaODFsYzJyNzEySmY4R3NVK0pLa1YxMVAxcitHclNndURZaG9md0lsdFZKMHdLaEtrUDJWZElmQjhpdFpFClltUUFIWVpvTzltVy9PTXhTeTZMbm05Z3RETD
N2pCcS9NajBuNU1iR2JreC9qSVJFOFNQNGtsUzh1MHEKY1dadzd3eVlQU1Y2YU9aSEEwQmFhY2lXckR2VXFDMVlpMms0SVFJREFRQUJBb0lCQUZ2MEg2bU5aVUxjaFFkdwpINisz
1PV3owUzVwVGpaSFkwam9jUUJISVl1L2dRSjkzeElxakhQNDh6ajMwWjdRVlp5ZTYzZm13cWJhCk16SU9nQUlvNzhJY1VoOENaQlR2Nmh0RmN1Qy9YSkV0cTYwMkhBT0dZRDRueT
OG1rWjZpeHRKQzREekkKV0pjSjc1aHFIWmZoY0FTVzludWVUeDM4bnphS2hNQWtWZzRiU2VVdEpuV2prNmFkKzV3bHVhc1dvWkMrbWtOMAo2S25jQ1g4UnRwVWRWSlhrZHJNVVd1
JDLzhSWlRyL0k1WlJNQkh0Ni9iUjk2dWgrd2FHSEtZYXhFTW53CmFDS2JoZXlxRGttWnhpQmJVTVZkSlg1cnVIRU1iVG1UNFRiYXBvOU1idUF5TUxBdGVQM0JObm10aE15SktaY3
RVBnRUNnWUVBM2FSTWEwRmJUUGpTZm1JT0xPWU9RRXNrK0l6YytSRWU2U05XRzMrNEh6cnkwQUlQenB0ZwpZNHEzSUtGT3dTdlJHcEdlVEw2WDFUNVpwRnNVL1FjRDRISDh2VXJ1
RuWGZoU0V6TWppK0VQUHRCL3VzCmlPZkFEblF2b3hCTEtIZFluTkloUmV6eEVES3ZhUkdhVGVQaGxHbzJYSXl6dHRPY0RDa3pVOGtDZ1lFQS80YmYKdWs4eWJzbTkzb1U2SWhnd2
bkpISk9aaVd6OHdab2F4Q2RRTHFvQnBjSWZWQUxxZTBwQUVQK1pTcnZlMApjVE9OUTN1eEovTzRIQnVuMWdOdW5BYVZzSVdIL3hvU0lXZUlPeXQreTIxU0xDSXJ4cURndXBCaUhK
I5CnB3SXM2SFBWQnNHL3FnUThlVzQza0NQRnFycVVVRm0yL21pN2Zaa0NnWUFqQmZkZGEzSjE4MHV2MU9iUWlYbnYKQnZETXBkVzFOZE5nbThqYmg0OGN3RC9tUnEwZ3hNU3lkaF
bTR2bWo3U2JLa2NPRVd0bm5jczBhUHFXSwpMZmoySUlVSHdFcWw1bkp2b3hObitEZHdqbHNDUys2em80dGh2VkpadmhWaVM1bWhEa05idlZsS2FmZjE3bVluCnNMN1FxUWYyRC9Y
ZlYmcyb1FLQmdGSUJ6a0d6OHlLZE9vTjB6SUJ1dFAzQ1FUL092MGgzYSsrTzZ4Q3MKZjVzQTQ3R1h2cjh2eU11UUdUTElwNlIrWGVEcmZJeXNGOVVsbUdpVERtNXU3Y0dBbm5vU2
UTFoSjVweQo2TVRjbzdmVDRRQnJBeE9mUkNCOEFCRGZFdjEwNkEzMlk4VkNqNGE0bmJPSjRvS3lQY0RUeWplTjQ5V1p4V0lHCjZNMVpBb0dBU0NIOE53dDNLVWdETWMrQUkzcW9M
BZUlZScnBPK3ArVHRBSjZIalNaUDQrNDhwTUVjVnAKNVN1ZXVod2VzcU0xY2F2TkgxZmhpUzQySFgzUkVNZEpNeGVBL3pMYVVRRk9EbmFQcTdCWVlhK2lhMS9Ib2FmMwo2ZHUxck
RTRKZyt5SVRpQTlQU1JpZDVrSUlSWXFESjBFUWRNZzUxU2V3TVpQL1k9Ci0tLS0tRU5EIFJTQSBQUklWQVRFIEtFWS0tLS0tCg==
```

```
2645@LAPTOP-1TVBND2B:~/.kube$
```



localhost:8080/manage/cloud/create

Dashboard > Manage Jenkins > Clouds > New cloud

□ Disable https certificate check

Kubernetes Namespace

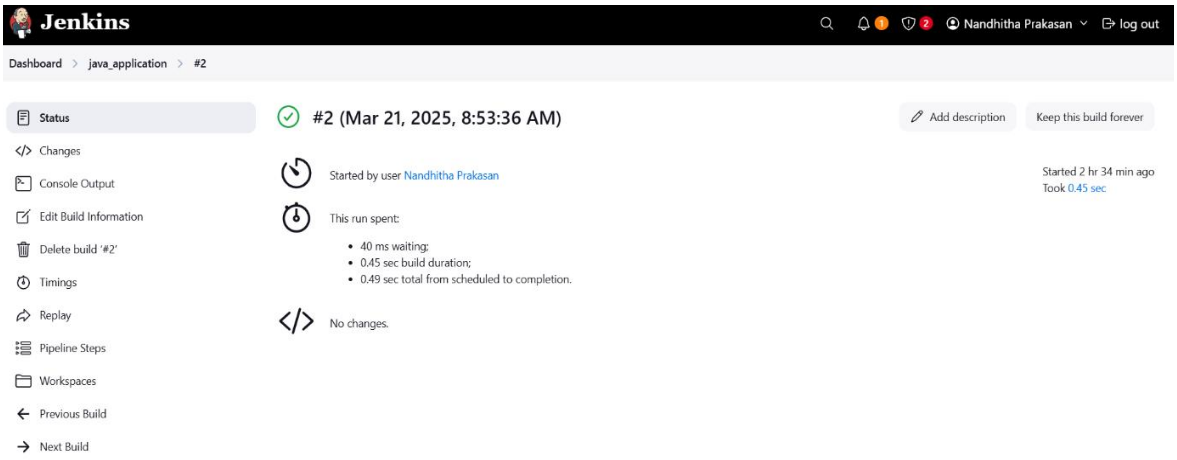
Agent Docker Registry

□ Inject restricted PSS security context in agent container definition

Credentials

config

Connected to Kubernetes v1.32.0



Dashboard > java_application > #2

#2 (Mar 21, 2025, 8:53:36 AM)

Started by user Nandhitha Prakasan

Started 2 hr 34 min ago
Took 0.45 sec

This run spent:

- 40 ms waiting;
- 0.45 sec build duration;
- 0.49 sec total from scheduled to completion.

No changes.

COMMANDS:

PIPELINE:

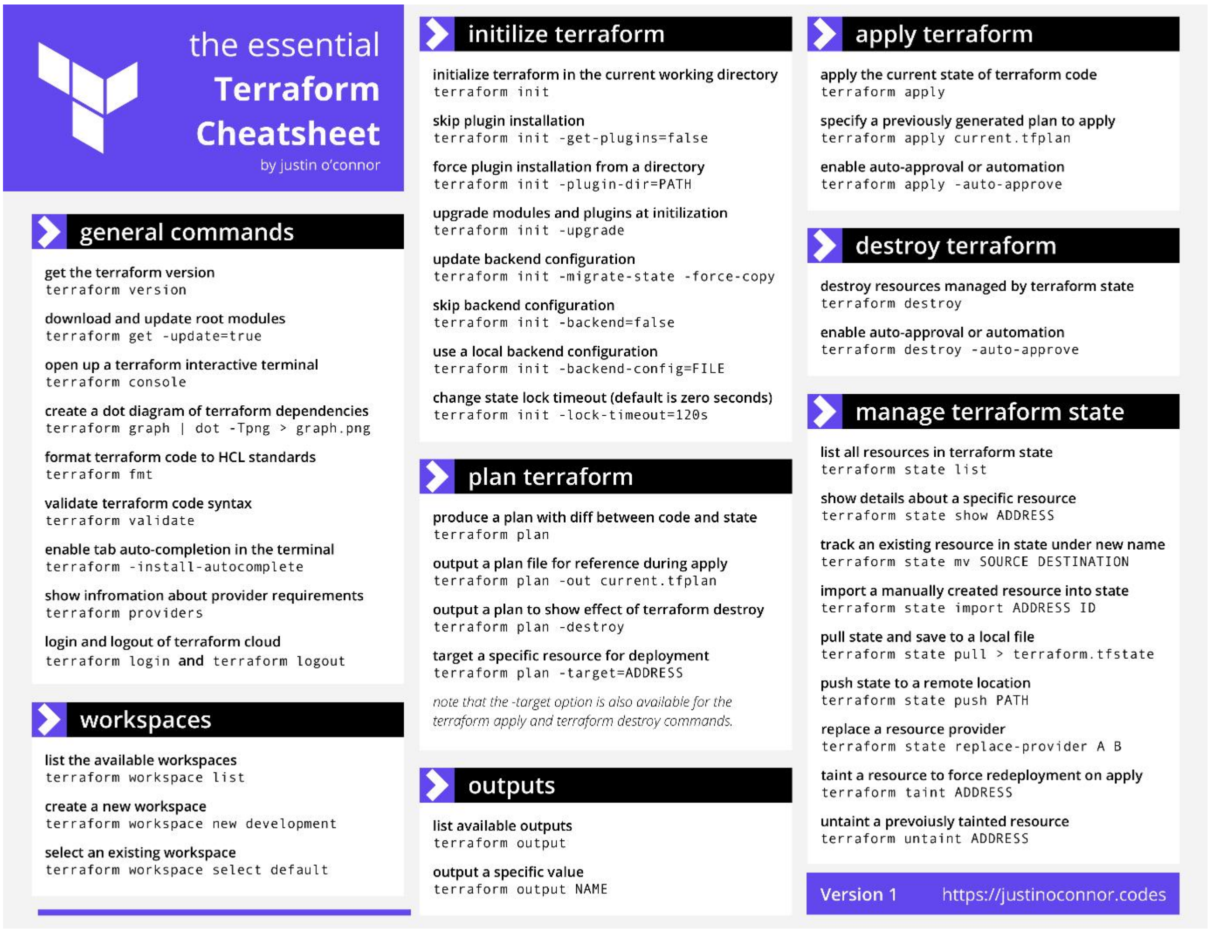
```
pipeline {
    agent any

    stages {
        stage('scm') {
            steps {
        git branch: 'master', url:'https://github.com/PriyaSenthilnathan/devops_main.git'
            }
        }
        stage('builb-clean') {
           steps {
              sh "mvn clean"
}
}
        stage('build-validate') {
           steps {
              sh "mvn validate"
}
}
        stage('build-com') {
           steps {
              sh "mvn compile"
}
}

        stage('build-test') {
           steps {
              sh "mvn test"
}
}
        stage('build-install') {
           steps {
              sh "mvn package"
}
}
stage('build to images') {
           steps {
              script{
                 sh 'docker build -t  priyadarshini.22cse154/mysimplewebapp .'
              }
    }
}
```

```
stage('push to hub') {
        steps {
          script{
            withDockerRegistry(credentialsId: 'Docker_cred', url:
'https://index.docker.io/v1/') {
             sh 'docker push priyadarshini22cse126/mysimplewebapp'
          }
        }
        }
}
     stage('Deploy App') {
        steps {
           withKubeConfig(caCertificate: '', clusterName: 'minikube', contextName:
'minikube', credentialsId: 'mukubeconfig_011', namespace: '',
restrictKubeConfigAccess: false, serverUrl: 'https://192.168.49.2:8443') {
           sh 'kubectl apply -f deployment.yml --validate=false'
        }
     }
     }
   stage('Test') {
    steps {
       withKubeConfig(caCertificate: '', clusterName: 'minikube', contextName:
'minikube', credentialsId: 'mukubeconfig_011', namespace: '',
restrictKubeConfigAccess: false, serverUrl: 'https://192.168.49.2:8443') {

     sh 'minikube service my-service --url | xargs curl'
  }
}
}
}
}
```



the essential Terraform Cheatsheet

by justin o'connor

general commands

get the terraform version
`terraform version`

download and update root modules
`terraform get -update=true`

open up a terraform interactive terminal
`terraform console`

create a dot diagram of terraform dependencies
`terraform graph | dot -Tpng > graph.png`

format terraform code to HCL standards
`terraform fmt`

validate terraform code syntax
`terraform validate`

enable tab auto-completion in the terminal
`terraform -install-autocomplete`

show infromation about provider requirements
`terraform providers`

login and logout of terraform cloud
`terraform login` and `terraform logout`

workspaces

list the available workspaces
`terraform workspace list`

create a new workspace
`terraform workspace new development`

select an existing workspace
`terraform workspace select default`

initilize terraform

initialize terraform in the current working directory
`terraform init`

skip plugin installation
`terraform init -get-plugins=false`

force plugin installation from a directory
`terraform init -plugin-dir=PATH`

upgrade modules and plugins at initilization
`terraform init -upgrade`

update backend configuration
`terraform init -migrate-state -force-copy`

skip backend configuration
`terraform init -backend=false`

use a local backend configuration
`terraform init -backend-config=FILE`

change state lock timeout (default is zero seconds)
`terraform init -lock-timeout=120s`

plan terraform

produce a plan with diff between code and state
`terraform plan`

output a plan file for reference during apply
`terraform plan -out current.tfplan`

output a plan to show effect of terraform destroy
`terraform plan -destroy`

target a specific resource for deployment
`terraform plan -target=ADDRESS`

note that the -target option is also available for the terraform apply and terraform destroy commands.

outputs

list available outputs
`terraform output`

output a specific value
`terraform output NAME`

apply terraform

apply the current state of terraform code
`terraform apply`

specify a previously generated plan to apply
`terraform apply current.tfplan`

enable auto-approval or automation
`terraform apply -auto-approve`

destroy terraform

destroy resources managed by terraform state
`terraform destroy`

enable auto-approval or automation
`terraform destroy -auto-approve`

manage terraform state

list all resources in terraform state
`terraform state list`

show details about a specific resource
`terraform state show ADDRESS`

track an existing resource in state under new name
`terraform state mv SOURCE DESTINATION`

import a manually created resource into state
`terraform state import ADDRESS ID`

pull state and save to a local file
`terraform state pull > terraform.tfstate`

push state to a remote location
`terraform state push PATH`

replace a resource provider
`terraform state replace-provider A B`

taint a resource to force redeployment on apply
`terraform taint ADDRESS`

untaint a prevoiusly tainted resource
`terraform untaint ADDRESS`

Version 1 https://justinoconnor.codes

TERRAFORM:

```
terraform {
  required_providers {
    aws = {
      source = "hashicorp/aws"
      version = "5.92.0"
    }
  }
}
provider "aws" {
  region = "us-east-1"
}

# Create a VPC
resource "aws_vpc" "myvpc" {
  cidr_block = "10.0.0.0/16"
```

```
}
 tags = {
   Name = "demovpc"
  }
resource "aws_subnet" "pubsub" {
  vpc_id     = aws_vpc.myvpc.id
  cidr_block = "10.0.1.0/24"
  availability_zone = "us-east-1a"

  tags = {
    Name = "sn1"
  }
}
resource "aws_subnet" "pub_sub" {
  vpc_id     = aws_vpc.myvpc.id
  cidr_block = "10.0.1.0/24"
  availability_zone = "us-east-1a"

  tags = {
    Name = "sn2"
  }
}
resource "aws_subnet" "prisub" {
  vpc_id     = aws_vpc.myvpc.id
  cidr_block = "10.0.1.0/24"
  availability_zone = "us-east-1a"

  tags = {
    Name = "sn3"
  }
}
resource "aws_subnet" "pri_sub" {
  vpc_id     = aws_vpc.myvpc.id
  cidr_block = "10.0.1.0/24"
  availability_zone = "us-east-1a"

  tags = {
    Name = "sn4"
  }
}
resource "aws_internet_gateway" "tfigw" {
  vpc_id = aws_vpc.myvpc.id

  tags = {
    Name = "tfigw"
  }
}
resource "aws_route_table" "tfpubrt" {
  vpc_id = aws_vpc.myvpc.id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_internet_gateway.tfigw.id
  }

  tags = {
    Name = "tfpublicroute"
  }
}
resource "aws_route_table_association" "pubsn1" {
  subnet_id      = aws_subnet.pubsub.id
  route_table_id = aws_route_table.tfpubrt.id
}
resource "aws_route_table_association" "pubsn2" {
  subnet_id      = aws_subnet.pub_sub.id
  route_table_id = aws_route_table.tfpubrt.id
}
resource "aws_eip" "tfeip" {
  domain   = "vpc"
}
resource "aws_nat_gateway" "tfnat" {
  allocation_id = aws_eip.tfeip.id
  subnet_id     = aws_subnet.pub_sub.id

  tags = {
    Name = "gw NAT"
  }
}
resource "aws_route_table" "tfprirt" {
  vpc_id = aws_vpc.myvpc.id

  route {
    cidr_block = "0.0.0.0/0"
    gateway_id = aws_nat_gateway.tfnat.id
  }

  tags = {
    Name = "tfprivateroute"
  }
}
resource "aws_route_table_association" "prisn3" {
  subnet_id      = aws_subnet.prisub.id
  route_table_id = aws_route_table.tfprirt.id
}
resource "aws_route_table_association" "prisn4" {
  subnet_id      = aws_subnet.pri_sub.id
  route_table_id = aws_route_table.tfprirt.id
}
resource "aws_security_group" "allow_tfsg" {
  name        = "allow_tfsg"
  description = "Allow TLS inbound traffic"
  vpc_id      = aws_vpc.myvpc.id
  ingress {
    description      = "HTTPS "
    from_port        = 443
    to_port          = 443
    protocol         = "tcp"
    cidr_blocks      = ["0.0.0.0/0"]
  }
   ingress {
    description      = "HTTP "
    from_port        = 80
    to_port          = 80
    protocol         = "tcp"
    cidr_blocks      = ["0.0.0.0/0"]
  }
   ingress {
    description      = "SSH"
    from_port        = 22
    to_port          = 22
    protocol         = "tcp"
    cidr_blocks      = ["0.0.0.0/0"]
  }

  egress {
    from_port        = 0
    to_port          = 0
    protocol         = "-1"
    cidr_blocks      = ["0.0.0.0/0"]
  }

  tags = {
    Name = "TfsecurityGroup"
  }
}
```

```
resource "aws_instance" "pub_ins" {
  ami                          = "ami-0fc5d935ebf8bc3bc"
  instance_type                = "t2.micro"
  subnet_id                    = aws_subnet.pub_sub.id
  vpc_security_group_ids       = [aws_security_group.allow_tfsg.id]
 key_name                      = "David"
 associate_public_ip_address   =  "true"
}
resource "aws_instance" "pri_ins" {
  ami                          = "ami-0fc5d935ebf8bc3bc"
  instance_type                = "t2.micro"
  subnet_id                    =  aws_subnet.prisub.id
  vpc_security_group_ids       = [aws_security_group.allow_tfsg.id]
  key_name                     = "David"
}

#terraform init
#terraform validate
#terraform plan
#terraform apply
#terraform destroy
```