DevOps-Project-BankApp-CICD-Monitoring-LogAggregartion-Multicloud-Multibranch-Pipeline-with-Webhook (AWSandAzure)

By Ritesh Kumar Singh

Email Address: - riteshkumarsingh9559@gmail.com

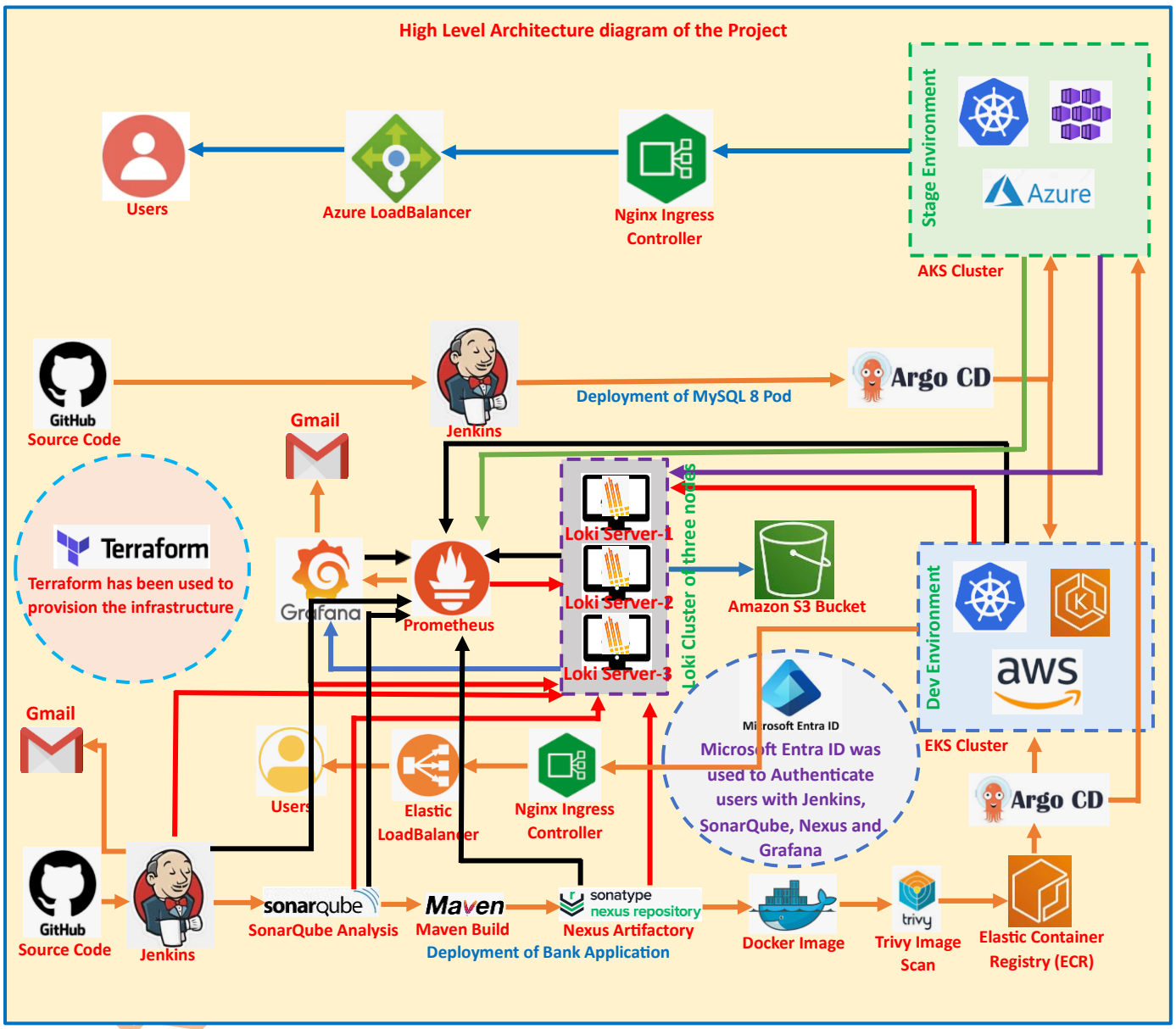
LinkedIn: - https://www.linkedin.com/in/ritesh-kumar-singh-41113128b/

GitHub: - https://github.com/singhritesh85

या कुन्देन्दुतुषारहारधवला या शुभ्रवस्त्रावृता
या वीणावरदण्डमण्डितकरा या श्वेतपद्मासना।
या ब्रह्माच्युत शंकरप्रभृतिभिर्देवैः सदा वन्दिता
सा मां पातु सरस्वती भगवती निःशेषजाड्यापहा ॥

DevOps-Project-BankApp-CICD-Monitoring-LogAggregartion-Multibranch-Pipeline-with-Webhook-Multicloud

Module-1: [Non-Production Environment Dev and Stage]



In the above diagram the black arrow (⟶) indicated the metrics exported by node exporter to the prometheus and red arrow (⟶) indicates logs extracted by the promtail to loki.

This DevOps Project deals with creation of Infrastructure using Terraform and setup of CICD Pipeline using Jenkins, Monitoring using Prometheus and Grafana and Log Aggregation using Loki, Promtail and Grafana. SonarQube was used for Code-Analysis and Maven was used as the Build Tool. Nexus Artifactory was used to keep the Artifacts as shown in the Architecture diagram above. Trivy was used for Docker Image Scan. The Docker Image was kept in the Elastic Container Registry (ECR) and which was deployed to EKS and AKS Cluster using the ArgoCD as shown in the high-level architecture diagram above. User was able to access the Application through the Ingress and hence the

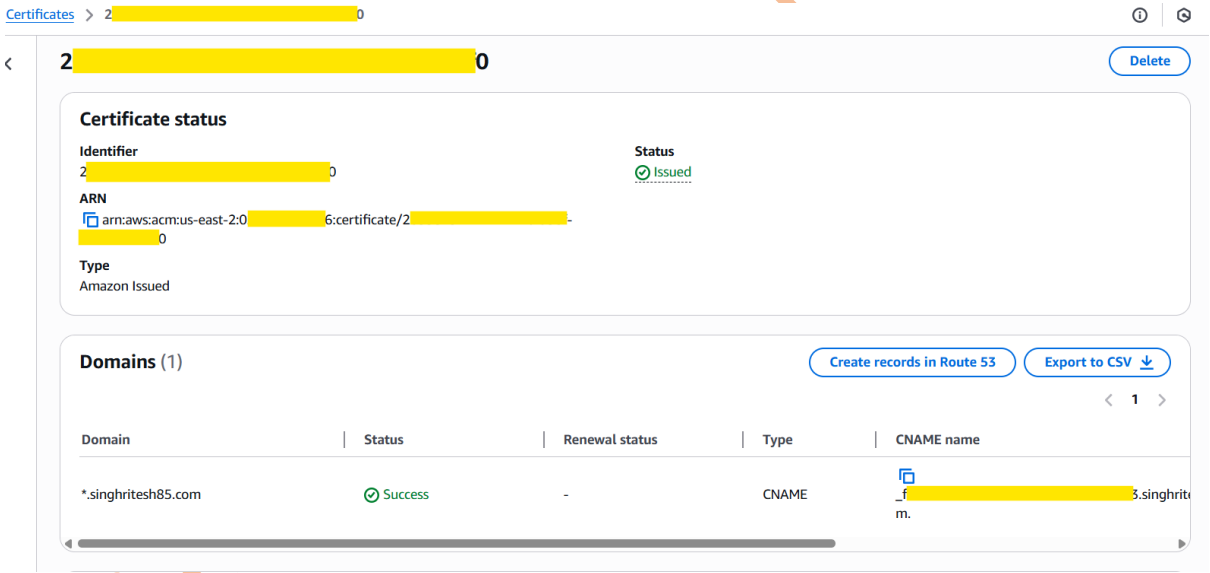
Kubernetes Service. The source code was present in the GitHub Repository https://github.com/singhritesh85/Bank-App-multibranch.git. Promtail and Node Exporter was installed on all the Loki Servers, Grafana, Prometheus, Jenkins (Master and Slave Nodes) using the boot-strapping script. For the EKS and AKS cluster the Promtail and Node Exporter was installed with the help of helm.

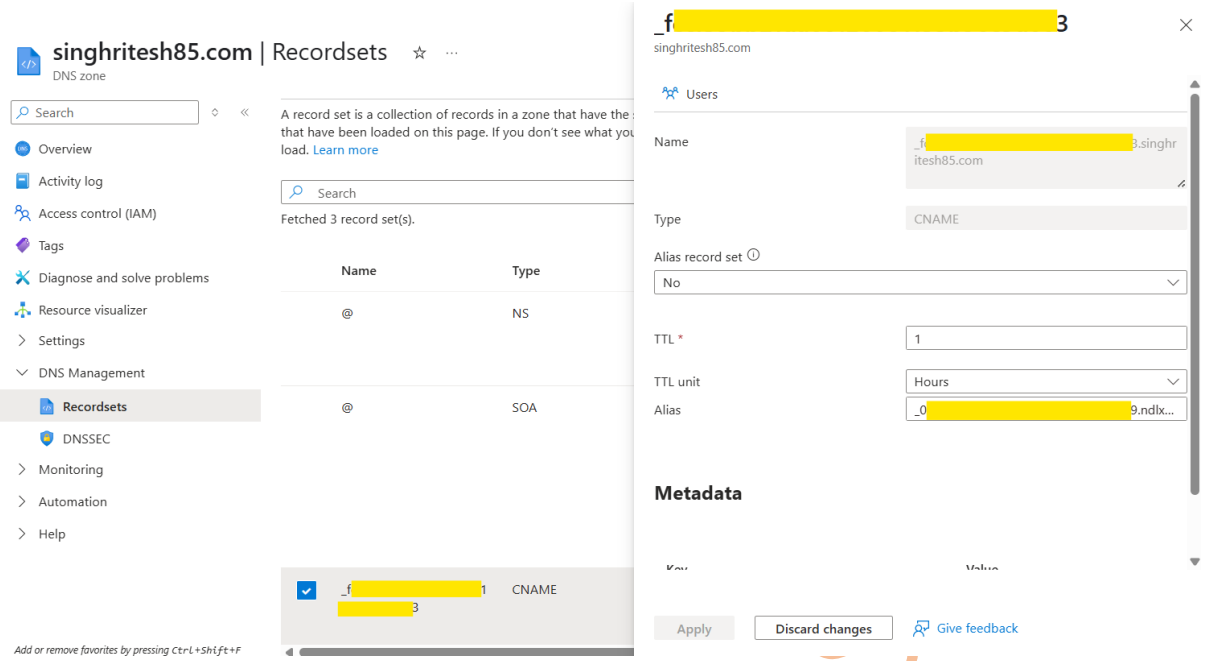
Promtail is Log Aggregation Agent for Loki and Node Exporter agent collects the metrics and forward to Loki and Prometheus respectively. The high-level architecture diagram of the project is as shown above. Installation of Jenkins-Master, Jenkins-Slave, Nexus, SonarQube, Loki Servers, Prometheus and Grafana had been done using the bootstrapping Script.

In this Project I had created multibranch Jenkins pipeline for non-production environment (dev and stage environment). The Dev Environment is available on EKS Cluster and Stage environment is available on AKS Cluster as shown in the diagram drawn above.

For this project the production environment is completely different and which was present of AKS Cluster. For deployment to production environment, I used Azure DevOps CI/CD which I explained in **module-2**.

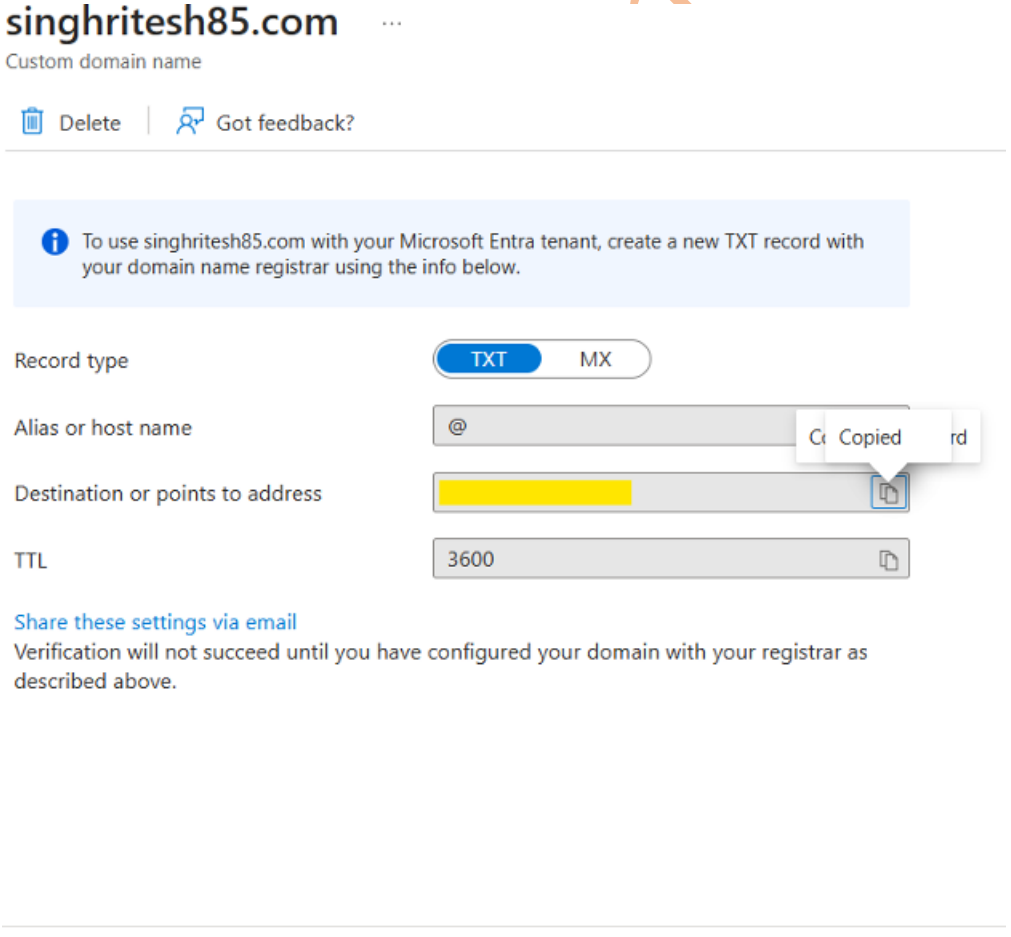
To validate the SSL Certificate generated from AWS certificate manager I used DNS validation and did the entry for Azure DNS Zone record set of type CNAME as shown in the screenshot attached below.





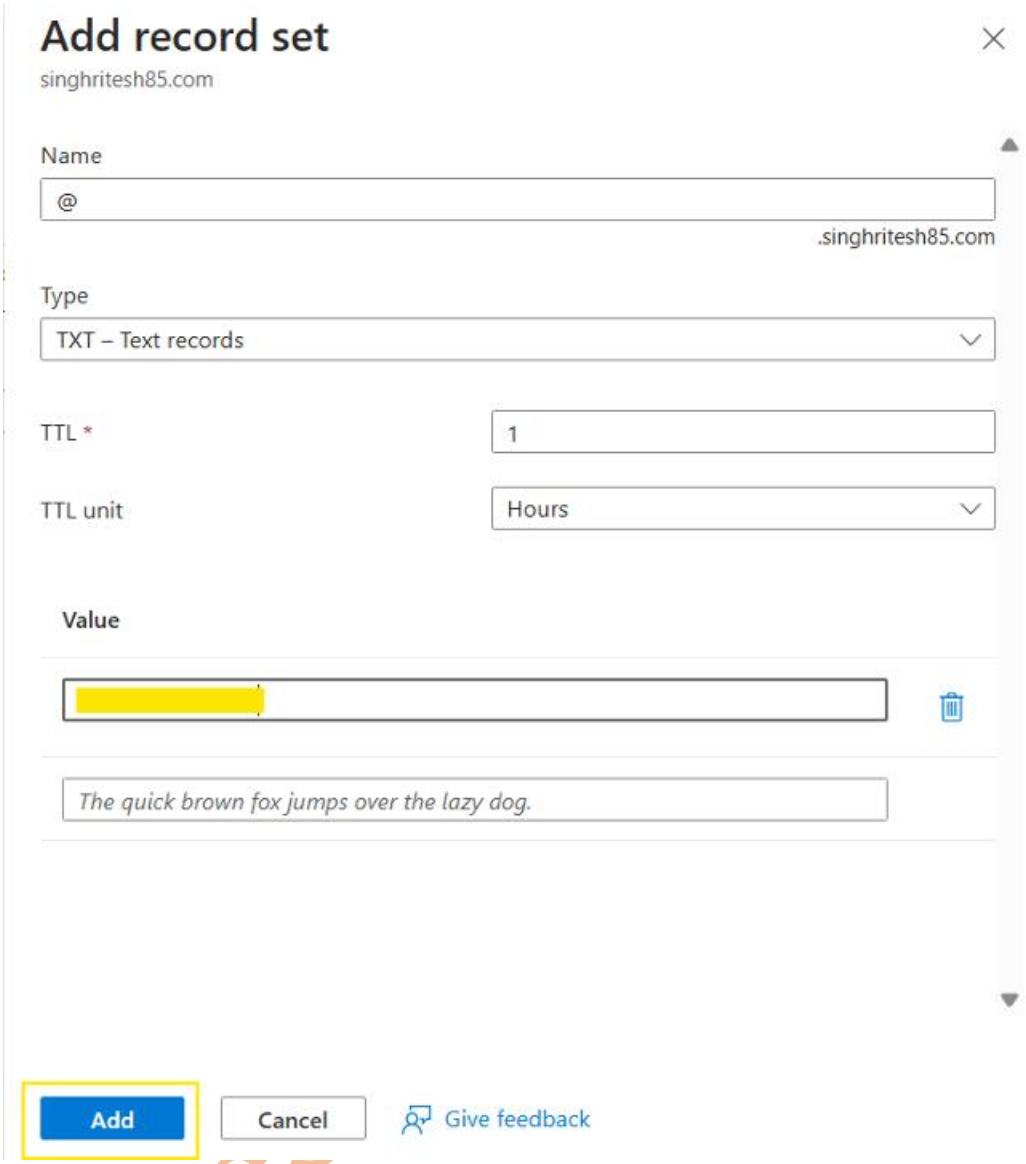
Then I wait for around 20-25 seconds and after that SSL Certificate was issued as shown in the first screenshot attached above.

In my Azure Active Directory, I created a custom domain singhritesh85.com then made this custom domain as primary domain as shown in the screenshot attached below.

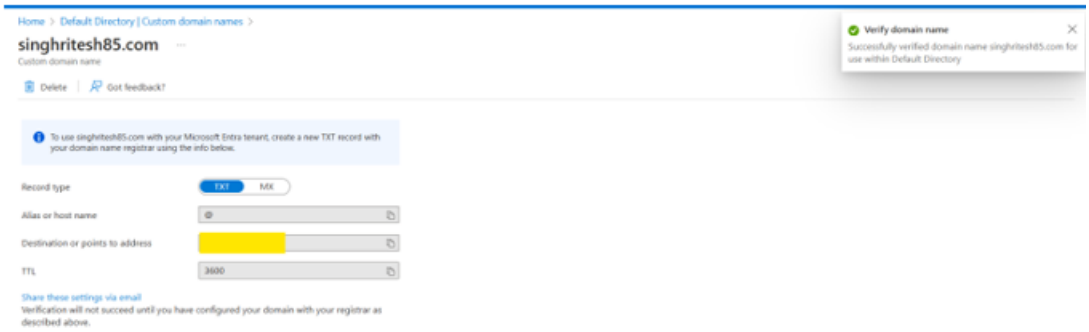


Ritesh Kumar Singh || Email Address: - riteshkumarsingh9559@gmail.com **|| LinkedIn: -** https://www.linkedin.com/in/ritesh-kumar-singh-41113128b/ **|| GitHub: -** https://github.com/singhritesh85

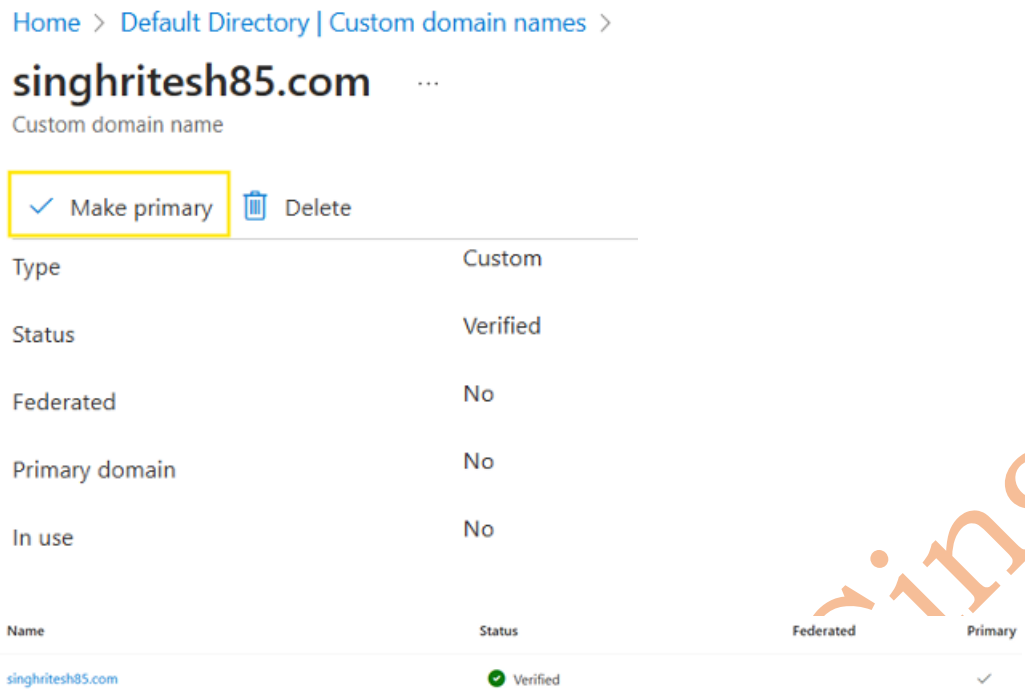
Do the entry in Azure DNS Zone to create the record set with Name and value and type TXT as shown in the screenshot attached below.



Then verified it and found it was verified successfully as shown in the screenshot attached below.



Then made this as a primary domain as shown in the screenshot attached below



Verified from above attached screenshot, in my Azure Entra ID, the primary domain is **singhritesh85.com**.

I installed terraform on Alma Linux2 Azure VM and State file was kept in the S3 Bucket and state lock had been achieved using the AWS DynamoDB. Before running the terraform script I ran the shell script present in the directory **terraform-differentbranch-multibranch-multicloud** as shown in the screenshot attached below. This shell script will install the aws cli, kubectl and helm.

```
[root@ terraform-multi-kubernetes-cluster-multicloud]# ./initial-setup.sh
```

Then I logged-out and login again ran the command aws configure as shown in the screenshot attached below. As it is an important step to Authenticate and Authorize the user before creating resources using terraform in the AWS Account.

```
[root@ terraform-multi-kubernetes-cluster-multicloud]# cd main/
[root@ main]# aws configure
AWS Access Key ID [****************    ]:
AWS Secret Access Key [****************    ]:
Default region name [          ]:
Default output format [    ]:
[root@ main]#
```

Then I installed Azure CLI and authenticated and authorize the user as shown in the screenshot attached below.

```
[root@ main]# yum install -y https://packages.microsoft.com/config/rhel/8/packages-microsoft-prod.rpm
```

```
[root@ main]# yum install azure-cli -y
```

```
[root@ main]# az login
To sign in, use a web browser to open the page                    and enter the code          to authenticate.
```

Then you can run the below commands

terraform init -------> initializes a working directory containing configuration files and installs plugins for required providers.

terraform validate --------> verify that terraform configuration file is correct or not

terraform plan ---------> Check which resources are going to be created.

Then you can run the command **terraform apply -auto-approve** -----> Finally, Create the resources.

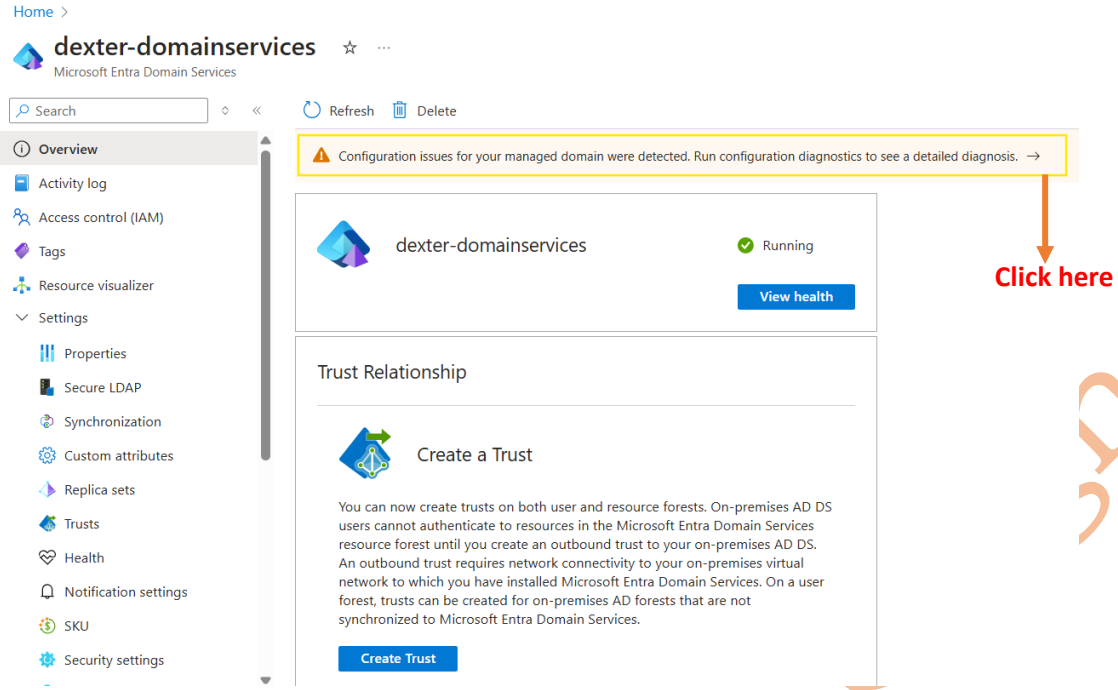
```
module.eks_cluster.azurerm_active_directory_domain_service.entra_ds: Still creating... [1h13m40s elapsed]
module.eks_cluster.azurerm_active_directory_domain_service.entra_ds: Still creating... [1h13m50s elapsed]
module.eks_cluster.azurerm_active_directory_domain_service.entra_ds: Creation complete after 1h13m56s [id=/subscriptions/
resourceGroups/aks-ds-rg/providers/Microsoft.AAD/domainServices/dexter-domainservices/initialReplicaSetId/ ]

Apply complete! Resources: 2 added, 5 changed, 0 destroyed.

Outputs:

acr_ec2_private_ip_alb_dns = {
  "EC2_Instance_Blaxbox_Exporter_Server_Private_IP_Address" = "
  "EC2_Instance_Grafana_Server_Private_IP_Address" = "
  "EC2_Instance_Jenkins_Master_Server_Private_IP_Address" = "
  "EC2_Instance_Jenkins_Slave_Server_Private_IP_Address" = "
  "EC2_Instance_Loki_Servers_Private_IP_Addresses" = [
    "
    "
    "
  ]
  "EC2_Instance_Prometheus_Server_Private_IP_Address" = "
  "EC2_Instance_SonarQube_Server_Private_IP_Address" = "
  "Grafana_ALB_DNS_Name" = "Grafana-.us-east-2.elb.amazonaws.com"
  "Jenkins_ALB_DNS_Name" = "jenkins-ms-.us-east-2.elb.amazonaws.com"
  "Loki_ALB_DNS_Name" = "Loki-.us-east-2.elb.amazonaws.com"
  "SonarQube_ALB_DNS_Name" = "SonarQube-.us-east-2.elb.amazonaws.com"
  "registry_id" = "
  "repository_url" = "02 6.dkr.ecr.us-east-2.amazonaws.com/bankapp"
}
```

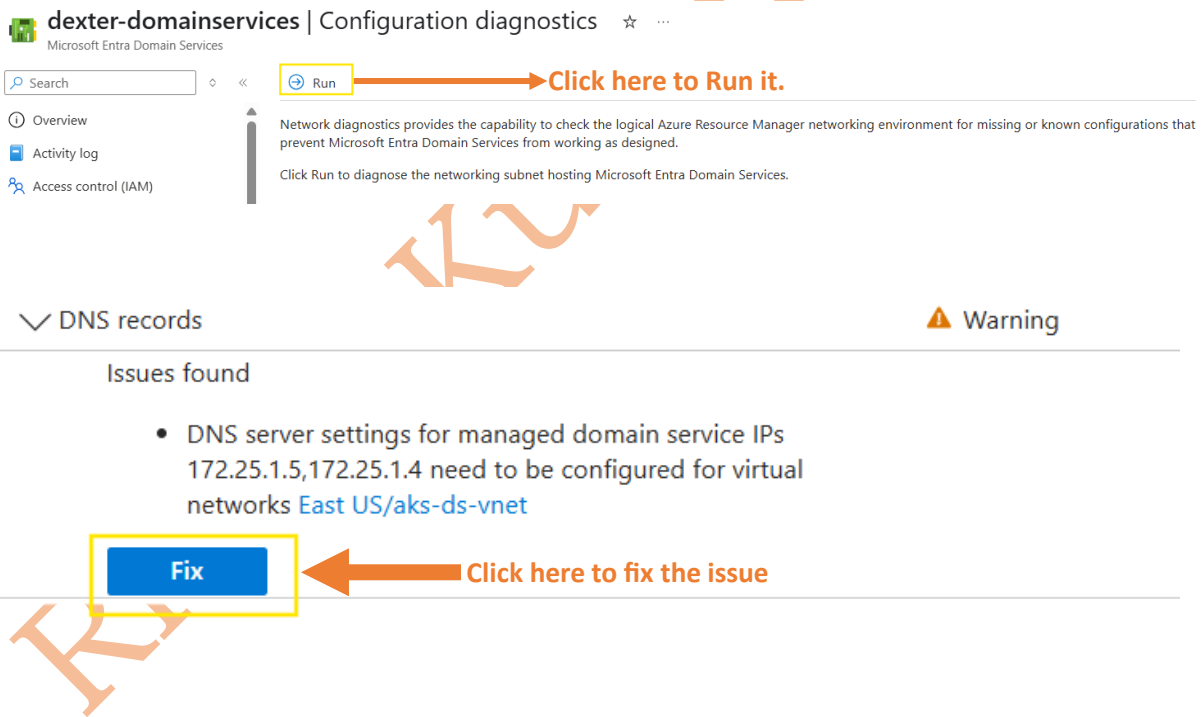
After creation of the resources a kubeconfig file was generated, I did below change in the kubeconfig file as shown in the screenshot attached below.

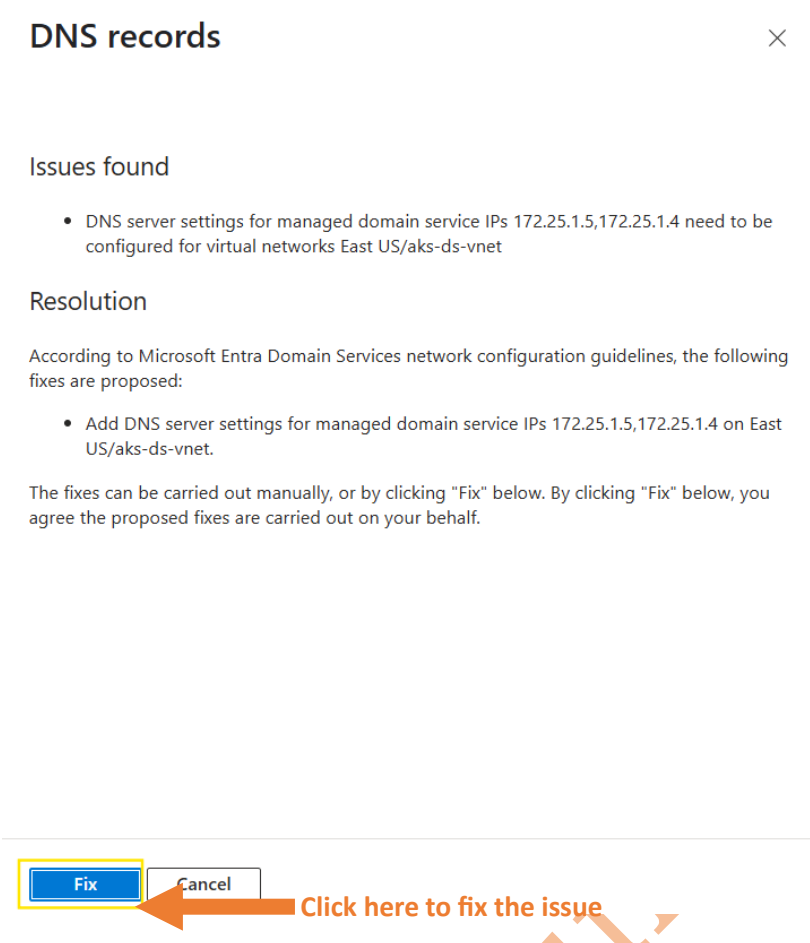
```
    server: https://aks-cluster-dns- .eastus. .io:443
  name: aks-cluster
- cluster:
    certificate-authority-data:
    server: https:// .us-east-2.eks.amazonaws.com
  name: eks-demo-cluster-dev
contexts:
- context:
    cluster: aks-cluster
    user: clusterUser_aks-rg_aks-cluster
  name: aks-cluster
- context:
    cluster: eks-demo-cluster-dev
    user: arn:aws:eks:us-east-2:02 5:cluster/eks-demo-cluster-dev
  name: eks-demo-cluster-dev
current-context: aks-cluster
kind: Config
preferences: {}
users:
- name: clusterUser_aks-rg_aks-cluster
  user:
    client-certificate-data:
@@
"~/.kube/config" 42L, 11795C                                          20,28         19%
```

It is my suggestion before any change in the kubeconfig file take a backup of the original kubeconfig file.

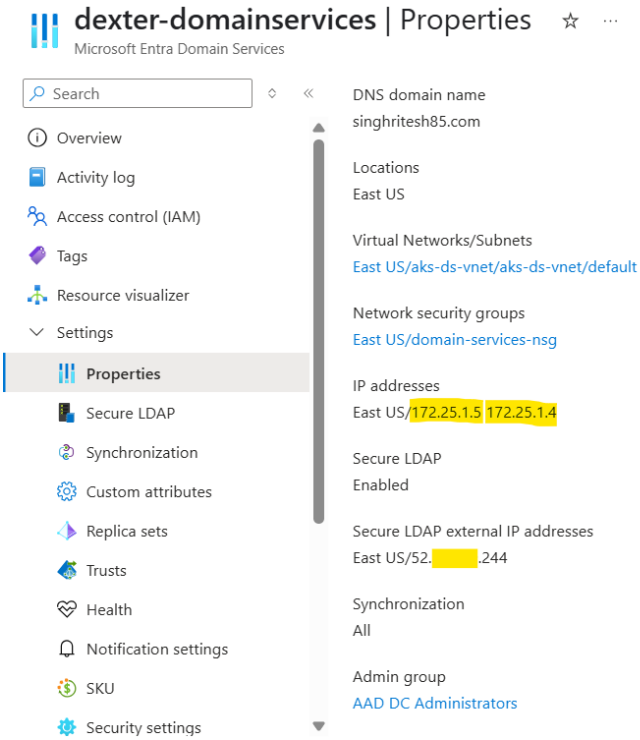
This terraform script took total 1 hour 15 minutes in all to get executed successfully as shown in the screenshot attached above. After that All the resources in AWS and Azure will be created. The Microsoft Entra Domain Services gets created and I did below change in the Microsoft Entra Domain Services as shown in the screenshot attached below.

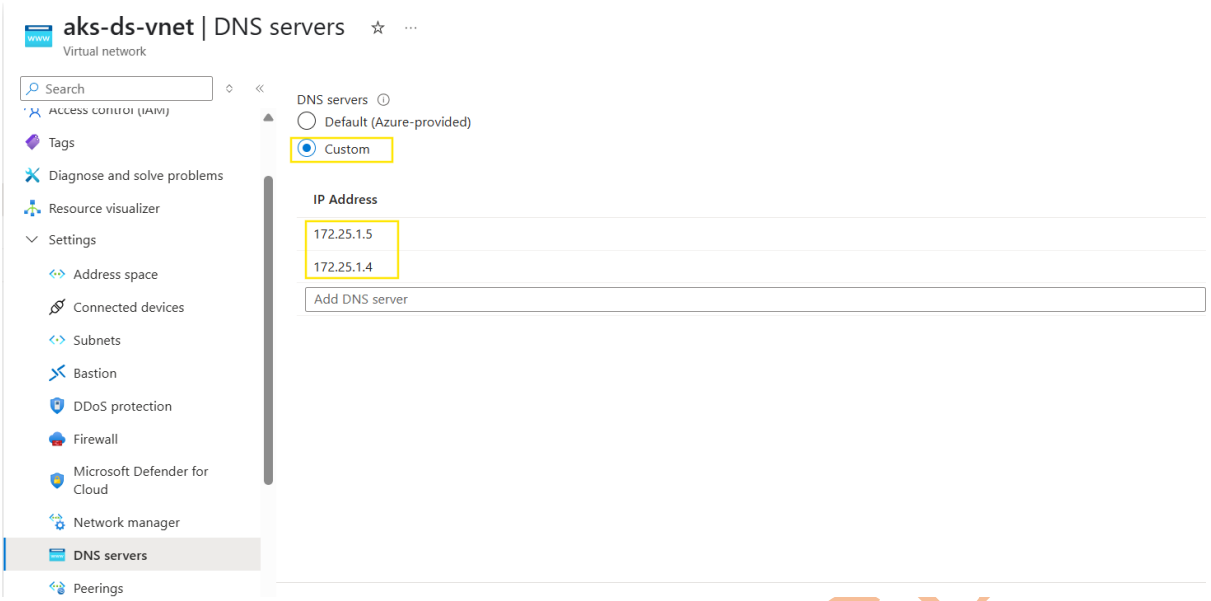

Home
dexter-domainservices
Microsoft Entra Domain Services
Search
Refresh
Delete
Overview
Activity log
Access control (IAM)
Tags
Resource visualizer
Settings
Properties
Secure LDAP
Synchronization
Custom attributes
Replica sets
Trusts
Health
Notification settings
SKU
Security settings
Configuration issues for your managed domain were detected. Run configuration diagnostics to see a detailed diagnosis.
dexter-domainservices
Running
View health
Click here
Trust Relationship
Create a Trust
You can now create trusts on both user and resource forests. On-premises AD DS users cannot authenticate to resources in the Microsoft Entra Domain Services resource forest until you create an outbound trust to your on-premises AD DS. An outbound trust requires network connectivity to your on-premises virtual network to which you have installed Microsoft Entra Domain Services. On a user forest, trusts can be created for on-premises AD forests that are not synchronized to Microsoft Entra Domain Services.
Create Trust


dexter-domainservices | Configuration diagnostics
Microsoft Entra Domain Services
Search
Overview
Activity log
Access control (IAM)
Run
Click here to Run it.
Network diagnostics provides the capability to check the logical Azure Resource Manager networking environment for missing or known configurations that prevent Microsoft Entra Domain Services from working as designed.
Click Run to diagnose the networking subnet hosting Microsoft Entra Domain Services.
DNS records
Warning
Issues found
DNS server settings for managed domain service IPs 172.25.1.5,172.25.1.4 need to be configured for virtual networks East US/aks-ds-vnet
Fix
Click here to fix the issue

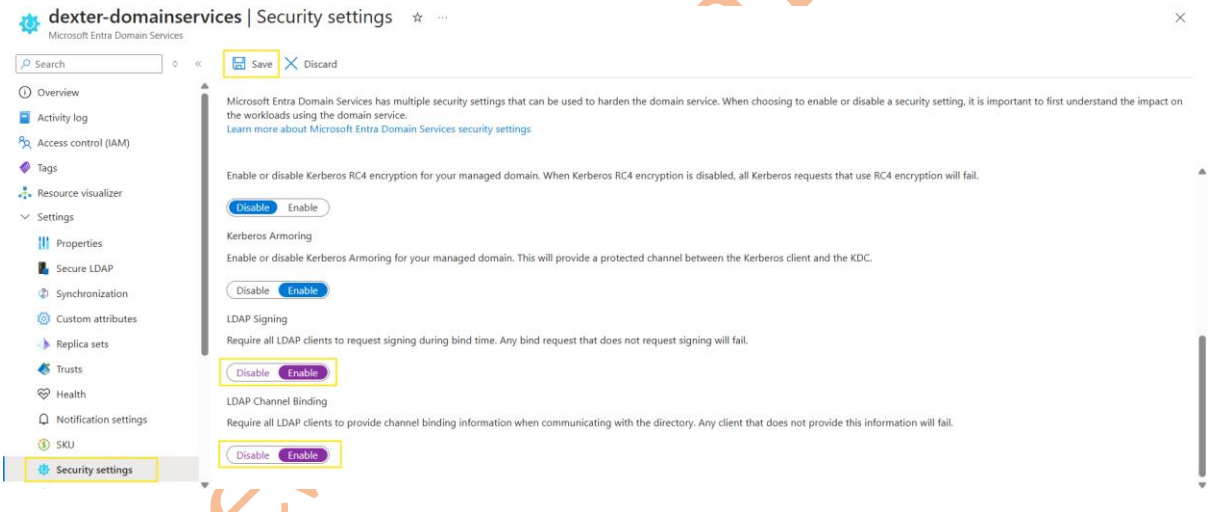


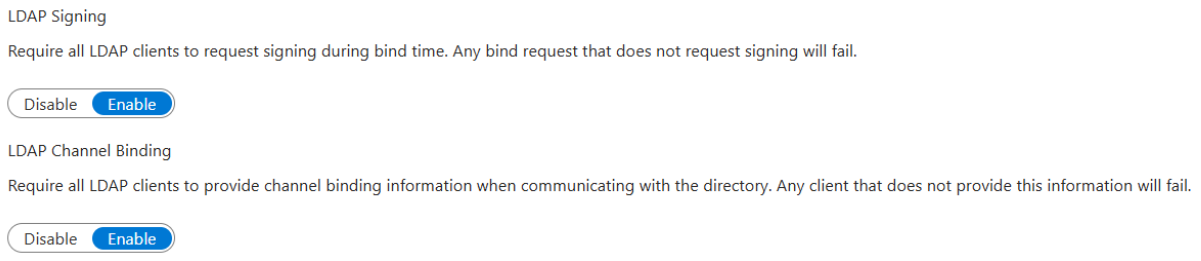
By fixing this issue through click this button will change the DNS Servers for VNet from Default (Azure-Provided) to Custom as shown in the screenshot attached below.



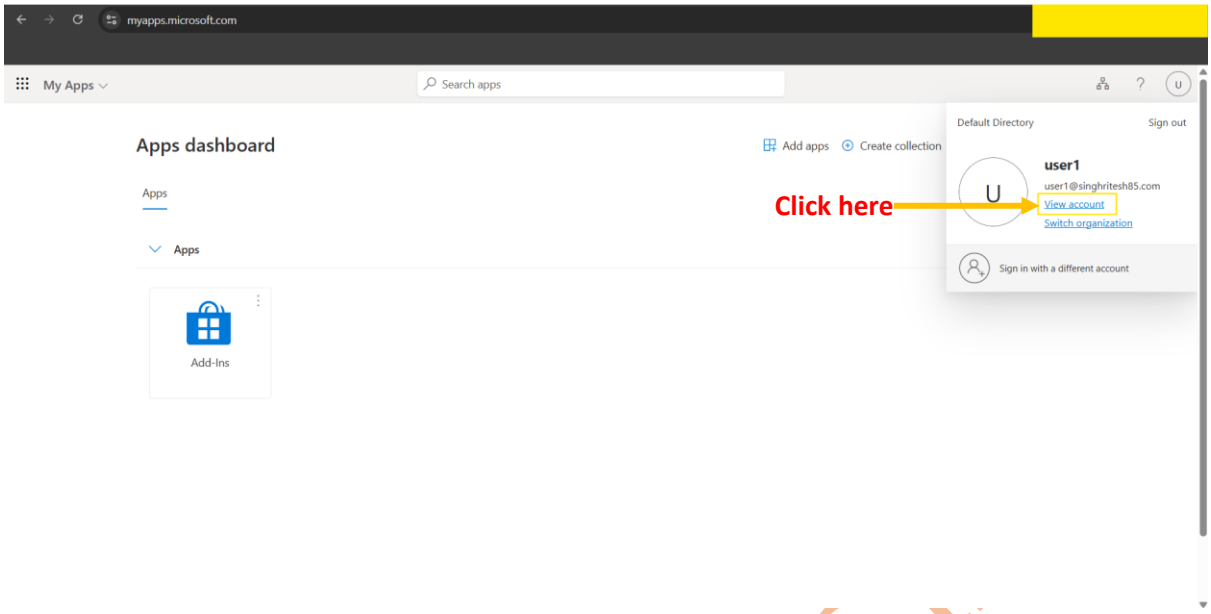


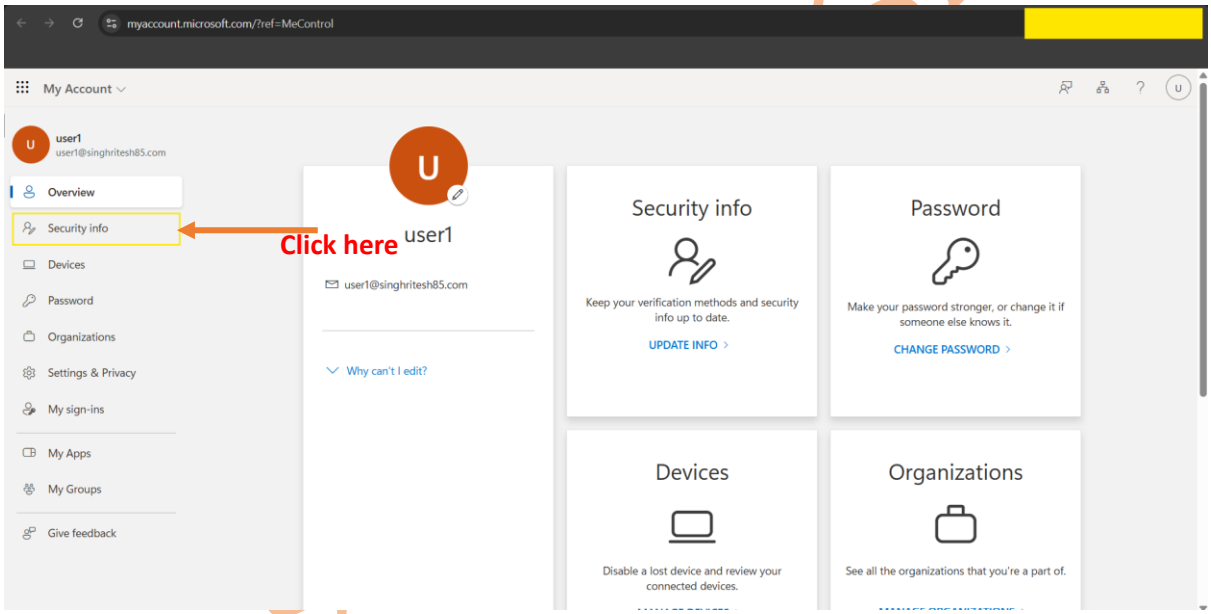
Then go to Security Setting of Microsoft Entra Domain Services and enable LDAP Signing and LDAP Channel Binding then save this setting as shown in the screenshot attached below.

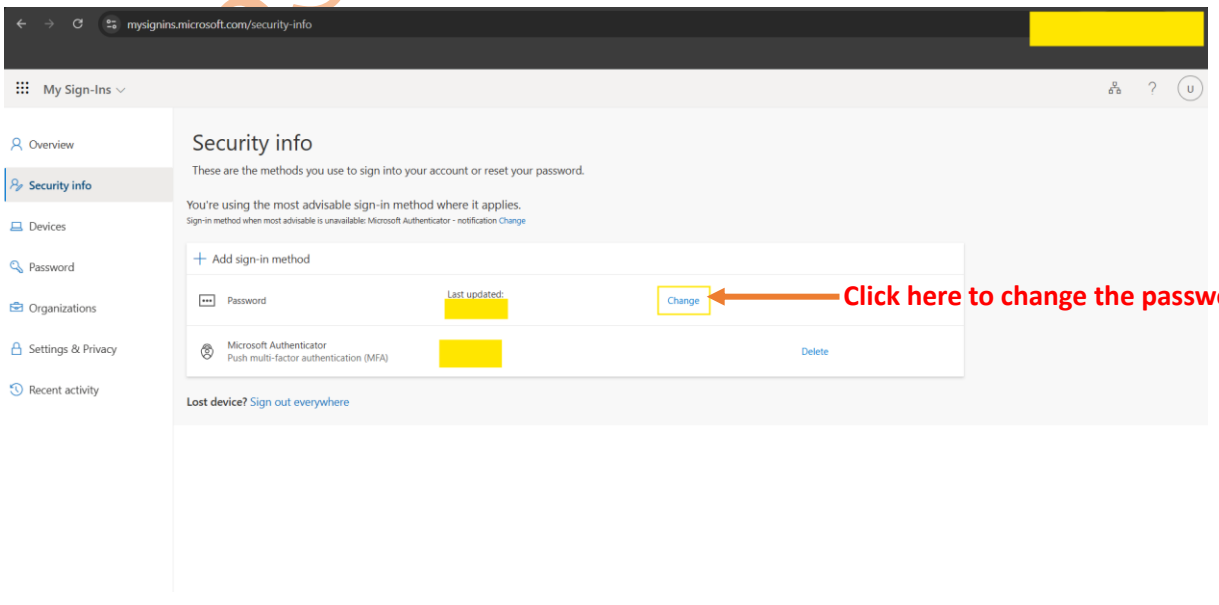


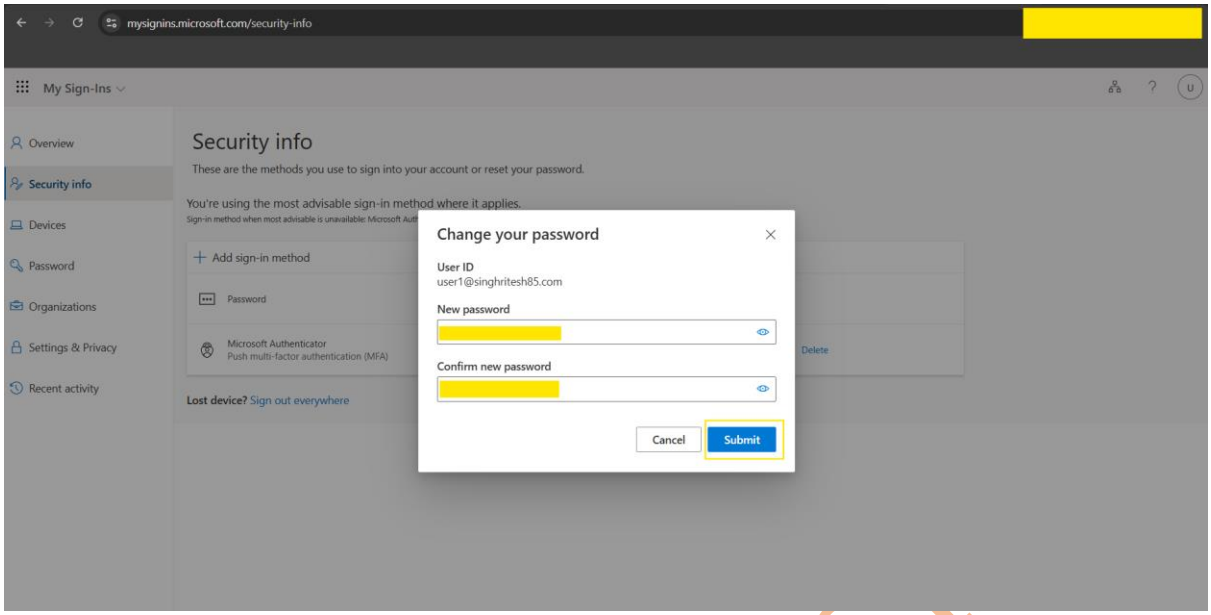


After successfully creating the Microsoft Entra Domain Services it is mandatory to change the password of user **user1** which was the member of Group **AAD DC Administrators**. To change the password of user **user1** I logged-in into the URL https://myapps.microsoft.com as shown in the screenshot attached below and changed the password.

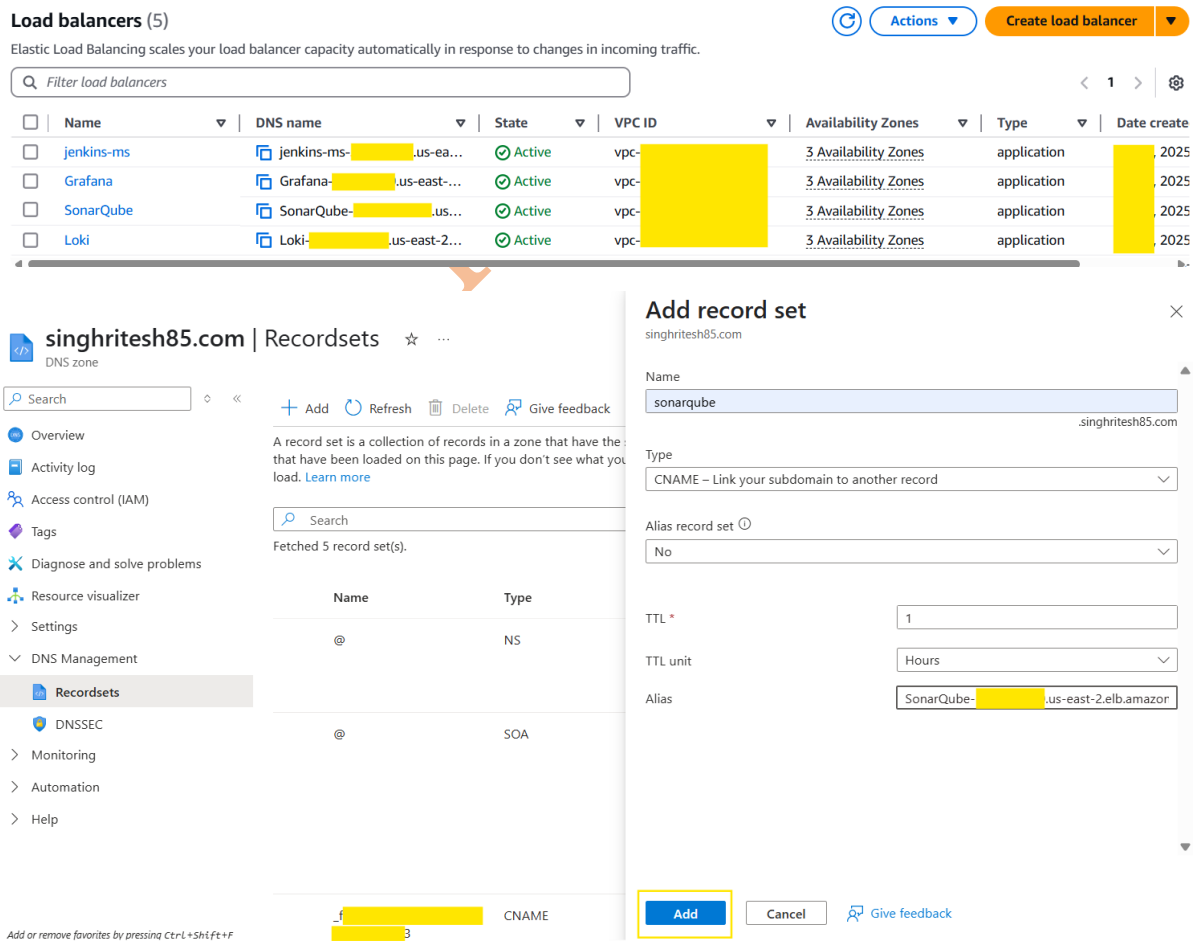

myapps.microsoft.com
My Apps
Search apps
Apps dashboard
Add apps
Create collection
Apps
Apps
Add-Ins
Default Directory
Sign out
user1
user1@singhritesh85.com
View account
Switch organization
Sign in with a different account
Click here


myaccount.microsoft.com/?ref=MeControl
My Account
user1
user1@singhritesh85.com
Overview
Security info
Devices
Password
Organizations
Settings & Privacy
My sign-ins
My Apps
My Groups
Give feedback
Click here
user1
user1@singhritesh85.com
Why can't I edit?
Security info
Keep your verification methods and security info up to date.
UPDATE INFO
Password
Make your password stronger, or change it if someone else knows it.
CHANGE PASSWORD
Devices
Disable a lost device and review your connected devices.
Organizations
See all the organizations that you're a part of.


mysignins.microsoft.com/security-info
My Sign-Ins
Overview
Security info
Devices
Password
Organizations
Settings & Privacy
Recent activity
Security info
These are the methods you use to sign into your account or reset your password.
You're using the most advisable sign-in method where it applies.
Add sign-in method
Password
Last updated:
Change
Microsoft Authenticator
Push multi-factor authentication (MFA)
Delete
Lost device? Sign out everywhere
Click here to change the password

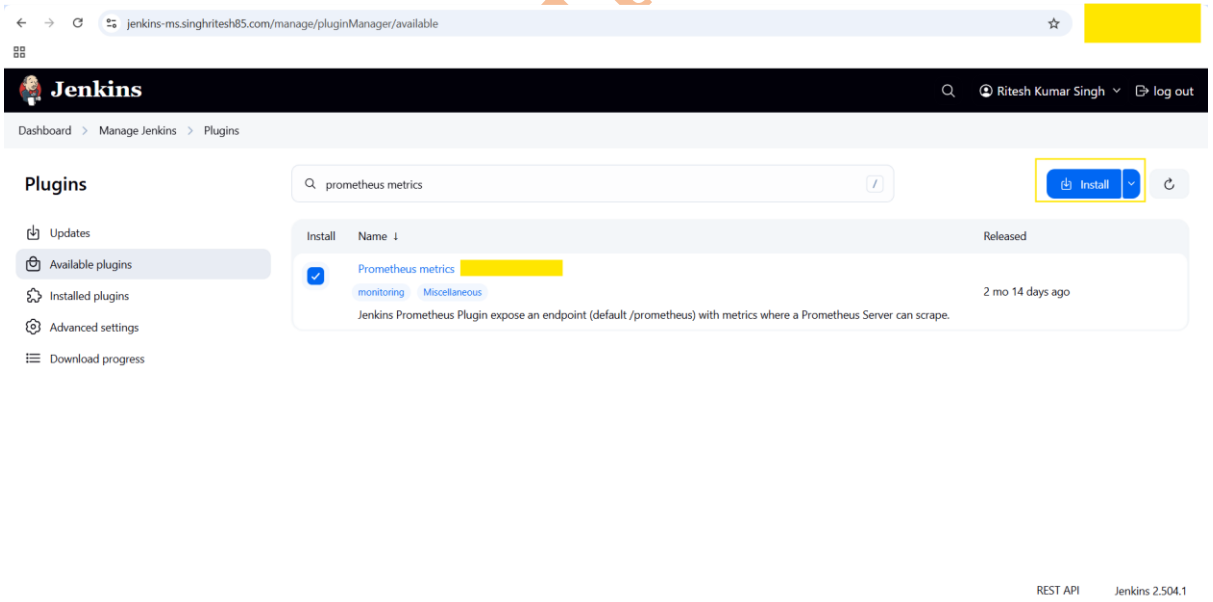


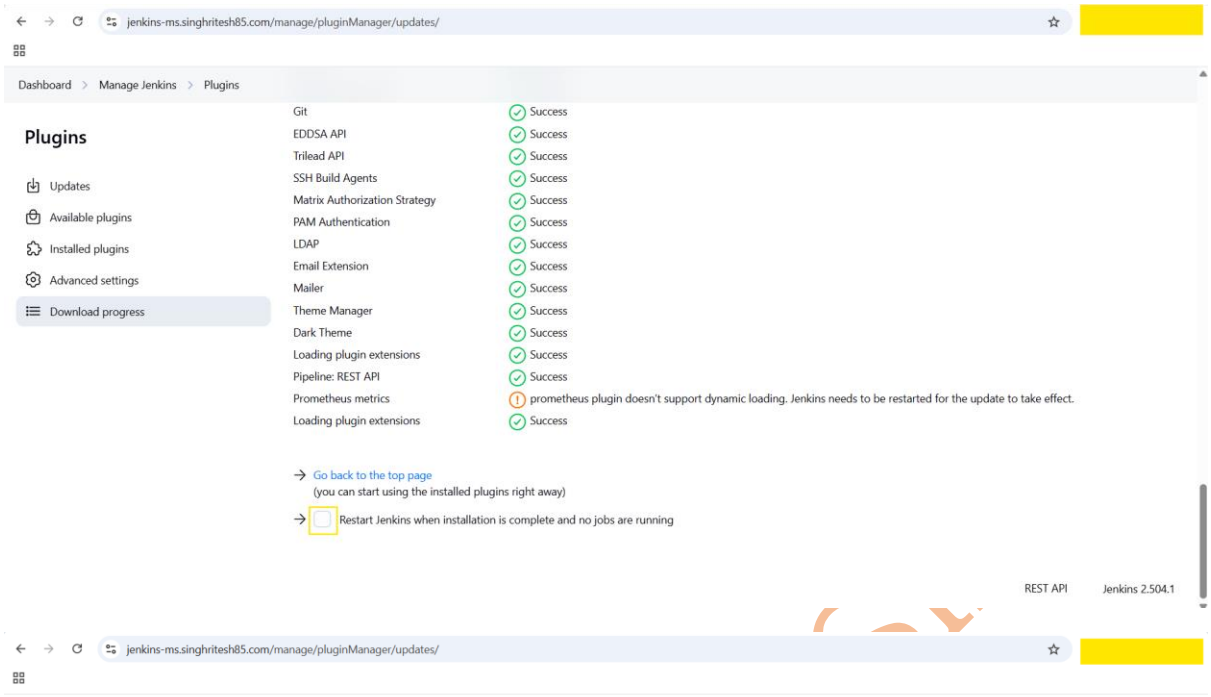
After running the terraform script LoadBalancers for SonarQube, Jenkins, Nexus, Loki and Grafana had been created as shown in the screenshot attached below. I created the record set of type CNAME for SonarQube, Jenkins, Nexus and Grafana in Azure DNS Zone using the DNS Name of the LoadBalancers for example I had shown below the creation of record set for SonarQube.

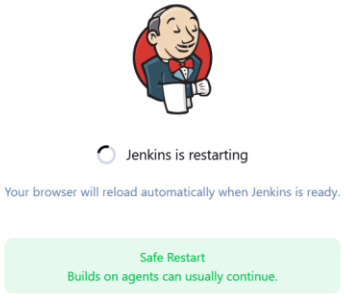


After doing the entry for all the DNS Names (of the LoadBalancers SonarQube, Jenkins, Nexus and Grafana) as shown in the screenshot attached below to create the record set. The final screenshot is as shown in the screenshot attached below.

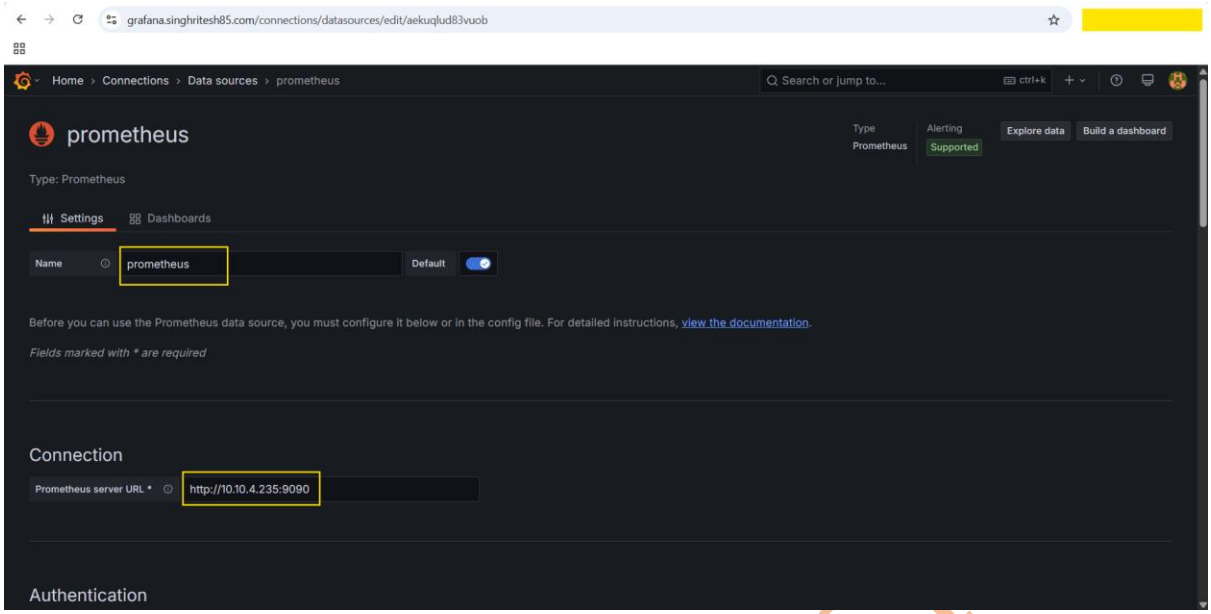
It is possible to monitor Jenkins Job using Prometheus and Grafana and for that I installed **Prometheus Metrics** plugin in Jenkins, after its installation I restarted the Jenkins as shown in the screenshot attached below.

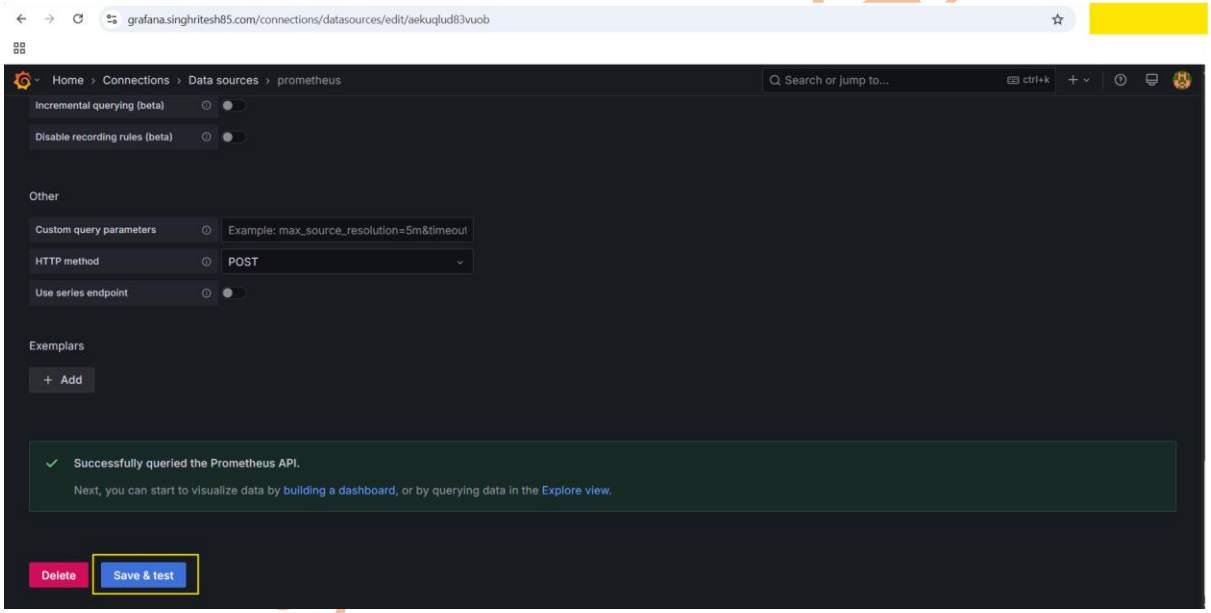


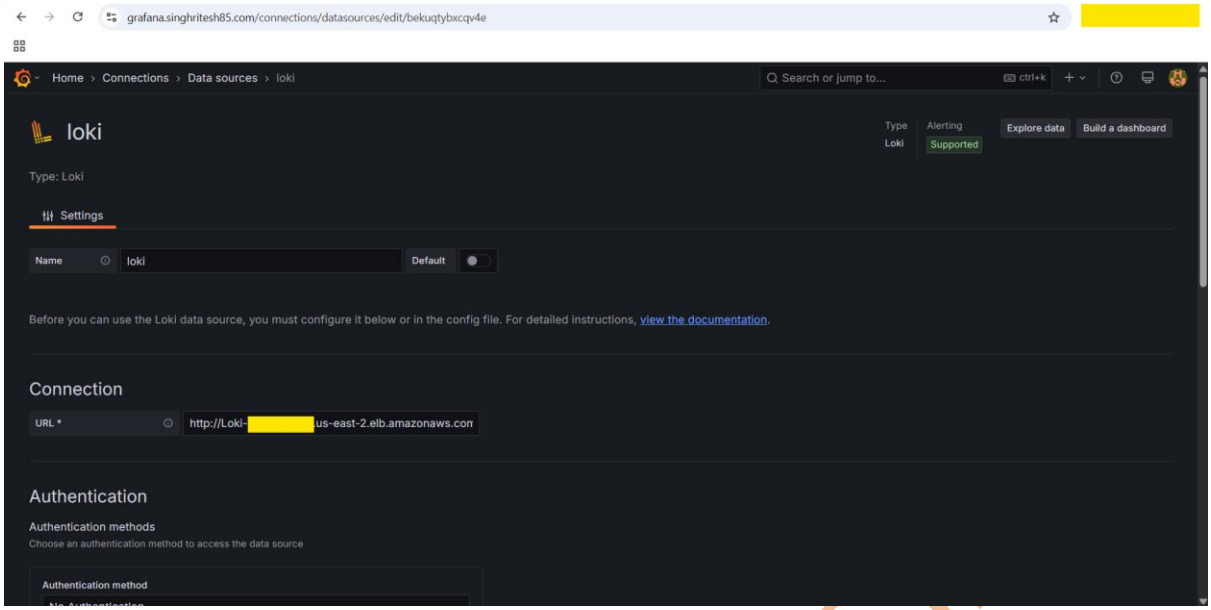


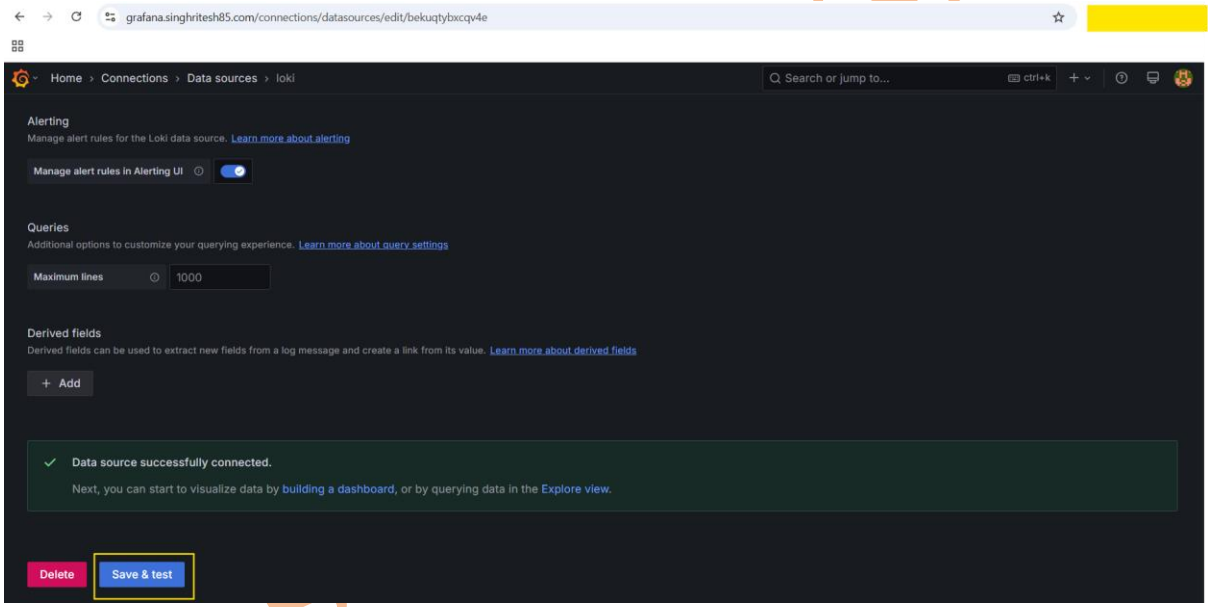


Now, login into the Grafana for the first time and update admin password then login into Grafana and created two data sources each for prometheus and loki as shown in the screenshot attached below.


grafana.singhritesh85.com/connections/datasources/edit/aekuqlud83vuob
Home › Connections › Data sources › prometheus
Search or jump to...
ctrl+k
prometheus
Type
Prometheus
Alerting
Supported
Explore data
Build a dashboard
Type: Prometheus
Settings
Dashboards
Name
prometheus
Default
Before you can use the Prometheus data source, you must configure it below or in the config file. For detailed instructions, view the documentation.
Fields marked with * are required
Connection
Prometheus server URL *
http://10.10.4.235:9090
Authentication


grafana.singhritesh85.com/connections/datasources/edit/aekuqlud83vuob
Home › Connections › Data sources › prometheus
Search or jump to...
ctrl+k
Incremental querying (beta)
Disable recording rules (beta)
Other
Custom query parameters
Example: max_source_resolution=5m&timeout
HTTP method
POST
Use series endpoint
Exemplars
+ Add
Successfully queried the Prometheus API.
Next, you can start to visualize data by building a dashboard, or by querying data in the Explore view.
Delete
Save & test

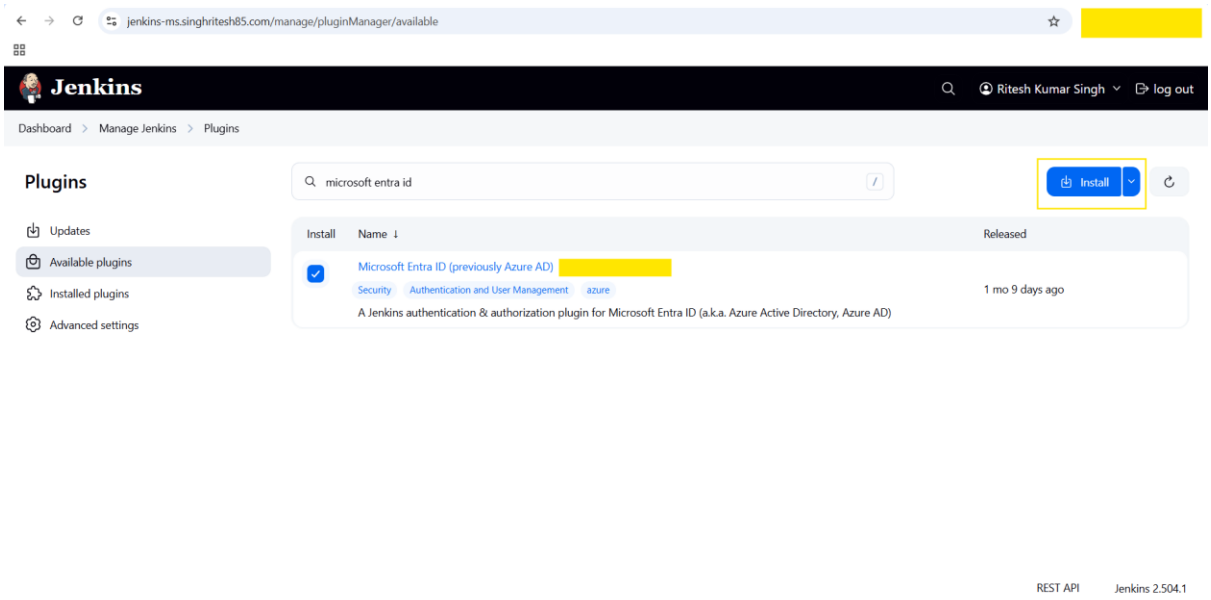




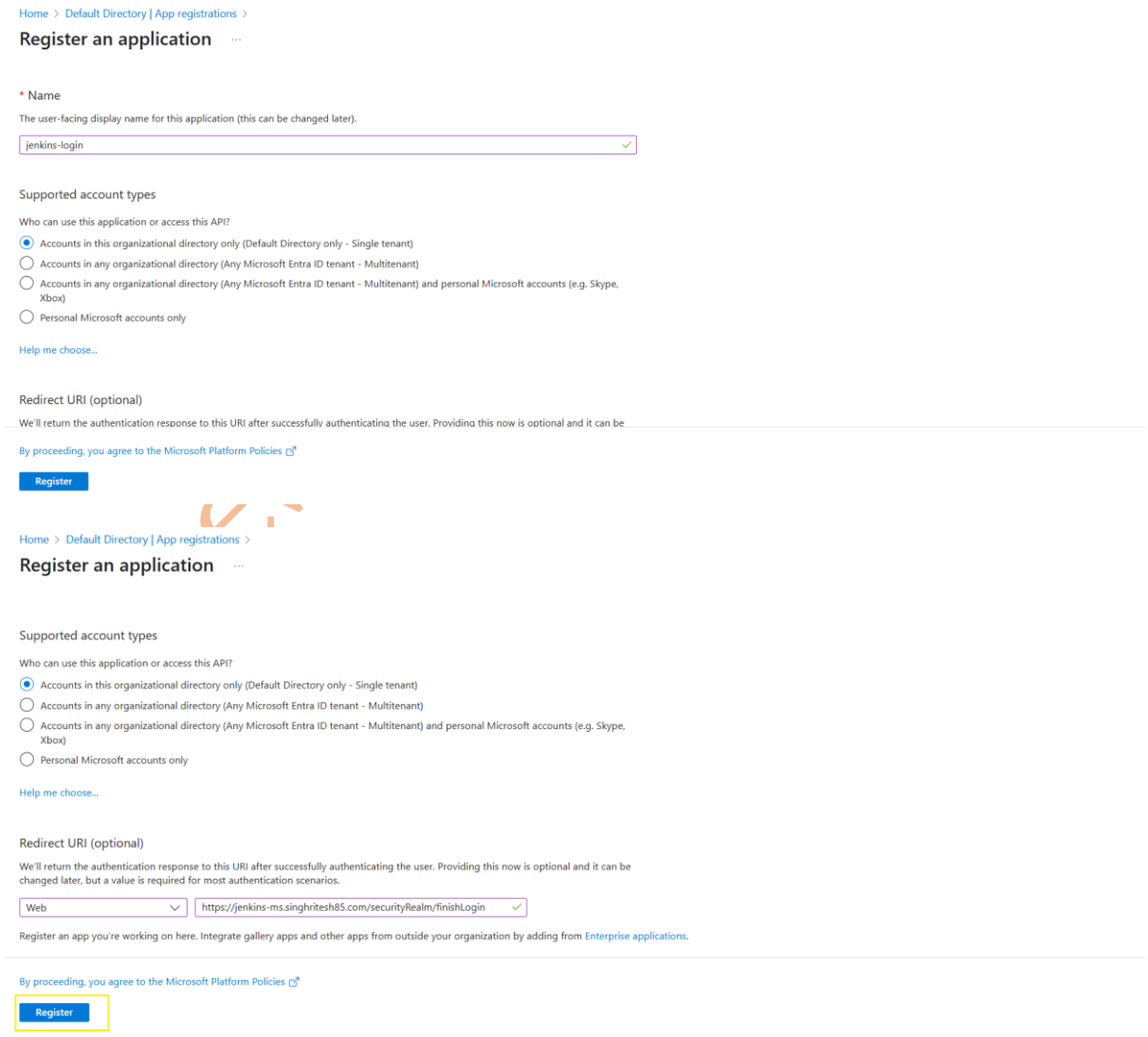
Now I will configure the Integration of Azure Entra ID with Jenkins, SonarQube and Grafana for Authentication

Configuration of Integration of Azure Entra ID with Jenkins

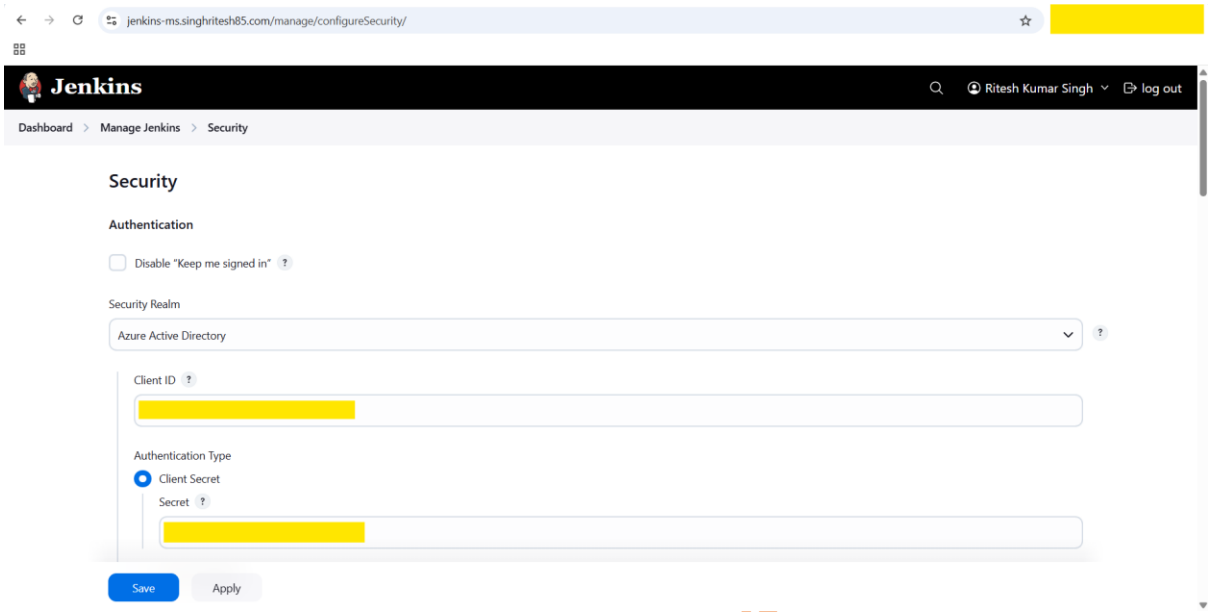
Search from the list of Available plugins for **Microsoft Entra ID** Plugin in Jenkins and install it as shown in the screenshot attached below.



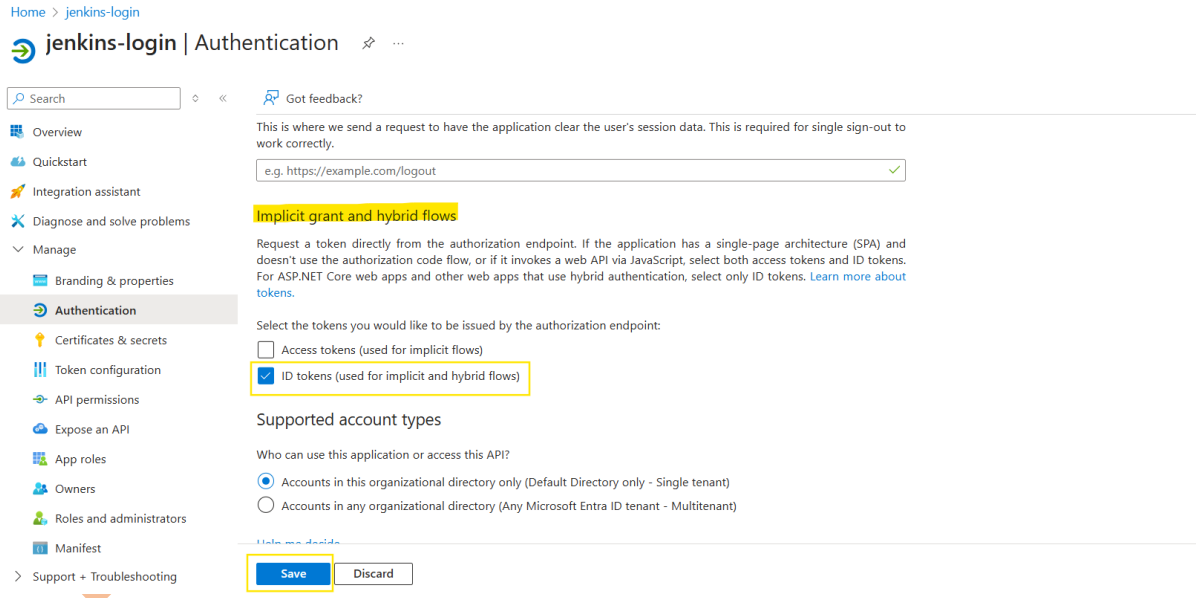
After installation of Azure Entra ID Plugin in Jenkins restart Jenkins and go to the Azure Portal and open Entra ID and create a new App Registration as shown in the screenshot attached below.



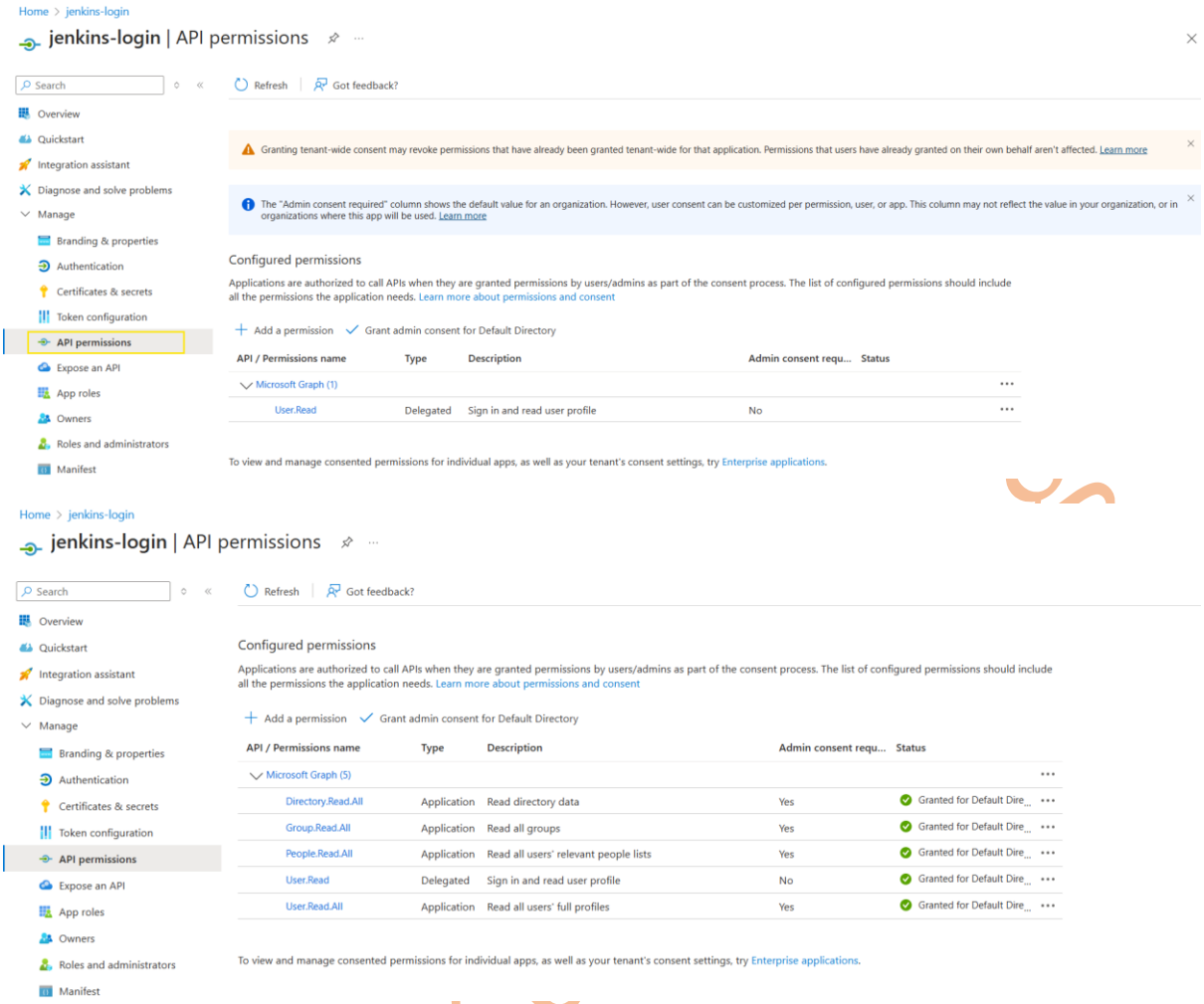
After App Registration in Azure go to Jenkins and **Manage Jenkins > Security > Security Realm and Select the Azure Active Directory** and provide the Client ID, Client Secret and Tenant ID then Save it as shown in screenshot attached below.



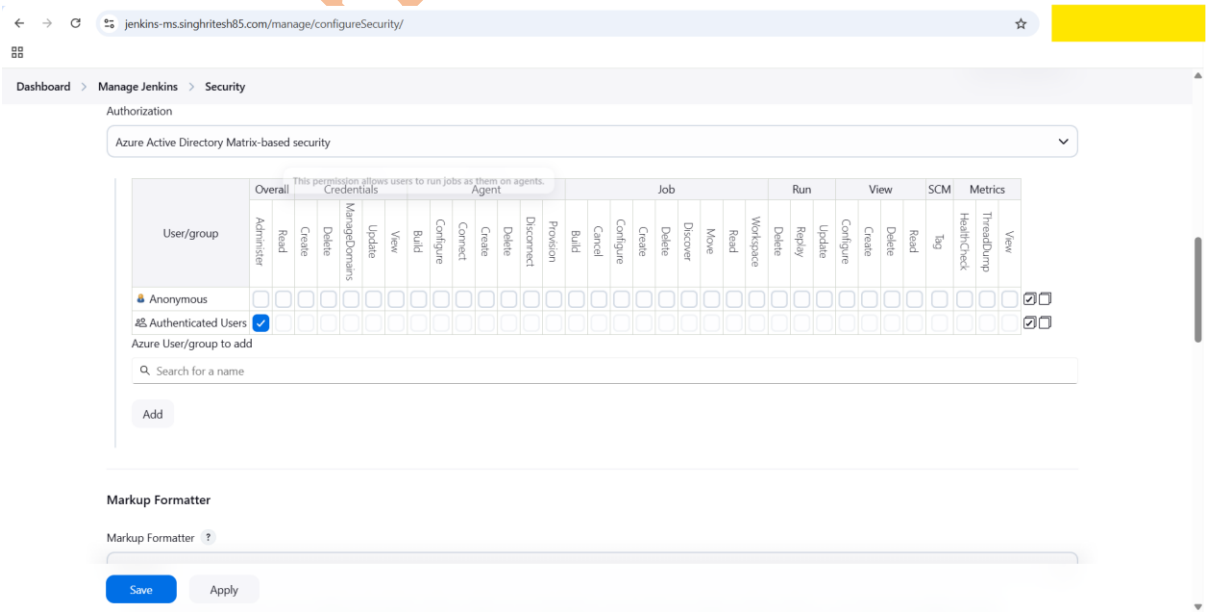
In the Registered Application go to Application and under **Implicit grant and hybrid flows** and select **ID Tokens** as shown in the screenshot attached below.



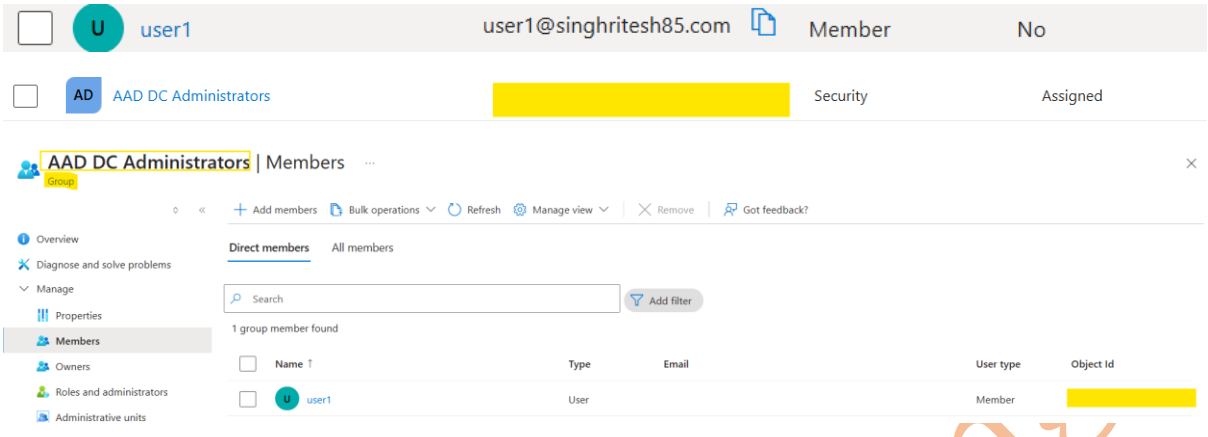
Now select API Permission > Add a permission > Microsoft Graph > Application Permissions People.Read.All, Group.Read.All, User.Read.All and Directory.Read.All in the created Registered App. Then Click on Grant admin consent as shown in the screenshot attached below



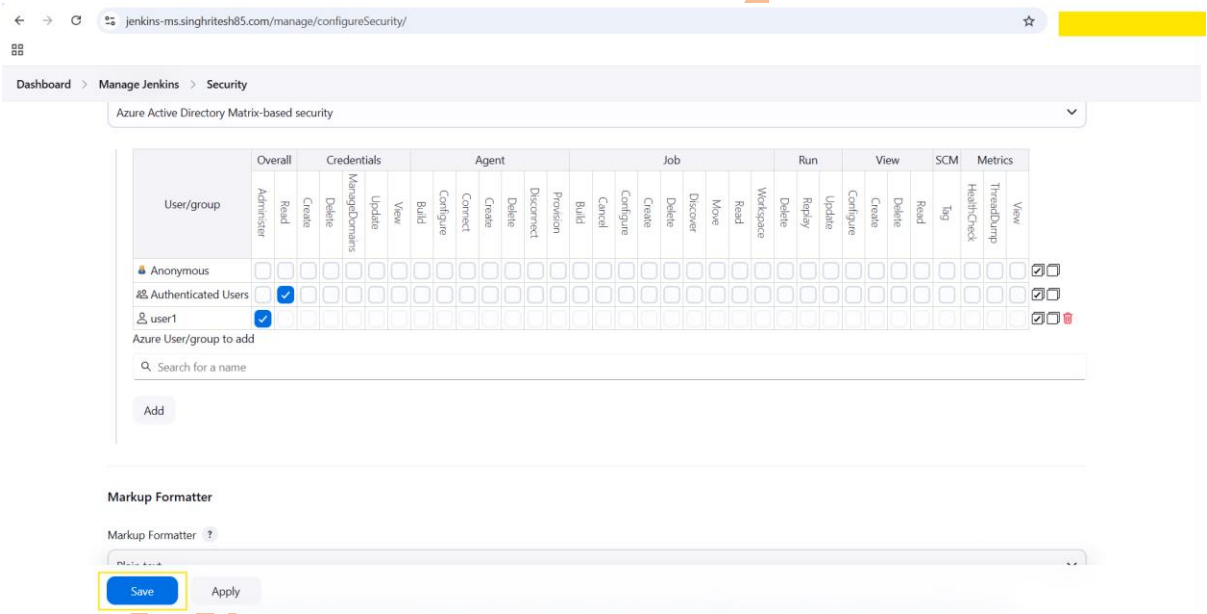
For Authorization select Azure Active Directory Matrix based Authorization as shown in the screenshot attached below, to login for the very first time provide Authenticated users as overall access and the login with a user and add other users if needed.



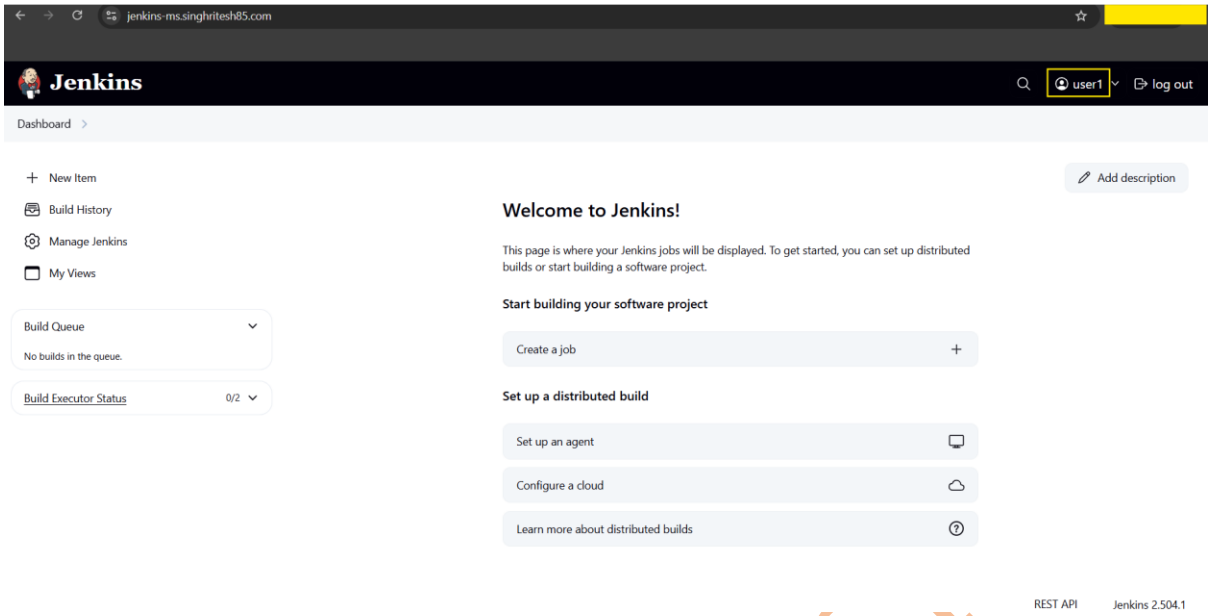
Below screenshot shows the user and group in Azure Entra ID Which I had created initially using the terraform.



For the very first time I logged-in with a user then I provided user1 as administrator privileges and removed the Authenticated user overall administrator access (and provided overall Read access) which I provided before first time logged-in as shown in the screenshot attached below.

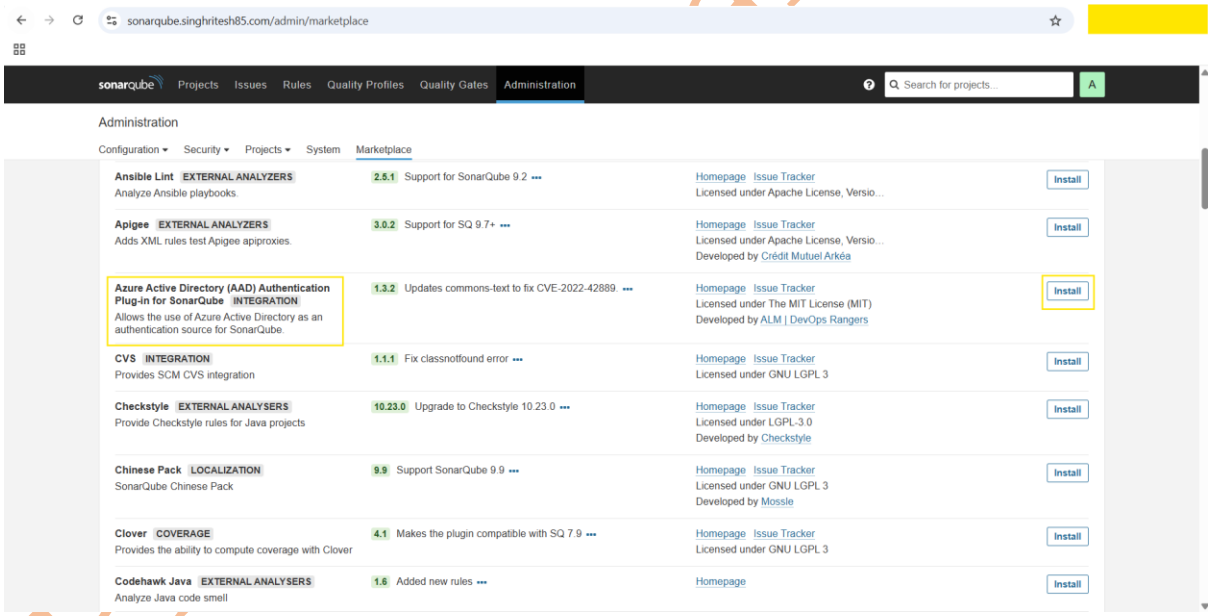


Below screenshot shows the access which I had when I logged-in with user1.

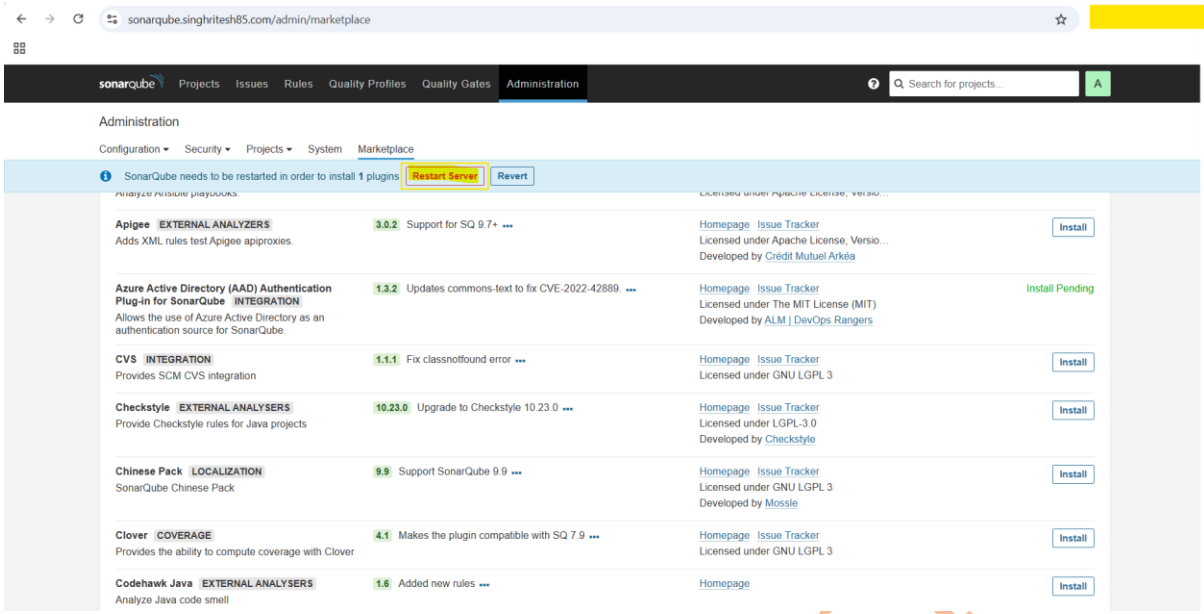


Integration of Azure Entra ID with SonarQube

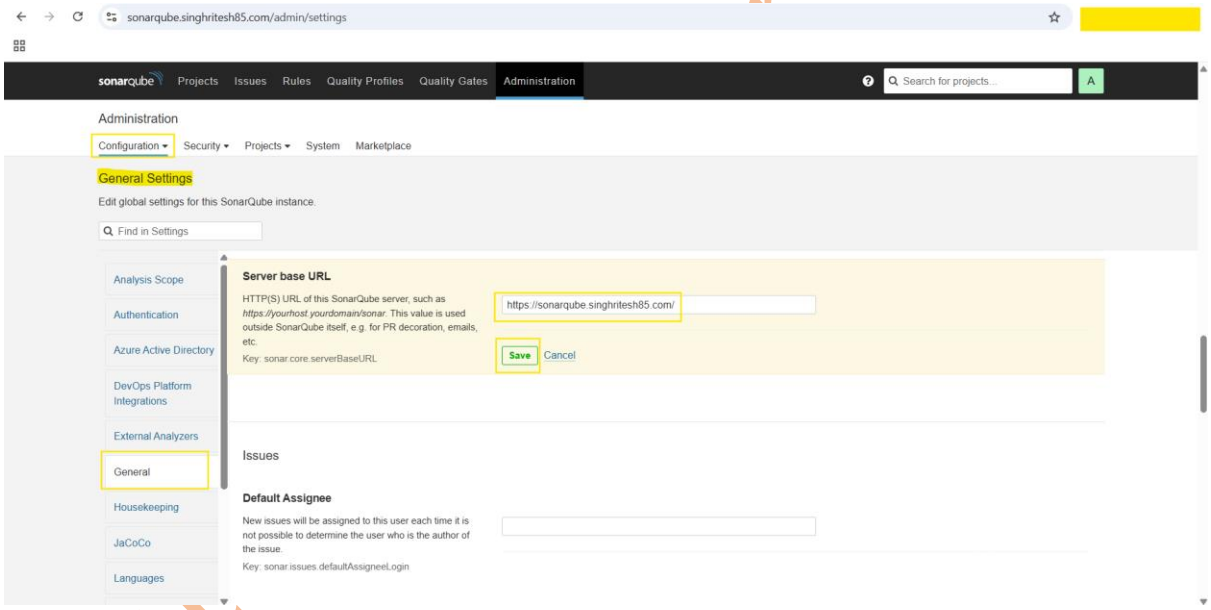
To Integrate Azure Active Directory with SonarQube I installed Azure Active Directory (AAD) Authentication Plug-in for SonarQube as shown in the screenshot attached below.



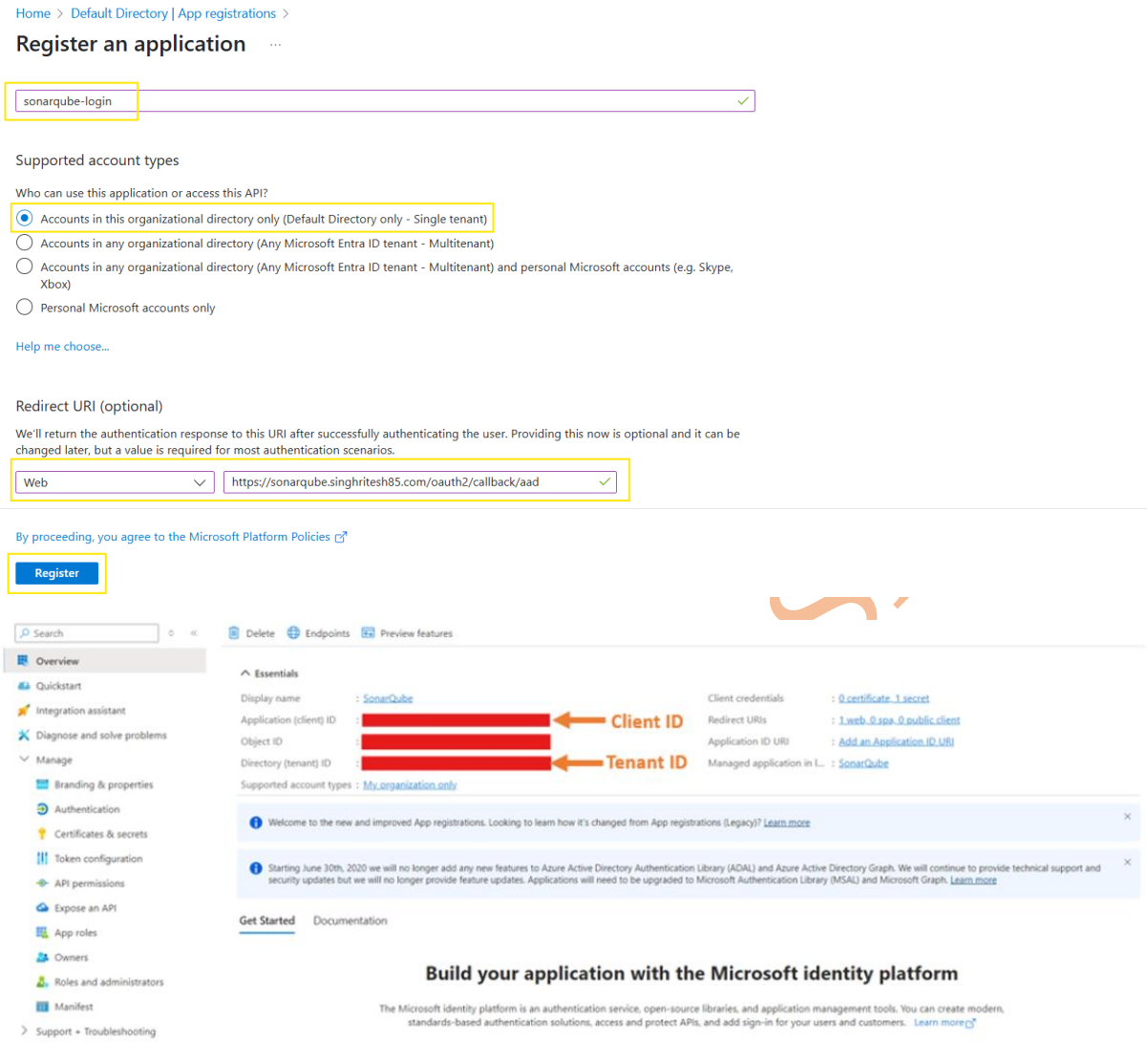
Then restarted the sonarqube server as shown in the screenshot attached below.



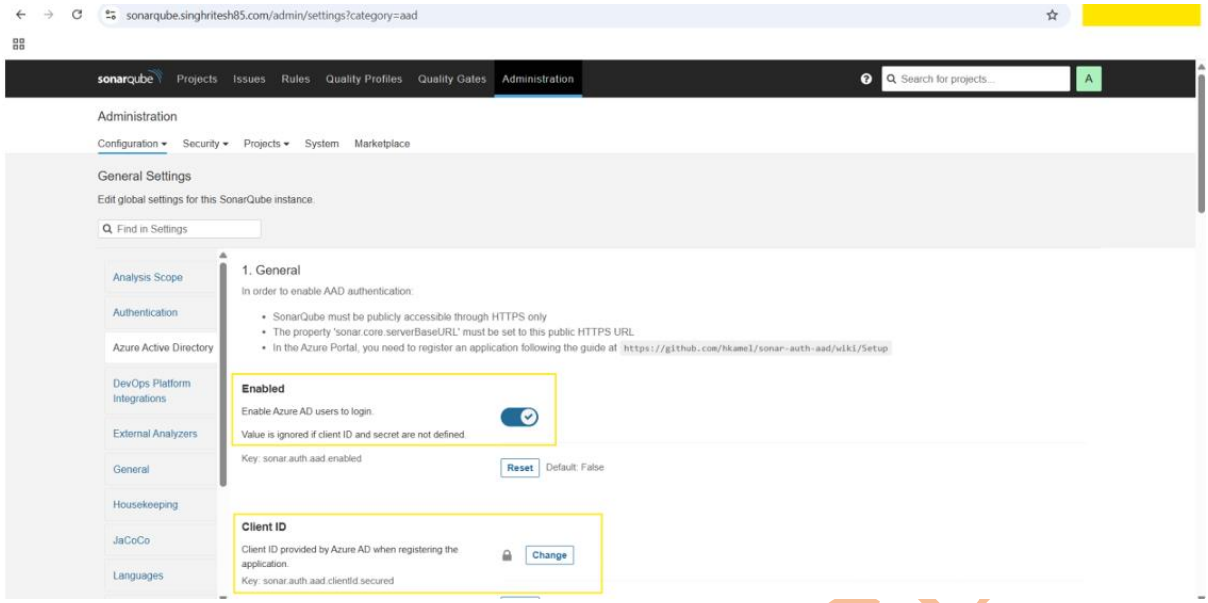
Go to SonarQube UI then to **Administration > Configuration > General Settings > General** and Add the Server Base URL as shown in the screenshot attached below.

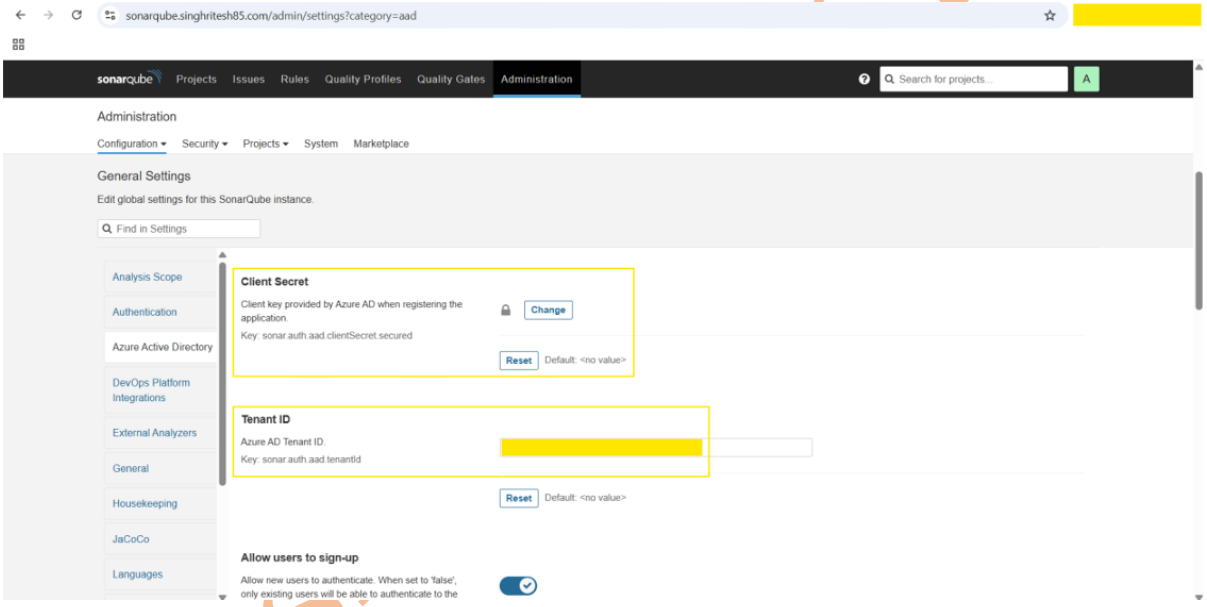


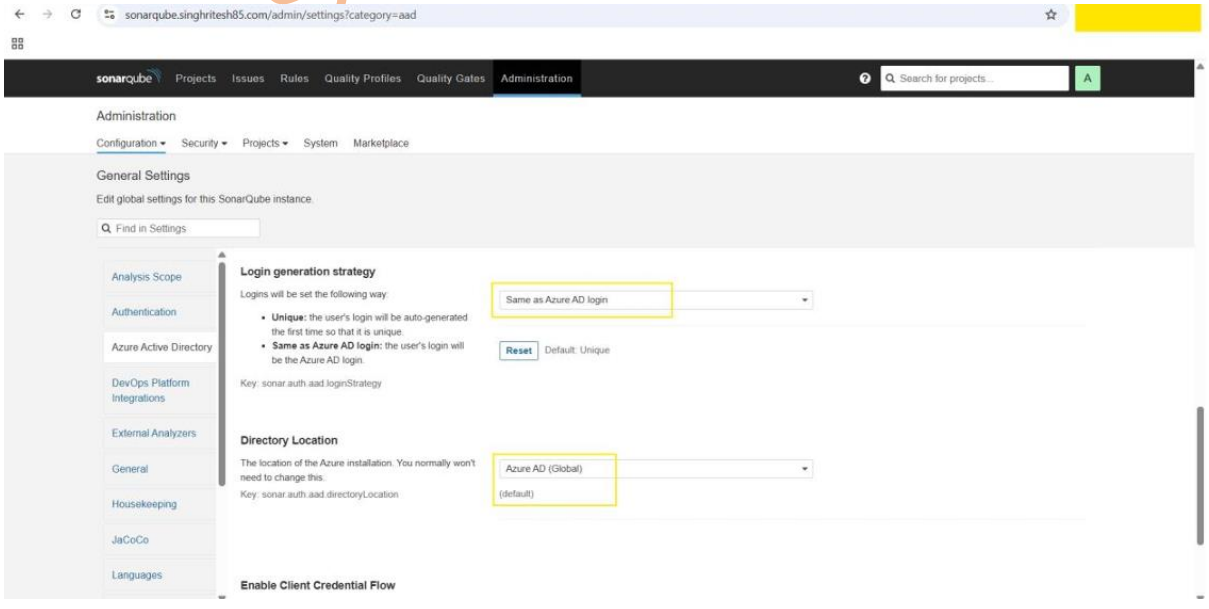
Create the App Registration in Azure Entra ID as shown in the screenshot attached below.



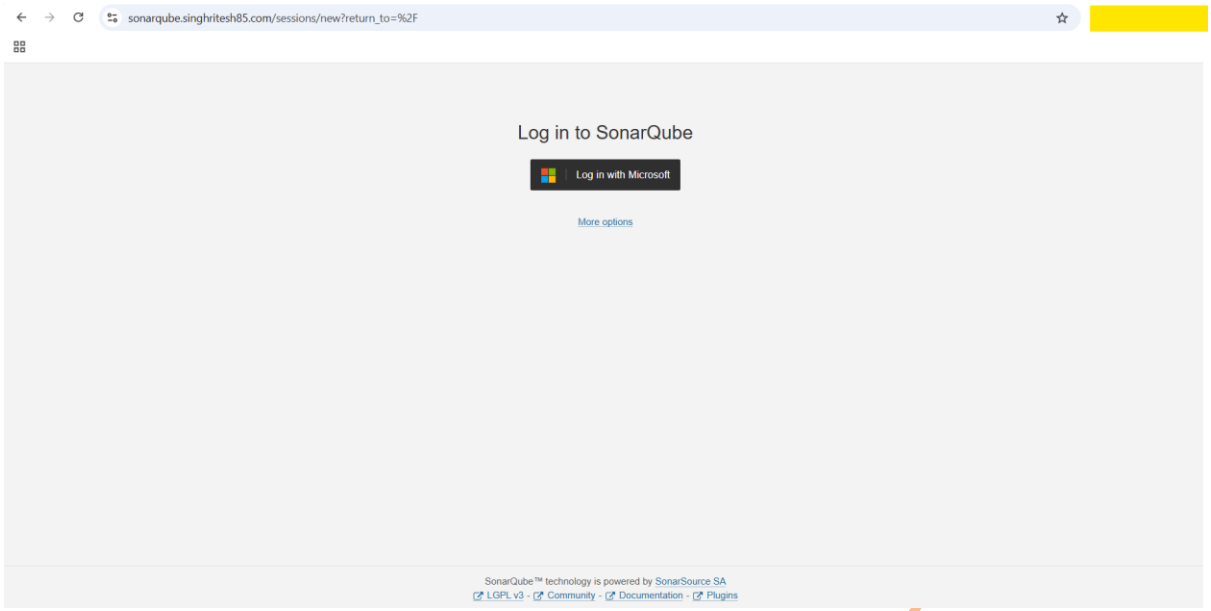
Now go to Sonarqube UI and then to **Administration > Configuration > General Settings > Azure Active Directory** as shown in the screenshot attached below and enable Azure AD users to login then provide the **client id**, **client secret** and **tenant id** as shown in the screenshot attached below.


sonarqube.singhritesh85.com/admin/settings?category=aad
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
Administration
Configuration
Security
Projects
System
Marketplace
General Settings
Edit global settings for this SonarQube instance.
Find in Settings
Analysis Scope
Authentication
Azure Active Directory
DevOps Platform Integrations
External Analyzers
General
Housekeeping
JaCoCo
Languages
1. General
In order to enable AAD authentication:
SonarQube must be publicly accessible through HTTPS only
The property 'sonar.core.serverBaseURL' must be set to this public HTTPS URL
In the Azure Portal, you need to register an application following the guide at https://github.com/hkamel/sonar-auth-aad/wiki/Setup
Enabled
Enable Azure AD users to login.
Value is ignored if client ID and secret are not defined.
Key: sonar.auth.aad.enabled
Reset
Default: False
Client ID
Client ID provided by Azure AD when registering the application.
Key: sonar.auth.aad.clientId.secured
Change

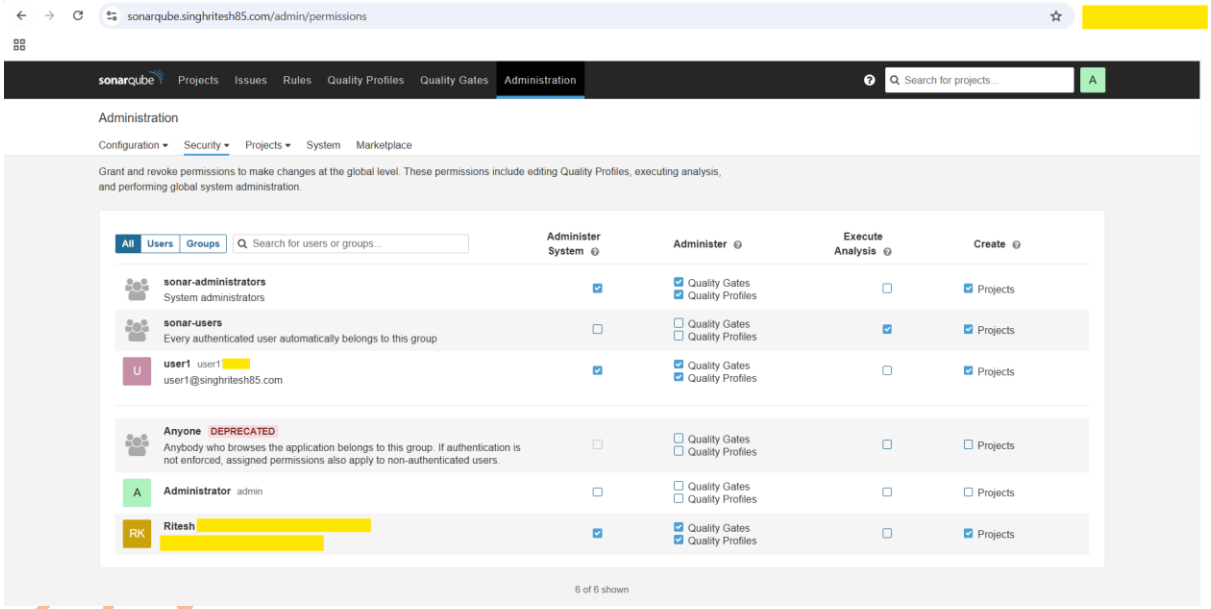

sonarqube.singhritesh85.com/admin/settings?category=aad
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
Administration
Configuration
Security
Projects
System
Marketplace
General Settings
Edit global settings for this SonarQube instance.
Find in Settings
Analysis Scope
Authentication
Azure Active Directory
DevOps Platform Integrations
External Analyzers
General
Housekeeping
JaCoCo
Languages
Client Secret
Client key provided by Azure AD when registering the application.
Key: sonar.auth.aad.clientSecret.secured
Change
Reset
Default: <no value>
Tenant ID
Azure AD Tenant ID.
Key: sonar.auth.aad.tenantId
Reset
Default: <no value>
Allow users to sign-up
Allow new users to authenticate. When set to 'false', only existing users will be able to authenticate to the


sonarqube.singhritesh85.com/admin/settings?category=aad
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
Administration
Configuration
Security
Projects
System
Marketplace
General Settings
Edit global settings for this SonarQube instance.
Find in Settings
Analysis Scope
Authentication
Azure Active Directory
DevOps Platform Integrations
External Analyzers
General
Housekeeping
JaCoCo
Languages
Login generation strategy
Logins will be set the following way:
Unique: the user's login will be auto-generated the first time so that it is unique.
Same as Azure AD login: the user's login will be the Azure AD login.
Same as Azure AD login
Reset
Default: Unique
Key: sonar.auth.aad.loginStrategy
Directory Location
The location of the Azure installation. You normally won't need to change this:
Key: sonar.auth.aad.directoryLocation
Azure AD (Global)
(default)
Enable Client Credential Flow

Then logout from SonarQube UI and login again as shown in the screenshot attached below.

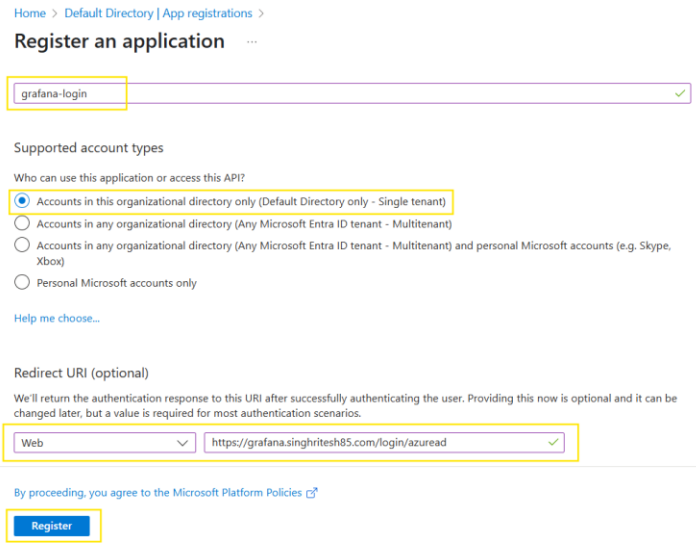


After the user will login once, you can change their permission as shown in screenshot attached below.

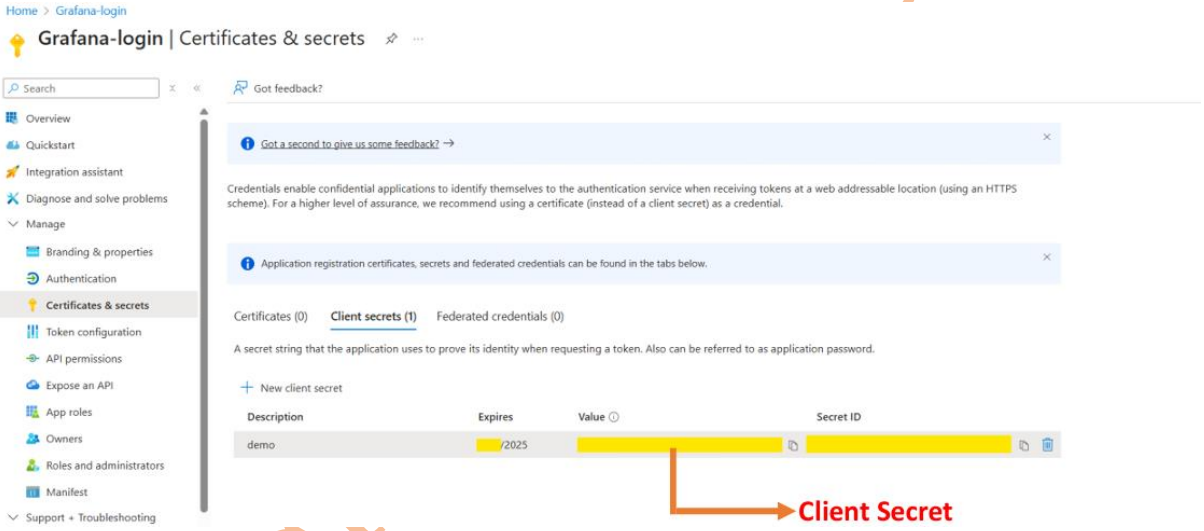


Integration of Azure Entra ID with Grafana

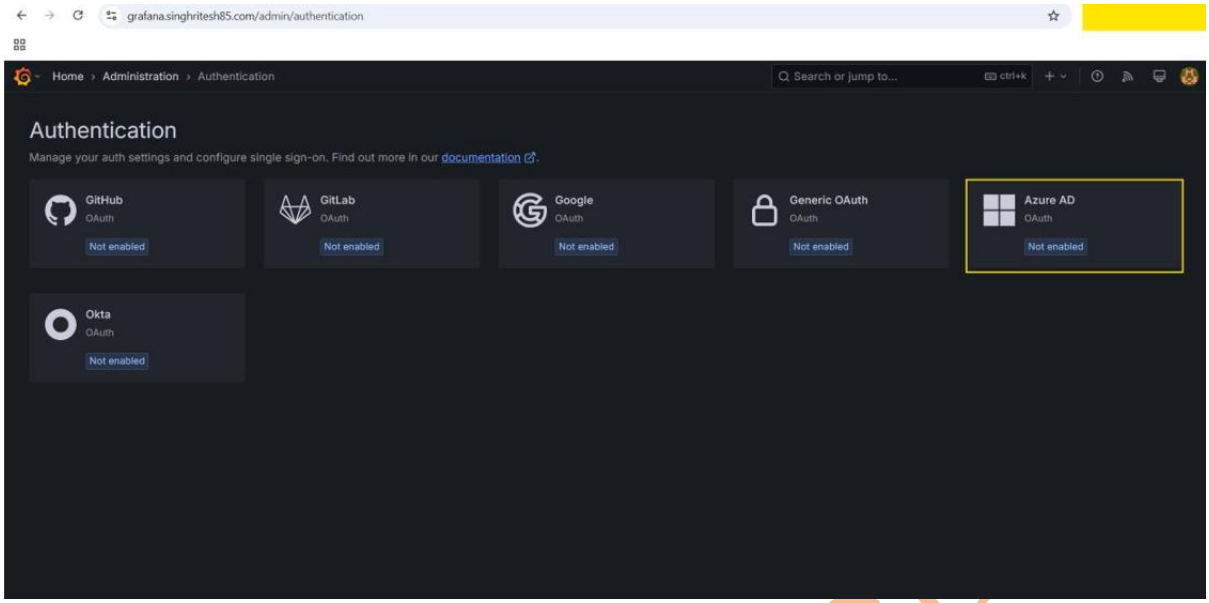
To integrate Azure Entra ID with Grafana I created an Azure Entra ID Service Principal (using APP Registration) as shown in the screenshot attached below. Go to **App Registration > New Registration** and configure a new Application as shown in the screenshot attached below.



Created the Secrets for Registered App Grafana-login in Azure Entra ID as shown in the screenshot attached below.

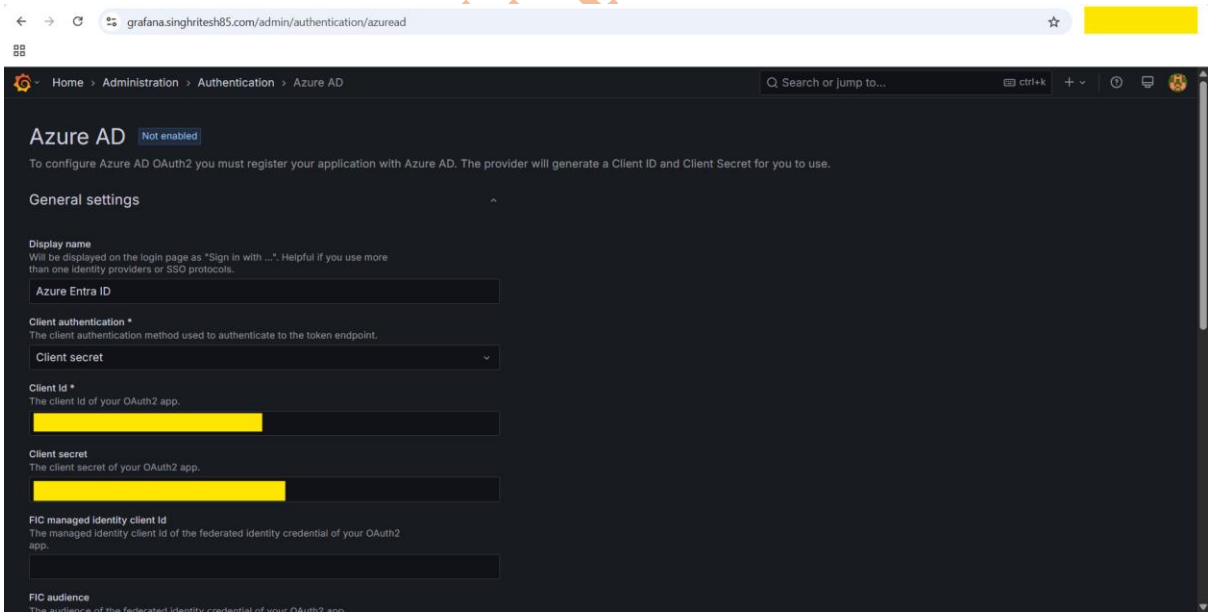


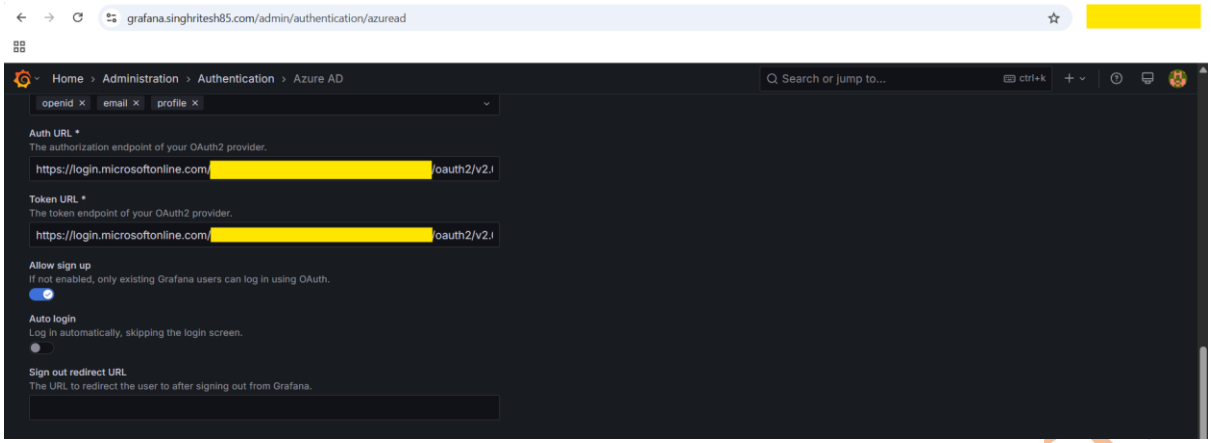
Now, login into the Grafana and update admin password then login into Grafana and Go to Grafana **Home > Administration > Authentication** and **enable the Azure AD OAuth** as shown in the screenshot attached below.

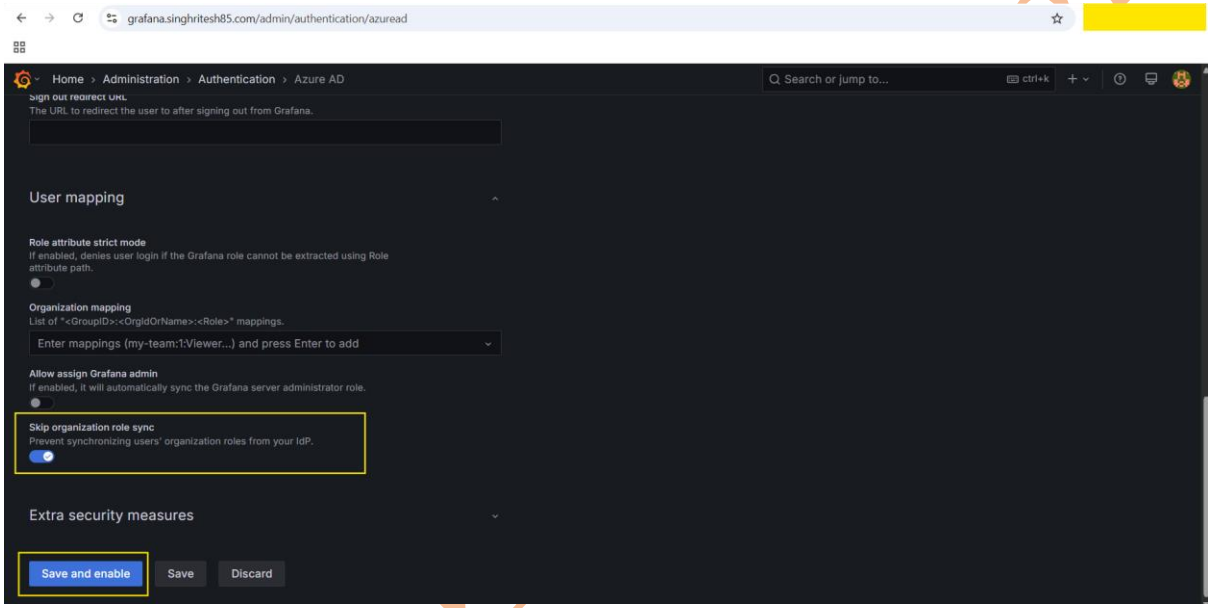


Provide the Client ID, Client Secret, Tenant ID and enable Allow Sign up and enable Skip organization role sync as shown in the screenshot attached below.

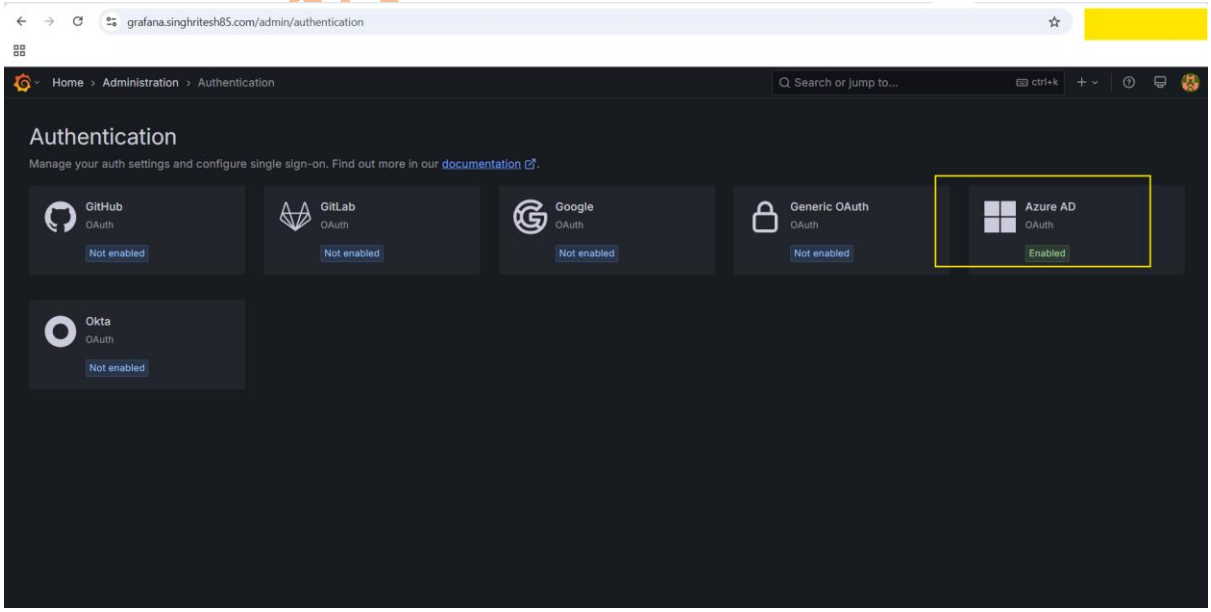
I enabled the **Skip organization role sync** otherwise Grafana will Sync the Azure Entra ID users with Main Org. Role as **Viewer** and Grafana Administrator cannot change it further but if I enabled **Skip organization role sync** then Grafana Administrator can change the viewer Role and can assign another Role as Editor or Admin. In this demonstration I created two users in Azure Entra ID user1 and user2. User1, I had provided as Administrator Role and user2 as default viewer access in Main Organisation (**Main Organisation always have Organisation ID 1**).







Now the Azure AD OAuth is enable as shown in the screenshot attached below.



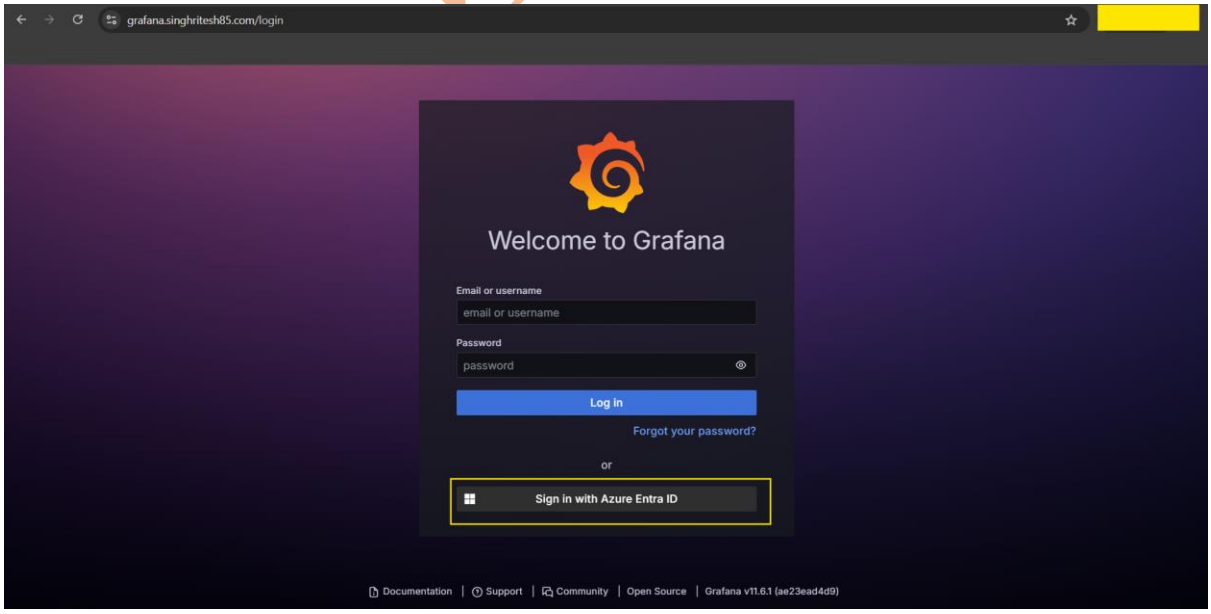
Now go to the Grafana Server and open the file **/etc/grafana/grafana.ini** and edit **root_url** under the [server] then restart the **grafana-server** service as shown in the screenshot attached below.

```
[root@                    ~]# vim /etc/grafana/grafana.ini
```

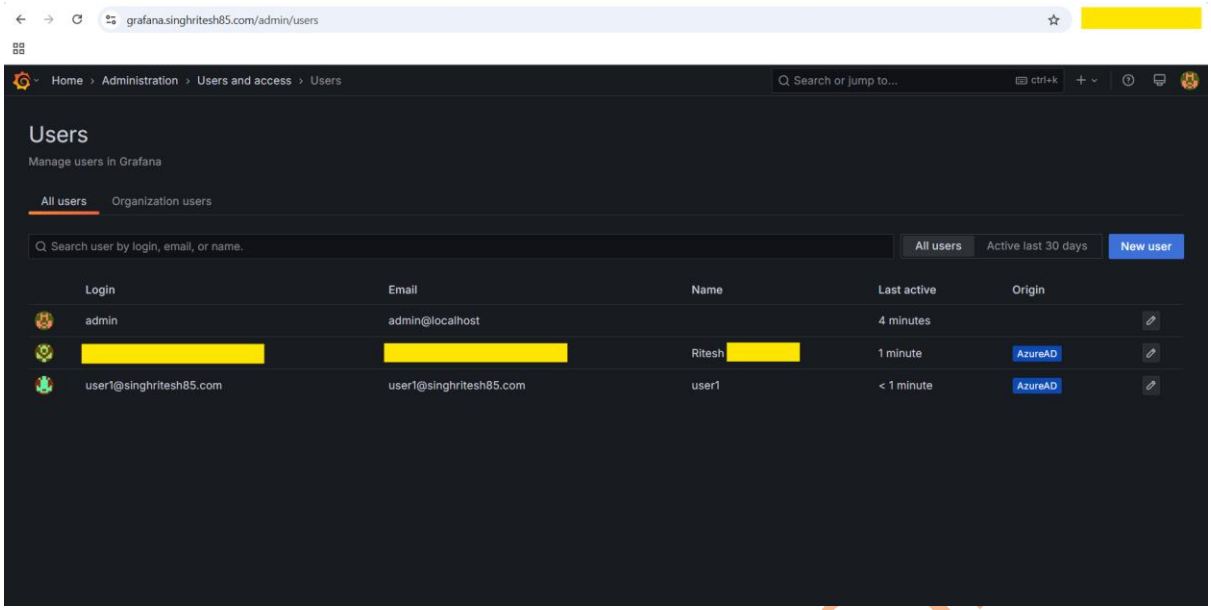
```
#################################### Server ####################################
[server]
# Protocol (http, https, h2, socket)
;protocol = http

# Minimum TLS version allowed. By default, this value is empty. Accepted values are: TLS1.2, TLS1.3. If nothing is set TLS1.2 would be taken
;min_tls_version = ""

# The ip address to bind to, empty will bind to all interfaces
;http_addr =

# The http port to use
;http_port = 3000

# The public facing domain name used to access grafana from a browser
;domain = localhost

# Redirect to correct domain if host header does not match domain
# Prevents DNS rebinding attacks
;enforce_domain = false

# The full public facing url you use in browser, used for redirects and emails
# If you use reverse proxy and sub path specify full url (with sub path)
root_url = https://grafana.singhritesh85.com      #%(protocol)s://%(domain)s:%(http_port)s/

# Serve Grafana from subpath specified in `root_url` setting. By default it is set to `false` for compatibility reasons.
;serve_from_sub_path = false

# Log web requests
;router_logging = false
```

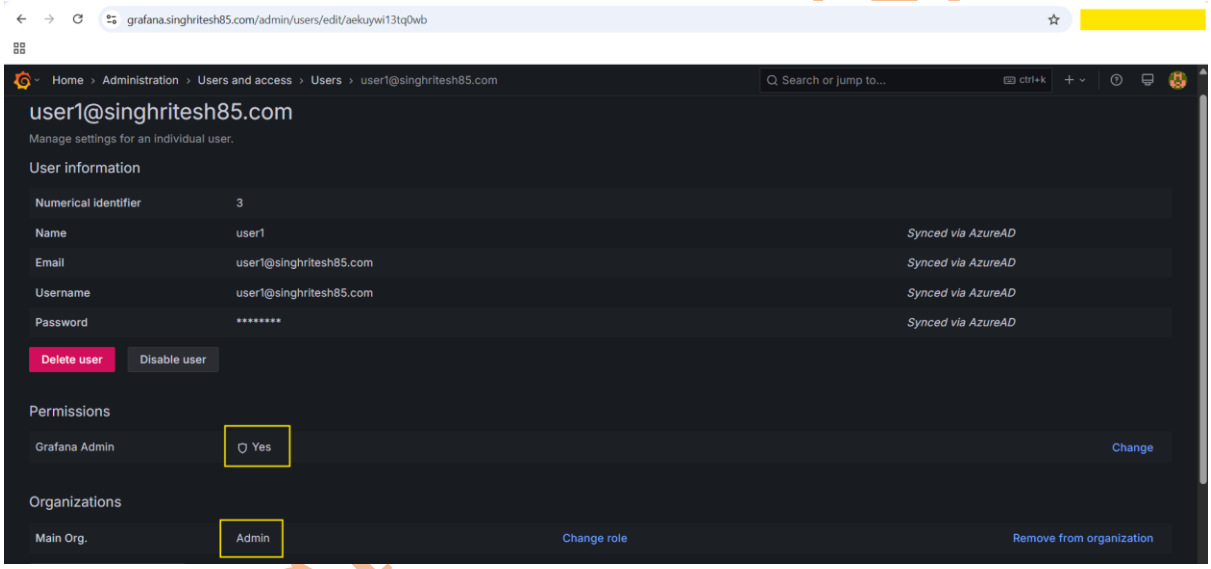
```
[root@              ~]# systemctl restart grafana-server.service
[root@              ~]# systemctl status grafana-server.service
● grafana-server.service - Grafana instance
   Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; vendor preset: disabled
   Active: active (running) since Sun 2025-              UTC; 7s ago
     Docs: http://docs.grafana.org
 Main PID: 4864 (grafana)
```

Now, Grafana login dashboard will show the option of sign-in with Azure Entra ID as shown in the screenshot attached below. I logged-in to the Grafana dashboard with the Azure Entra ID user as shown in the screenshot attached below.



For the first time when a user logged-in, they had viewer access which Grafana Administrator can change as per the requirement as shown in the screenshot attached below.



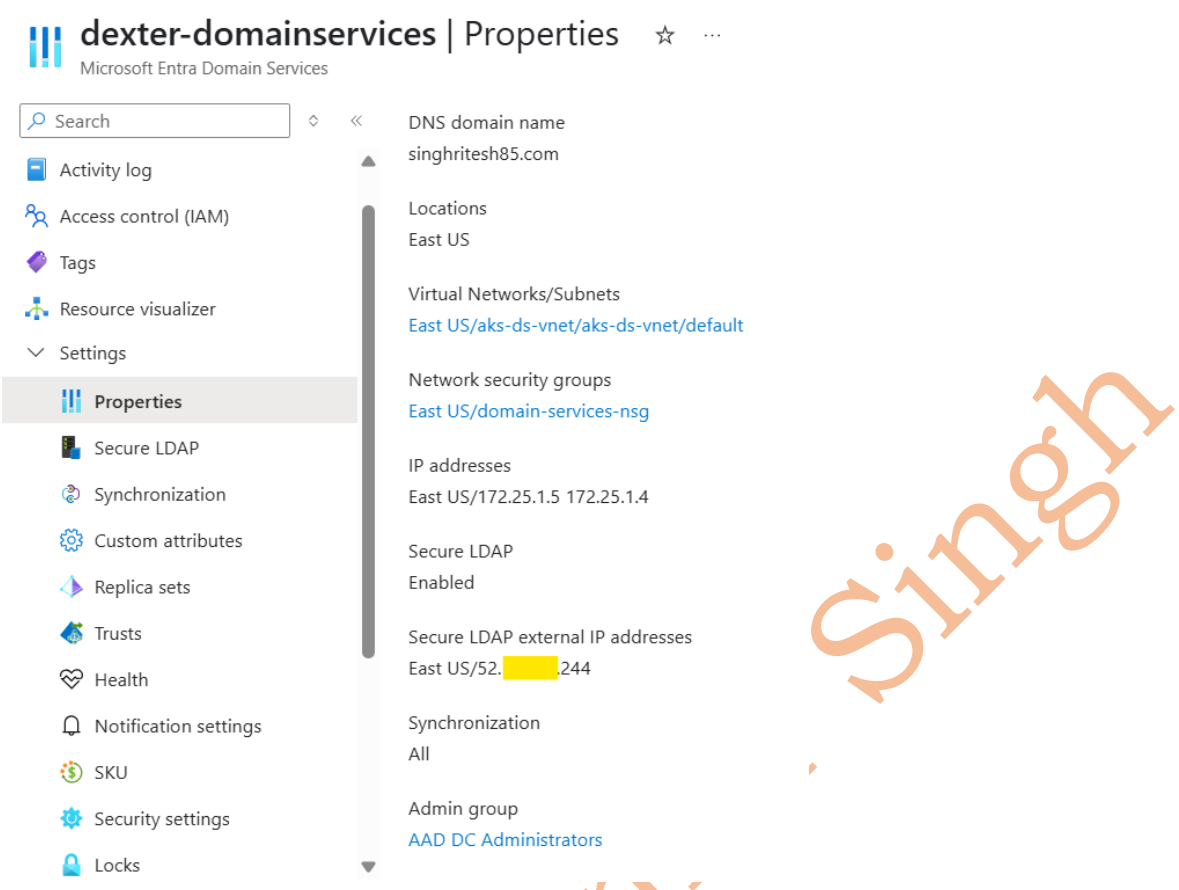


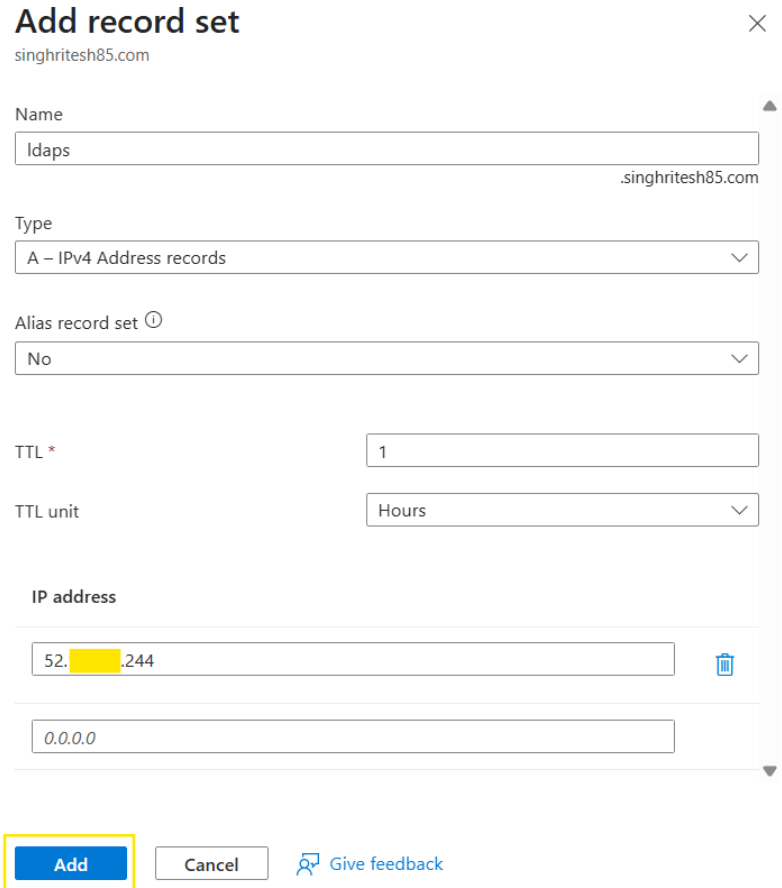
After that user1 will refresh Grafana UI and their Access will be reflected on the UI as well.

Integration of Azure Entra ID with Sonatype Nexus3 Repository

I had integrated Azure Entra ID with Sonatype Nexus3 Repository using the Secure LDAP authentication. To achieve this, I had to use the Microsoft Entra Domain Services (formerly known as Azure Active Directory Domain Services) which I had created earlier using the terraform script.

First this you should do here is go to **Azure Microsoft Entra Domain Services > Properties** and note down the LDAP external IP address and do the entry in Azure DNS Zone to create the Record Set of A Type as shown in the screenshot attached below.


dexter-domainservices | Properties
Microsoft Entra Domain Services
Search
Activity log
Access control (IAM)
Tags
Resource visualizer
Settings
Properties
Secure LDAP
Synchronization
Custom attributes
Replica sets
Trusts
Health
Notification settings
SKU
Security settings
Locks
DNS domain name
singhritesh85.com
Locations
East US
Virtual Networks/Subnets
East US/aks-ds-vnet/aks-ds-vnet/default
Network security groups
East US/domain-services-nsg
IP addresses
East US/172.25.1.5 172.25.1.4
Secure LDAP
Enabled
Secure LDAP external IP addresses
East US/52. .244
Synchronization
All
Admin group
AAD DC Administrators

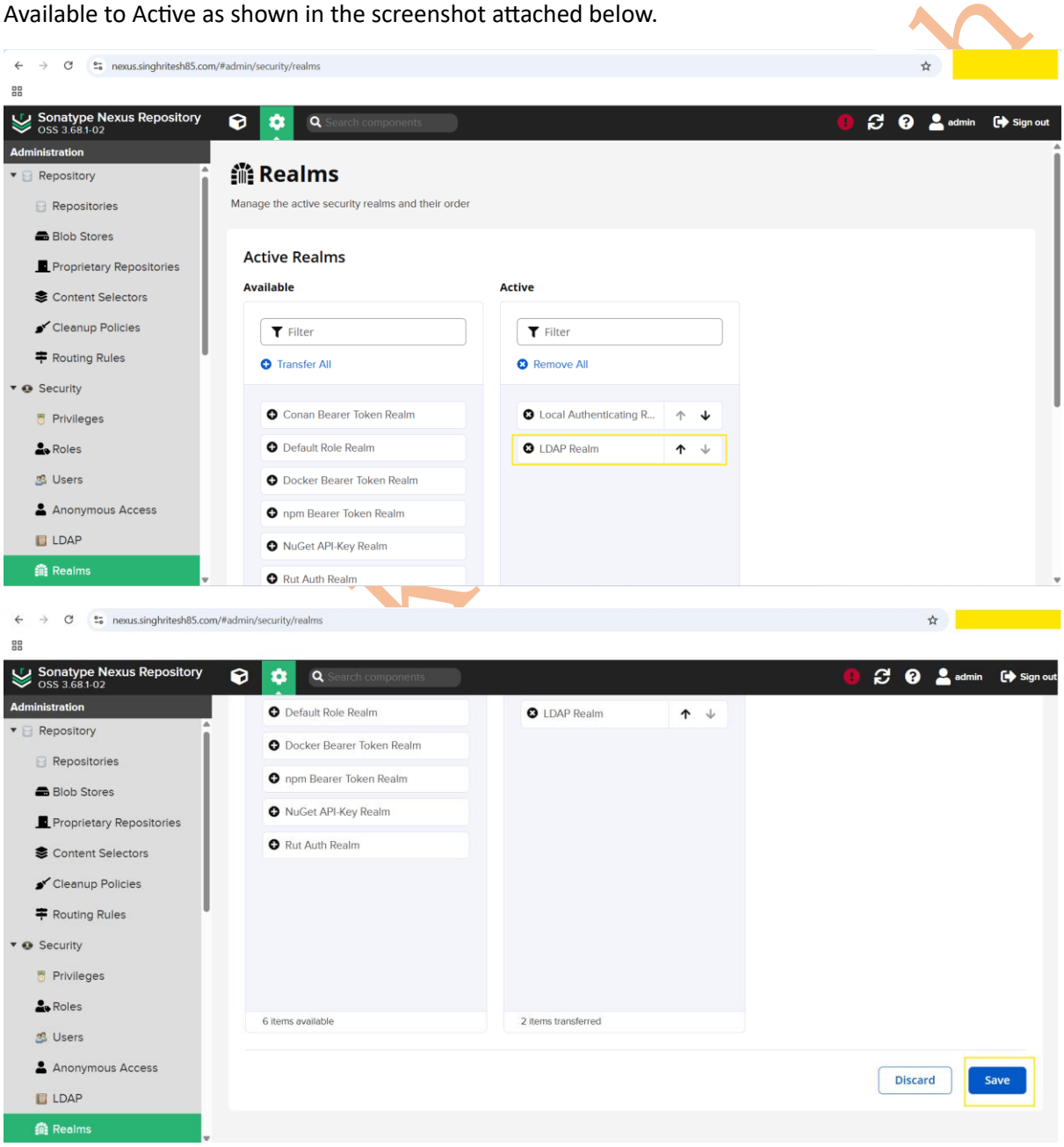

Add record set
singhritesh85.com
Name
ldaps
.singhritesh85.com
Type
A – IPv4 Address records
Alias record set
No
TTL *
1
TTL unit
Hours
IP address
52. .244
0.0.0.0

Add
Cancel
Give feedback

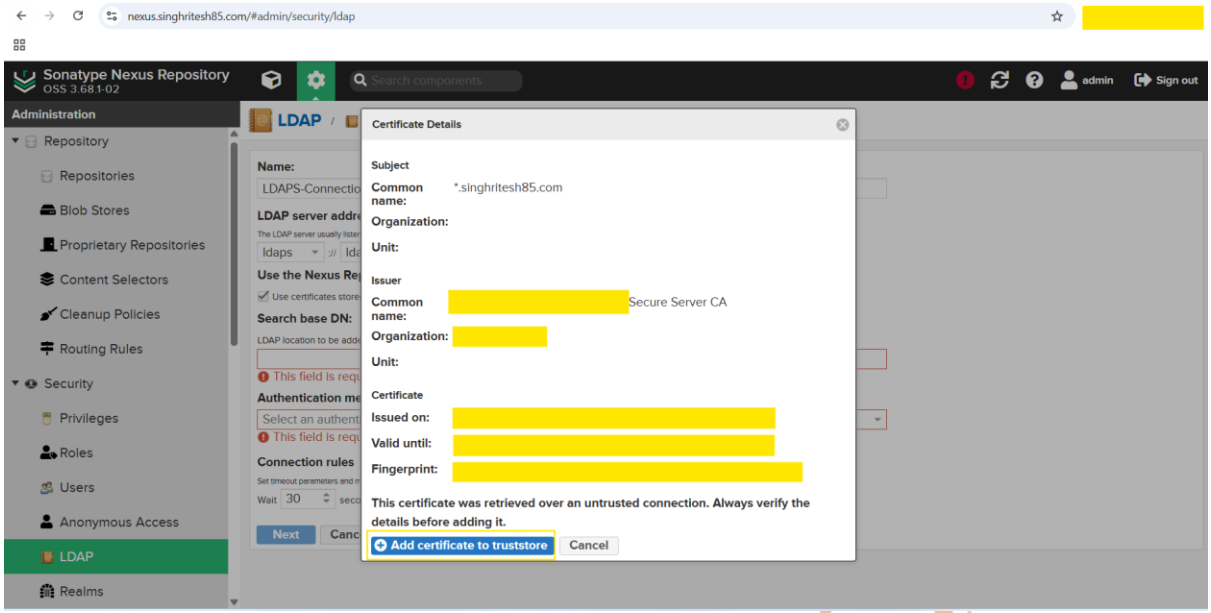
I had updated the **/etc/resolv.conf** file present on **Nexus-Server** and used the Google's Public DNS Server as shown in the screenshot attached below.

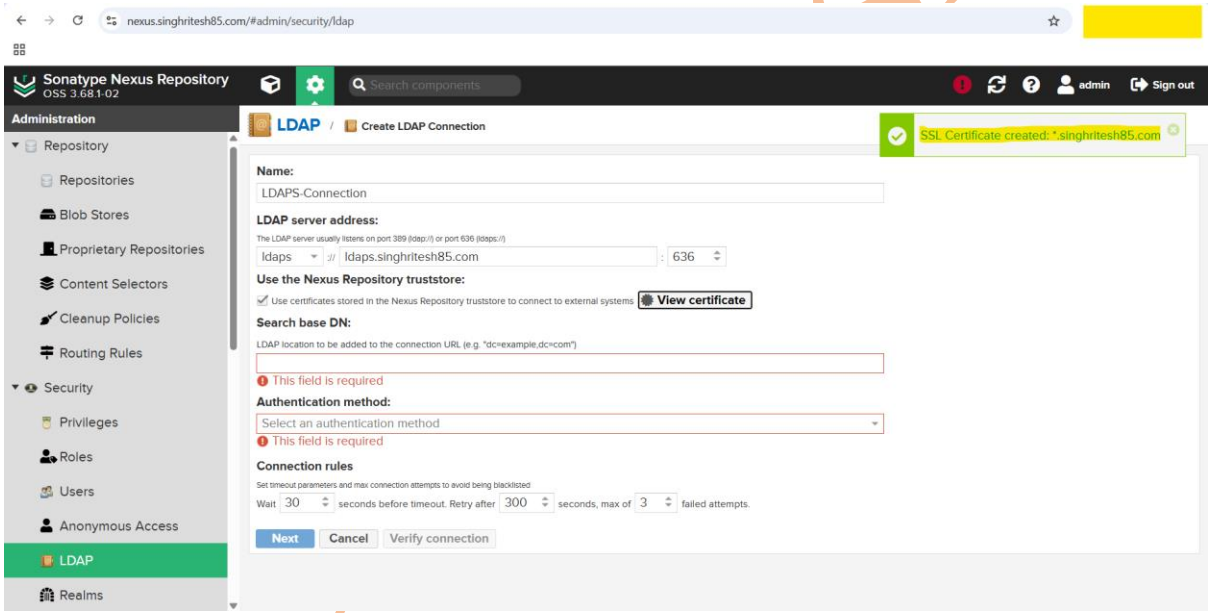
```
[root@                    ~]# cat /etc/resolv.conf
; generated by /usr/sbin/dhclient-script
search us-east-2.compute.internal
options timeout:2 attempts:5
nameserver 8.8.8.8     #10.10.0.2
```

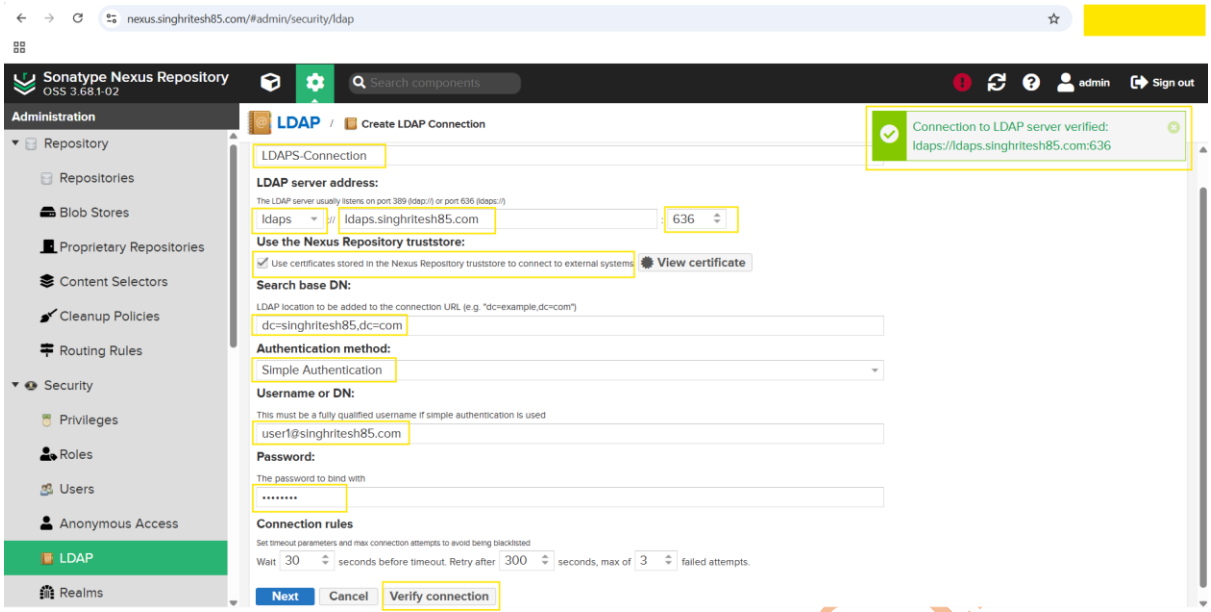
Login into the Nexus and go to **Administration > Security > Realms** and make the LDAP Realm from Available to Active as shown in the screenshot attached below.

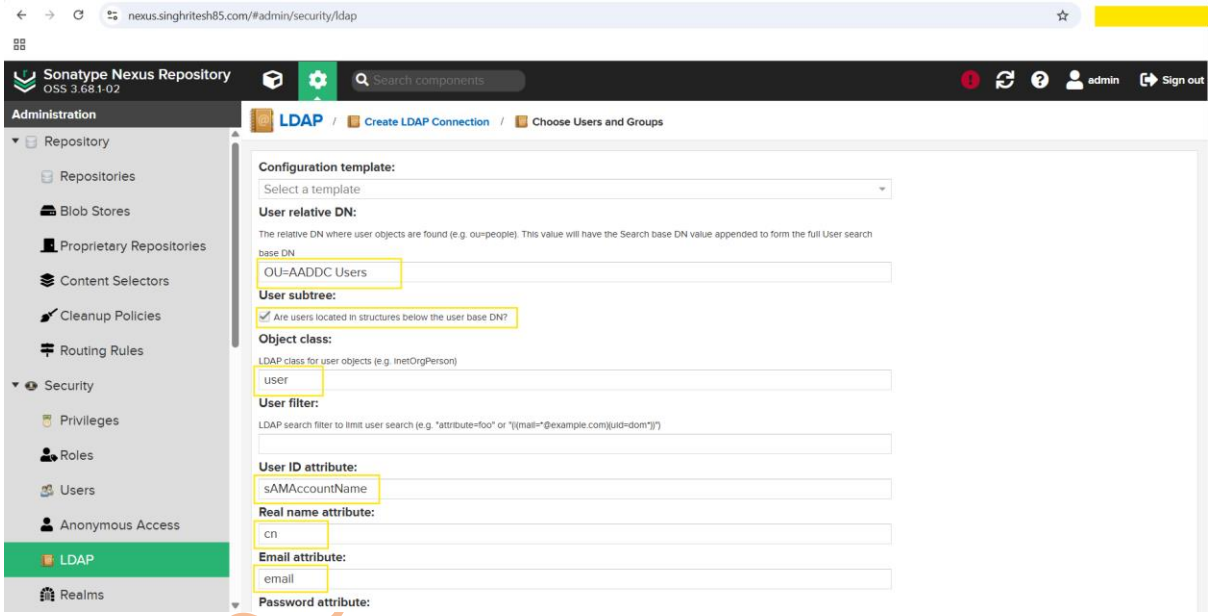


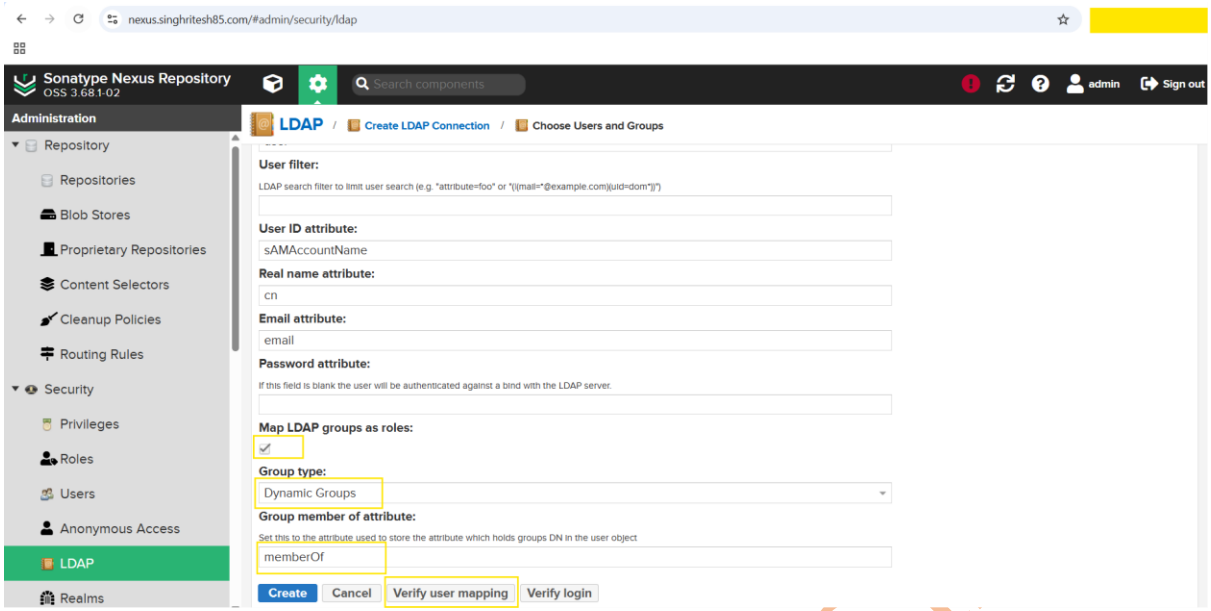
Now go to the **Administration > Security > LDAP** and then **Create Connection**. While creating the connection add the certificate to the truststore as shown in the screenshot attached below.

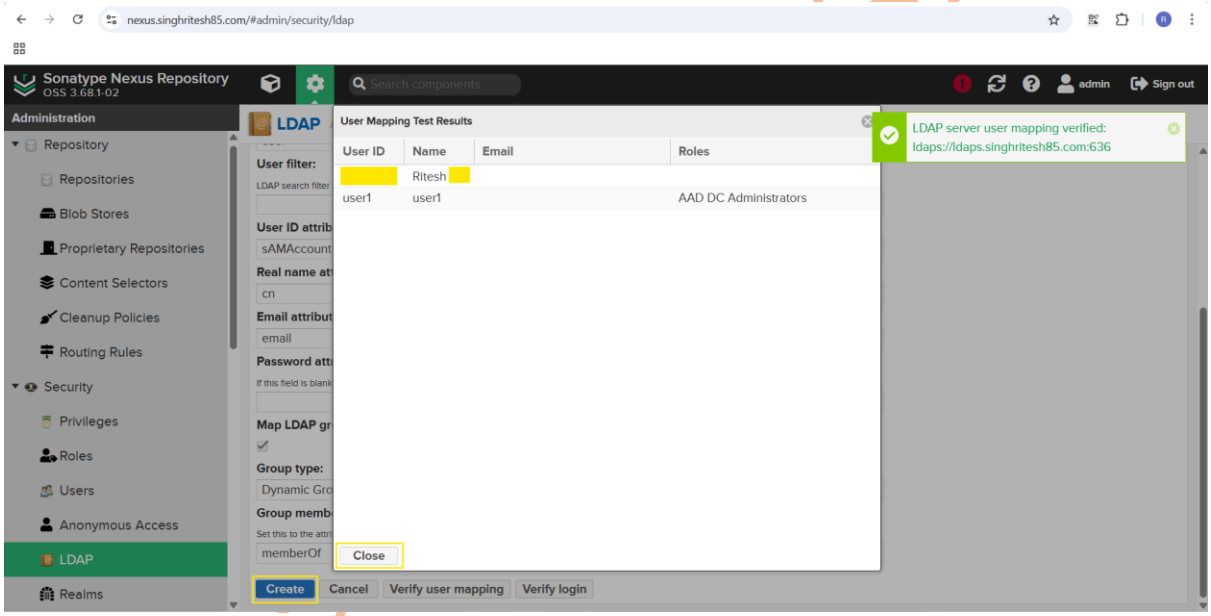

nexus.singhritesh85.com/#admin/security/ldap
Sonatype Nexus Repository
OSS 3.68.1-02
admin
Sign out
Administration
Repository
Repositories
Blob Stores
Proprietary Repositories
Content Selectors
Cleanup Policies
Routing Rules
Security
Privileges
Roles
Users
Anonymous Access
LDAP
Realms
LDAP
Certificate Details
Subject
Common name: *.singhritesh85.com
Organization:
Unit:
Issuer
Common name: Secure Server CA
Organization:
Unit:
Certificate
Issued on:
Valid until:
Fingerprint:
This certificate was retrieved over an untrusted connection. Always verify the details before adding it.
Add certificate to truststore
Cancel


nexus.singhritesh85.com/#admin/security/ldap
Sonatype Nexus Repository
OSS 3.68.1-02
admin
Sign out
Administration
LDAP
Create LDAP Connection
SSL Certificate created: *.singhritesh85.com
Name:
LDAPS-Connection
LDAP server address:
ldaps
ldaps.singhritesh85.com
636
Use the Nexus Repository truststore:
Use certificates stored in the Nexus Repository truststore to connect to external systems
View certificate
Search base DN:
This field is required
Authentication method:
Select an authentication method
This field is required
Connection rules
Wait 30 seconds before timeout. Retry after 300 seconds, max of 3 failed attempts.
Next
Cancel
Verify connection

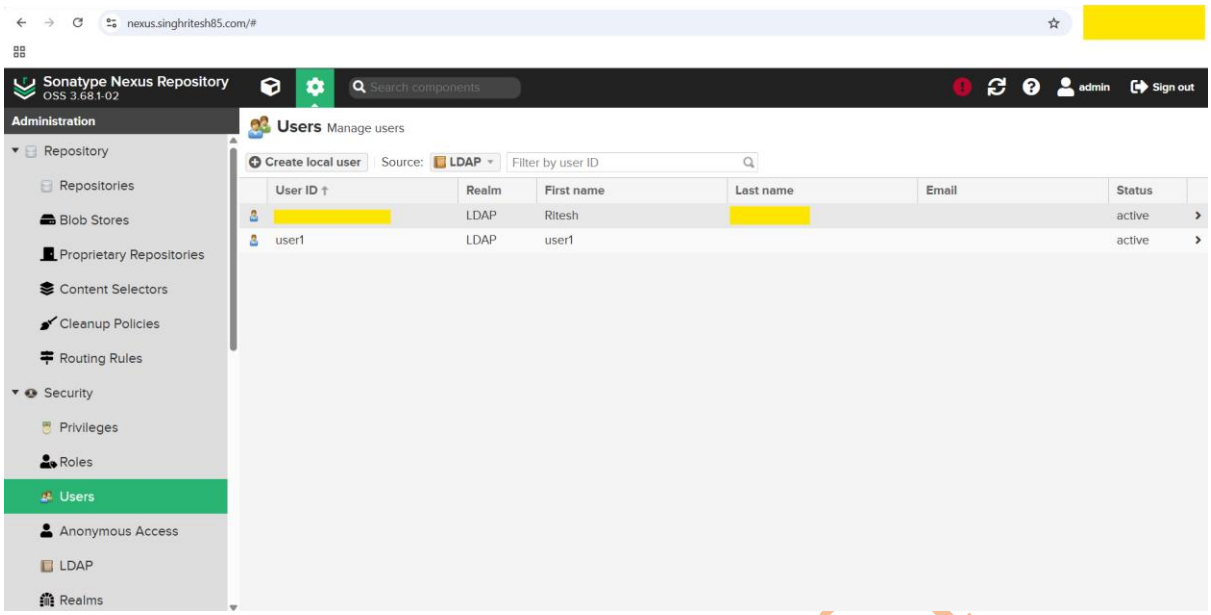

nexus.singhritesh85.com/#admin/security/ldap
Sonatype Nexus Repository
OSS 3.68.1-02
Search components
admin
Sign out
Administration
Repository
Repositories
Blob Stores
Proprietary Repositories
Content Selectors
Cleanup Policies
Routing Rules
Security
Privileges
Roles
Users
Anonymous Access
LDAP
Realms
LDAP / Create LDAP Connection
Connection to LDAP server verified:
ldaps://ldaps.singhritesh85.com:636
LDAPS-Connection
LDAP server address:
The LDAP server usually listens on port 389 (ldap://) or port 636 (ldaps://)
ldaps
ldaps.singhritesh85.com
636
Use the Nexus Repository truststore:
Use certificates stored in the Nexus Repository truststore to connect to external systems
View certificate
Search base DN:
LDAP location to be added to the connection URL (e.g. "dc=example,dc=com")
dc=singhritesh85,dc=com
Authentication method:
Simple Authentication
Username or DN:
This must be a fully qualified username if simple authentication is used
user1@singhritesh85.com
Password:
The password to bind with
Connection rules
Set timeout parameters and max connection attempts to avoid being blacklisted
Wait 30 seconds before timeout. Retry after 300 seconds, max of 3 failed attempts.
Next
Cancel
Verify connection


nexus.singhritesh85.com/#admin/security/ldap
Sonatype Nexus Repository
OSS 3.68.1-02
Search components
admin
Sign out
Administration
Repository
Repositories
Blob Stores
Proprietary Repositories
Content Selectors
Cleanup Policies
Routing Rules
Security
Privileges
Roles
Users
Anonymous Access
LDAP
Realms
LDAP / Create LDAP Connection / Choose Users and Groups
Configuration template:
Select a template
User relative DN:
The relative DN where user objects are found (e.g. ou=people). This value will have the Search base DN value appended to form the full User search base DN
OU=AADDC Users
User subtree:
Are users located in structures below the user base DN?
Object class:
LDAP class for user objects (e.g. inetOrgPerson)
user
User filter:
LDAP search filter to limit user search (e.g. "attribute=foo" or "(|(mail=*@example.com)(uid=dom*))")
User ID attribute:
sAMAccountName
Real name attribute:
cn
Email attribute:
email
Password attribute:

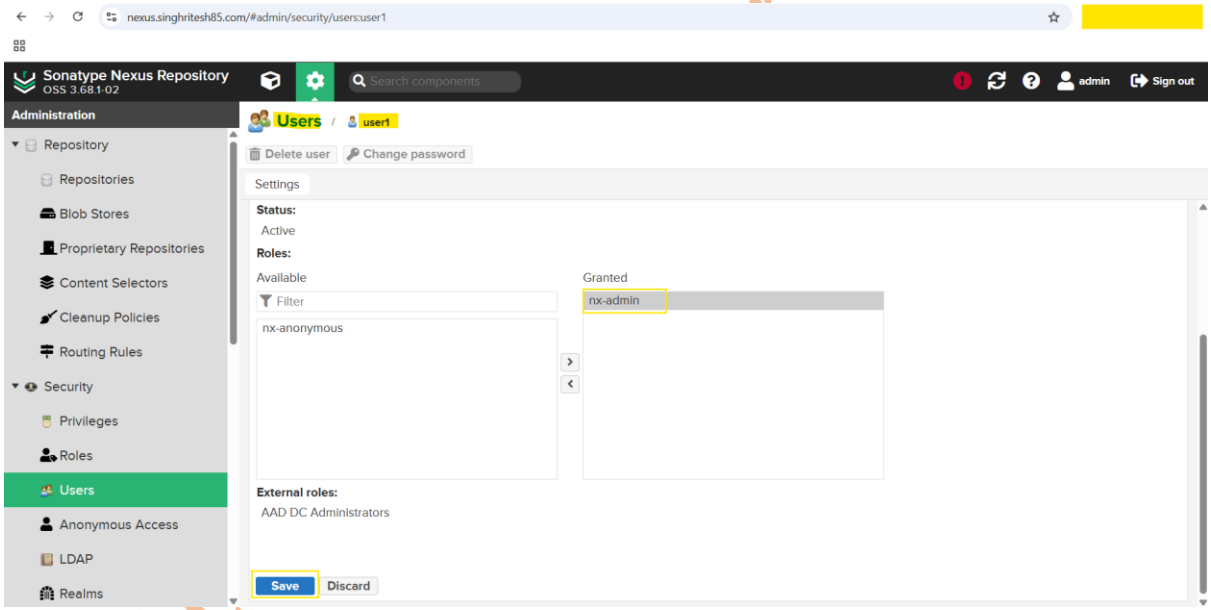




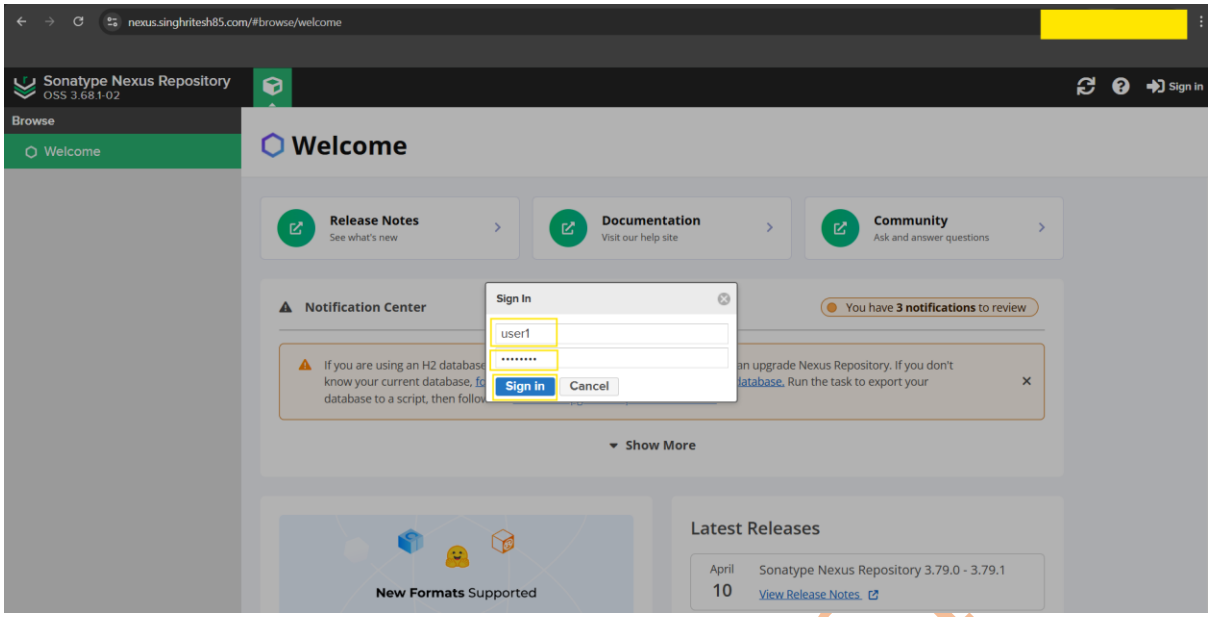
Now go to Administration > Users and Source: LDAP then you will see your Azure Entra ID Users will be listed here as shown in the screenshot attached below.

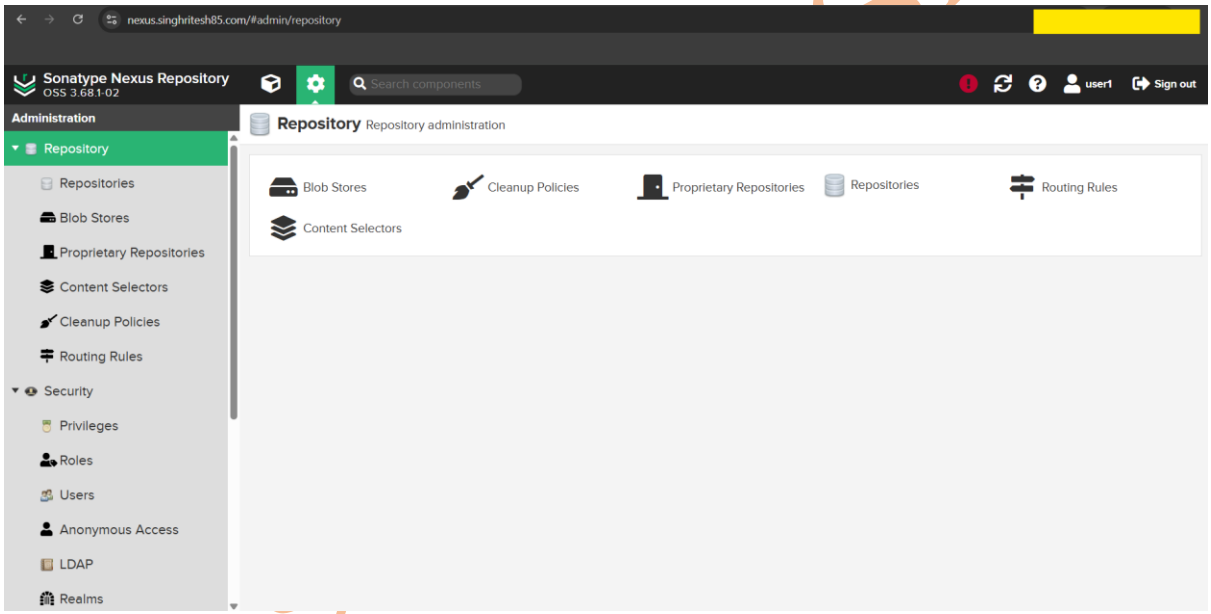


Then I logged-in as user1 and the Nexus Administrator will provide the sufficient privileges to user1. In this project I provide user1 as Administrator privilege as shown in the screenshot attached below.

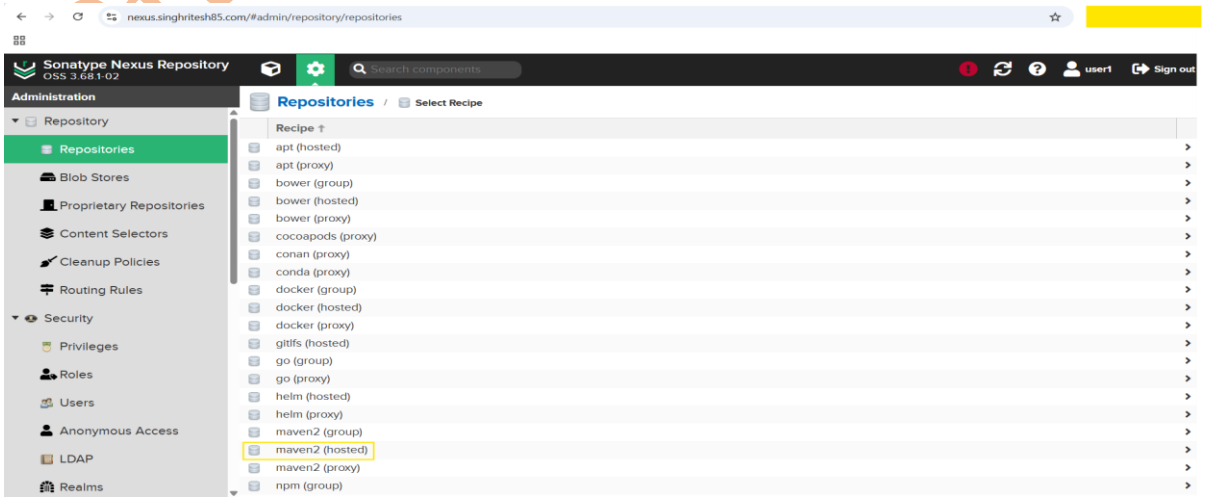


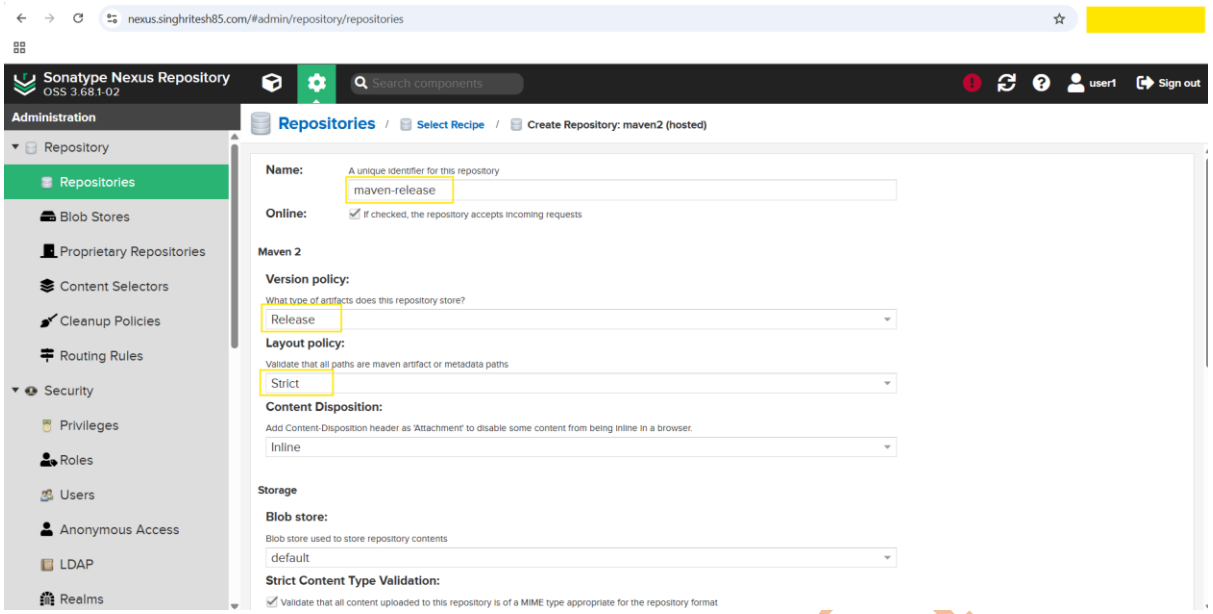
Finally, I logged-in as user1 as shown in the screenshot attached below and found that user got the access as provided above.

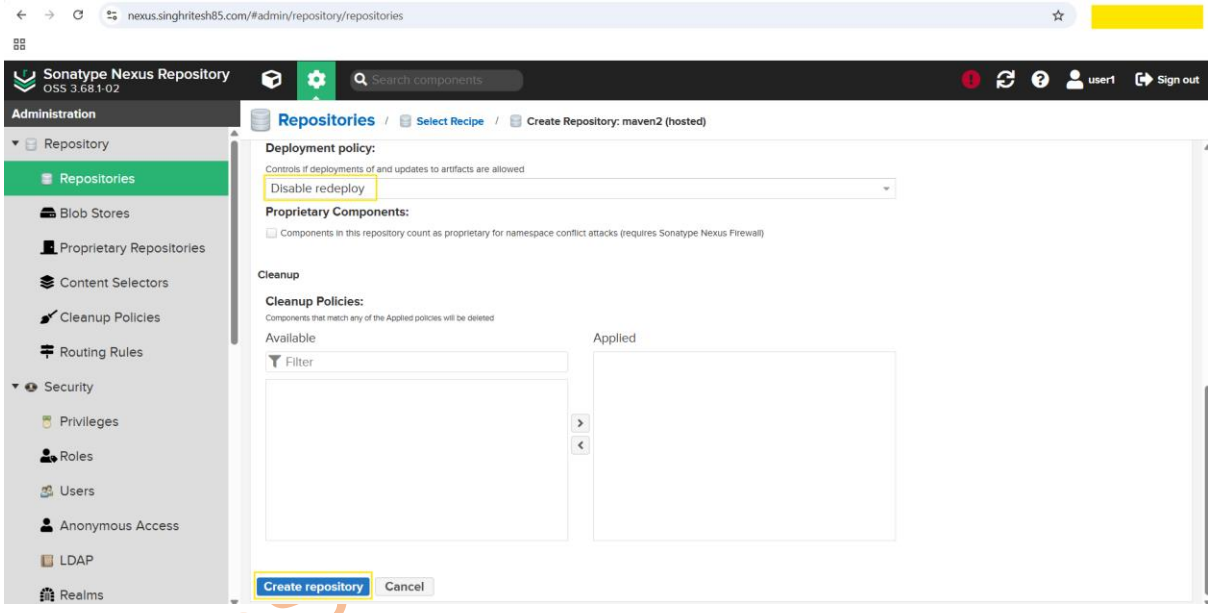


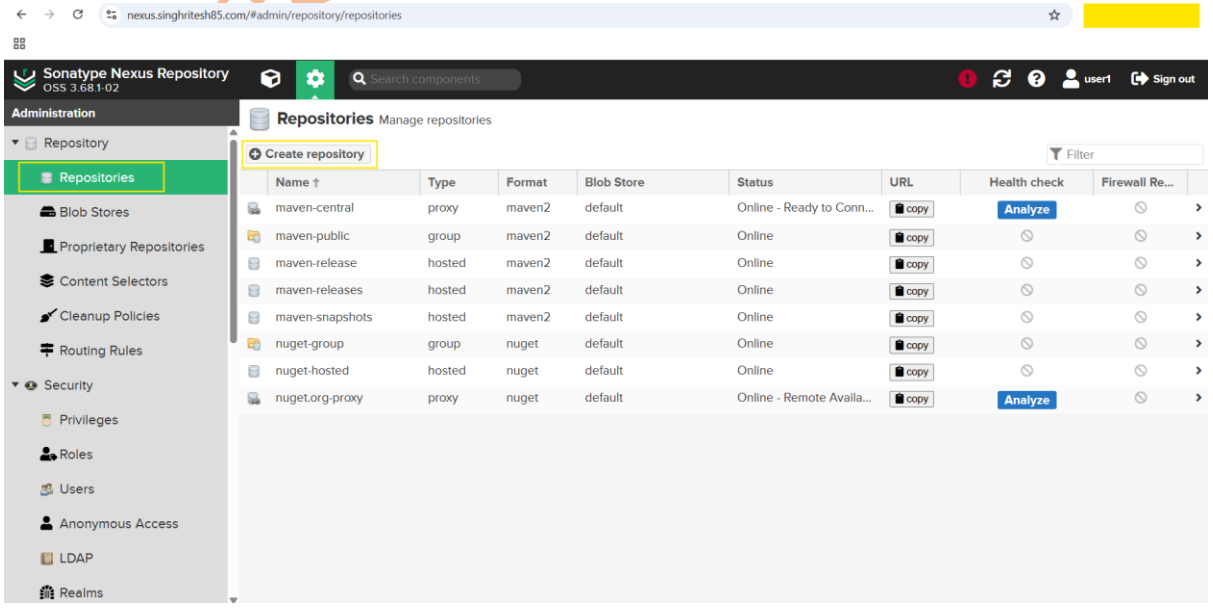


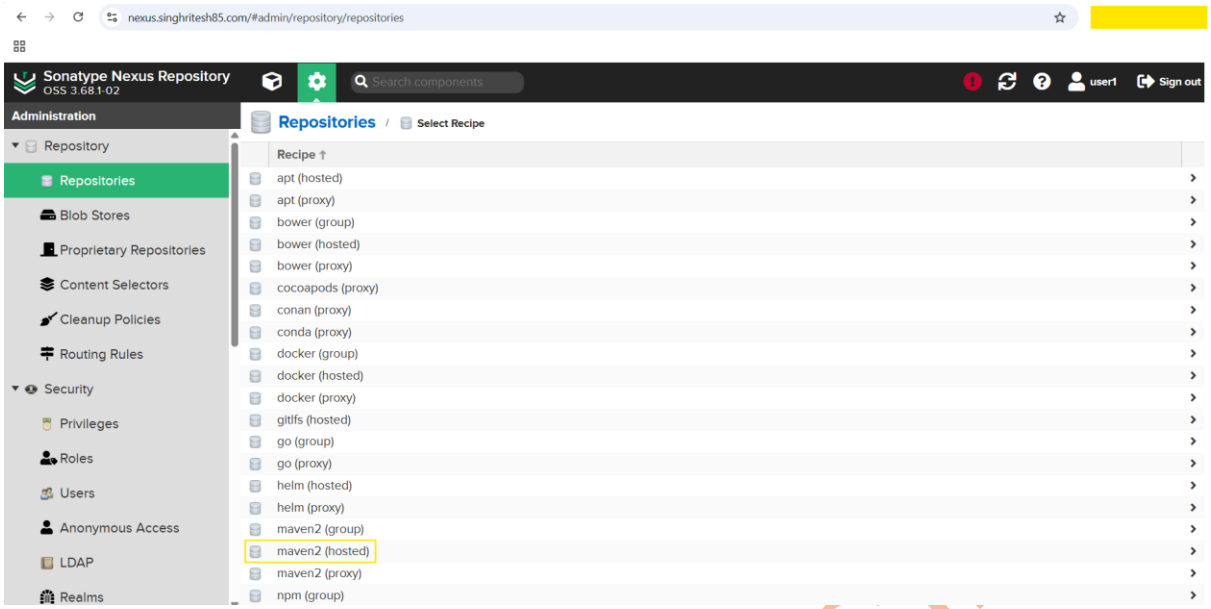
For this project in Nexus, I created two repositories named as **maven-release** and **maven-snapshot** as shown in the screenshot attached below.

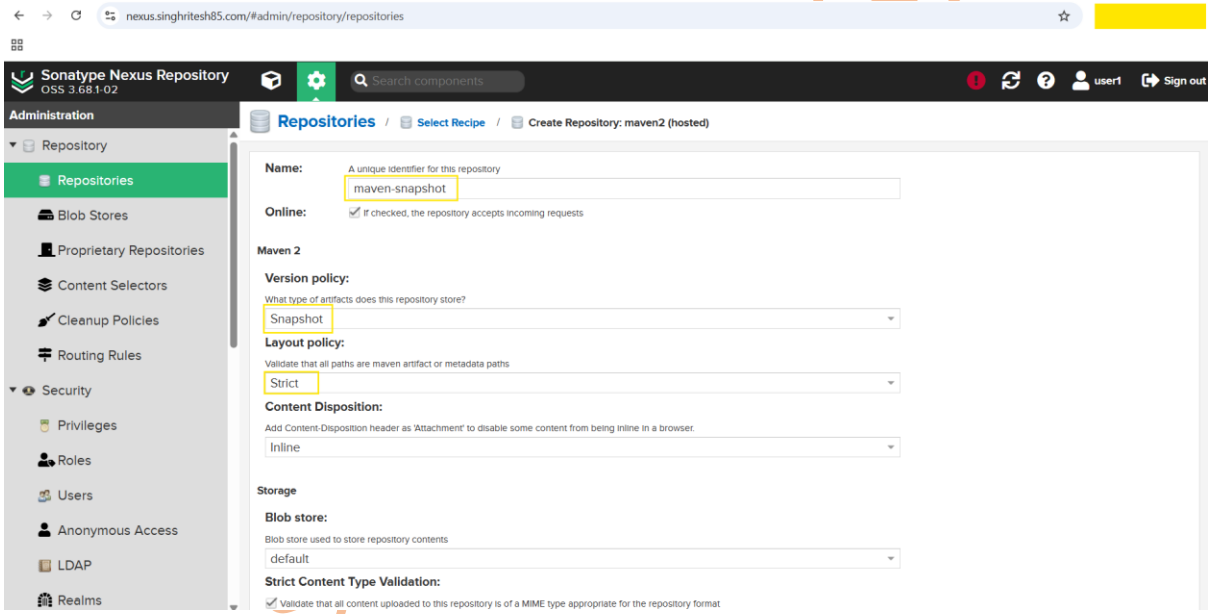


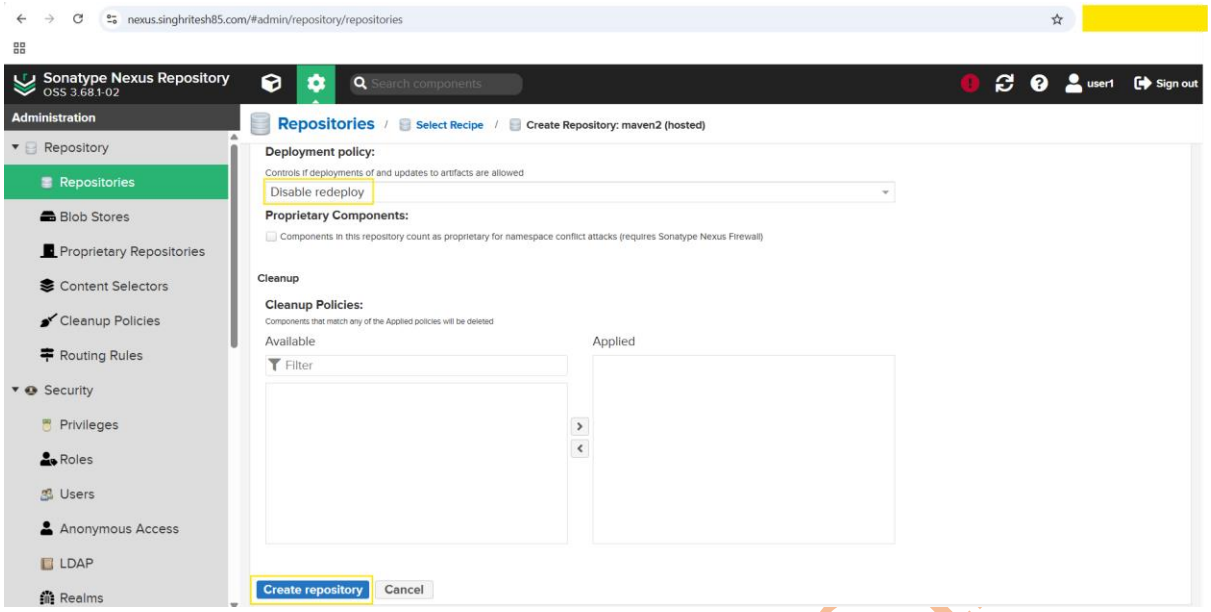

nexus.singhritesh85.com/#admin/repository/repositories
Sonatype Nexus Repository
OSS 3.68.1-02
Search components
user1
Sign out
Administration
Repository
Repositories
Blob Stores
Proprietary Repositories
Content Selectors
Cleanup Policies
Routing Rules
Security
Privileges
Roles
Users
Anonymous Access
LDAP
Realms
Repositories / Select Recipe / Create Repository: maven2 (hosted)
Name:
A unique identifier for this repository
maven-release
Online:
If checked, the repository accepts incoming requests
Maven 2
Version policy:
What type of artifacts does this repository store?
Release
Layout policy:
Validate that all paths are maven artifact or metadata paths
Strict
Content Disposition:
Add Content-Disposition header as 'Attachment' to disable some content from being Inline in a browser.
Inline
Storage
Blob store:
Blob store used to store repository contents
default
Strict Content Type Validation:
Validate that all content uploaded to this repository is of a MIME type appropriate for the repository format


nexus.singhritesh85.com/#admin/repository/repositories
Sonatype Nexus Repository
OSS 3.68.1-02
Search components
user1
Sign out
Administration
Repository
Repositories
Blob Stores
Proprietary Repositories
Content Selectors
Cleanup Policies
Routing Rules
Security
Privileges
Roles
Users
Anonymous Access
LDAP
Realms
Repositories / Select Recipe / Create Repository: maven2 (hosted)
Deployment policy:
Controls if deployments of and updates to artifacts are allowed
Disable redeploy
Proprietary Components:
Components in this repository count as proprietary for namespace conflict attacks (requires Sonatype Nexus Firewall)
Cleanup
Cleanup Policies:
Components that match any of the Applied policies will be deleted
Available
Filter
Applied
Create repository
Cancel

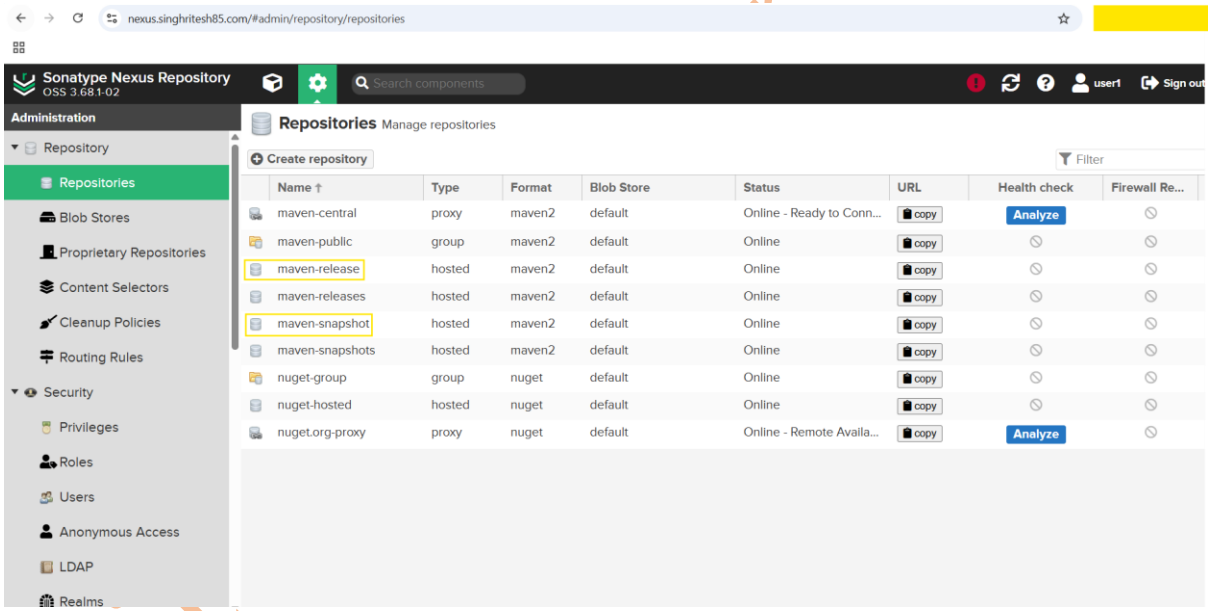

nexus.singhritesh85.com/#admin/repository/repositories
Sonatype Nexus Repository
OSS 3.68.1-02
Search components
user1
Sign out
Administration
Repository
Repositories
Blob Stores
Proprietary Repositories
Content Selectors
Cleanup Policies
Routing Rules
Security
Privileges
Roles
Users
Anonymous Access
LDAP
Realms
Repositories Manage repositories
Create repository
Filter
Name
Type
Format
Blob Store
Status
URL
Health check
Firewall Re...
maven-central proxy maven2 default Online - Ready to Conn... copy Analyze
maven-public group maven2 default Online copy
maven-release hosted maven2 default Online copy
maven-releases hosted maven2 default Online copy
maven-snapshots hosted maven2 default Online copy
nuget-group group nuget default Online copy
nuget-hosted hosted nuget default Online copy
nuget.org-proxy proxy nuget default Online - Remote Availa... copy Analyze


nexus.singhritesh85.com/#admin/repository/repositories
Sonatype Nexus Repository
OSS 3.68.1-02
Search components
user1
Sign out
Administration
Repository
Repositories
Blob Stores
Proprietary Repositories
Content Selectors
Cleanup Policies
Routing Rules
Security
Privileges
Roles
Users
Anonymous Access
LDAP
Realms
Repositories / Select Recipe
Recipe
apt (hosted)
apt (proxy)
bower (group)
bower (hosted)
bower (proxy)
cocoapods (proxy)
conan (proxy)
conda (proxy)
docker (group)
docker (hosted)
docker (proxy)
gitlfs (hosted)
go (group)
go (proxy)
helm (hosted)
helm (proxy)
maven2 (group)
maven2 (hosted)
maven2 (proxy)
npm (group)


nexus.singhritesh85.com/#admin/repository/repositories
Sonatype Nexus Repository
OSS 3.68.1-02
Search components
user1
Sign out
Administration
Repository
Repositories
Blob Stores
Proprietary Repositories
Content Selectors
Cleanup Policies
Routing Rules
Security
Privileges
Roles
Users
Anonymous Access
LDAP
Realms
Repositories / Select Recipe / Create Repository: maven2 (hosted)
Name:
A unique identifier for this repository
maven-snapshot
Online:
If checked, the repository accepts incoming requests
Maven 2
Version policy:
What type of artifacts does this repository store?
Snapshot
Layout policy:
Validate that all paths are maven artifact or metadata paths
Strict
Content Disposition:
Add Content-Disposition header as 'Attachment' to disable some content from being inline in a browser.
Inline
Storage
Blob store:
Blob store used to store repository contents
default
Strict Content Type Validation:
Validate that all content uploaded to this repository is of a MIME type appropriate for the repository format



Finally, the two repositories in Nexus named as **maven-release** and **maven-snapshot** had been created as shown in the screenshot attached below.



Installation of Nginx Ingress Controller in EKS Cluster and AKS Cluster

To install Nginx ingress controller in EKS Cluster use the Command as provided below and use you own SSL Certificate ARN instead of arn:aws:acm:us-east-2:02XXXXXXXXX6:certificate/XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX.

kubectl create ns ingress-nginx
helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
helm repo update

helm install ingress-nginx ingress-nginx/ingress-nginx -n ingress-nginx --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-ssl-cert"=arn:aws:acm:us-east-2:02XXXXXXXXX6:certificate/XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-connection-idle-timeout"="60" --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-cross-zone-load-balancing-enabled"="true" --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-type"="elb" --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-backend-protocol"=http --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-ssl-ports"=https --set controller.service.targetPorts.https=http --set-string controller.config.use-forwarded-headers="true"

```
[root@                ~]# kubectl config use-context eks-demo-cluster-dev
Switched to context "eks-demo-cluster-dev".
[root@                ~]# kubectl create ns ingress-nginx
namespace/ingress-nginx created
[root@                ~]# helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
"ingress-nginx" has been added to your repositories
[root@                ~]# helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "ingress-nginx" chart repository
Update Complete. ⎈Happy Helming!⎈
[root@                ~]# helm install ingress-nginx ingress-nginx/ingress-nginx -n ingress-nginx --set controller.service.annotations."service\.beta\.kubern
etes\.io/aws-load-balancer-ssl-cert"=arn:aws:acm:us-east-2:02        6:certificate/        --set controller.service.annotations.
"service\.beta\.kubernetes\.io/aws-load-balancer-connection-idle-timeout"="60" --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-ba
lancer-cross-zone-load-balancing-enabled"="true" --set controller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-type"="elb" --set contr
oller.service.annotations."service\.beta\.kubernetes\.io/aws-load-balancer-backend-protocol"=http --set controller.service.annotations."service\.beta\.kuberne
tes\.io/aws-load-balancer-ssl-ports"=https --set controller.service.targetPorts.https=http --set-string controller.config.use-forwarded-headers="true"
```

```
[root@                ~]# kubectl get all -n ingress-nginx
NAME                                        READY   STATUS    RESTARTS   AGE
pod/ingress-nginx-controller-               1/1     Running   0          2m45s

NAME                                 TYPE           CLUSTER-IP     EXTERNAL-IP                                                              PORT(S)
                AGE
service/ingress-nginx-controller             LoadBalancer   10.     .196   a                                .us-east-2.elb.amazonaws.com   80:32339/T
CP,443:32132/TCP   2m45s
service/ingress-nginx-controller-admission   ClusterIP      10.     16     <none>                                                          443/TCP
                2m45s

NAME                                    READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/ingress-nginx-controller   1/1     1            1           2m45s

NAME                                              DESIRED   CURRENT   READY   AGE
replicaset.apps/ingress-nginx-controller-         1         1         1       2m45s
```

To install Nginx Ingress controller in AKS Cluster use the commands as written below.

kubectl create ns ingress-nginx

helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx

helm repo update

helm install ingress-nginx ingress-nginx/ingress-nginx -n ingress-nginx

helm upgrade ingress-nginx ingress-nginx/ingress-nginx --namespace ingress-nginx --set controller.service.annotations."service\.beta\.kubernetes\.io/azure-load-balancer-health-probe-request-path"=/healthz --set controller.service.externalTrafficPolicy=Local

```
[root@                    ~]# kubectl config use-context aks-cluster
Switched to context "aks-cluster".
[root@                    ~]# kubectl create ns ingress-nginx
namespace/ingress-nginx created
[root@                    ~]# helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
"ingress-nginx" already exists with the same configuration, skipping
[root@                    ~]# helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "ingress-nginx" chart repository
Update Complete. ⎈Happy Helming!⎈
[root@                    ~]# helm install ingress-nginx ingress-nginx/ingress-nginx -n ingress-nginx
```

```
[root@                    ~]# helm upgrade ingress-nginx ingress-nginx/ingress-nginx --namespace ingress-nginx --set controller.service.annotations."service\.bet
a\.kubernetes\.io/azure-load-balancer-health-probe-request-path"=/healthz --set controller.service.externalTrafficPolicy=Local
```

```
[root@                    ~]# kubectl get all -n ingress-nginx
NAME                                        READY   STATUS    RESTARTS   AGE
pod/ingress-nginx-controller-               1/1     Running   0          97s

NAME                                         TYPE           CLUSTER-IP     EXTERNAL-IP     PORT(S)                      AGE
service/ingress-nginx-controller             LoadBalancer   10.    .145    20.       .52   80:30130/TCP,443:32694/TCP   98s
service/ingress-nginx-controller-admission   ClusterIP      10.    81      <none>          443/TCP                      98s

NAME                                    READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/ingress-nginx-controller   1/1     1            1           98s

NAME                                               DESIRED   CURRENT   READY   AGE
replicaset.apps/ingress-nginx-controller-          1         1         1       98s
```

Installation of ArgoCD in EKS Cluster

In this project I used ArgoCD and the ArgoCD CLI for Deployment, I installed the ArgoCD as shown in the screenshot attached below.

kubectl create namespace argocd

kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install.yaml

```
[root@                    ~]# kubectl config use-context eks-demo-cluster-dev
Switched to context "eks-demo-cluster-dev".
[root@                    ~]# kubectl create namespace argocd
namespace/argocd created
[root@                    ~]# kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install.yaml
```

The ingress rule for ArgoCD is as shown in the screenshot attached below.

```
[root@                    ~]# kubectl get ing -n argocd --context=eks-demo-cluster-dev --watch
NAME              CLASS   HOSTS                    ADDRESS                                                                  PORTS   AGE
minimal-ingress   nginx   argocd.singhritesh85.com                                                .us-east-2.elb.amazonaws.com   80      20s
```

```
[root@                ~]# cat argocd-ingress-rule.yaml
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: minimal-ingress
  namespace: argocd
  annotations:
    kubernetes.io/ingress.class: nginx
    nginx.ingress.kubernetes.io/backend-protocol: "HTTPS"    ###  You can use this option for this particular case for ArgoCD but not for all
#    nginx.ingress.kubernetes.io/ssl-redirect: "false"
spec:
  ingressClassName: nginx
  rules:
  - host: argocd.singhritesh85.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: argocd-server   ### Provide your service Name
            port:
              number: 80     #### Provide your service port for this particular example you can also choose 443
```

cat argocd-ingress-rule.yaml

apiVersion: networking.k8s.io/v1

kind: Ingress

metadata:

name: minimal-ingress

namespace: argocd

annotations:

kubernetes.io/ingress.class: nginx

nginx.ingress.kubernetes.io/backend-protocol: "HTTPS" ### You can use this option for this particular case for ArgoCD but not for all

nginx.ingress.kubernetes.io/ssl-redirect: "false"

spec:

ingressClassName: nginx

rules:

- host: argocd.singhritesh85.com

http:

paths:

- path: /

pathType: Prefix

backend:

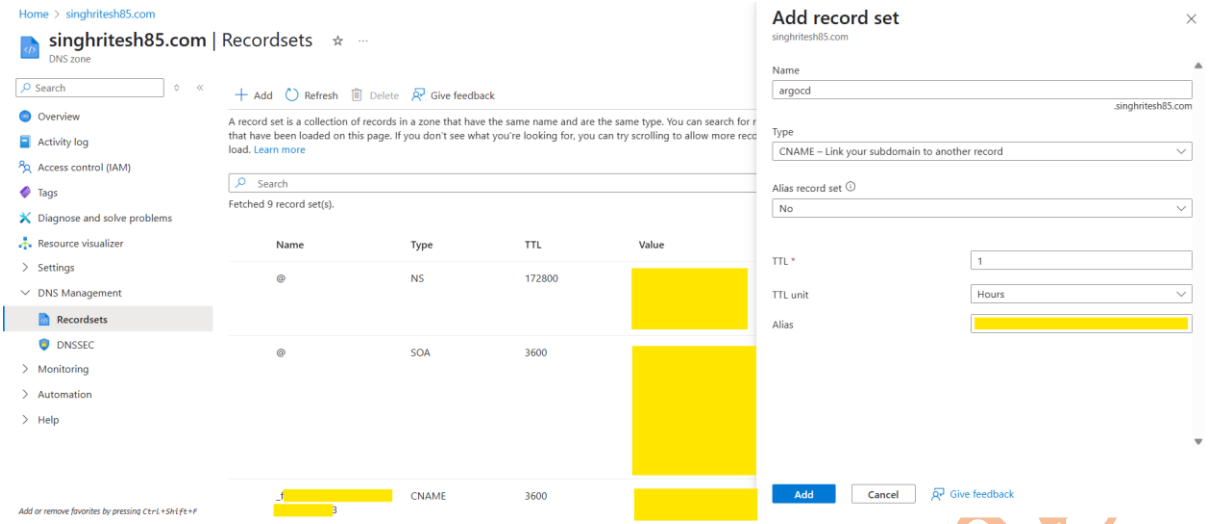
service:

name: argocd-server ### Provide your service Name

port:

number: 80 #### Provide your service port for this particular example you can also choose 443

The entry for DNS Name corresponding to HOST **argocd.singhritesh85.com** in Azure DNS Zone to create the record set is as shown in the screenshot attached below.

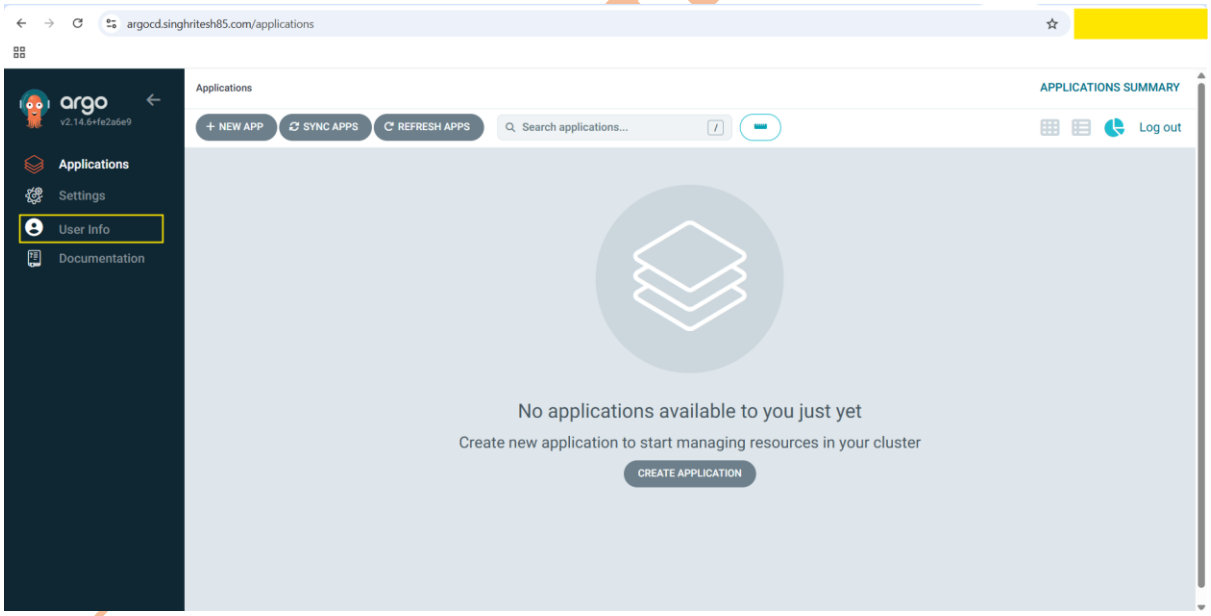


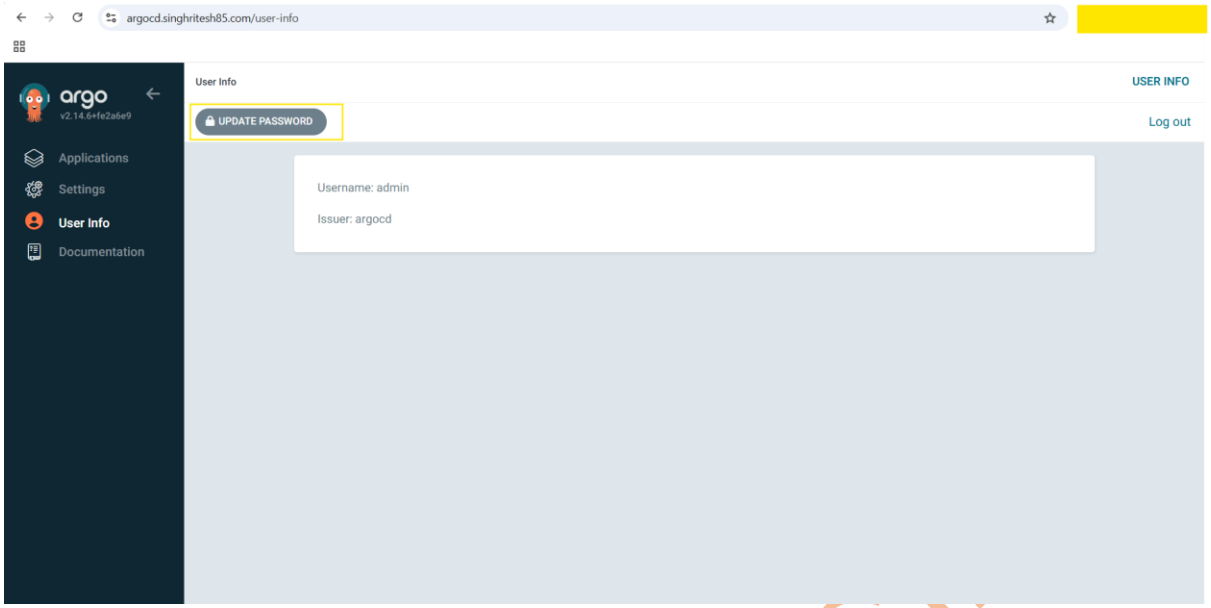
I generated the password for ArgoCD is as shown in the screenshot attached below.

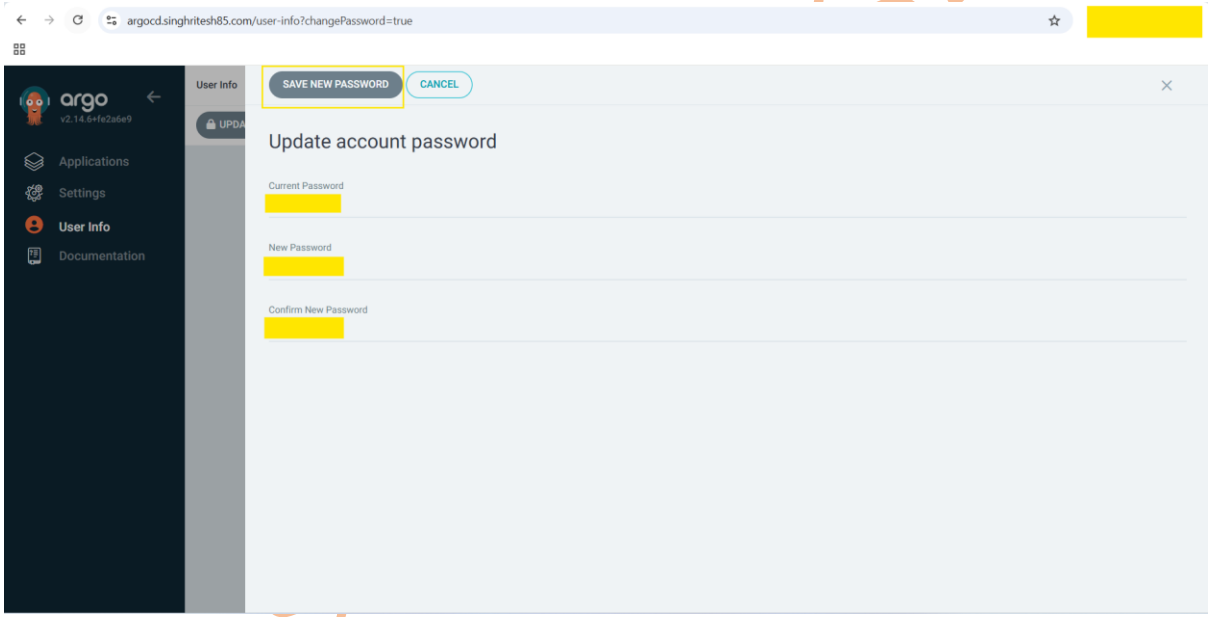
kubectl -n argocd get secret argocd-initial-admin-secret -o jsonpath='{.data.password}' | base64 -d

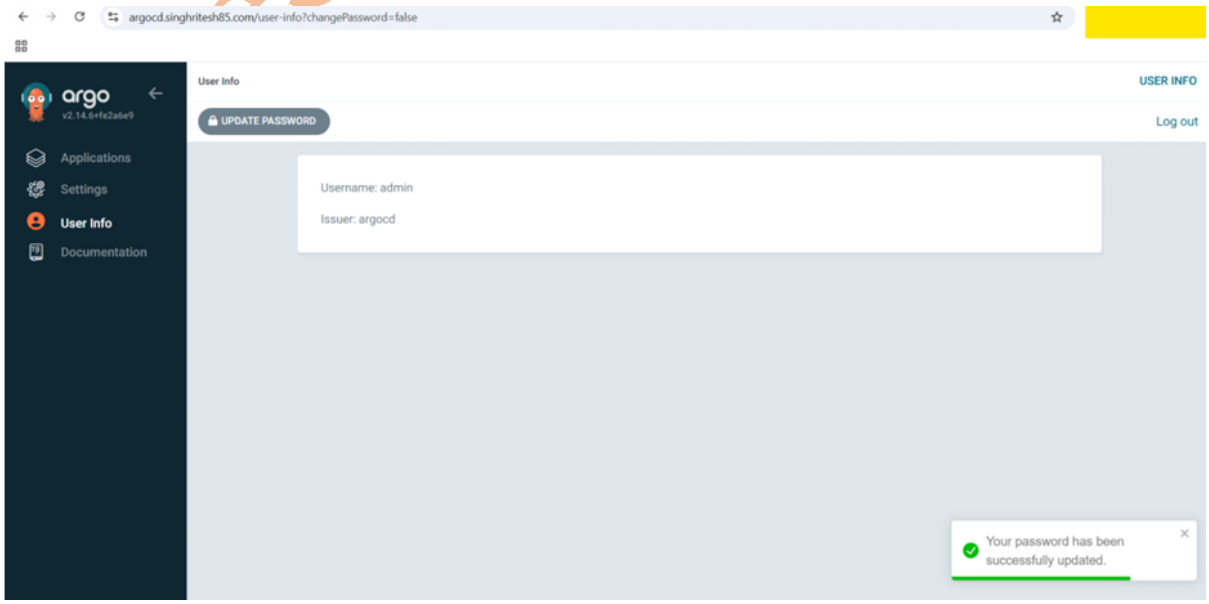
```
[root@                ~]# kubectl -n argocd get secret argocd-initial-admin-secret -o jsonpath='{.data.password}' | base64 -d
VT1-tqWnrrL3zGmz[root@                ~]#
[root@                ~]#
```

After login into the ArgoCD for the first time I updated the ArgoCD Password as shown in the screenshot attached below.




argocd.singhritesh85.com/user-info
argo
v2.14.6+fe2a6e9
Applications
Settings
User Info
Documentation
User Info
USER INFO
UPDATE PASSWORD
Log out
Username: admin
Issuer: argocd


argocd.singhritesh85.com/user-info?changePassword=true
argo
v2.14.6+fe2a6e9
Applications
Settings
User Info
Documentation
User Info
SAVE NEW PASSWORD
CANCEL
Update account password
Current Password
New Password
Confirm New Password


argocd.singhritesh85.com/user-info?changePassword=false
argo
v2.14.6+fe2a6e9
Applications
Settings
User Info
Documentation
User Info
USER INFO
UPDATE PASSWORD
Log out
Username: admin
Issuer: argocd
Your password has been successfully updated.

Installation of ArgoCD CLI using the commands as shown in the screenshot attached below.

curl -sSL -o argocd-linux-amd64 https://github.com/argoproj/argo-cd/releases/latest/download/argocd-linux-amd64

sudo install -m 555 argocd-linux-amd64 /usr/local/bin/argocd

rm argocd-linux-amd64

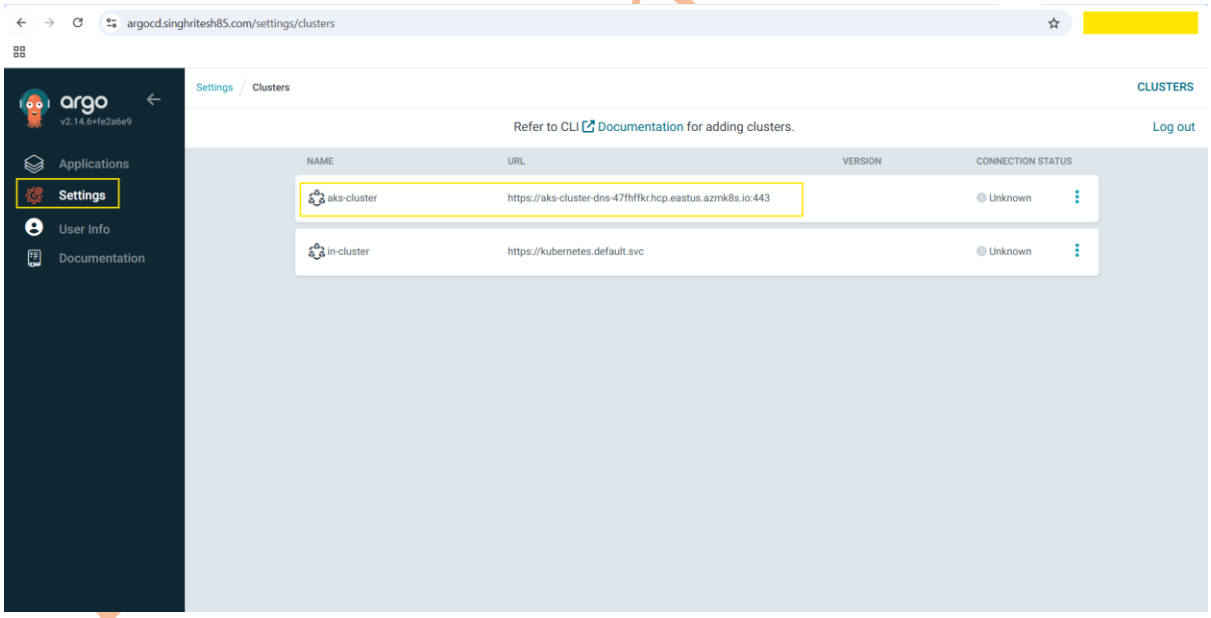
The ArgoCD CLI was installed on Jenkins Slave Node.

```
[jenkins@            ~]$ curl -sSL -o argocd-linux-amd64 https://github.com/argoproj/argo-cd/releases/latest/download/argocd-linux-amd64
[jenkins@            ~]$ sudo install -m 555 argocd-linux-amd64 /usr/local/bin/argocd
[jenkins@            ~]$
[jenkins@            ~]$ rm argocd-linux-amd64
```

I had added the aks-cluster to the ArgoCD using the command **argocd cluster add aks-cluster** as shown in the screenshot attached below.

```
[root@               ~]# argocd login argocd.singhritesh85.com --username admin --password Admin@123 --skip-test-tls  --grpc-web
'admin:login' logged in successfully
Context 'argocd.singhritesh85.com' updated
[root@               ~]# argocd cluster add aks-cluster
WARNING: This will create a service account `argocd-manager` on the cluster referenced by context `aks-cluster` with full cluster level privileges. Do you wan
t to continue [y/N]? y
INFO[0001] ServiceAccount "argocd-manager" already exists in namespace "kube-system"
INFO[0001] ClusterRole "argocd-manager-role" updated
INFO[0002] ClusterRoleBinding "argocd-manager-role-binding" updated
Cluster 'https://aks-cluster-dns-        .hcp.eastus.azmk8s.io:443' added
```

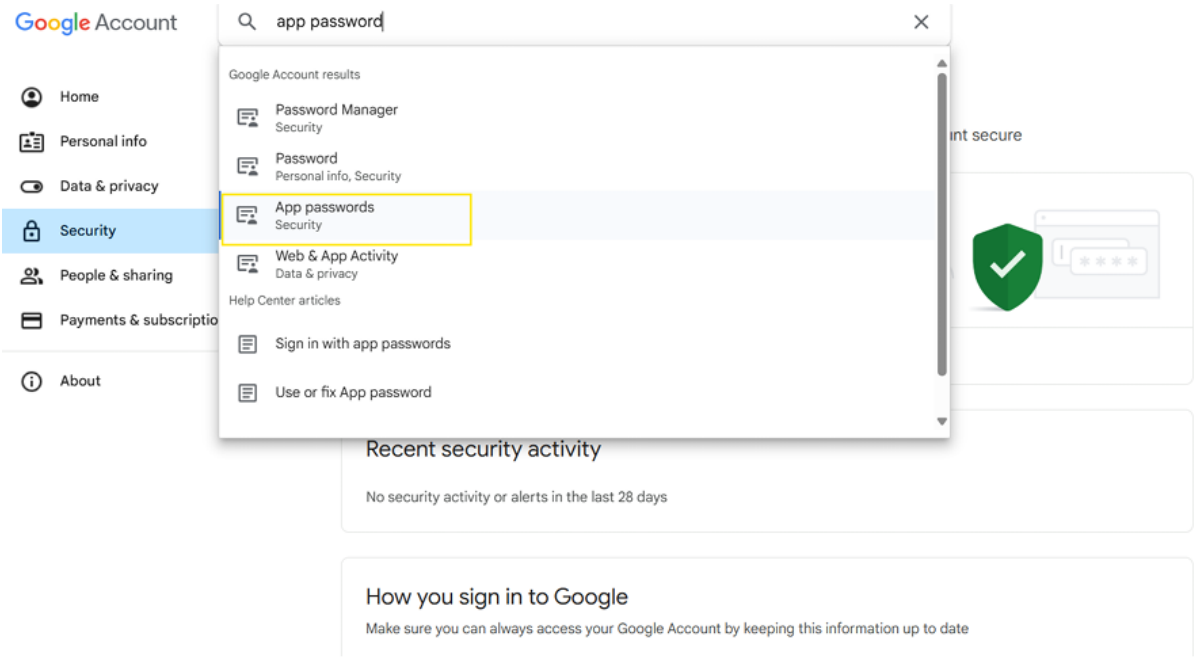
In ArgoCD UI I went to **Settings > Clusters** and you will find the two clusters one is default cluster (EKS Cluster) and another is aks-cluster as shown in the screenshot attached below.

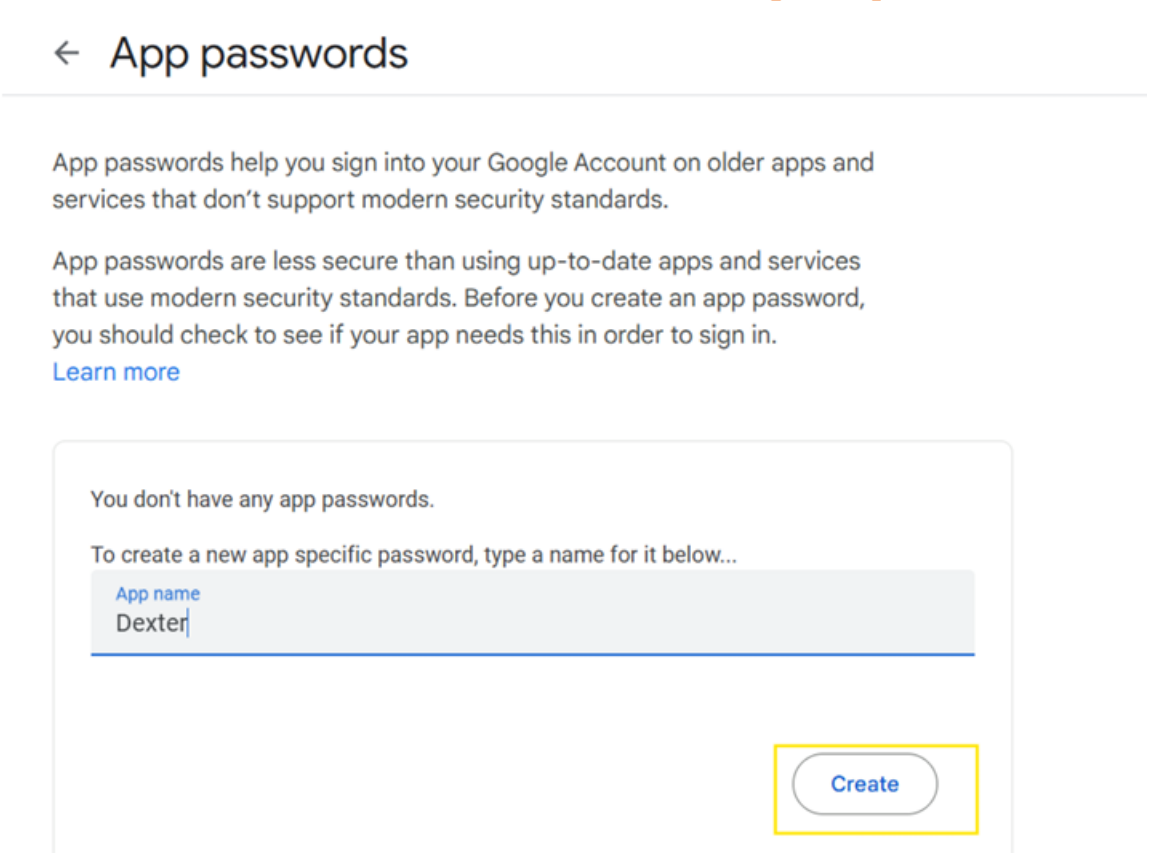


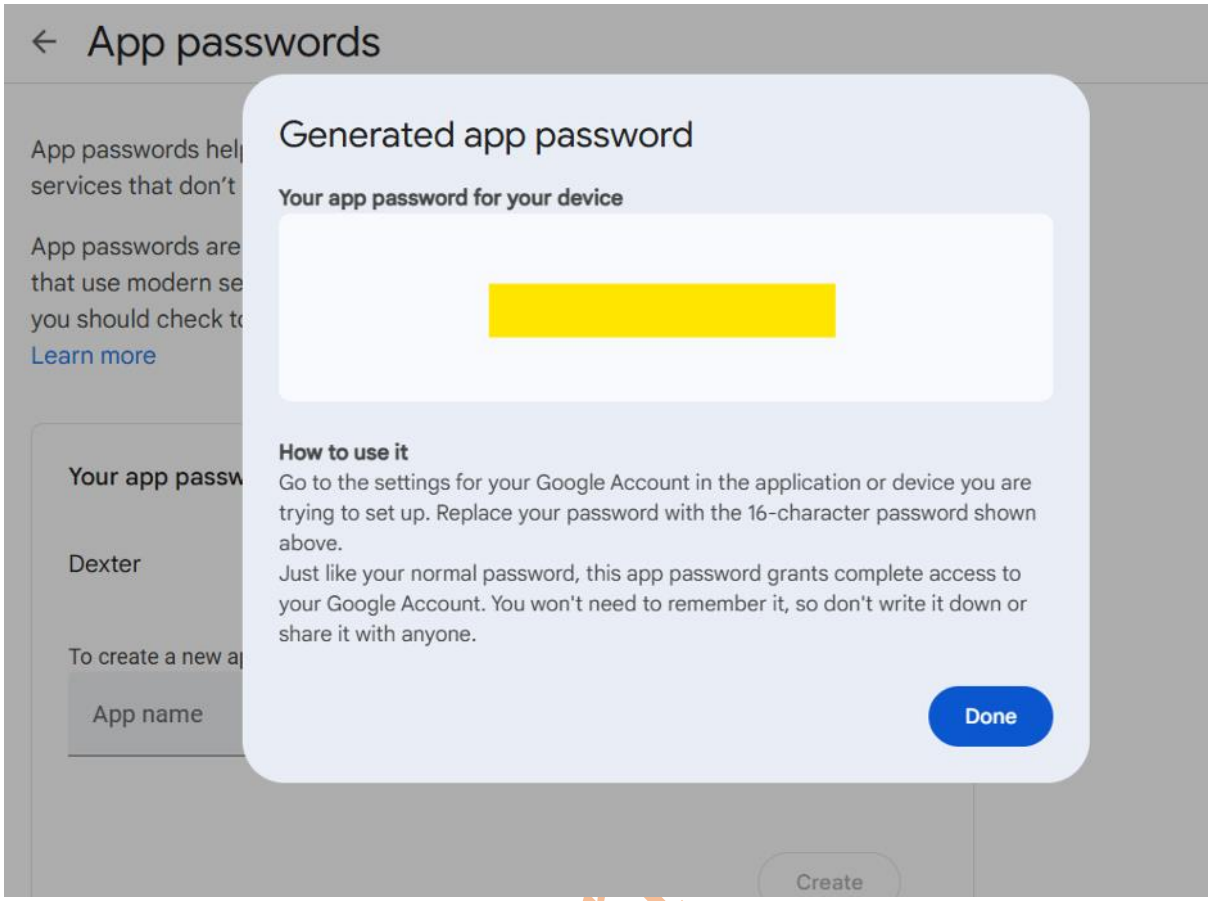
Configuration of Email to Send notification on Group Email-ID using Jenkins and Grafana

To configure Gmail to send notification to group Email ID I should have App Password for my Gmail account as shown in the screenshot attached below.

Go to your **Gmail Account > Manage your Google Account > Security** and then search for **app password** and click on **App Passwords** as shown in the screenshot attached below.


Google Account
app password
Google Account results
Password Manager
Security
Password
Personal info, Security
App passwords
Security
Web & App Activity
Data & privacy
Help Center articles
Sign in with app passwords
Use or fix App password
Home
Personal info
Data & privacy
Security
People & sharing
Payments & subscriptio
About
Recent security activity
No security activity or alerts in the last 28 days
How you sign in to Google
Make sure you can always access your Google Account by keeping this information up to date

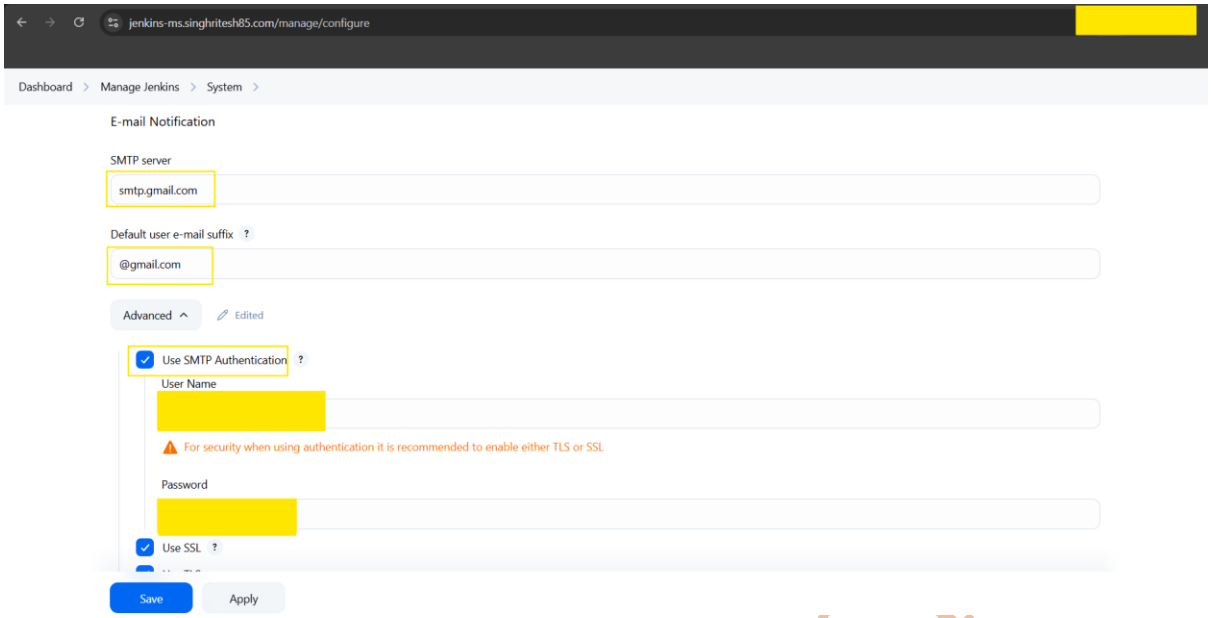

App passwords
App passwords help you sign into your Google Account on older apps and services that don't support modern security standards.
App passwords are less secure than using up-to-date apps and services that use modern security standards. Before you create an app password, you should check to see if your app needs this in order to sign in.
Learn more
You don't have any app passwords.
To create a new app specific password, type a name for it below...
App name
Dexter
Create

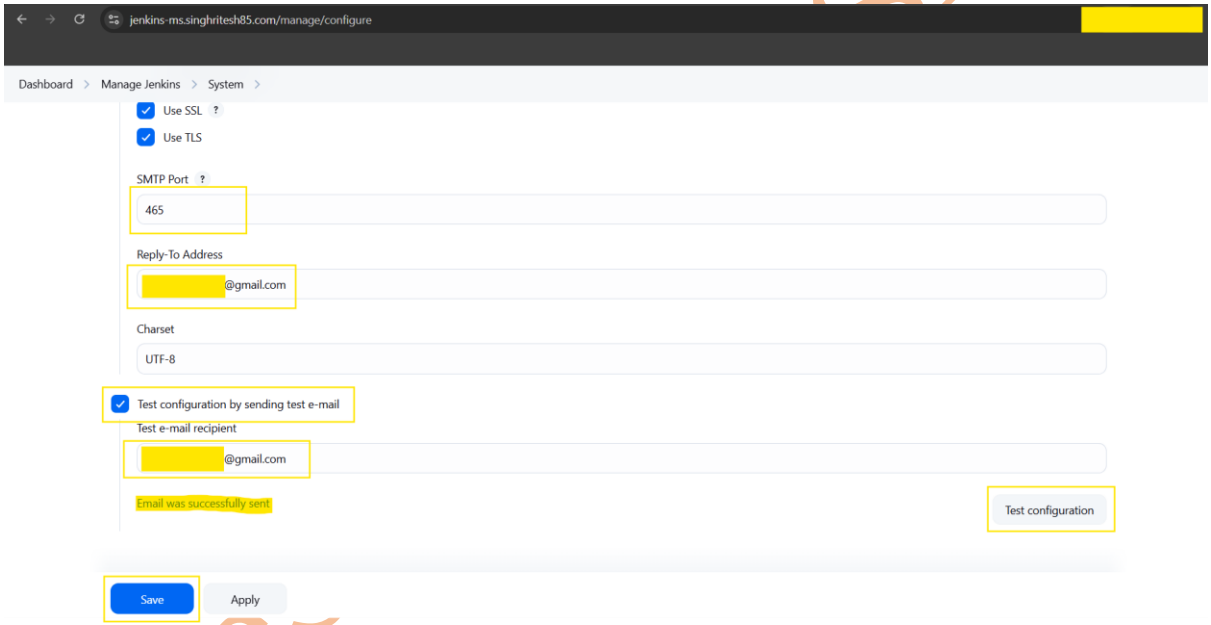


I had deleted this App Password after completion of this project, this App password does not exist anymore. You can use your own Gmail Account's App Passwords.

It is Possible to Send notification to group Email ID using Jenkins and Grafana through Amazon Simple Email Services (Amazon SES). To know more you can refer the project present in my GitHub Repo https://github.com/singhritesh85/DevOps-Project-2tier-WebApp-Deployment.git. However, for this project I used Gmail App Passwords as explained above.

Now, configuration of email to Send notification on Group Email-ID using Jenkins and Grafana is as discussed below.





To configure Alerts in Grafana, first I created contact points with the Email ID and changed smtp settings in the configuration file **/etc/grafana/grafana.ini** of Grafana as shown in the screenshot attached below.

```
################################### SMTP / Emailing ##########################
[smtp]
enabled = true
host = smtp.gmail.com:587
user = 
# If the password contains # or ; you have to wrap it with triple quotes. Ex """#password;"""
password = 
;cert_file =
;key_file =
skip_verify = true
from_address = 
from_name = Multibranch Pipeline Grafana Alert for Bankapp
# EHLO identity in SMTP dialog (defaults to instance_name)
;ehlo_identity = dashboard.example.com
# SMTP startTLS policy (defaults to 'OpportunisticStartTLS')
;startTLS_policy = NoStartTLS
# Enable trace propagation in e-mail headers, using the 'traceparent', 'tracestate' and (optionally) 'baggage' fields (defaults to false)
;enable_tracing = false
```

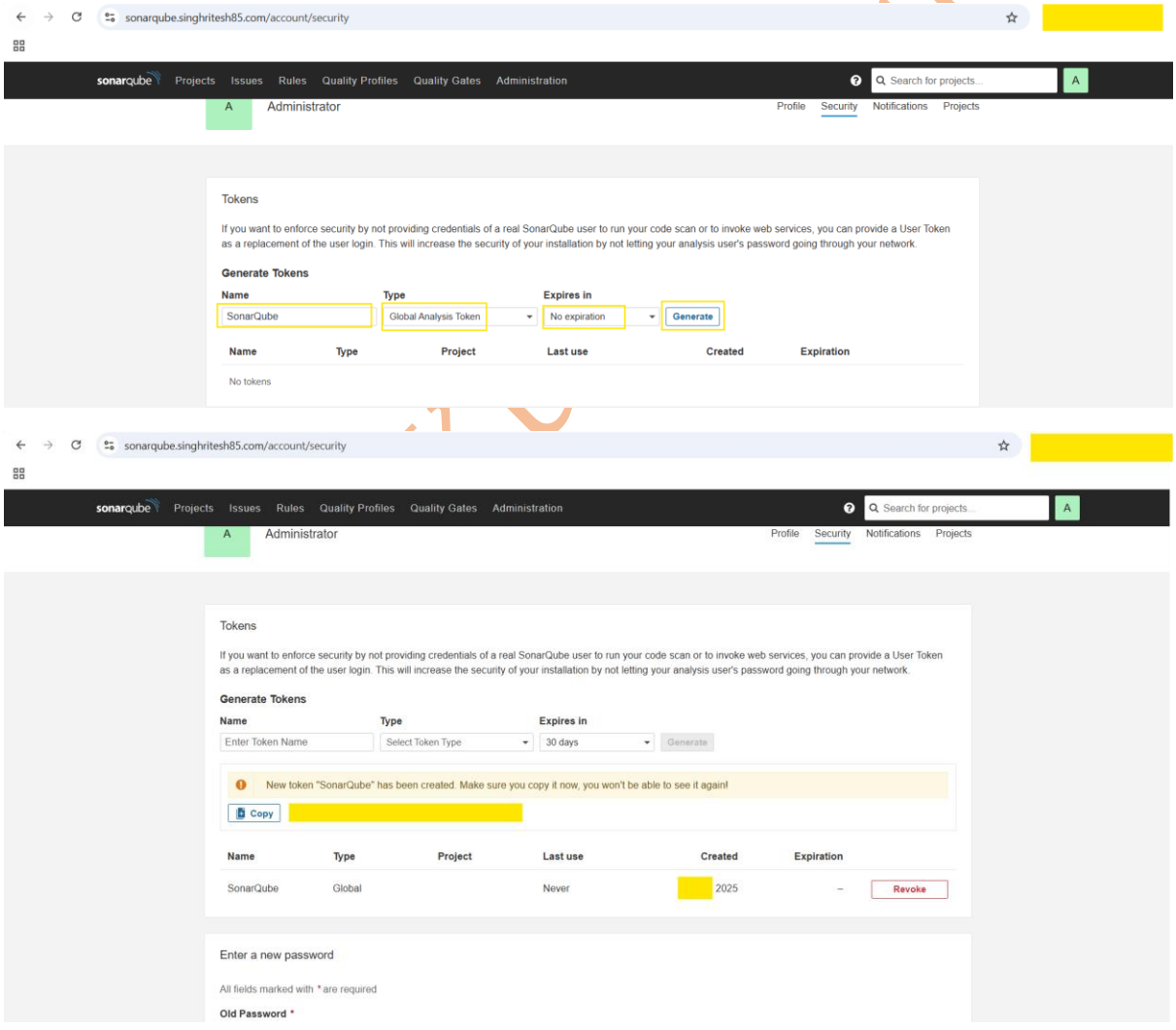
Then restart the **grafana-server** service as shown in the screenshot attached below.

```
[root@                ~]# systemctl restart grafana-server.service
[root@                ~]# systemctl status grafana-server.service
● grafana-server.service - Grafana instance
   Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; vendor preset: disabled)
   Active: active (running) since    2025-
     Docs: http://docs.grafana.org
 Main PID: 6543 (grafana)
```

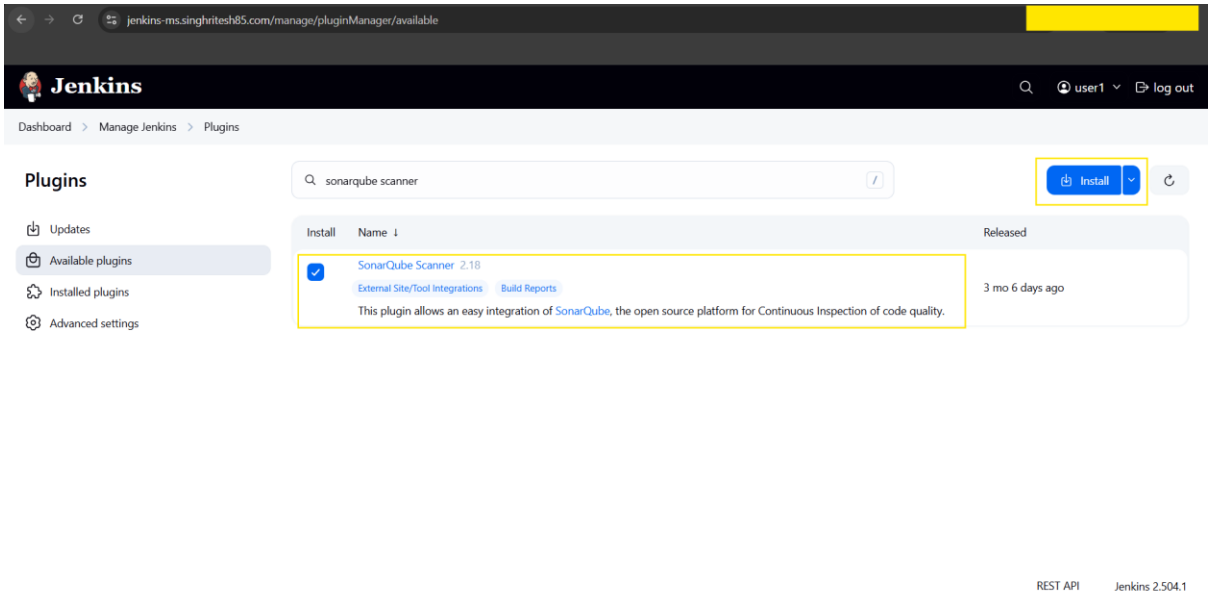
Creation of Alerts in Grafana I will discuss later here I will discuss first integration of Jenkins with SonarQube and Sonatype Nexus3 Repository then will create the Jenkins Job and deploy the Bank Application. To do so I used the Jenkinsfile as shown in the screenshot attached below. This Jenkinsfile is also available in my GitHub Repo https://github.com/singhritesh85/DevOps-Project-BankApplication-Multibranch-MultiCloud.git.

Integration of Jenkins with SonarQube

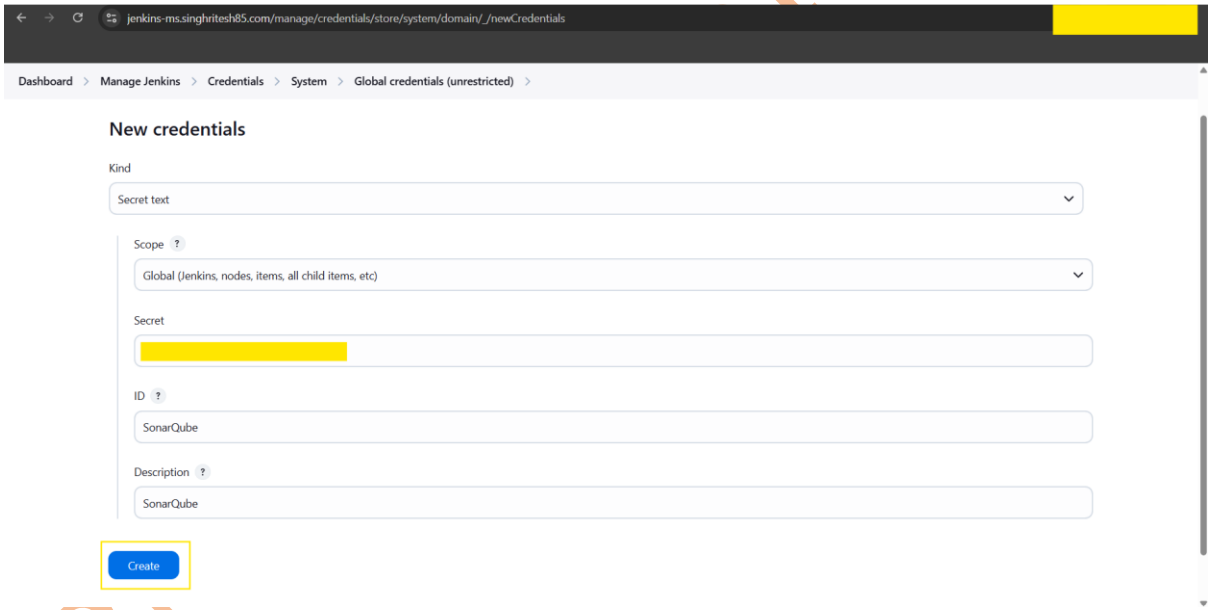
To Integrate Jenkins with SonarQube I need a Security Token in SonarQube which I created as shown in the screenshot attached below.



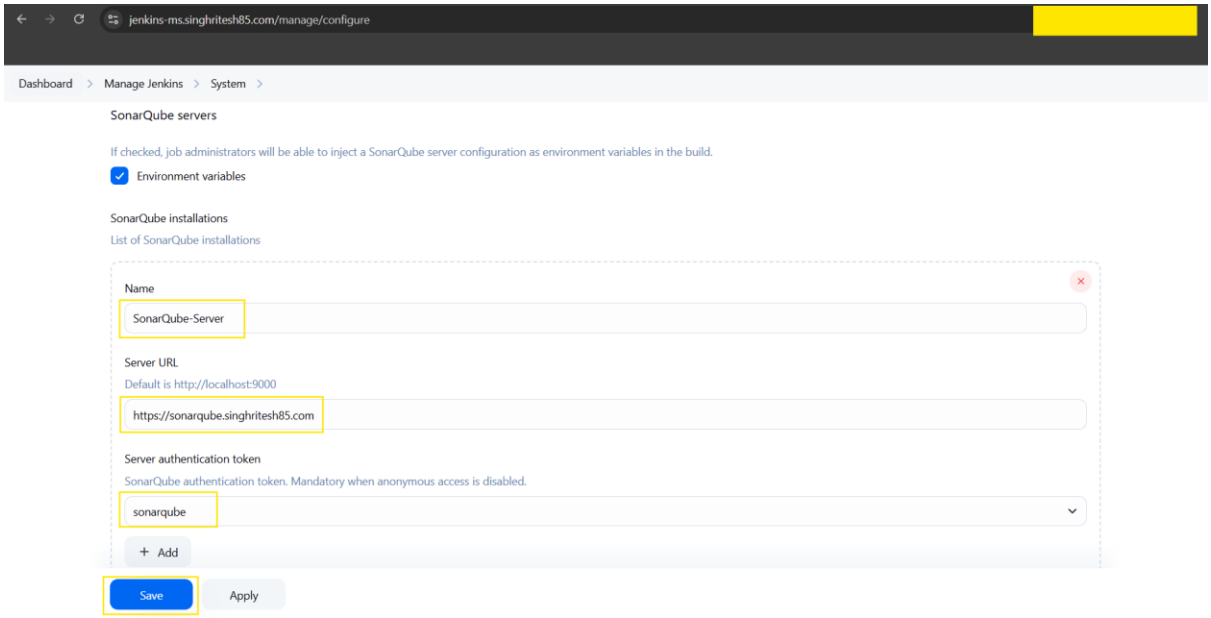
Now go to Jenkins **Manage Jenkins > System > Plugins > Available Plugins** and install the **SonarQube Scanner** Plugin as shown in the screenshot attached below.



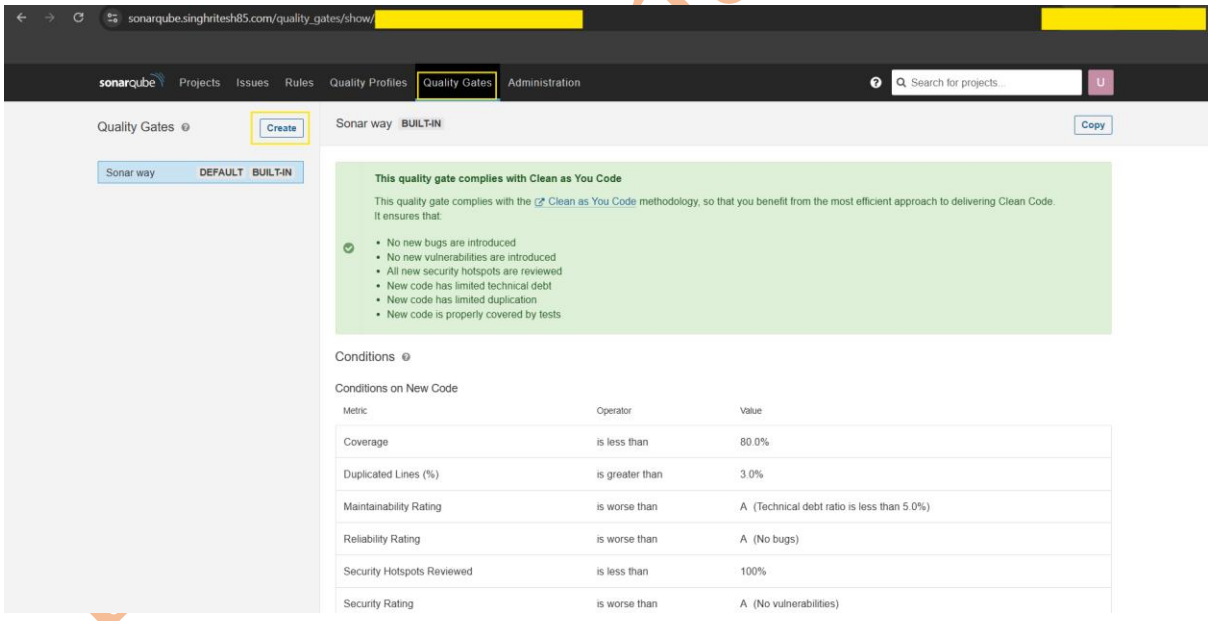
Do not restart the Jenkins after installation of **SonarQube-Scanner** Plugin. Then create a Jenkins Credential of kind Secret text with the SonarQube Secret Token which I created earlier as shown in the screenshot attached below.

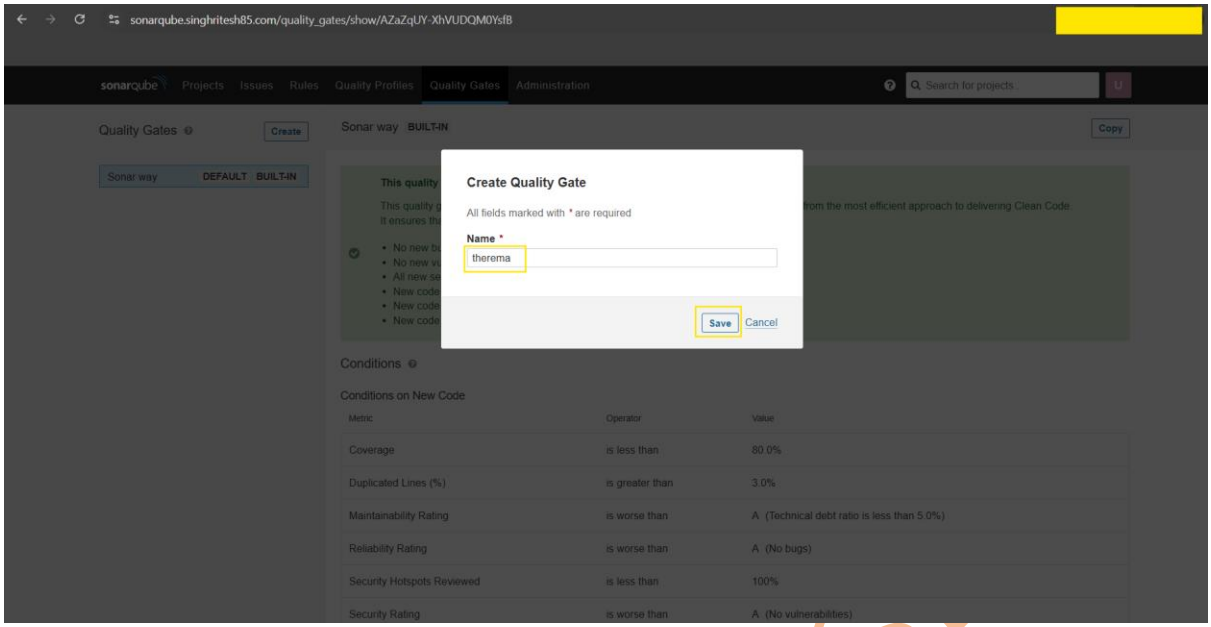


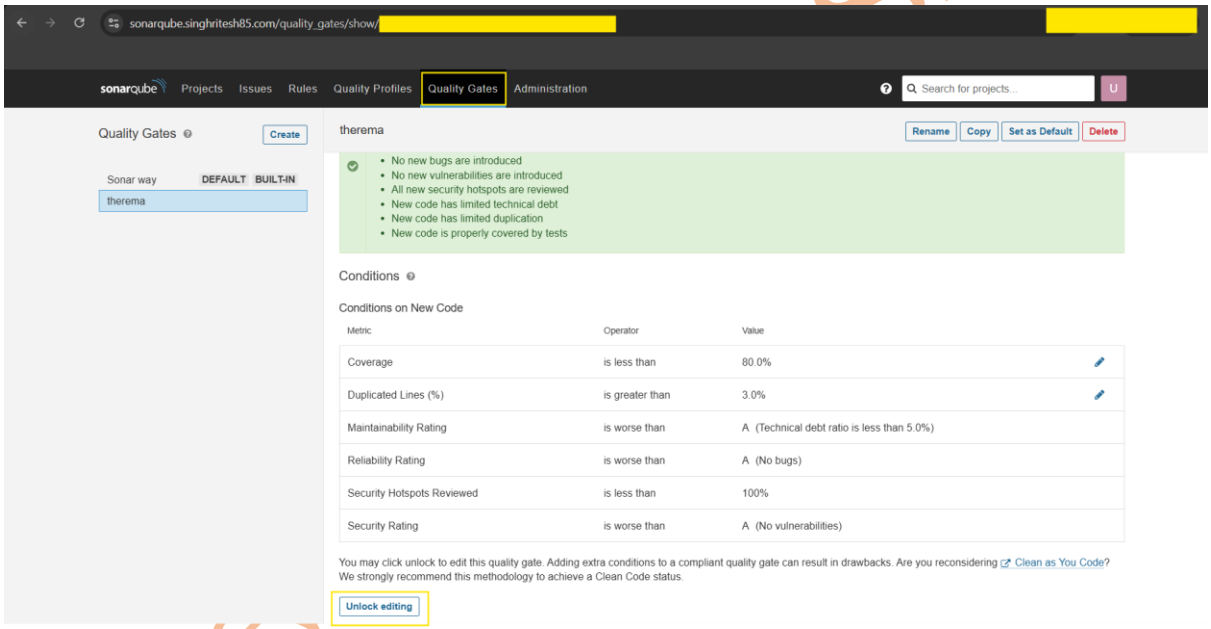
Now go to Jenkins **Manage Jenkins > System** and search for SonarQube and do the configuration as shown in the screenshot attached below.

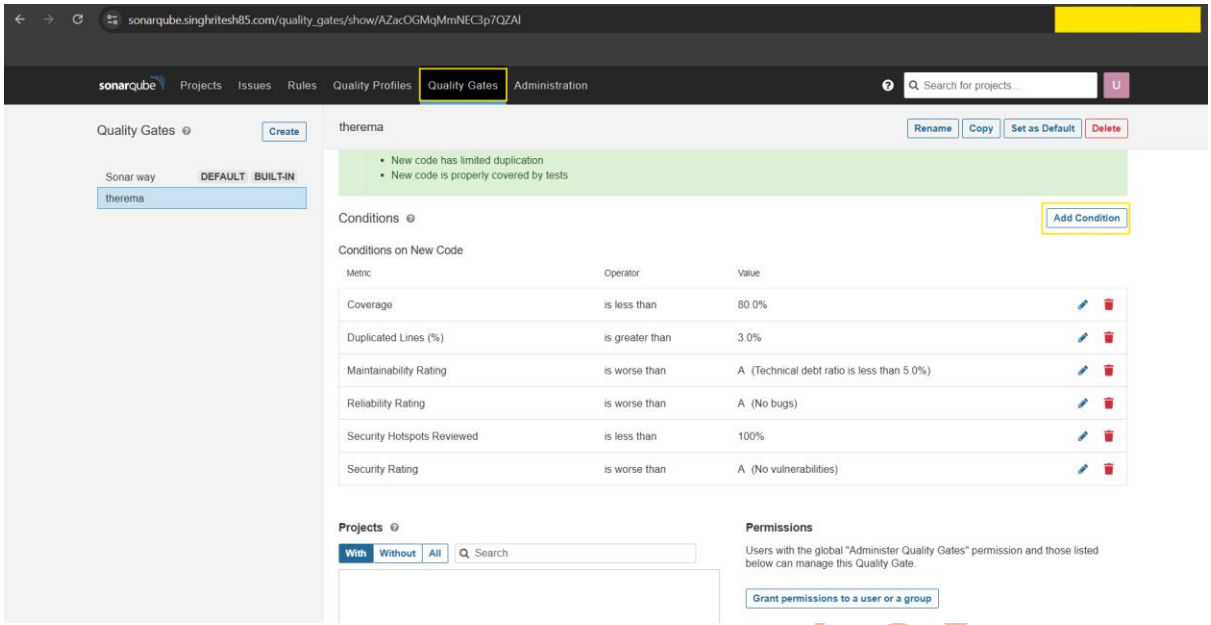


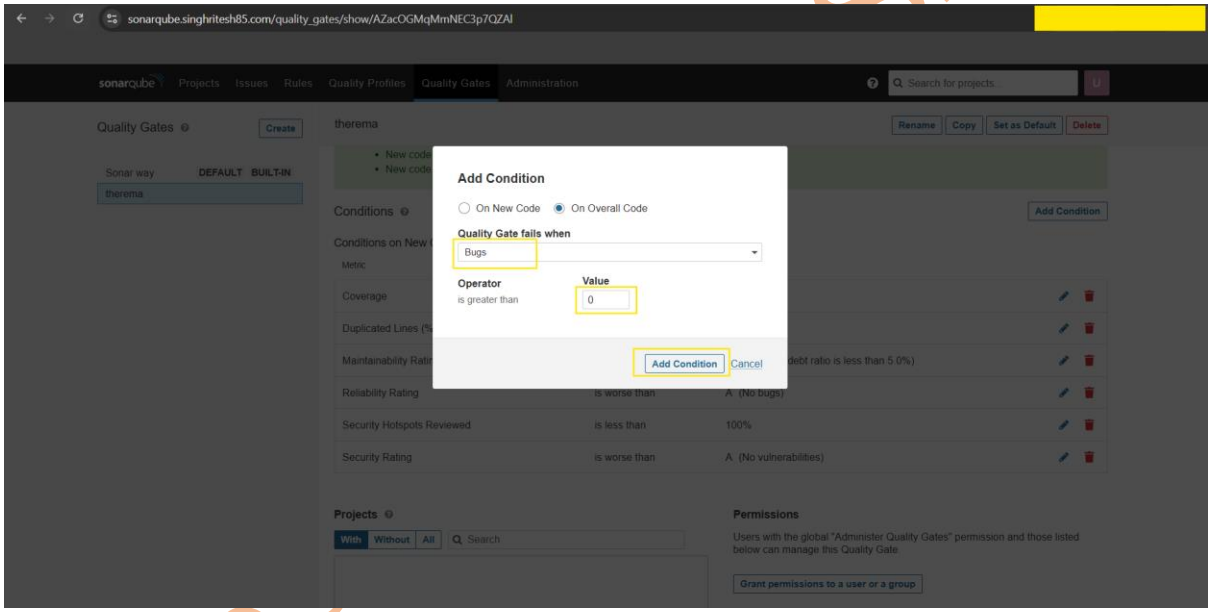
In SonarQube the **Quality Gate** is the predefined condition and threshold which decides the minimum Quality Standards of the code. During Jenkins Pipeline I had tested whether the Code passed the Quality Gate or not. If it does not pass the Quality Gate then Jenkins Job will fail there itself and do not move for further stages. Below screenshot shows how I created the Quality Gate in SonarQube for this Project.

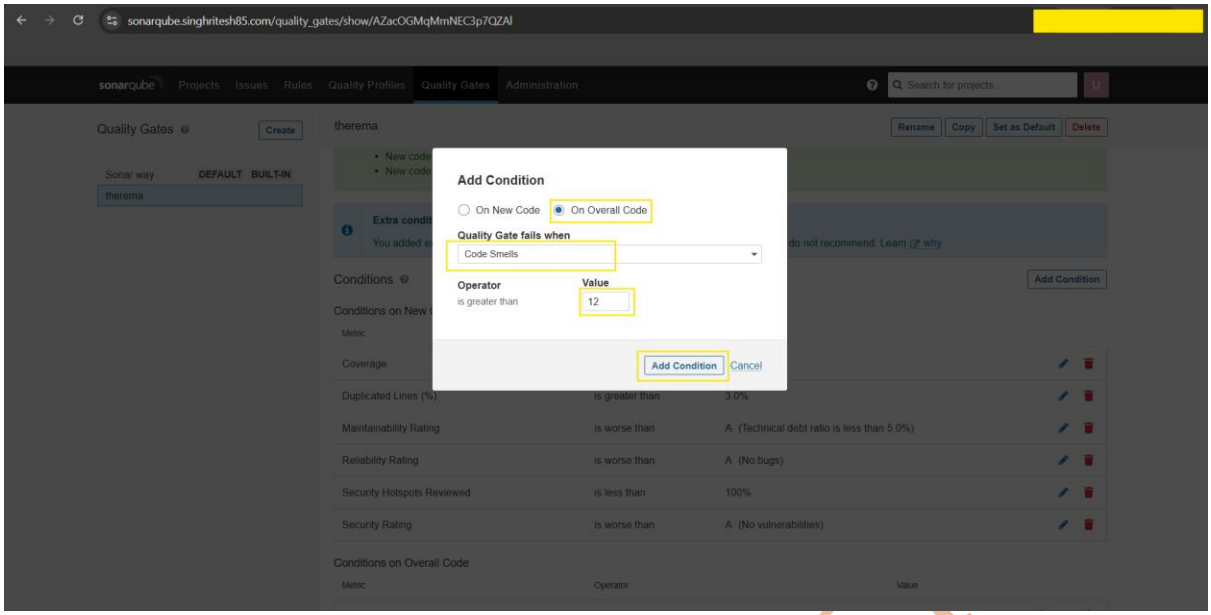


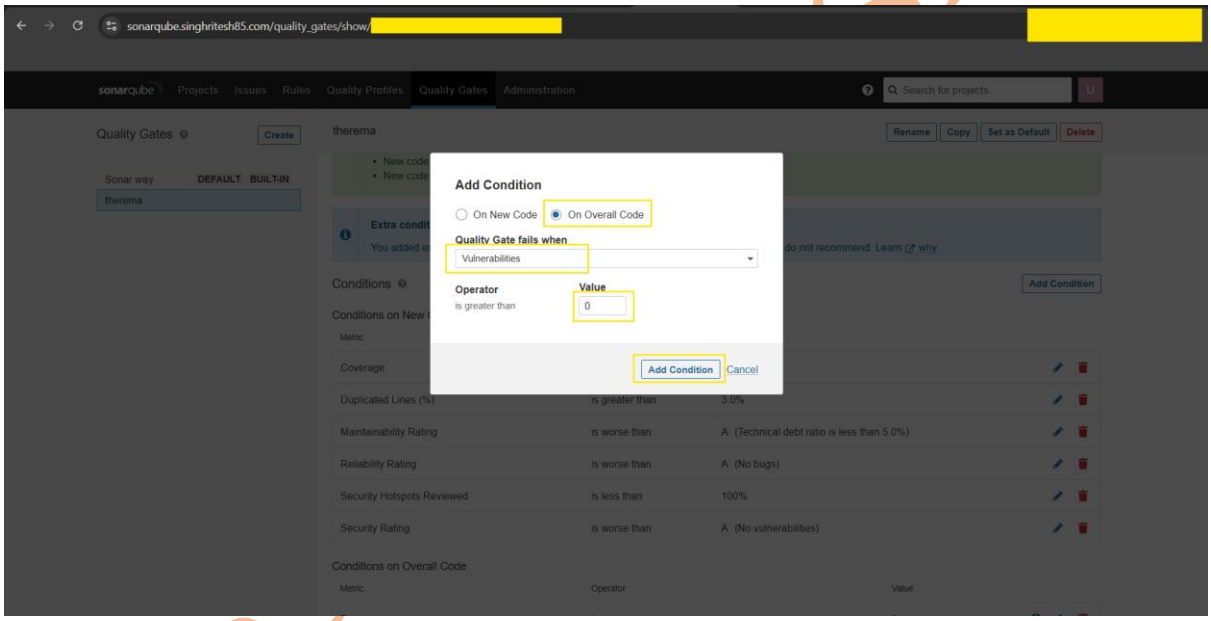

sonarqube.singhritesh85.com/quality_gates/show/AZaZqUY-XhVUDQM0YsfB
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
Quality Gates
Create
Sonar way BUILT-IN
Copy
Sonar way DEFAULT BUILT-IN
Create Quality Gate
All fields marked with * are required
Name *
therema
Save
Cancel
Conditions
Conditions on New Code
Metric
Operator
Value
Coverage
is less than
80.0%
Duplicated Lines (%)
is greater than
3.0%
Maintainability Rating
is worse than
A (Technical debt ratio is less than 5.0%)
Reliability Rating
is worse than
A (No bugs)
Security Hotspots Reviewed
is less than
100%
Security Rating
is worse than
A (No vulnerabilities)


sonarqube.singhritesh85.com/quality_gates/show/
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
Quality Gates
Create
Sonar way DEFAULT BUILT-IN
therema
therema
Rename
Copy
Set as Default
Delete
No new bugs are introduced
No new vulnerabilities are introduced
All new security hotspots are reviewed
New code has limited technical debt
New code has limited duplication
New code is properly covered by tests
Conditions
Conditions on New Code
Metric
Operator
Value
Coverage
is less than
80.0%
Duplicated Lines (%)
is greater than
3.0%
Maintainability Rating
is worse than
A (Technical debt ratio is less than 5.0%)
Reliability Rating
is worse than
A (No bugs)
Security Hotspots Reviewed
is less than
100%
Security Rating
is worse than
A (No vulnerabilities)
You may click unlock to edit this quality gate. Adding extra conditions to a compliant quality gate can result in drawbacks. Are you reconsidering Clean as You Code?
We strongly recommend this methodology to achieve a Clean Code status.
Unlock editing

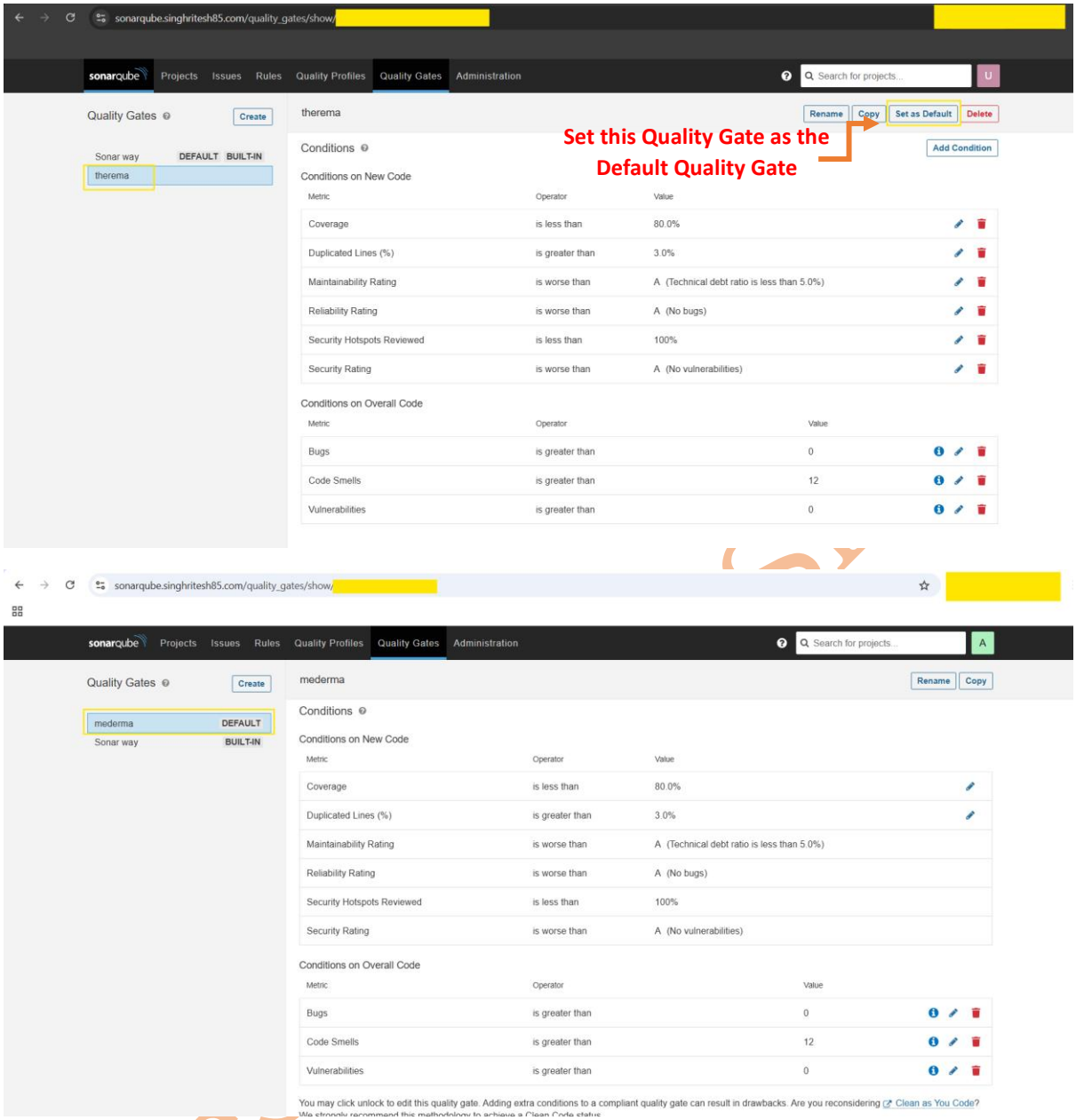

sonarqube.singhritesh85.com/quality_gates/show/AZacOGMqMmNEC3p7QZAl
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
Quality Gates
Create
Sonar way
DEFAULT
BUILT-IN
therema
therema
Rename
Copy
Set as Default
Delete
New code has limited duplication
New code is properly covered by tests
Conditions
Add Condition
Conditions on New Code
Metric
Operator
Value
Coverage
is less than
80.0%
Duplicated Lines (%)
is greater than
3.0%
Maintainability Rating
is worse than
A (Technical debt ratio is less than 5.0%)
Reliability Rating
is worse than
A (No bugs)
Security Hotspots Reviewed
is less than
100%
Security Rating
is worse than
A (No vulnerabilities)
Projects
With
Without
All
Search
Permissions
Users with the global "Administer Quality Gates" permission and those listed below can manage this Quality Gate.
Grant permissions to a user or a group


sonarqube.singhritesh85.com/quality_gates/show/AZacOGMqMmNEC3p7QZAl
Add Condition
On New Code
On Overall Code
Quality Gate fails when
Bugs
Operator
is greater than
Value
0
Add Condition
Cancel

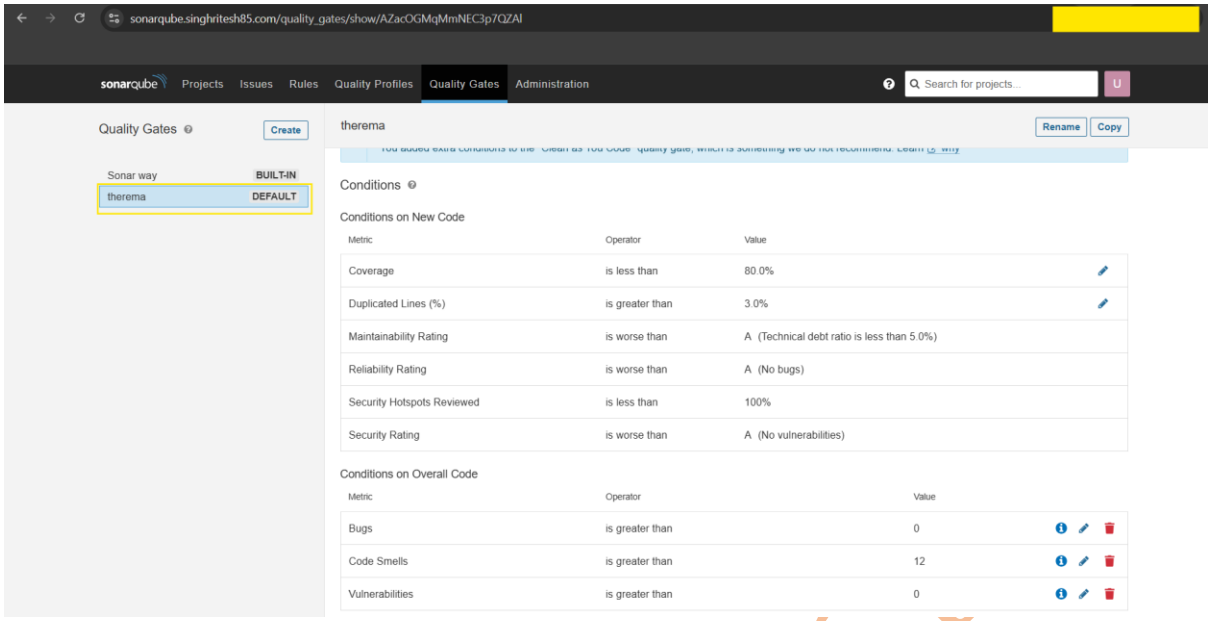




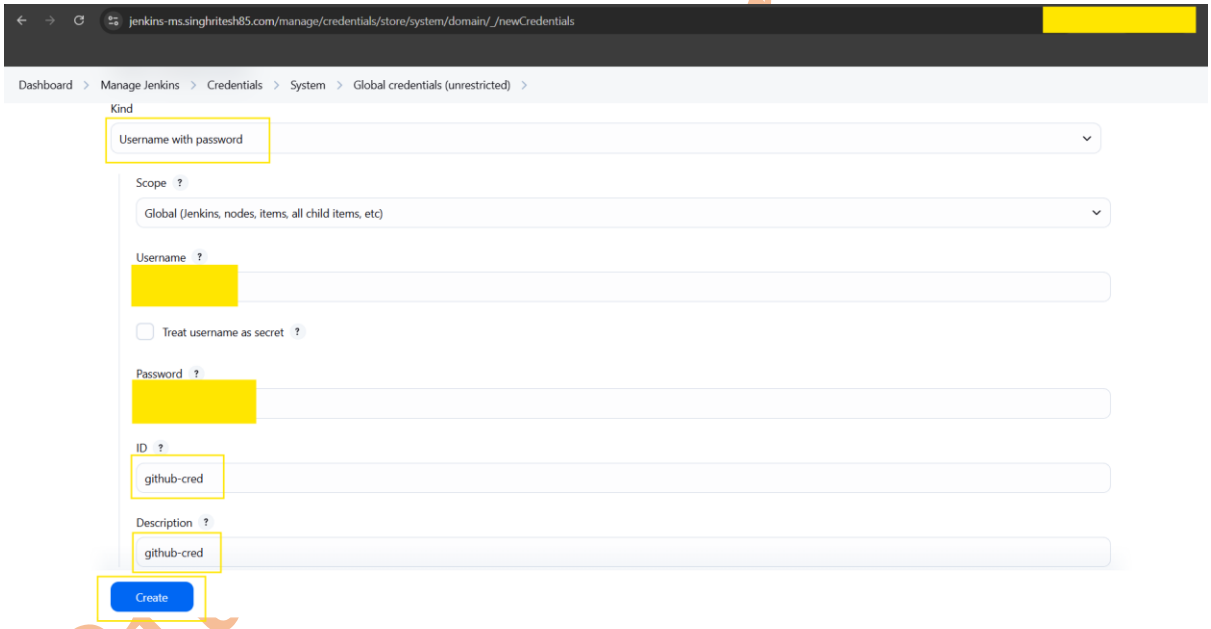
Finally, the predefine condition/threshold for Quality Gate of the SonarQube is as shown in the screenshot attached below.


sonarqube.singhritesh85.com/quality_gates/show/
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
Quality Gates
Create
Sonar way
DEFAULT
BUILT-IN
therema
therema
Rename
Copy
Set as Default
Delete
Set this Quality Gate as the Default Quality Gate
Conditions
Add Condition
Conditions on New Code
Metric
Operator
Value
Coverage
is less than
80.0%
Duplicated Lines (%)
is greater than
3.0%
Maintainability Rating
is worse than
A (Technical debt ratio is less than 5.0%)
Reliability Rating
is worse than
A (No bugs)
Security Hotspots Reviewed
is less than
100%
Security Rating
is worse than
A (No vulnerabilities)
Conditions on Overall Code
Bugs
is greater than
0
Code Smells
is greater than
12
Vulnerabilities
is greater than
0

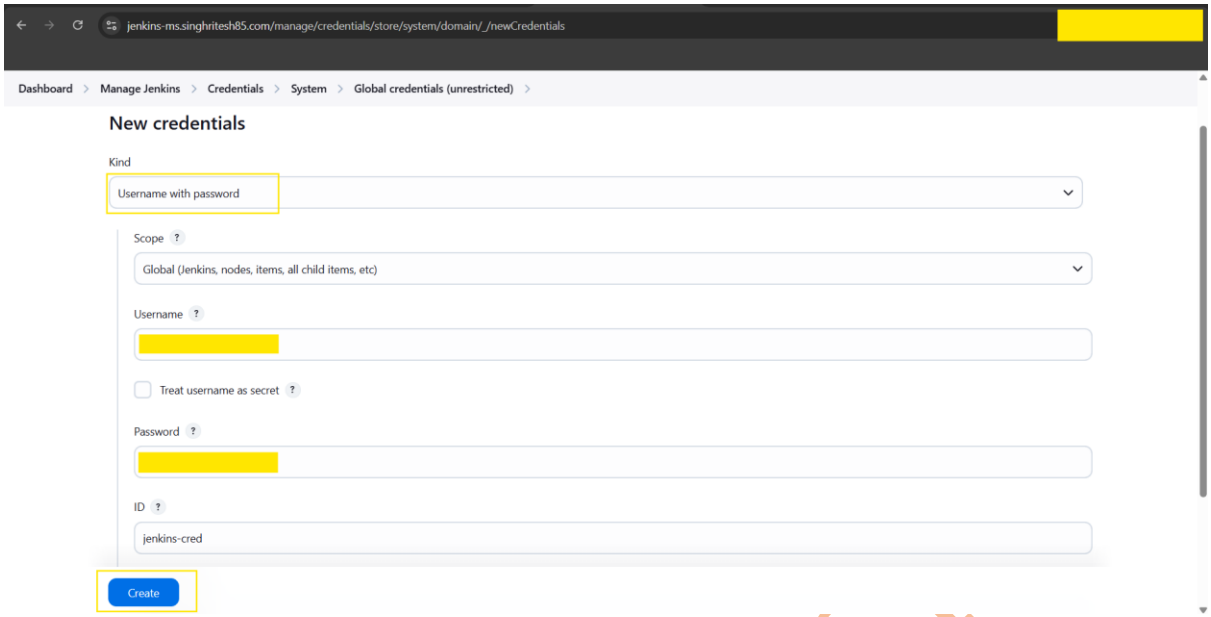
sonarqube.singhritesh85.com/quality_gates/show/
sonarqube
Projects
Issues
Rules
Quality Profiles
Quality Gates
Administration
Search for projects...
Quality Gates
Create
mederma
DEFAULT
Sonar way
BUILT-IN
mederma
Rename
Copy
Conditions
Conditions on New Code
Metric
Operator
Value
Coverage
is less than
80.0%
Duplicated Lines (%)
is greater than
3.0%
Maintainability Rating
is worse than
A (Technical debt ratio is less than 5.0%)
Reliability Rating
is worse than
A (No bugs)
Security Hotspots Reviewed
is less than
100%
Security Rating
is worse than
A (No vulnerabilities)
Conditions on Overall Code
Bugs
is greater than
0
Code Smells
is greater than
12
Vulnerabilities
is greater than
0
You may click unlock to edit this quality gate. Adding extra conditions to a compliant quality gate can result in drawbacks. Are you reconsidering Clean as You Code?



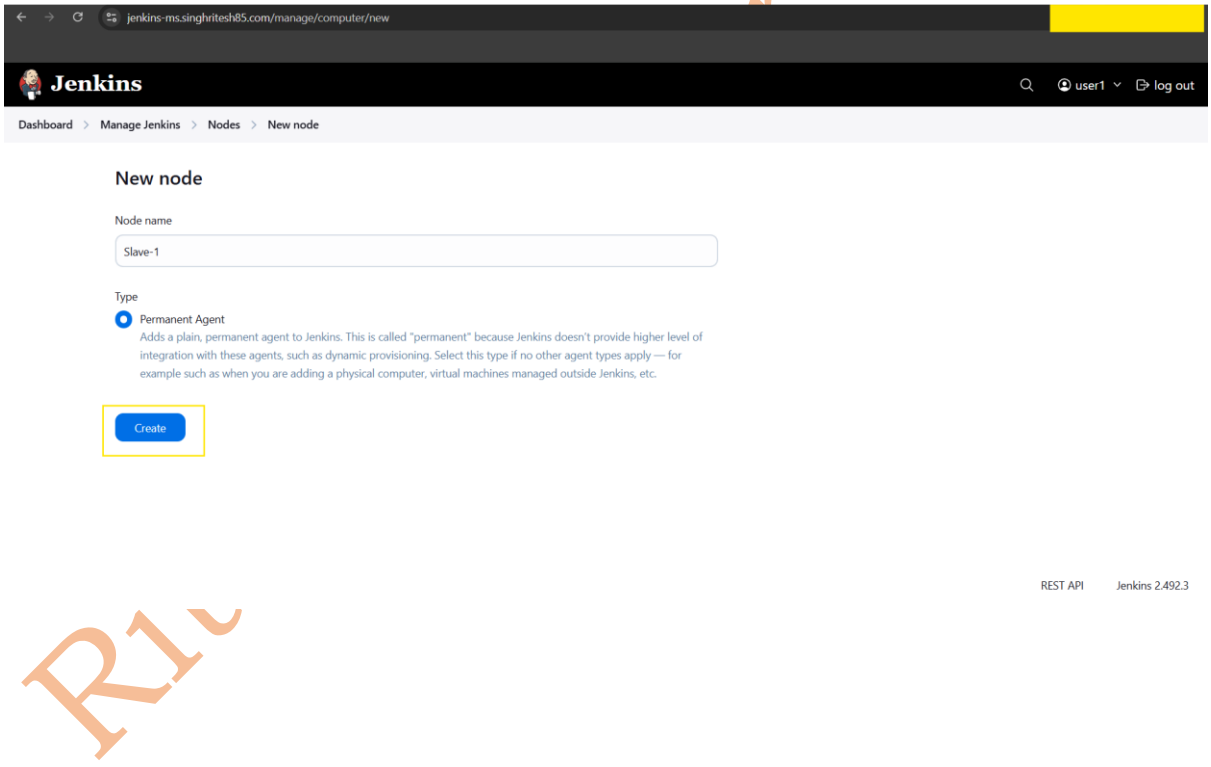
I had created the credential named as github-cred in Jenkins for Authenticating GitHub with Jenkins as shown in the screenshot attached below.

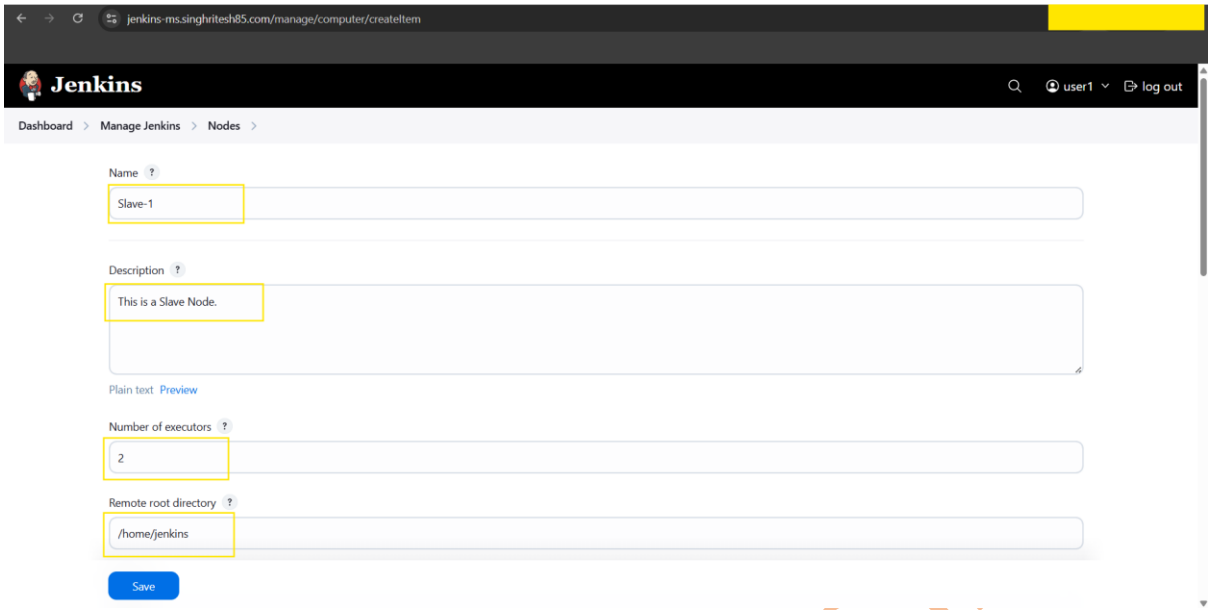


Create a Jenkins Secret to integrate Jenkins Slave Node with Jenkins Master as shown in the screenshot attached below.

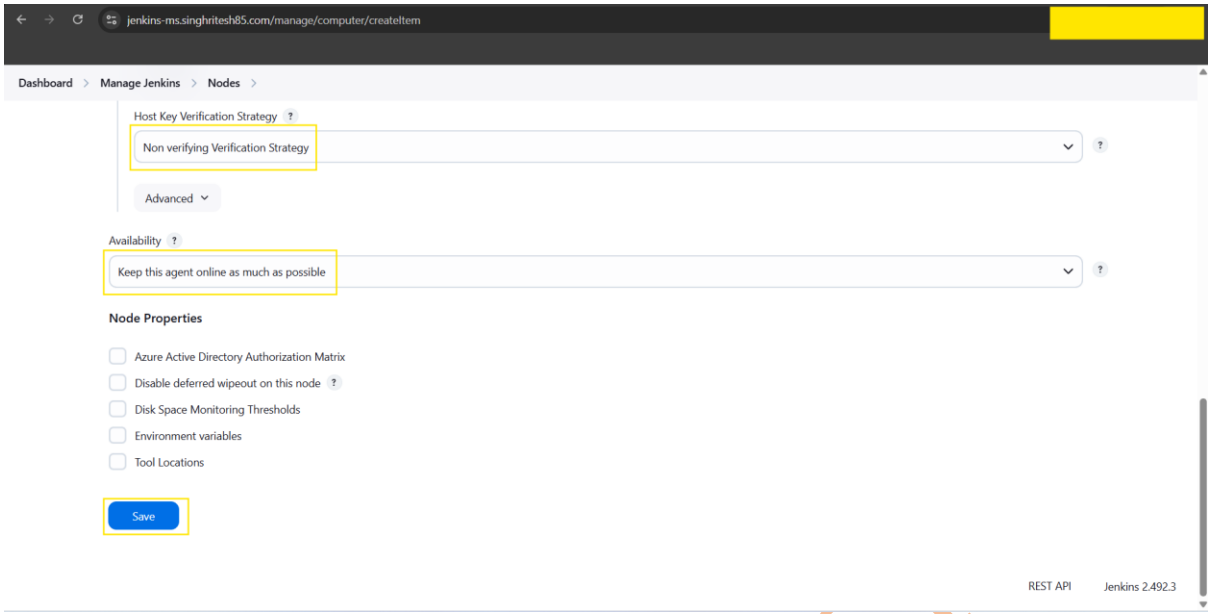


I had integrated a Slave node in Jenkins with the name of **Slave-1**. To do so go to **Manage Jenkins > Nodes > New Node** as shown in the screenshot attached below.

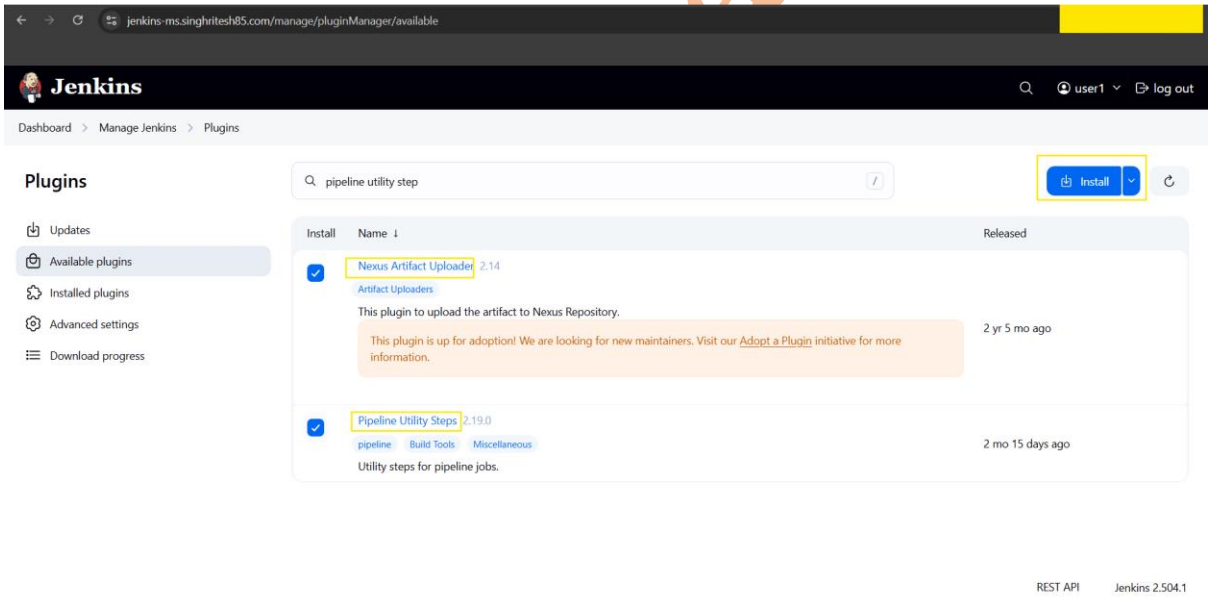



jenkins-ms.singhritesh85.com/manage/computer/createItem
Jenkins
user1
log out
Dashboard
Manage Jenkins
Nodes
Name
Slave-1
Description
This is a Slave Node.
Plain text
Preview
Number of executors
2
Remote root directory
/home/jenkins
Save

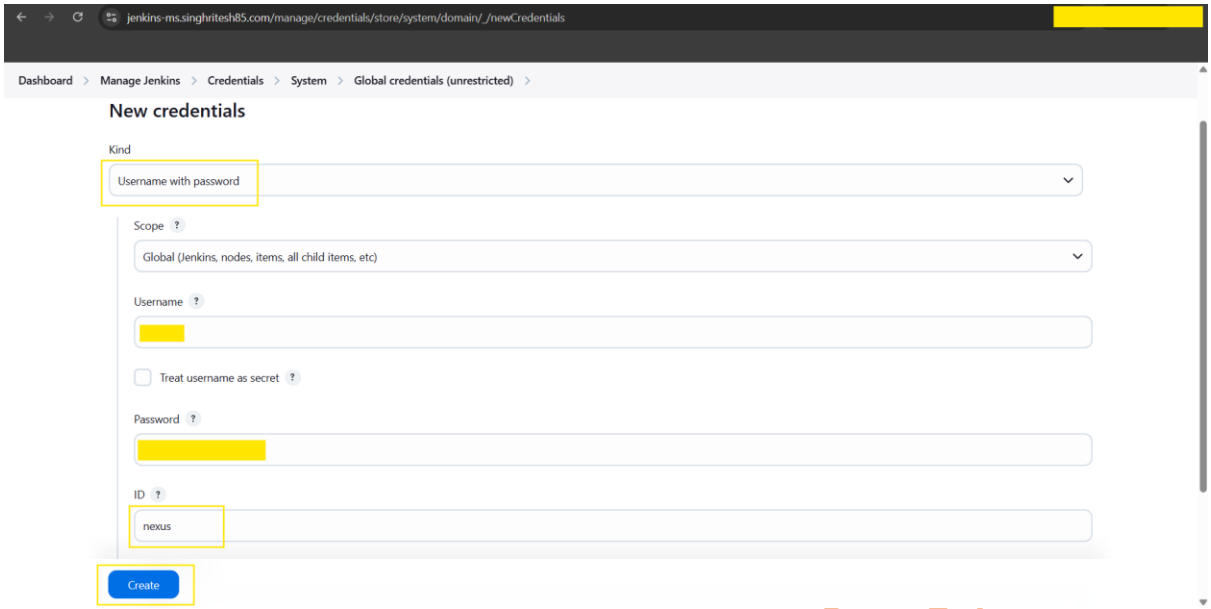

jenkins-ms.singhritesh85.com/manage/computer/createItem
Dashboard
Manage Jenkins
Nodes
Labels
Slave-1
Usage
Use this node as much as possible
Launch method
Launch agents via SSH
Host
Private IP Address of Jenkins Slave
Credentials
Add
Host Key Verification Strategy
Save



Finally, the Slave-1 is in Online mode. To **integrate Jenkins with Sonatype Nexus3** and **to use it in Jenkins Pipeline** I installed the **Nexus Artifact Uploader** and **Pipeline Utility Step** Plugin which is as shown in the screenshot attached below. **Do not restart the Jenkins** after installation of these two plugins.



I had created a credential in Jenkins for Sonatype Nexus3 as shown in the screenshot attached below.



I had provided restricted access to the deployment user **jenkins** using Service Account, Role and Role Binding as shown in the screenshot attached below. The deployment user had all the accesses in the namespaces **bankapp** and **mysql** but does not had access for the entire EKS or AKS cluster. That means deployment user **jenkins** access was restricted to the namespaces **bankapp** and **mysql** in the EKS and AKS Cluster.

```
[root@                ~]# kubectl apply -f sa-role-rolebinding-aks.yaml --context=aks-cluster
namespace/bankapp created
serviceaccount/jenkins-bankapp-aks created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding created
[root@                ~]# kubectl apply -f sa-role-rolebinding.yaml --context=eks-demo-cluster-dev
namespace/bankapp created
serviceaccount/jenkins-bankapp created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding created
```

```
[root@                    ~]# cat sa-role-rolebinding.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-bankapp
  namespace: bankapp
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: bankapp
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: bankapp
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: bankapp
  kind: ServiceAccount
  name: jenkins-bankapp
```

```
[root@                    ~]# cat sa-role-rolebinding-aks.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-bankapp-aks
  namespace: bankapp
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: bankapp
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: bankapp
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: bankapp
  kind: ServiceAccount
  name: jenkins-bankapp-aks
```

```
cat sa-role-rolebinding.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-bankapp
  namespace: bankapp
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: bankapp
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: bankapp
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: bankapp
  kind: ServiceAccount
  name: jenkins-bankapp
```

```
cat sa-role-rolebinding-aks.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-bankapp-aks
  namespace: bankapp
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: bankapp
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: bankapp
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: bankapp
  kind: ServiceAccount
  name: jenkins-bankapp-aks
```

Created Kubernetes Secrets Which Token was utilized in the kubeconfig file (which was shared with the deployment user jenkins) as shown in the screenshot attached below.

cat secrets.yaml

apiVersion: v1

kind: Secret

type: kubernetes.io/service-account-token

metadata:

name: mysecretname

namespace: bankapp

annotations:

kubernetes.io/service-account.name: jenkins-bankapp

cat secrets-aks.yaml

apiVersion: v1

kind: Secret

type: kubernetes.io/service-account-token

metadata:

name: mysecretname-aks

namespace: bankapp

annotations:

kubernetes.io/service-account.name: jenkins-bankapp-aks

```
[root@[redacted] ~]# kubectl describe secrets mysecretname -n bankapp --context=eks-demo-cluster-dev
Name:         mysecretname
Namespace:    bankapp
Labels:       <none>
Annotations:  kubernetes.io/service-account.name: jenkins-bankapp
              kubernetes.io/service-account.uid: [redacted]

Type:  kubernetes.io/service-account-token

Data
====
token:      e[redacted]Q
ca.crt:     1107 bytes
namespace:  7 bytes
```

```
[root@[redacted] ~]# cat secrets.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname
  namespace: bankapp
  annotations:
    kubernetes.io/service-account.name: jenkins-bankapp
[root@[redacted] ~]# cat secrets-aks.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-aks
  namespace: bankapp
  annotations:
    kubernetes.io/service-account.name: jenkins-bankapp-aks
```

```
[root@                    ~]# cat sa-role-rolebinding-mysql.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-mysql
  namespace: mysql
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: mysql
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: mysql
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: mysql
  kind: ServiceAccount
  name: jenkins-mysql
```

```
[root@                    ~]# cat sa-role-rolebinding-mysql-aks.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-mysql-aks
  namespace: mysql
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: mysql
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: mysql
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: mysql
  kind: ServiceAccount
  name: jenkins-mysql-aks
```

```
[root@                ~]# kubectl apply -f sa-role-rolebinding-mysql-aks.yaml --context=aks-cluster
namespace/mysql created
serviceaccount/jenkins-mysql-aks created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding created
[root@                ~]# kubectl apply -f sa-role-rolebinding-mysql.yaml --context=eks-demo-cluster-dev
namespace/mysql created
serviceaccount/jenkins-mysql created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding created
```

```
cat sa-role-rolebinding-mysql.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-mysql
  namespace: mysql
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: mysql
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: mysql
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: mysql
  kind: ServiceAccount
  name: jenkins-mysql
```

```
cat sa-role-rolebinding-mysql-aks.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: jenkins-mysql-aks
  namespace: mysql
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: mysql
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: mysql
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: mysql
  kind: ServiceAccount
  name: jenkins-mysql-aks
```

Created Kubernetes Secrets Which Token was utilized in the kubeconfig file (which was shared with the deployment user jenkins) as shown in the screenshot attached below.

cat secrets-mysql.yaml

apiVersion: v1

kind: Secret

type: kubernetes.io/service-account-token

metadata:

name: mysecretname-mysql

namespace: mysql

annotations:

kubernetes.io/service-account.name: jenkins-mysql

cat secrets-mysql-aks.yaml

apiVersion: v1

kind: Secret

type: kubernetes.io/service-account-token

metadata:

name: mysecretname-mysql-aks

namespace: mysql

annotations:

kubernetes.io/service-account.name: jenkins-mysql-aks

```
[root@                ~]# kubectl apply -f secrets-mysql-aks.yaml --context=aks-cluster
secret/mysecretname-mysql-aks created
[root@                ~]# kubectl apply -f secrets-mysql.yaml --context=eks-demo-cluster-dev
secret/mysecretname-mysql created
```

```
[root@                    ~]# cat secrets-mysql.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-mysql
  namespace: mysql
  annotations:
    kubernetes.io/service-account.name: jenkins-mysql
[root@                    ~]# cat secrets-mysql-aks.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-mysql-aks
  namespace: mysql
  annotations:
    kubernetes.io/service-account.name: jenkins-mysql-aks
```

```
[root@              ~]# kubectl describe secrets mysecretname-mysql-aks -n mysql --context=aks-cluster
Name:         mysecretname-mysql-aks
Namespace:    mysql
Labels:       <none>
Annotations:  kubernetes.io/service-account.name: jenkins-mysql-aks
              kubernetes.io/service-account.uid:

Type:  kubernetes.io/service-account-token

Data
====
ca.crt:     1765 bytes
namespace:  5 bytes
token:      e
A
```

```
[root@              ~]# kubectl describe secrets mysecretname-mysql -n mysql --context=eks-demo-cluster-dev
Name:         mysecretname-mysql
Namespace:    mysql
Labels:       <none>
Annotations:  kubernetes.io/service-account.name: jenkins-mysql
              kubernetes.io/service-account.uid:

Type:  kubernetes.io/service-account-token

Data
====
ca.crt:     1107 bytes
namespace:  5 bytes
token:      e
Q
```

Below is the screenshot of kubeconfig file which I shared with the deployment user **jenkins**. The deployment user **jenkins** will create a directory named as **.kube** and will keep the kubeconfig file at this path as shown in the screenshot attached. The 600 permissions had been provided to the file .kube/config using the command **chmod 600 ~/.kube/config**.

```
cat ~/.kube/config

apiVersion: v1

clusters:

- cluster:

    certificate-authority-data:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXX

    server: https://aks-cluster-dns-XXXXXXXX.hcp.eastus.azmk8s.io:443

  name: aks-cluster

- cluster:

    certificate-authority-data:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXX

    server: https://XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX.gr7.us-east-2.eks.amazonaws.com

  name: eks-demo-cluster-dev

contexts:

- context:

    cluster: aks-cluster

    user: jenkins-aks

  name: aks-cluster-bankapp

- context:

    cluster: aks-cluster

    user: jenkins-mysql-aks

  name: aks-cluster-mysql

- context:

    cluster: eks-demo-cluster-dev

    user: jenkins

  name: eks-demo-cluster-dev-bankapp

- context:

    cluster: eks-demo-cluster-dev

    user: jenkins-mysql

  name: eks-demo-cluster-dev-mysql

current-context: eks-demo-cluster-dev-bankapp

kind: Config
```

preferences: {}

users:

- name: jenkins

  user:

    token: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

- name: jenkins-mysql

  user:

    token: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

- name: jenkins-aks

  user:

    token: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

- name: jenkins-mysql-aks

  user:

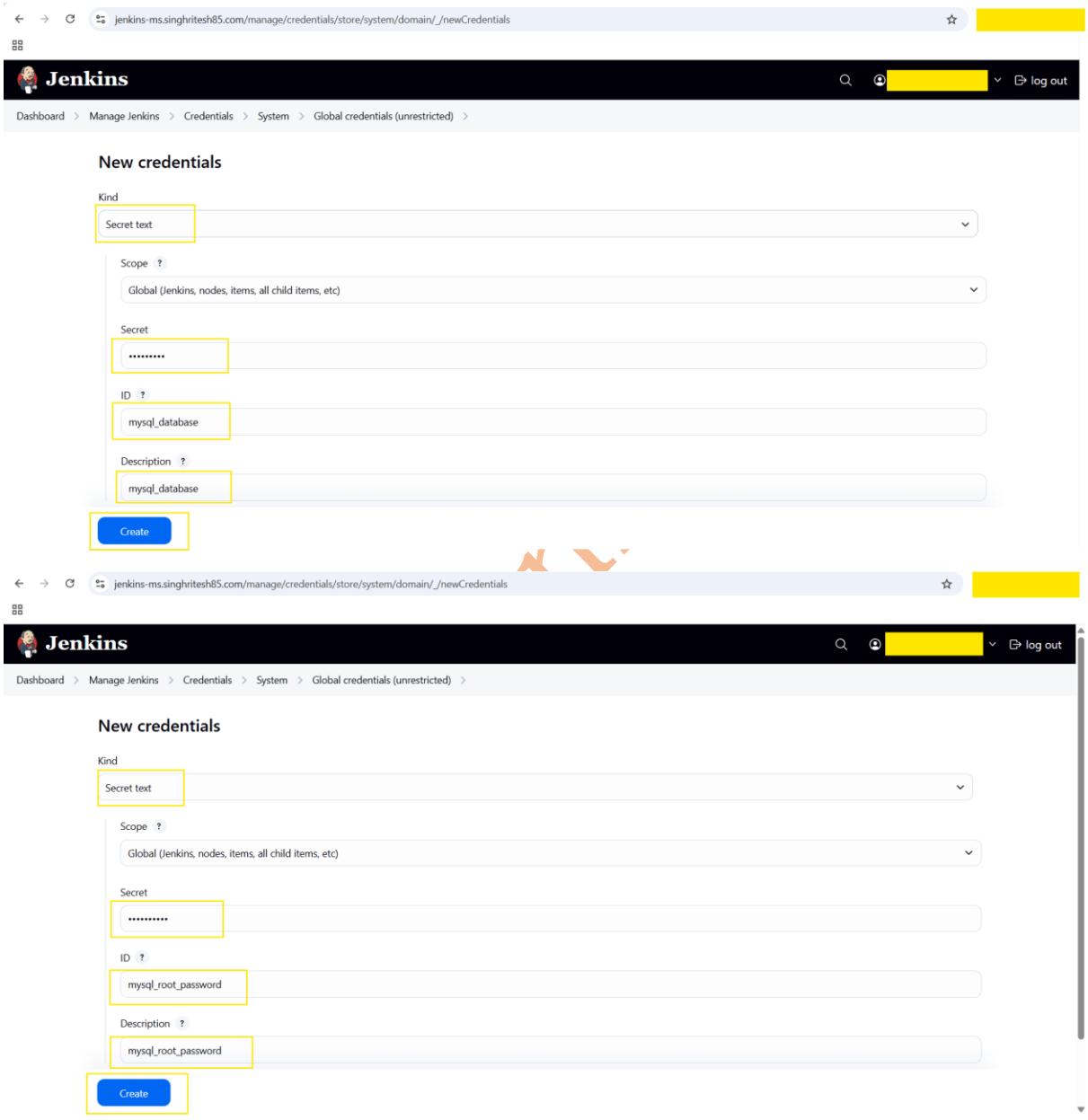
    token: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

After getting the kubeconfig file the deployment user **jenkins** checked its access in EKS and AKS Cluster and found that they have only the access inside the **bankapp** and **mysql** namespaces and not for the entire EKS and AKS Cluster as shown in the screenshot attached below.

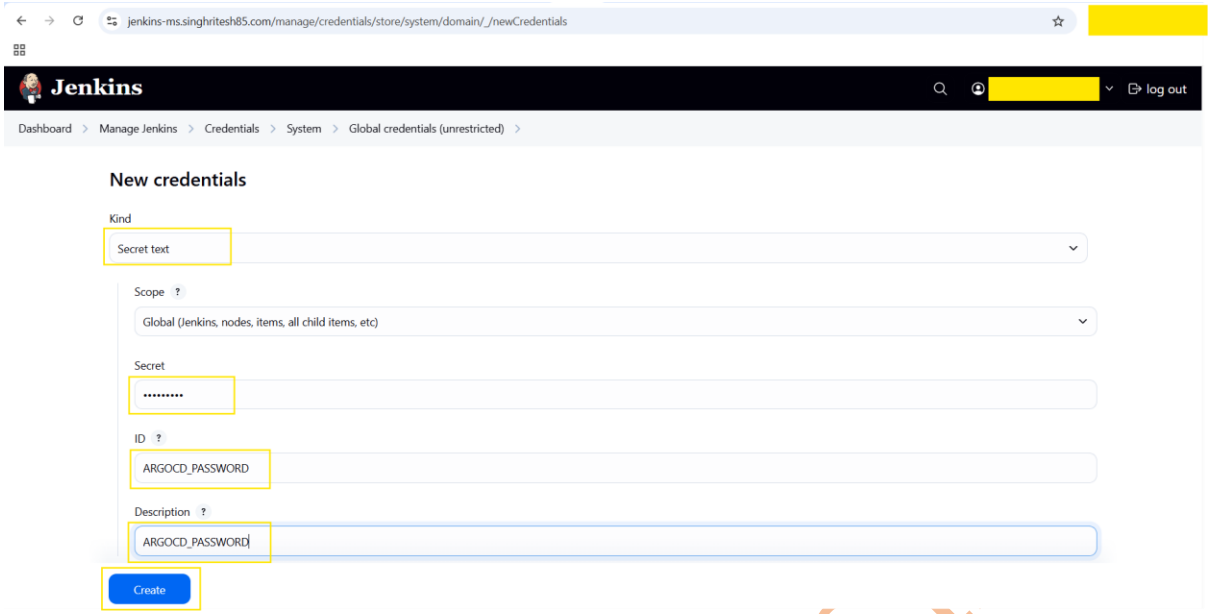
```
[jenkins@ ~]$ kubectl config get-contexts
CURRENT   NAME                          CLUSTER                AUTHINFO            NAMESPACE
          aks-cluster-bankapp           aks-cluster            jenkins-aks
          aks-cluster-mysql             aks-cluster            jenkins-mysql-aks
*         eks-demo-cluster-dev-bankapp  eks-demo-cluster-dev   jenkins
          eks-demo-cluster-dev-mysql    eks-demo-cluster-dev   jenkins-mysql
[jenkins@ ~]$ kubectl get nodes --context=aks-cluster-bankapp
Error from server (Forbidden): nodes is forbidden: User "system:serviceaccount:bankapp:jenkins-bankapp-aks" cannot list resource "nodes" in API group "" at th
e cluster scope
[jenkins@ ~]$ kubectl get nodes --context=aks-cluster-mysql
Error from server (Forbidden): nodes is forbidden: User "system:serviceaccount:mysql:jenkins-mysql-aks" cannot list resource "nodes" in API group "" at the cl
uster scope
[jenkins@ ~]$ kubectl get nodes --context=eks-demo-cluster-dev-bankapp
Error from server (Forbidden): nodes is forbidden: User "system:serviceaccount:bankapp:jenkins-bankapp" cannot list resource "nodes" in API group "" at the cl
uster scope
[jenkins@ ~]$ kubectl get nodes --context=eks-demo-cluster-dev-mysql
Error from server (Forbidden): nodes is forbidden: User "system:serviceaccount:mysql:jenkins-mysql" cannot list resource "nodes" in API group "" at the cluste
r scope
[jenkins@ ~]$ kubectl get all -n bankapp --context=aks-cluster-bankapp
No resources found in bankapp namespace.
[jenkins@ ~]$ kubectl get all -n mysql --context=aks-cluster-mysql
No resources found in mysql namespace.
[jenkins@ ~]$ kubectl get all -n bankapp --context=eks-demo-cluster-dev-bankapp
No resources found in bankapp namespace.
[jenkins@ ~]$ kubectl get all -n mysql --context=eks-demo-cluster-dev-mysql
No resources found in mysql namespace.
```

Creation of multibranch pipeline in Jenkins

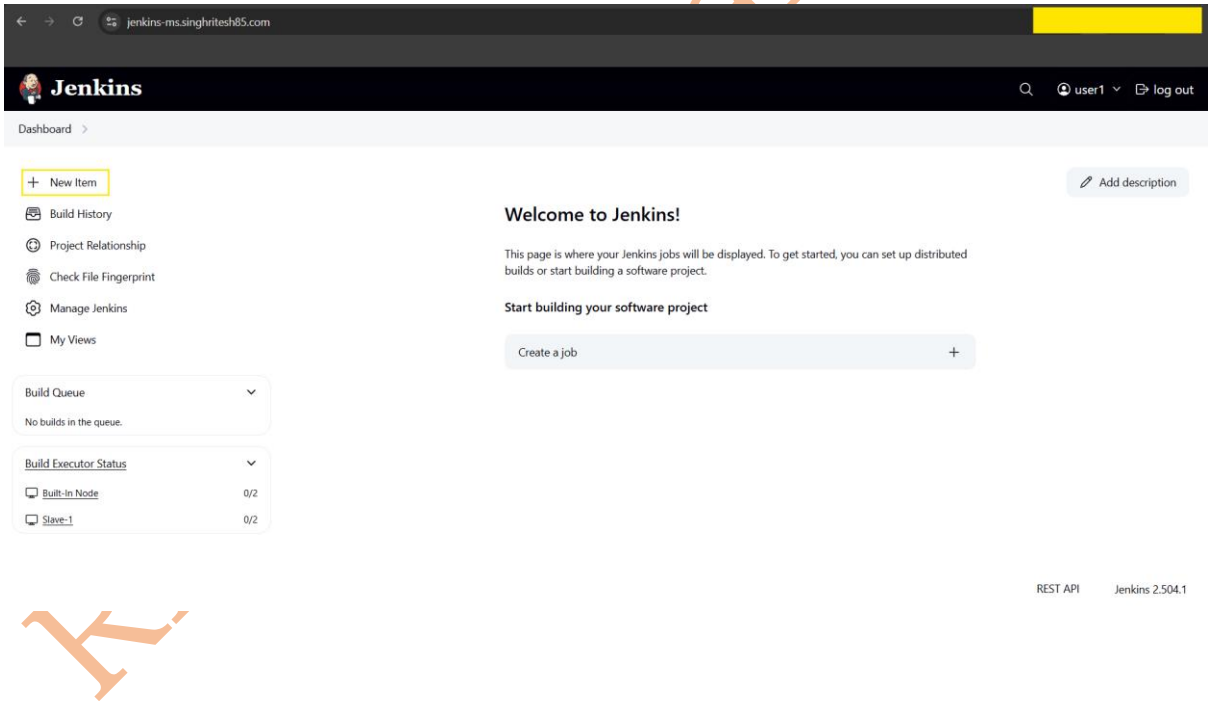
Before creation of multibranch pipeline I created Jenkins Credentails to store the MYSQL_DATABASE name and MYSQL_ROOT_PASSWORD which I will be utilized in the Jenkinsfile. As it is not a good practice to hardcode the password or sensitive information in the Jenkinsfile So I had created the Jenkins Credentials. Below screenshot shows how I had created the Jenkins Credentails of kind secret text.

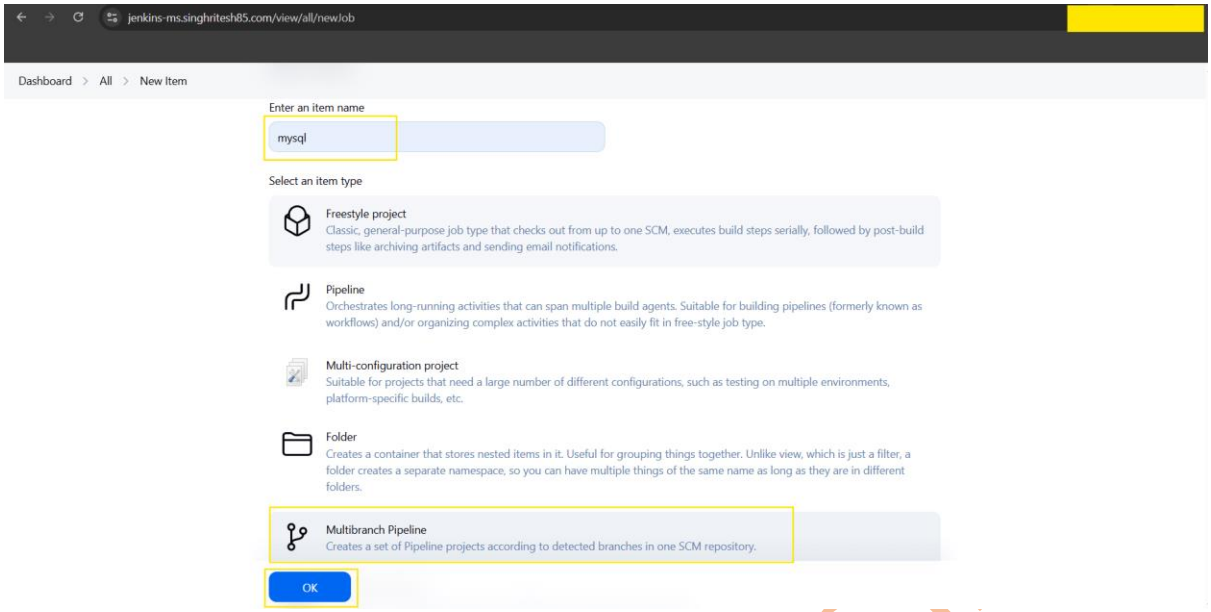


I had created the Jenkins credentials for ArgoCD Password as show shown in the screenshot attached below which I had utilized in the Jenkinsfile.

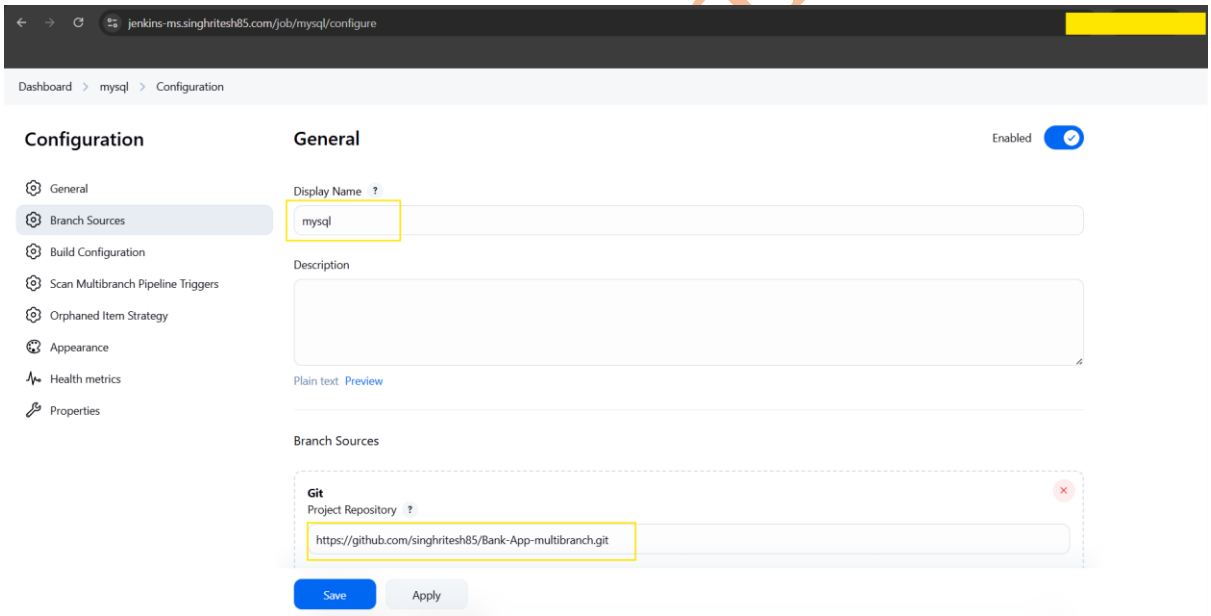


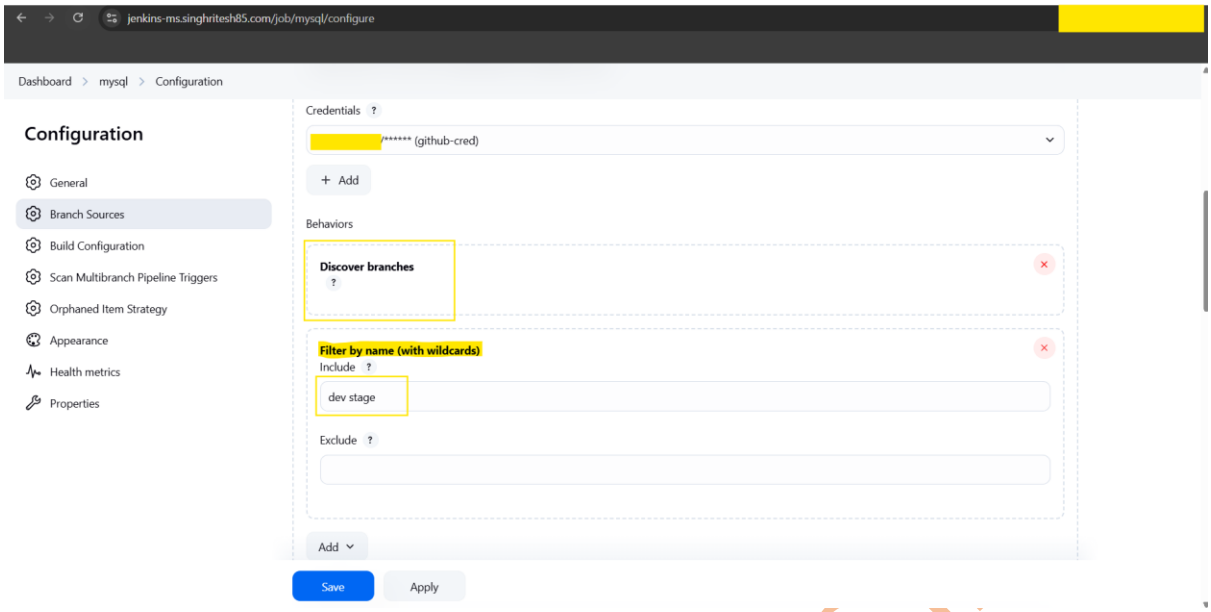
Then Created a multi branch pipeline in Jenkins for dev and stage branch as shown in the screenshot attached below.

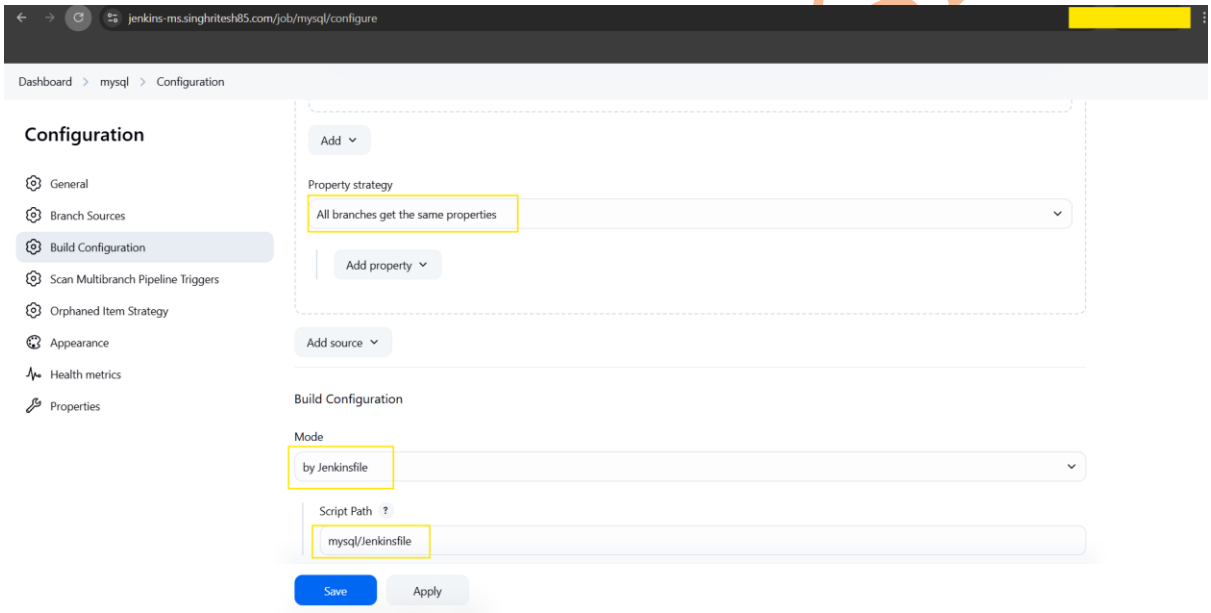




Then first, I created the multibranch pipeline for MySQL Pods as shown in the screenshot attached below.





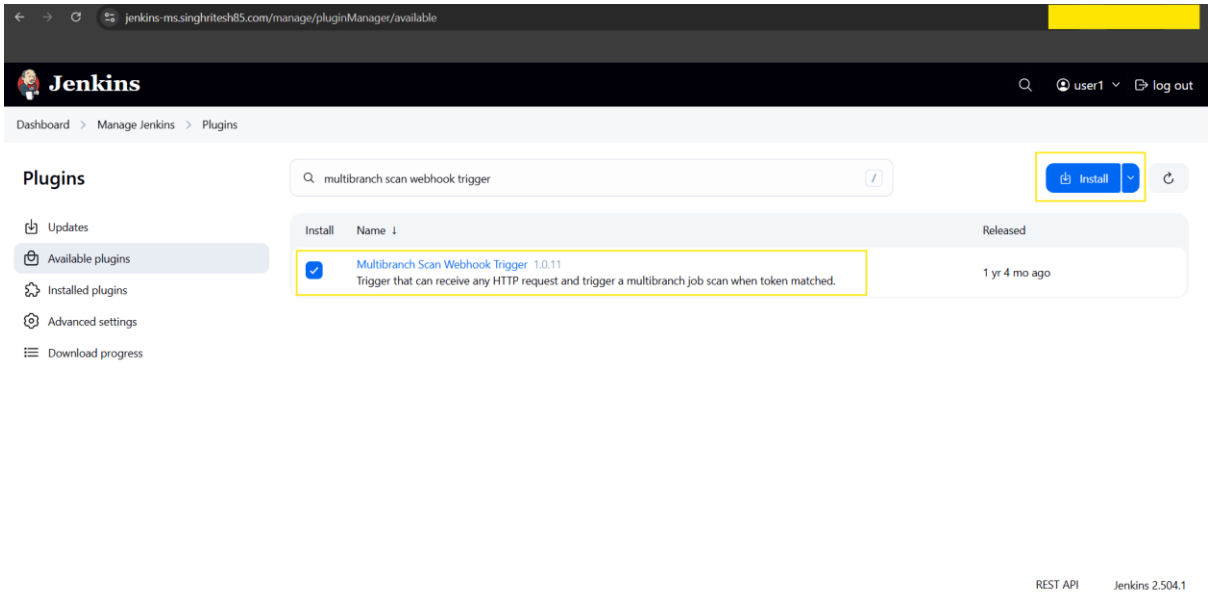


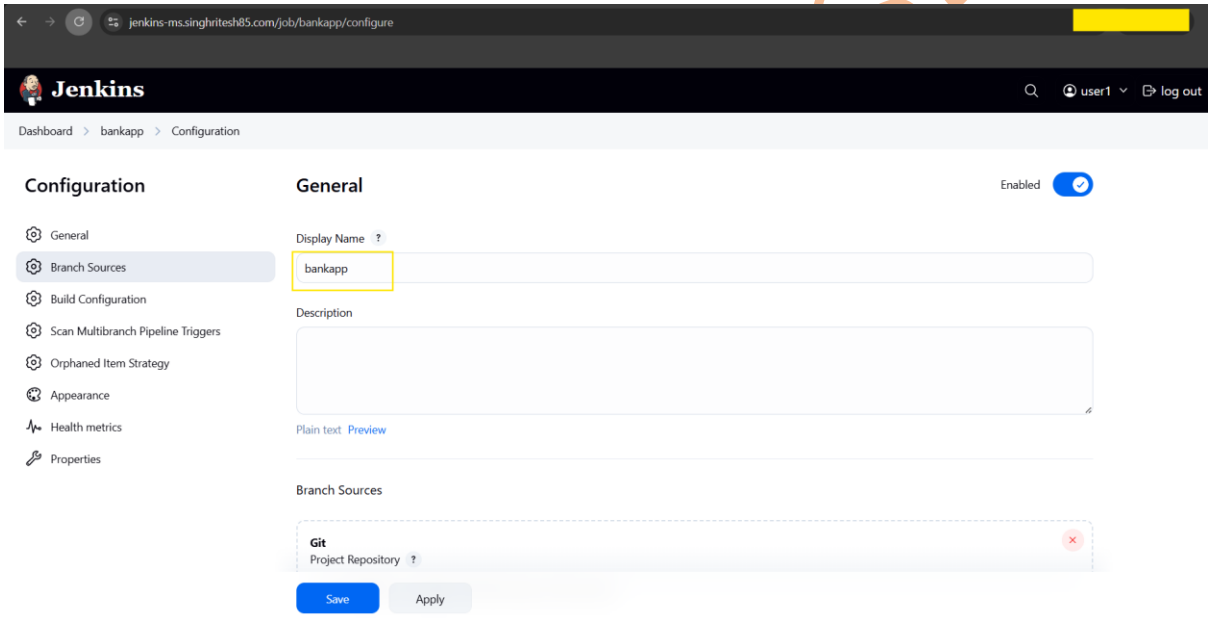
Before creating the pipeline for bankapp you need to create a kubernetes secrets to pull docker image from private ECR (Elastic Container Registry) Repository on AKS (Azure Kubernetes Services).

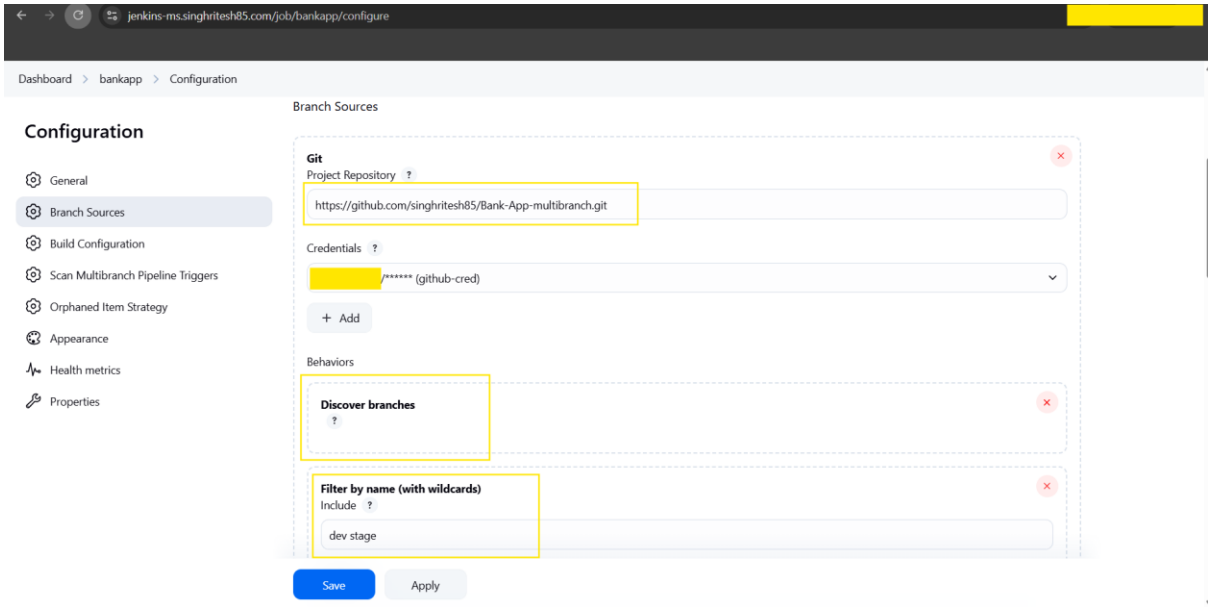
kubectl create secret docker-registry regcred --docker-server=02XXXXXXXXX6.dkr.ecr.us-east-2.amazonaws.com --docker-username=AWS --docker-password=$(aws ecr get-login-password) --namespace=bankapp --context=aks-cluster

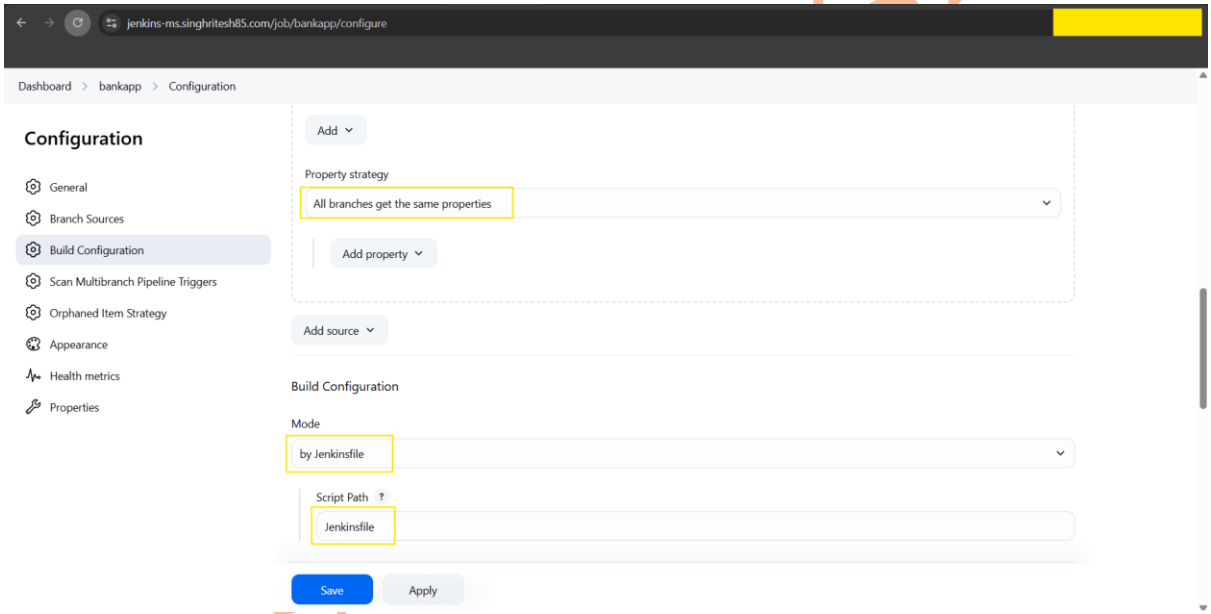
```
[root@                ~]# kubectl create secret docker-registry regcred --docker-server=02        6.dkr.ecr.us-east-2.amazonaws.com --docker-username=AWS --
docker-password=$(aws ecr get-login-password) --namespace=bankapp --context=aks-cluster
secret/regcred created
```

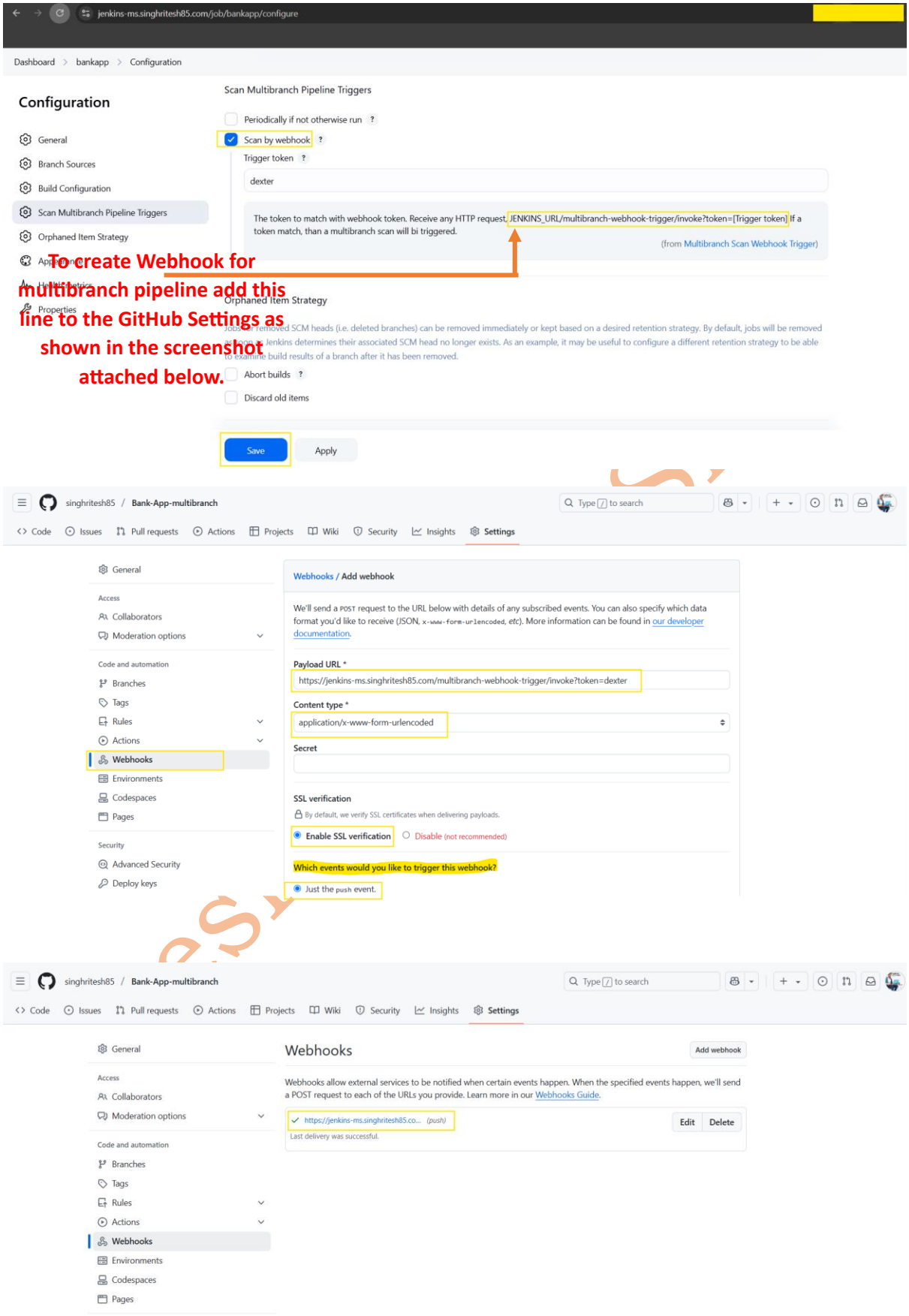
To create a Webhook you need to install the plugin **multibranch scan webhook trigger** as shown in the screenshot attached below.


jenkins-ms.singhritesh85.com/manage/pluginManager/available
Jenkins
user1
log out
Dashboard > Manage Jenkins > Plugins
Plugins
multibranch scan webhook trigger
Install
Updates
Available plugins
Installed plugins
Advanced settings
Download progress
Install
Name ↓
Released
Multibranch Scan Webhook Trigger 1.0.11
Trigger that can receive any HTTP request and trigger a multibranch job scan when token matched.
1 yr 4 mo ago
REST API
Jenkins 2.504.1


jenkins-ms.singhritesh85.com/job/bankapp/configure
Jenkins
user1
log out
Dashboard > bankapp > Configuration
Configuration
General
Branch Sources
Build Configuration
Scan Multibranch Pipeline Triggers
Orphaned Item Strategy
Appearance
Health metrics
Properties
General
Enabled
Display Name
bankapp
Description
Plain text Preview
Branch Sources
Git
Project Repository
Save
Apply

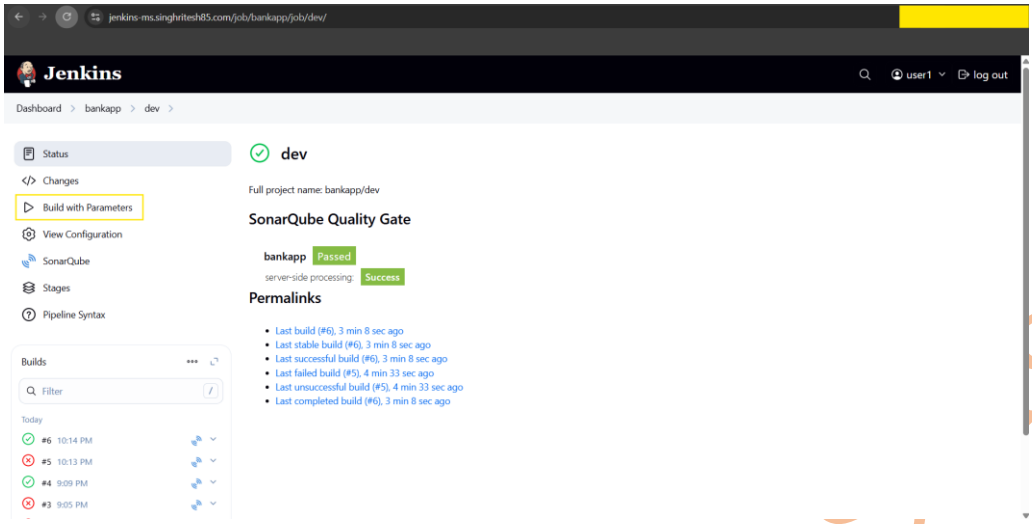

jenkins-ms.singhritesh85.com/job/bankapp/configure
Dashboard > bankapp > Configuration
Configuration
General
Branch Sources
Build Configuration
Scan Multibranch Pipeline Triggers
Orphaned Item Strategy
Appearance
Health metrics
Properties
Branch Sources
Git
Project Repository
https://github.com/singhritesh85/Bank-App-multibranch.git
Credentials
/****** (github-cred)
+ Add
Behaviors
Discover branches
Filter by name (with wildcards)
Include
dev stage
Save
Apply

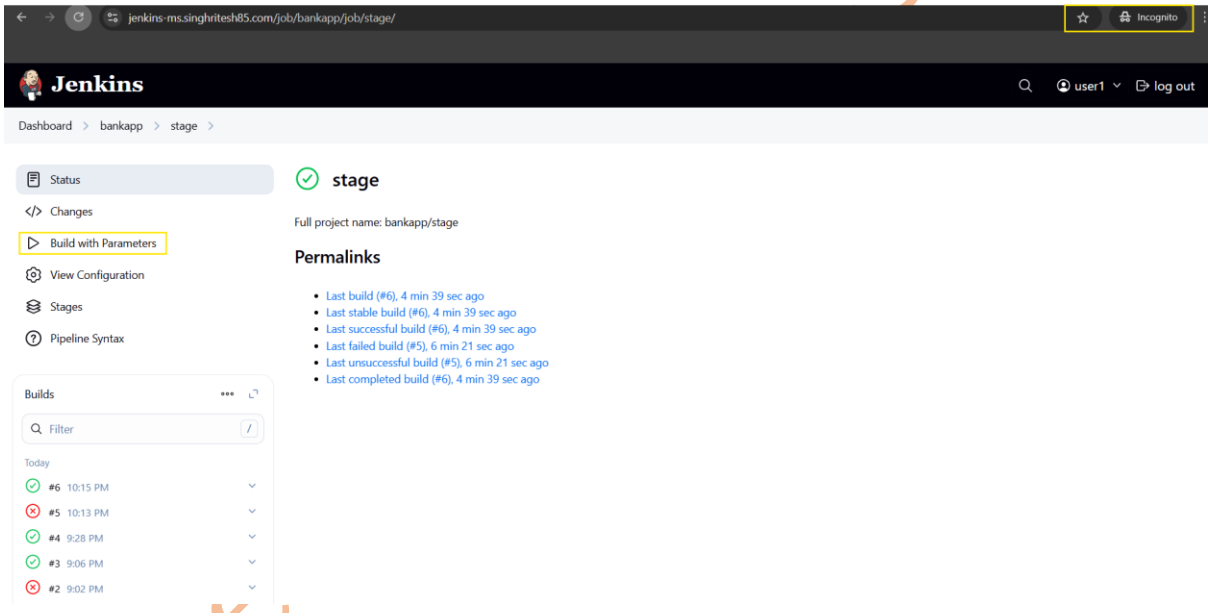

jenkins-ms.singhritesh85.com/job/bankapp/configure
Dashboard > bankapp > Configuration
Configuration
General
Branch Sources
Build Configuration
Scan Multibranch Pipeline Triggers
Orphaned Item Strategy
Appearance
Health metrics
Properties
Add
Property strategy
All branches get the same properties
Add property
Add source
Build Configuration
Mode
by Jenkinsfile
Script Path
Jenkinsfile
Save
Apply

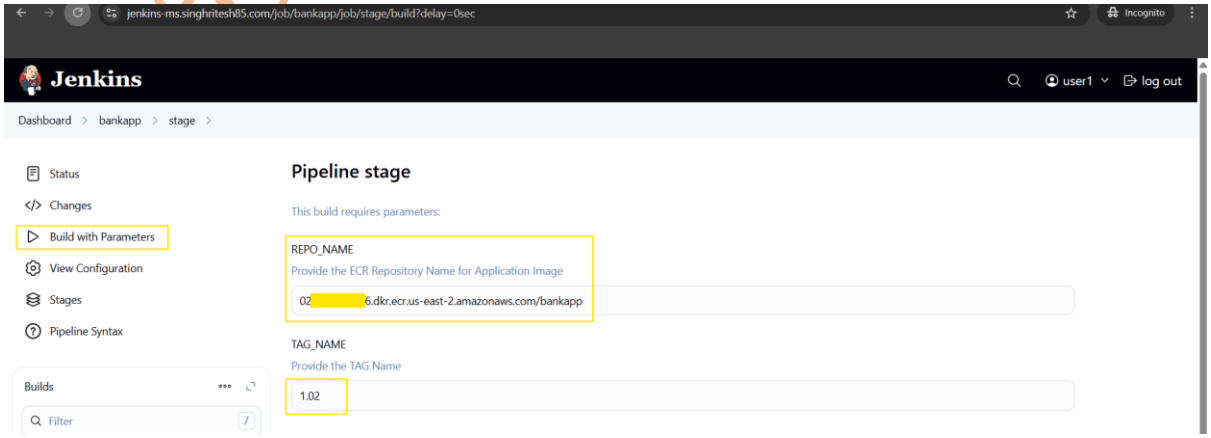


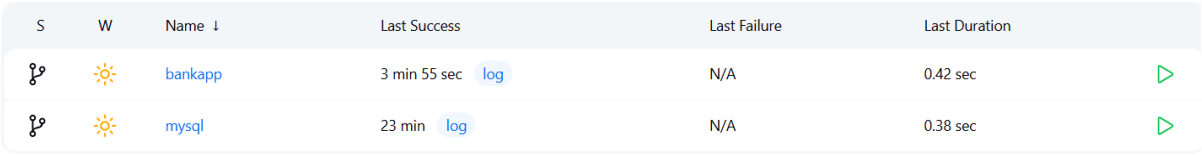
In this project for the demonstration purpose, I had configured webhook however to run the Jenkins Job **bankapp/dev** and **bankapp/stage** I need to run those Jenkins Job manually because I was

supposed to pass the string parameter REPO_NAME and TAG_NAME to run those Jenkins Jobs. It was mandatory to pass the parameters to run bankapp Jenkins multibranch pipeline to dev and stage pipeline as shown in the screenshot attached below.









S	W	Name ↓	Last Success	Last Failure	Last Duration
		bankapp	3 min 55 sec log	N/A	0.42 sec
		mysql	23 min log	N/A	0.38 sec

GitHub Webhook, trigger I had applied to only multibranch Jenkins Job bankapp but not to the MySQL Jenkin pods.

```
[root@                ~]# kubectl get pods -n mysql --context=aks-cluster --watch
NAME                READY   STATUS    RESTARTS      AGE
mysql-primary-0     1/1     Running   0             95m
mysql-secondary-0   1/1     Running   1 (93m ago)   95m
[root@                ~]#
[root@Terraform-Server ~]# kubectl get pods -n bankapp --context=eks-demo-cluster-dev --watch
NAME                          READY   STATUS    RESTARTS   AGE
bankapp-                      1/1     Running   0          75m
```

```
[root@                ~]# kubectl get pods -n mysql --context=aks-cluster --watch
NAME                READY   STATUS    RESTARTS      AGE
mysql-primary-0     1/1     Running   0             96m
mysql-secondary-0   1/1     Running   1 (94m ago)   96m
^C[root@                ~]#
[root@                ~]# kubectl get pods -n bankapp --context=eks-demo-cluster-dev --watch
NAME                          READY   STATUS    RESTARTS   AGE
bankapp-folo-                 1/1     Running   0          76m
[root@                ~]#
```

To access the Bank Application, I created the ingress rule using the yaml manifests file as shown in the screenshot attached below.

```
[root@                ~]# cat ingress-rule-dev.yaml
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: bankapp-ingress-dev
  namespace: bankapp
  annotations:
    kubernetes.io/ingress.class: nginx
spec:
  ingressClassName: nginx
  rules:
  - host: bankapp-dev.singhritesh85.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: bankapp-folo
            port:
              number: 80
[root@                ~]# kubectl apply -f ingress-rule-dev.yaml --context=eks-demo-cluster-dev
```

```
[root@                ~]# cat ingress-rule-stage.yaml
# kubectl create secret tls ingress-tls --key mykey.key --cert STAR_singhritesh85_com.crt --namespace bankapp --context=aks-cluster
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: bankapp-ingress-stage
  namespace: bankapp
  annotations:
    kubernetes.io/ingress.class: nginx
spec:
  ingressClassName: nginx
  tls:
  - hosts:
    - bankapp-stage.singhritesh85.com
    secretName: ingress-tls
  rules:
  - host: bankapp-stage.singhritesh85.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: bankapp-folo
            port:
              number: 80
```

```
[root@                ~]# kubectl apply -f ingress-rule-stage.yaml --context=aks-cluster
ingress.networking.k8s.io/bankapp-ingress-stage created
```

```
[root@                ~]# kubectl get ing -A --context=aks-cluster
NAMESPACE   NAME                    CLASS   HOSTS                          ADDRESS         PORTS   AGE
bankapp     bankapp-ingress-stage   nginx   bankapp-stage.singhritesh85.com   20.       .52   80      80s
[root@                ~]# kubectl get ing -A --context=eks-demo-cluster-dev
NAMESPACE   NAME                CLASS   HOSTS                       ADDRESS                                                          PORTS   AGE
argocd      minimal-ingress     nginx   argocd.singhritesh85.com    a                                        5.us-east-2.elb.amazonaws.com   80      7h4
6m
bankapp     bankapp-ingress-dev nginx   bankapp-dev.singhritesh85.com   a                                     6.us-east-2.elb.amazonaws.com   80      117
s
```

I did the entry for HOST with LoadBalancer External IP/DNS Name in the Azure DNS Zone to create the Record Set of A Type and CNAME Type respectively for Stage and Dev Environment as shown in the screenshot attached below.

Add record set

singhritesh85.com

Name

bankapp-stage

.singhritesh85.com

Type

A – IPv4 Address records

Alias record set ⓘ

No

TTL *

1

TTL unit

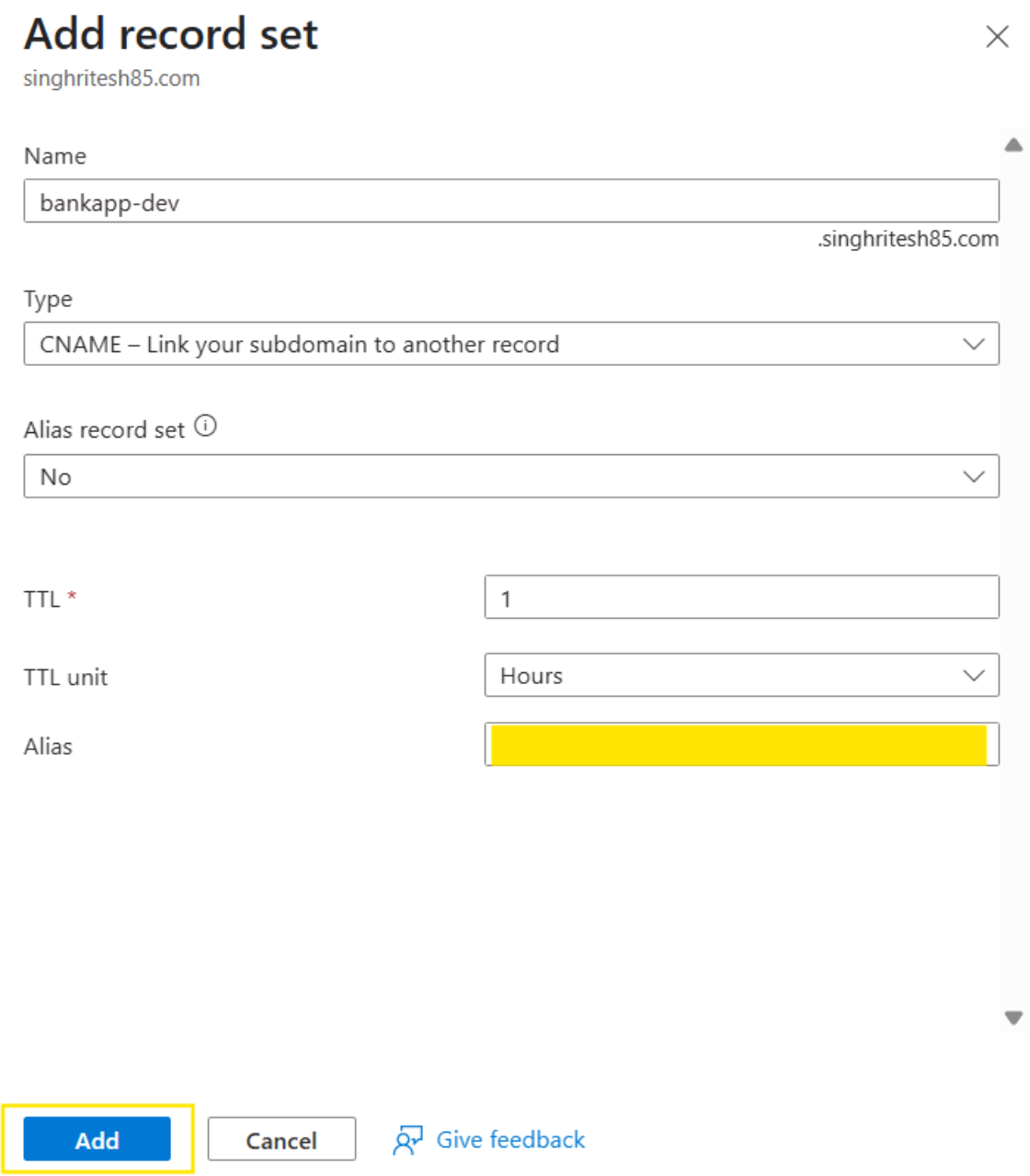
Hours

IP address

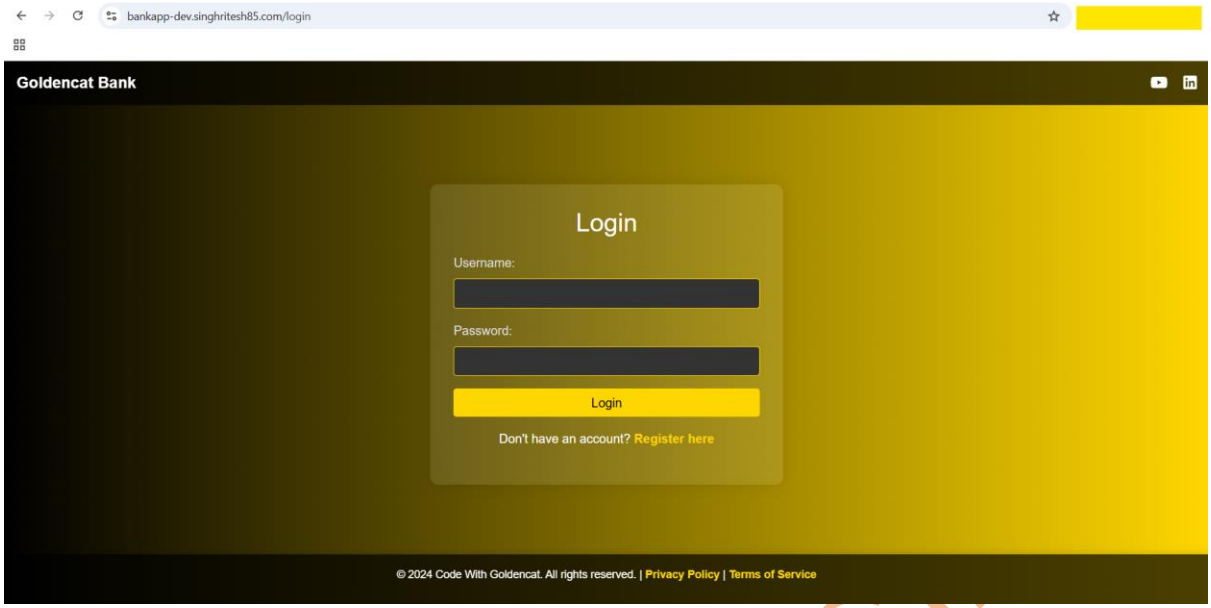
20 .52

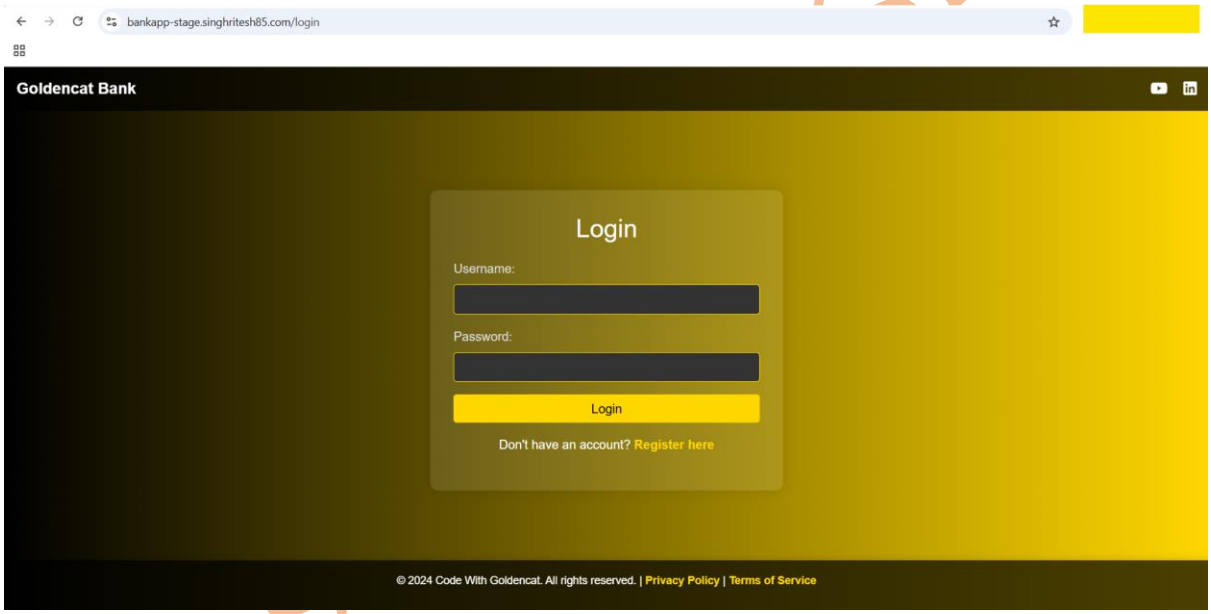
0.0.0.0

Add Cancel Give feedback

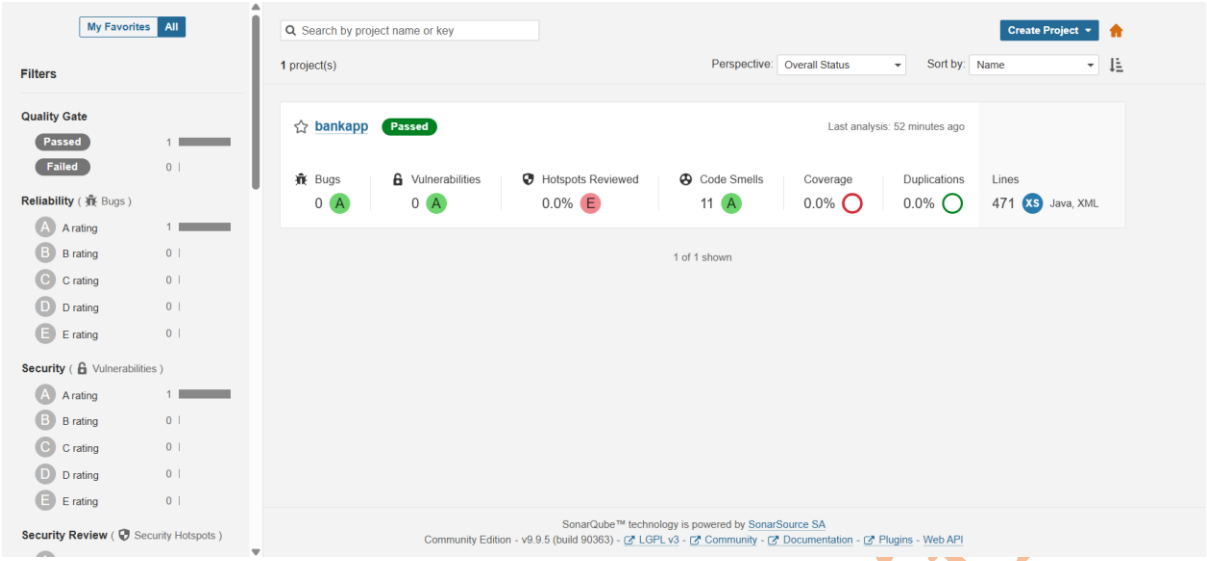


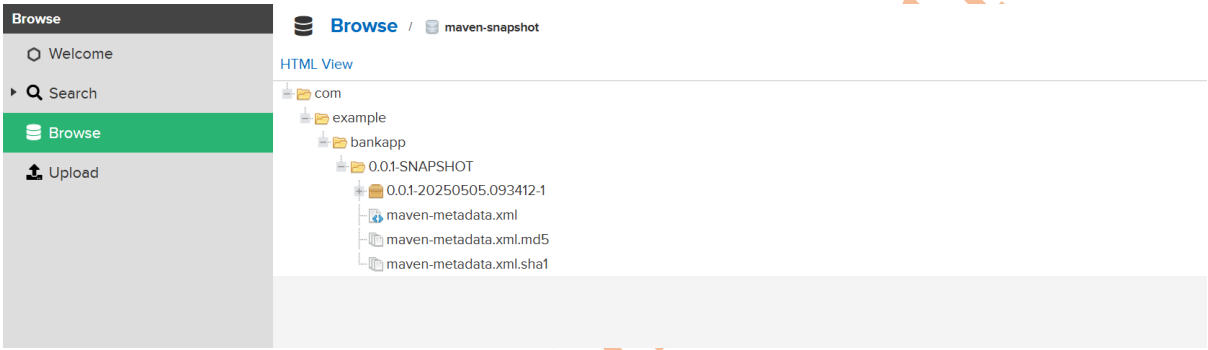
Finally, the Bank Application was accessible through the URL as shown in the screenshot attached below.

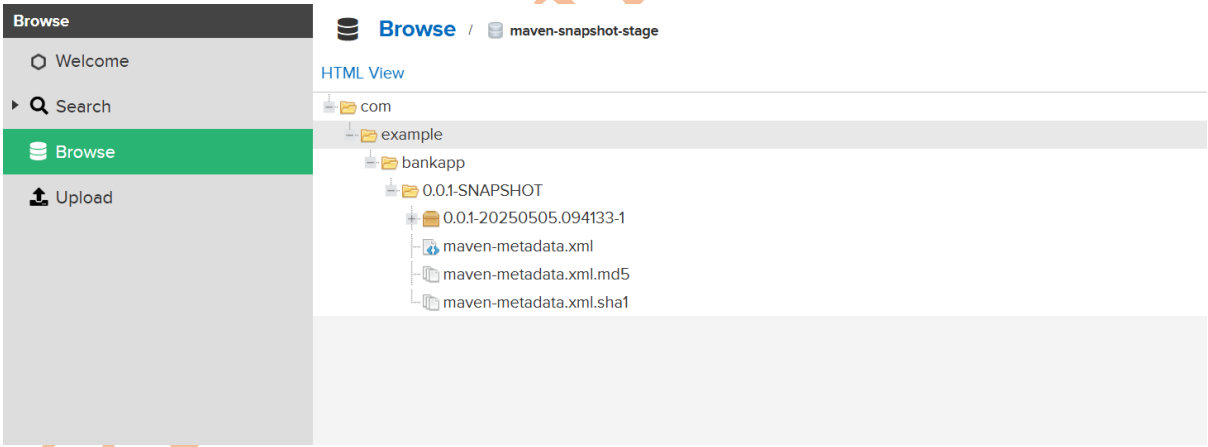


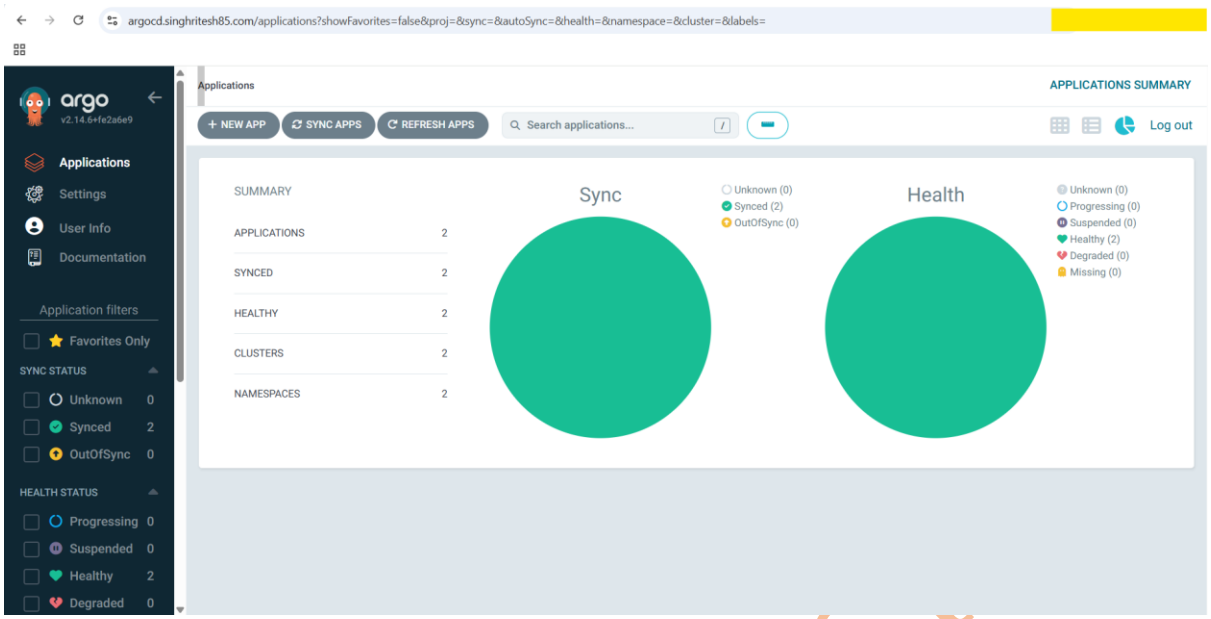


The SonarQube, Nexus Artifactory and Trivy Image Scan had been done in the non-production environment as shown in the Jenkinsfile present in the GitHub Repo https://github.com/singhritesh85/Bank-App-multibranch.git at **dev** and **stage** branch. The screenshot of **SonarQube**, **Nexus Artifactory** and **ArgoCD** after running the two CICD pipeline successfully is as shown in the screenshot attached below.

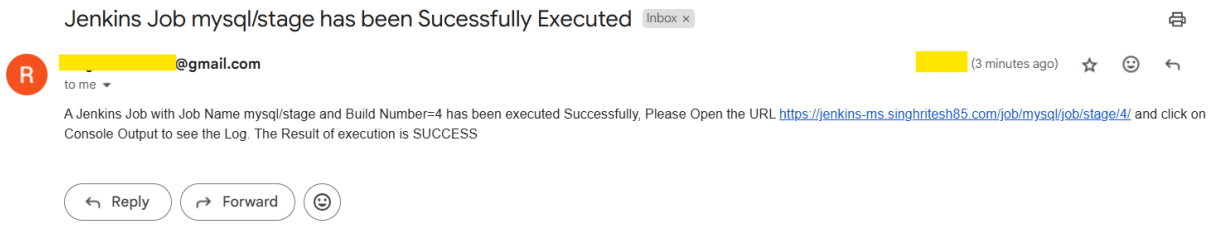

My Favorites
All
Filters
Quality Gate
Passed
Failed
Reliability (Bugs)
Security (Vulnerabilities)
Security Review (Security Hotspots)
Search by project name or key
Create Project
1 project(s)
Perspective: Overall Status
Sort by: Name
bankapp
Passed
Last analysis: 52 minutes ago
Bugs
Vulnerabilities
Hotspots Reviewed
Code Smells
Coverage
Duplications
Lines
0.0%
471
Java, XML
1 of 1 shown
SonarQube™ technology is powered by SonarSource SA
Community Edition - v9.9.5 (build 90363) - LGPL v3 - Community - Documentation - Plugins - Web API

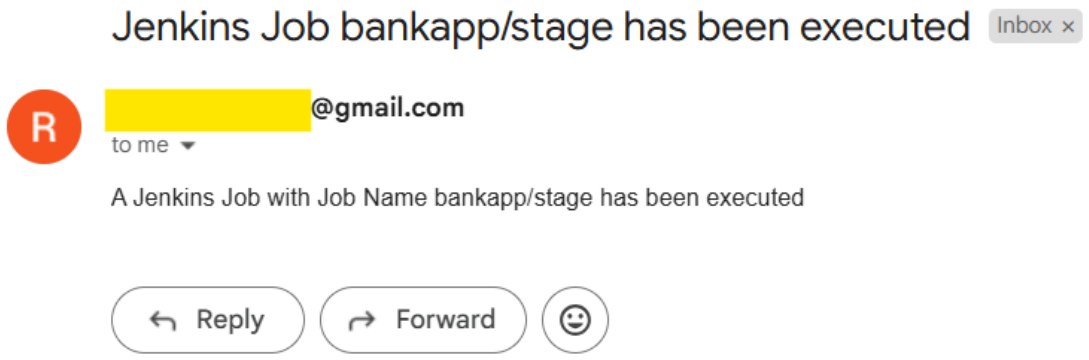

Browse
Welcome
Search
Browse
Upload
Browse / maven-snapshot
HTML View
com
example
bankapp
0.0.1-SNAPSHOT
0.0.1-20250505.093412-1
maven-metadata.xml
maven-metadata.xml.md5
maven-metadata.xml.sha1

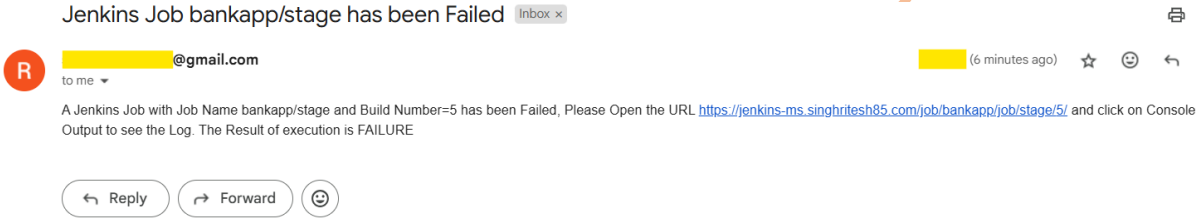

Browse
Welcome
Search
Browse
Upload
Browse / maven-snapshot-stage
HTML View
com
example
bankapp
0.0.1-SNAPSHOT
0.0.1-20250505.094133-1
maven-metadata.xml
maven-metadata.xml.md5
maven-metadata.xml.sha1

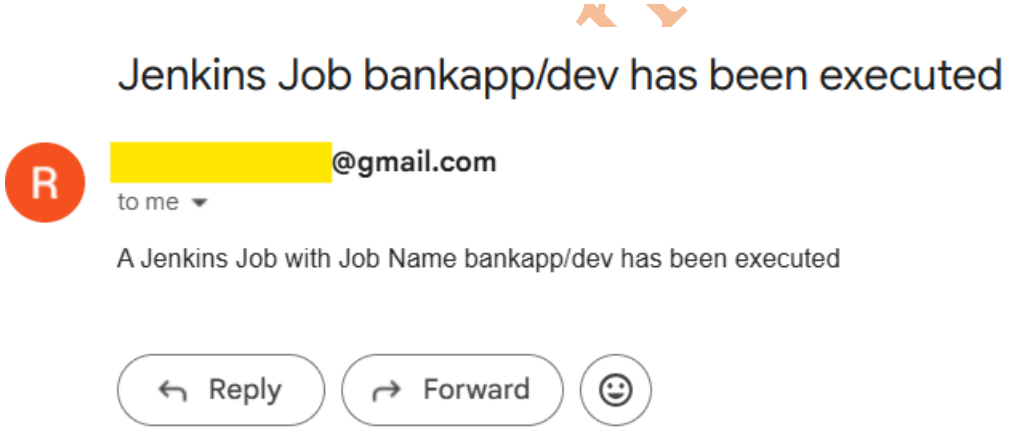


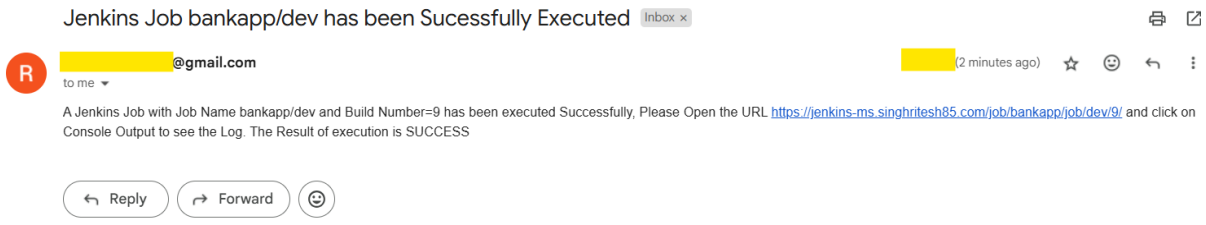
After Execution of Jenkins Job, I received the below notification on the group email Id which I configure for Jenkins.


Jenkins Job mysql/stage has been Sucessfully Executed
Inbox ×
@gmail.com
(3 minutes ago)
to me
A Jenkins Job with Job Name mysql/stage and Build Number=4 has been executed Successfully, Please Open the URL https://jenkins-ms.singhritesh85.com/job/mysql/job/stage/4/ and click on Console Output to see the Log. The Result of execution is SUCCESS
Reply
Forward


Jenkins Job bankapp/stage has been executed
Inbox ×
@gmail.com
to me
A Jenkins Job with Job Name bankapp/stage has been executed
Reply
Forward


Jenkins Job bankapp/stage has been Failed
Inbox ×
@gmail.com
(6 minutes ago)
to me
A Jenkins Job with Job Name bankapp/stage and Build Number=5 has been Failed, Please Open the URL https://jenkins-ms.singhritesh85.com/job/bankapp/job/stage/5/ and click on Console Output to see the Log. The Result of execution is FAILURE
Reply
Forward


Jenkins Job bankapp/dev has been executed
@gmail.com
to me
A Jenkins Job with Job Name bankapp/dev has been executed
Reply
Forward


Jenkins Job bankapp/dev has been Sucessfully Executed
Inbox ×
@gmail.com
(2 minutes ago)
to me
A Jenkins Job with Job Name bankapp/dev and Build Number=9 has been executed Successfully, Please Open the URL https://jenkins-ms.singhritesh85.com/job/bankapp/job/dev/9/ and click on Console Output to see the Log. The Result of execution is SUCCESS
Reply
Forward

Installation of node-exporter and promtail had been done using the helm chart in the EKS and AKS Cluster as written below.

helm repo add prometheus-community https://*prometheus-community.github.io/helm-charts*

kubectl create ns node-exporter

helm install my-prometheus-node-exporter prometheus-community/prometheus-node-exporter --version 4.37.1 --set service.type=LoadBalancer -n node-exporter

```
[root@ ~]# kubectl config use-context eks-demo-cluster-dev
Switched to context "eks-demo-cluster-dev".
[root@ ~]# helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
"prometheus-community" has been added to your repositories
[root@ ~]# kubectl create ns node-exporter
namespace/node-exporter created
[root@ ~]# helm install my-prometheus-node-exporter prometheus-community/prometheus-node-exporter --version 4.37.1 --set service.type=LoadBala
ncer -n node-exporter
```

```
[root@ ~]# kubectl config use-context aks-cluster
Switched to context "aks-cluster".
[root@ ~]# helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
"prometheus-community" already exists with the same configuration, skipping
[root@ ~]# kubectl create ns node-exporter
namespace/node-exporter created
[root@ ~]# helm install my-prometheus-node-exporter prometheus-community/prometheus-node-exporter --version 4.37.1 --set service.type=LoadBala
ncer -n node-exporter
```

Below screenshot shows the Kubernetes Service which was created for node-exporter using the helm chart.

```
[root@ ~]# kubectl get svc -n node-exporter --context=aks-cluster
NAME                          TYPE           CLUSTER-IP     EXTERNAL-IP      PORT(S)          AGE
my-prometheus-node-exporter   LoadBalancer   10.    .141    135.     .160   9100:31055/TCP   3m3s
[root@ ~]# kubectl get svc -n node-exporter --context=eks-demo-cluster-dev
NAME                          TYPE           CLUSTER-IP     EXTERNAL-IP                                                              PORT(S)          AGE
my-prometheus-node-exporter   LoadBalancer   10.    .137    a                                                6.us-east-2.elb.amazonaws.com   9100:32322/TCP   5m13s
```

Then I had updated the prometheus configuration file **/etc/prometheus/prometheus.yml** as shown in the screenshot attached below.

```
  - job_name: "EKS"
    static_configs:
      - targets: ["a                                                6.us-east-2.elb.amazonaws.com:9100"]
  - job_name: "AKS"
    static_configs:
      - targets: ["135.        .160:9100"]
```

```
[root@ ~]# systemctl restart prometheus.service
[root@ ~]# systemctl status prometheus.service
● prometheus.service - Prometheus
   Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2025-05-05 12:05:04 UTC; 7s ago
```

I installed the promtail using the helm chart as written below. First, I cloned the helm chart present in GitHub Repo.

git clone https://github.com/singhritesh85/helm-chart-promtail.git

After cloning helm chart from GitHub, I updated the values.yaml file of promtail helm chart with Loki Servers Private IP Addresses as shown in the screenshot attached below.

kubectl create ns promtail && helm upgrade --install promtail ./helm-chart-promtail -f ./helm-chart-promtail/values.yaml -n promtail

kubectl get pods -n promtail --watch

```
# -- The log level of the Promtail server
# Must be reference in `config.file` to configure `server.log_level`
# See default config in `values.yaml`
logLevel: info
# -- The log format of the Promtail server
# Must be reference in `config.file` to configure `server.log_format`
# Valid formats: `logfmt, json`
# See default config in `values.yaml`
logFormat: logfmt
# -- The port of the Promtail server
# Must be reference in `config.file` to configure `server.http_listen_port`
# See default config in `values.yaml`
serverPort: 3101
# -- The config of clients of the Promtail server
# Must be reference in `config.file` to configure `clients`
# @default -- See `values.yaml`
clients:
  - url: http://10.    .165:3100/loki/api/v1/push
  - url: http://10.    .14:3100/loki/api/v1/push
  - url: http://10.    .195:3100/loki/api/v1/push

# -- Configures where Promtail will save it's positions file, to resume reading after restarts.
# Must be referenced in `config.file` to configure `positions`
positions:
  filename: /run/promtail/positions.yaml
# -- The config to enable tracing
enableTracing: false
# -- A section of reusable snippets that can be reference in `config.file`.
# Custom snippets may be added in order to reduce redundancy.
```

```
[root@                ~]# git clone https://github.com/singhritesh85/helm-chart-promtail.git
```

```
[root@                ~]# kubectl create ns promtail && helm upgrade --install promtail ./helm-chart-promtail -f ./helm-chart-promtail/values.yaml -n promtai
l
namespace/promtail created
Release "promtail" does not exist. Installing it now.
NAME: promtail
LAST DEPLOYED:                2025
NAMESPACE: promtail
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
**********************************************************************
 Welcome to Grafana Promtail
 Chart version: 6.16.6
 Promtail version: 3.0.0
**********************************************************************

Verify the application is working by running these commands:
* kubectl --namespace promtail port-forward daemonset/promtail 3101
* curl http://127.0.0.1:3101/metrics
[root@Terraform-Server ~]# kubectl config get-contexts
CURRENT   NAME                  CLUSTER               AUTHINFO                                                NAMESPACE
*         aks-cluster           aks-cluster           clusterUser_aks-rg_aks-cluster
          eks-demo-cluster-dev  eks-demo-cluster-dev  arn:aws:eks:us-east-2:02        6:cluster/eks-demo-cluster-dev
```

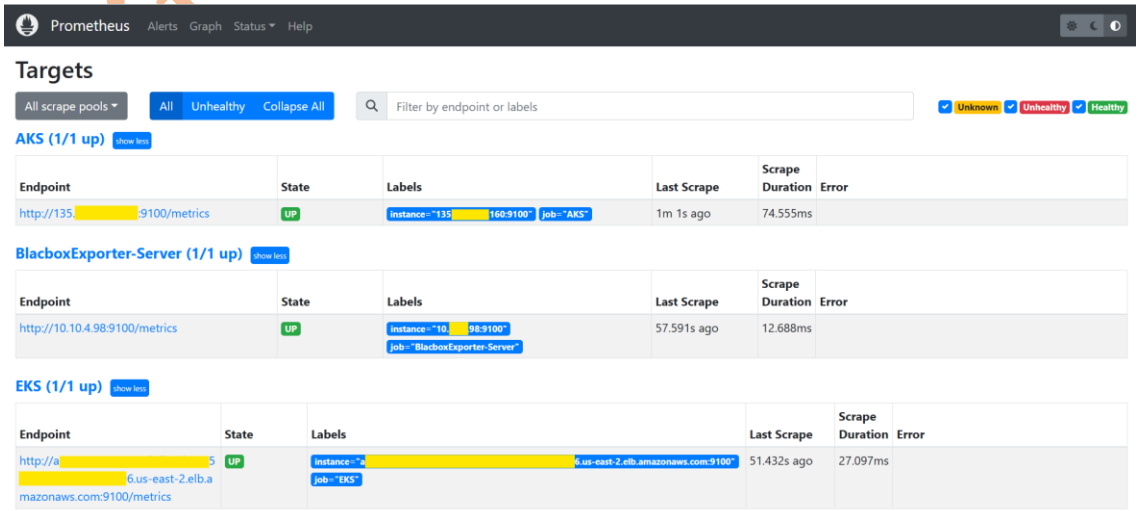
```
[root@ ~]# kubectl config use-context eks-demo-cluster-dev
Switched to context "eks-demo-cluster-dev".
[root@ ~]# kubectl create ns promtail && helm upgrade --install promtail ./helm-chart-promtail -f ./helm-chart-promtail/values.yaml -n promtai
l
namespace/promtail created
Release "promtail" does not exist. Installing it now.
NAME: promtail
LAST DEPLOYED: 2025
NAMESPACE: promtail
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
**********************************************************************
 Welcome to Grafana Promtail
 Chart version: 6.16.6
 Promtail version: 3.0.0
**********************************************************************

Verify the application is working by running these commands:
* kubectl --namespace promtail port-forward daemonset/promtail 3101
* curl http://127.0.0.1:3101/metrics
```

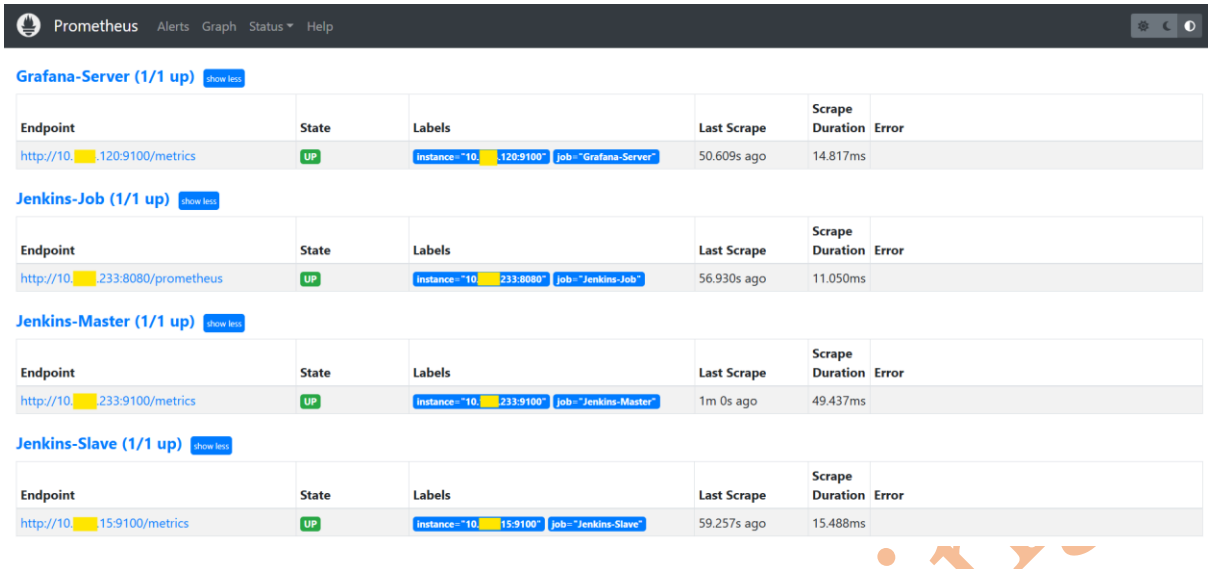
```
[root@ ~]# kubectl get pods -n promtail --context=aks-cluster
NAME              READY   STATUS    RESTARTS   AGE
promtail-         1/1     Running   0          3m3s
promtail-         1/1     Running   0          3m3s
[root@ ~]# kubectl get pods -n promtail --context=eks-demo-cluster-dev
NAME              READY   STATUS    RESTARTS   AGE
promtail-         1/1     Running   0          79s
promtail-         1/1     Running   0          79s
```

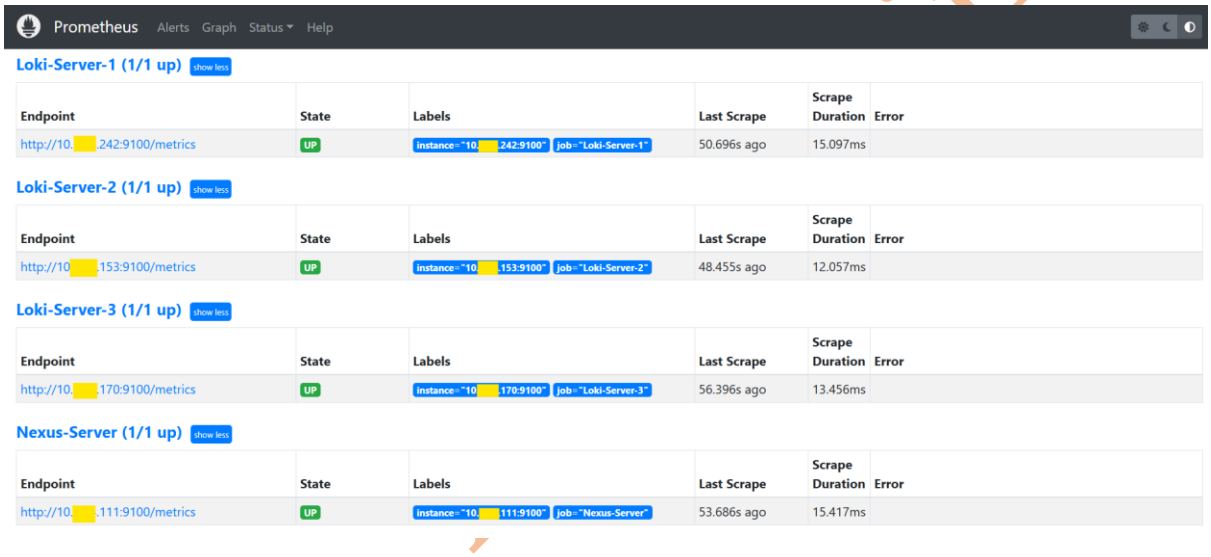
Monitoring Using Prometheus and Grafana and Log Aggregation using Loki

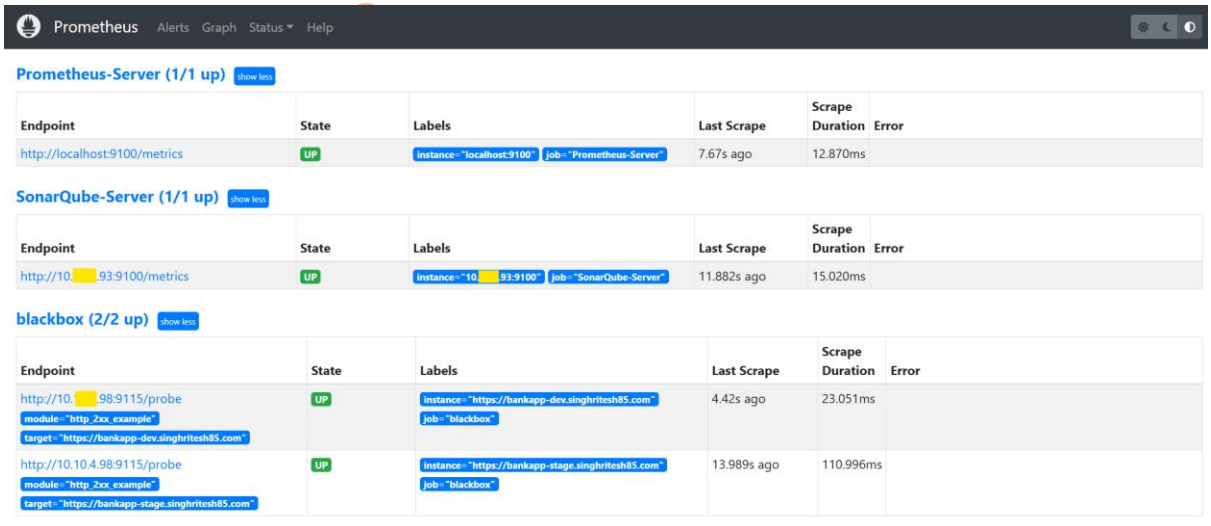
For Monitoring Tool I had used Prometheus and Grafana. To monitor SonarQube I had used SonarQube-Prometheus-Exporter which was installed using terraform at the path **/opt/sonarqube/extensions/plugins**. It was downloaded from the link https://github.com/dmeiners88/sonarqube-prometheus-exporter/releases/download/v1.0.0-SNAPSHOT-2018-07-04/sonar-prometheus-exporter-1.0.0-SNAPSHOT.jar. These steps had been covered in the terraform **user_data_sonarqube.sh**. It is basically a bootstrap script for SonarQube Server. For Monitoring Jenkins, you need to install the plugin **Prometheus metrics** and then restart Jenkins, these steps already been discussed at the starting. The configuration for prometheus had already been done in the terraform. I had taken sonarqube username and password as **admin** and **Admin123** respectively, you can choose as of your own choice and update the terraform script accordingly (Prometheus needs username and password to extract the metrics from SonarQube). I had provided the terraform script with this GitHub Repository. I had already integrated Prometheus and Loki as a data source for Grafana which was already discussed above.

Here I checked the prometheus console and I found all the Targets was UP as shown in the screenshot attached below.




Prometheus Alerts Graph Status Help
Grafana-Server (1/1 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://10.[redacted].120:9100/metrics UP instance="10.[redacted].120:9100" job="Grafana-Server" 50.609s ago 14.817ms
Jenkins-Job (1/1 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://10.[redacted].233:8080/prometheus UP instance="10.[redacted].233:8080" job="Jenkins-Job" 56.930s ago 11.050ms
Jenkins-Master (1/1 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://10.[redacted].233:9100/metrics UP instance="10.[redacted].233:9100" job="Jenkins-Master" 1m 0s ago 49.437ms
Jenkins-Slave (1/1 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://10.[redacted].15:9100/metrics UP instance="10.[redacted].15:9100" job="Jenkins-Slave" 59.257s ago 15.488ms


Prometheus Alerts Graph Status Help
Loki-Server-1 (1/1 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://10.[redacted].242:9100/metrics UP instance="10.[redacted].242:9100" job="Loki-Server-1" 50.696s ago 15.097ms
Loki-Server-2 (1/1 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://10.[redacted].153:9100/metrics UP instance="10.[redacted].153:9100" job="Loki-Server-2" 48.455s ago 12.057ms
Loki-Server-3 (1/1 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://10.[redacted].170:9100/metrics UP instance="10.[redacted].170:9100" job="Loki-Server-3" 56.396s ago 13.456ms
Nexus-Server (1/1 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://10.[redacted].111:9100/metrics UP instance="10.[redacted].111:9100" job="Nexus-Server" 53.686s ago 15.417ms


Prometheus Alerts Graph Status Help
Prometheus-Server (1/1 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://localhost:9100/metrics UP instance="localhost:9100" job="Prometheus-Server" 7.67s ago 12.870ms
SonarQube-Server (1/1 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://10.[redacted].93:9100/metrics UP instance="10.[redacted].93:9100" job="SonarQube-Server" 11.882s ago 15.020ms
blackbox (2/2 up) show less
Endpoint State Labels Last Scrape Scrape Duration Error
http://10.[redacted].98:9115/probe module="http_2xx_example" target="https://bankapp-dev.singhritesh85.com" UP instance="https://bankapp-dev.singhritesh85.com" job="blackbox" 4.42s ago 23.051ms
http://10.10.4.98:9115/probe module="http_2xx_example" target="https://bankapp-stage.singhritesh85.com" UP instance="https://bankapp-stage.singhritesh85.com" job="blackbox" 13.989s ago 110.996ms

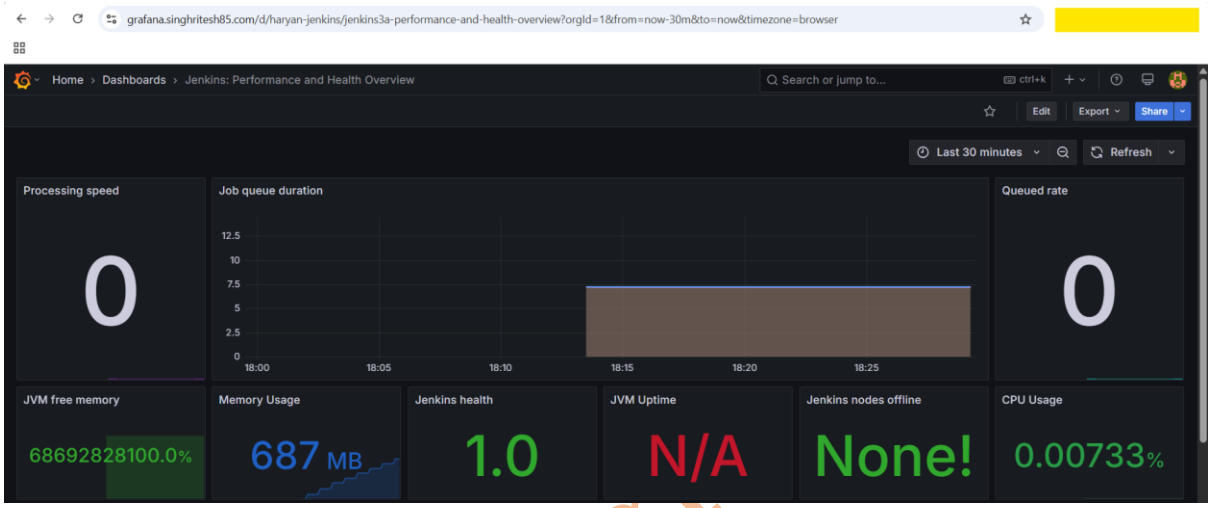
prometheus (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	13.644s ago	6.975ms	

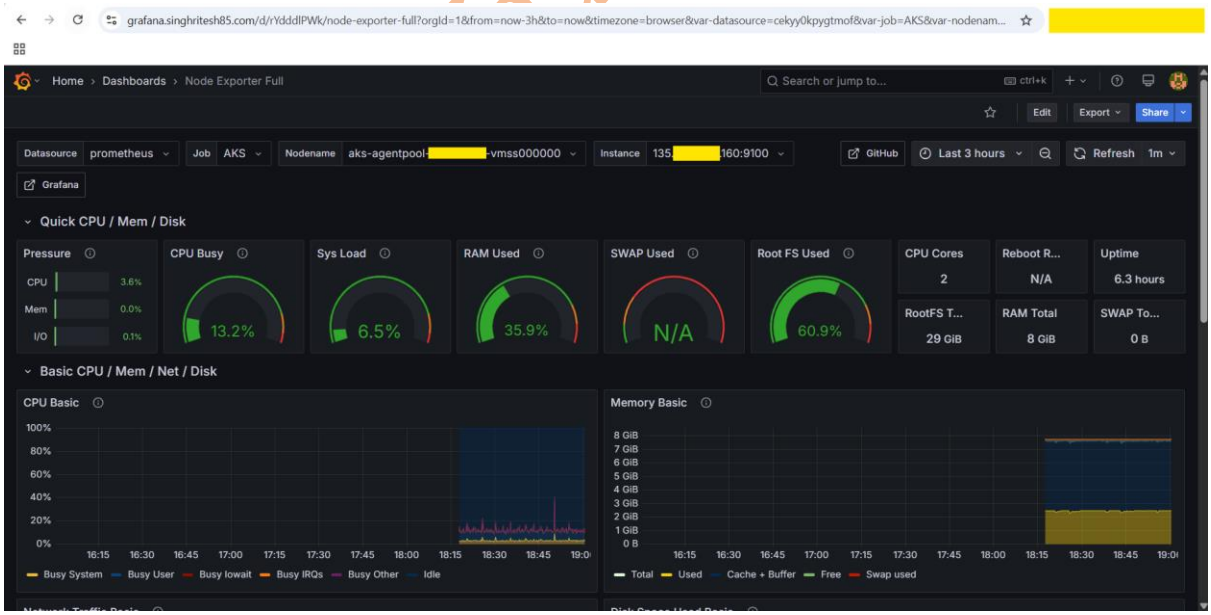
sonarqube (1/1 up) show less

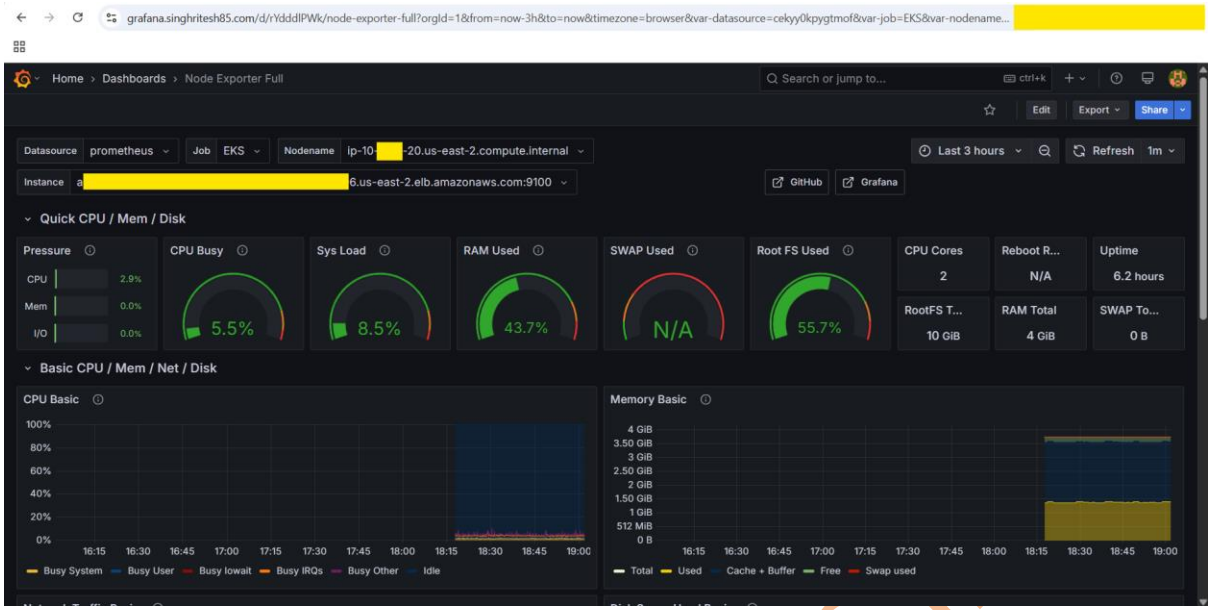
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://10.[redacted]93:9000/api/prometheus/metrics	UP	instance="10.[redacted]93:9000" job="sonarqube"	5.755s ago	163.547ms	

For Monitoring Jenkins Job using Prometheus I created the Grafana Dashboard using the Grafana ID **9964**.

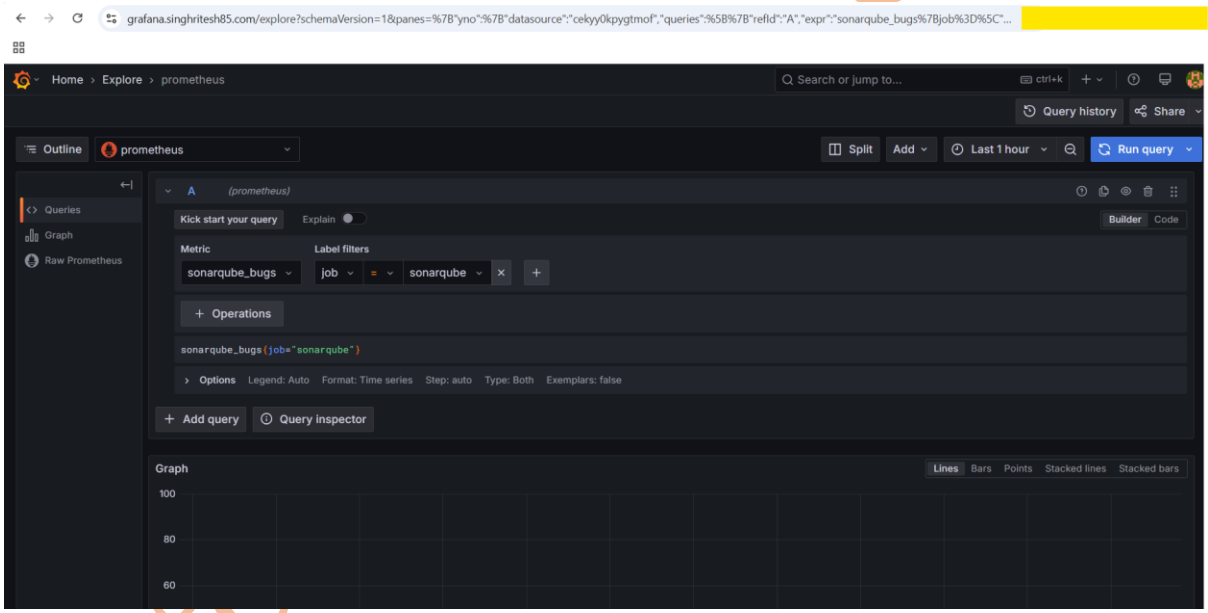


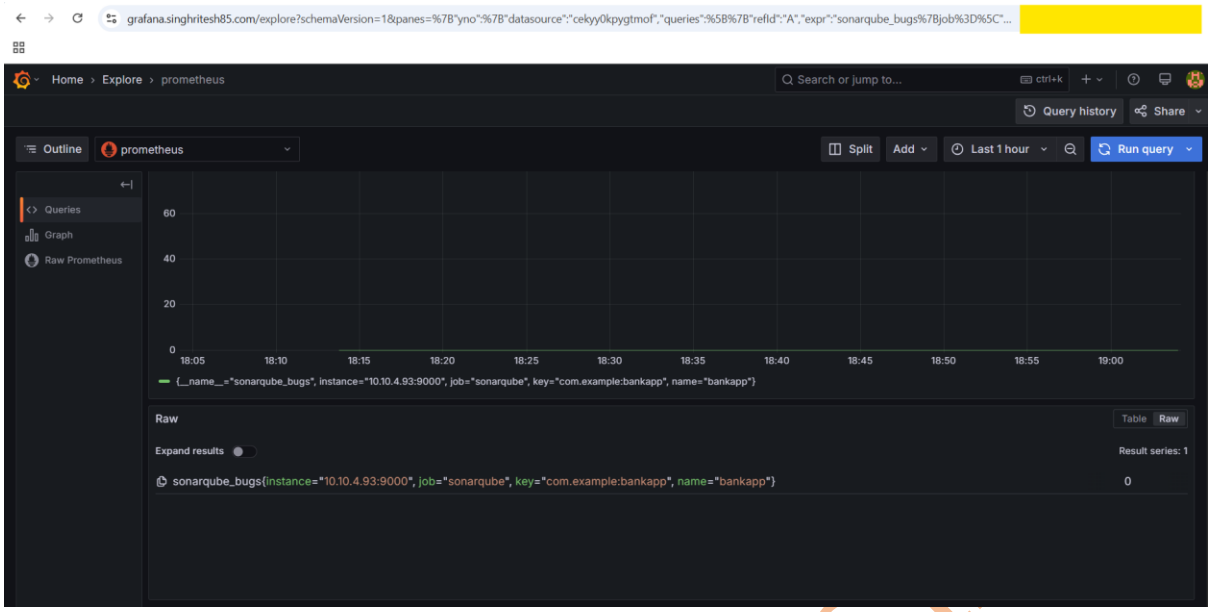
For Monitoring all the Servers and EKS and AKS Cluster health using the Node Exporter I used Grafana ID **1860**.



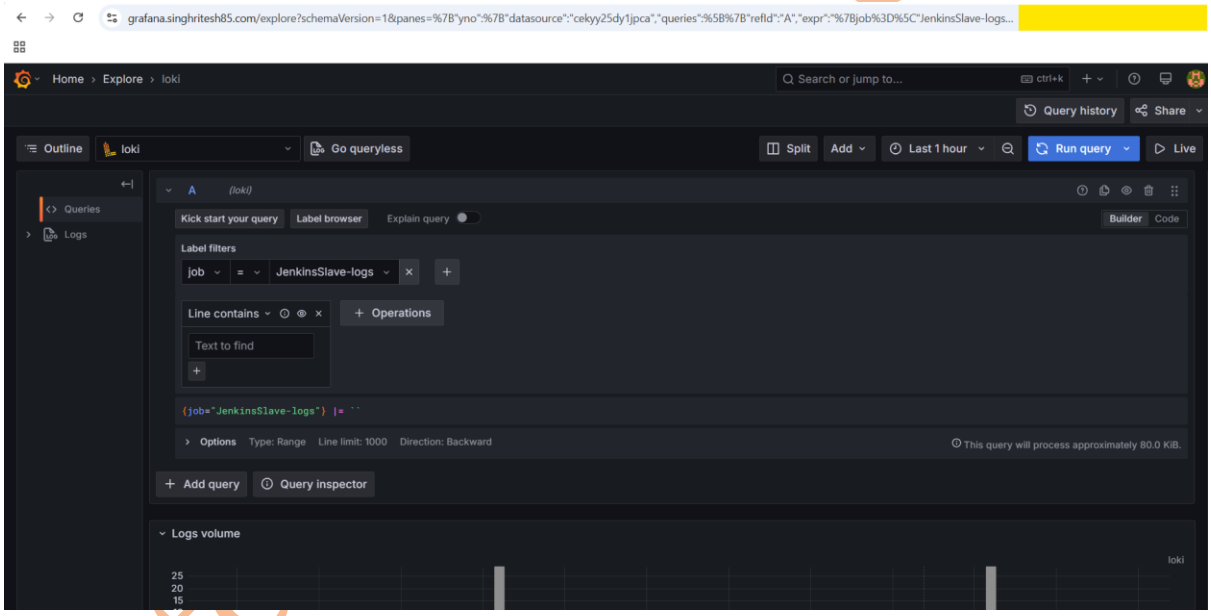


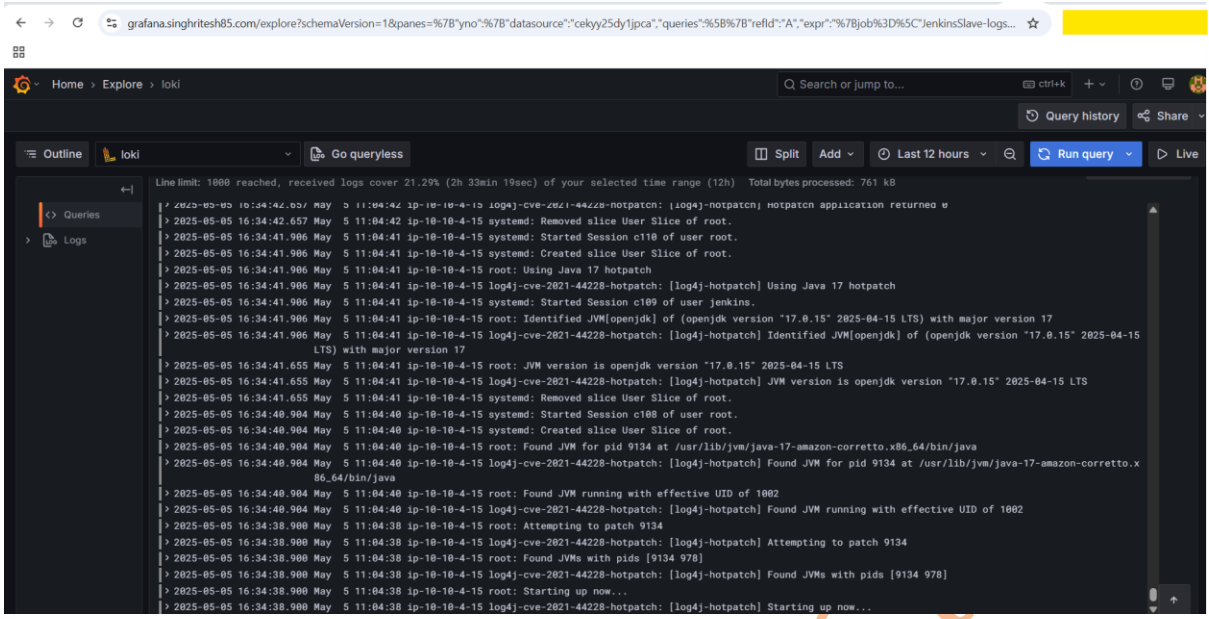
Grafana Metrics for SonarQube I started exploring as shown in the screenshot attached below.





Logs using Loki through Grafana I started exploring as shown in the screenshot attached below.





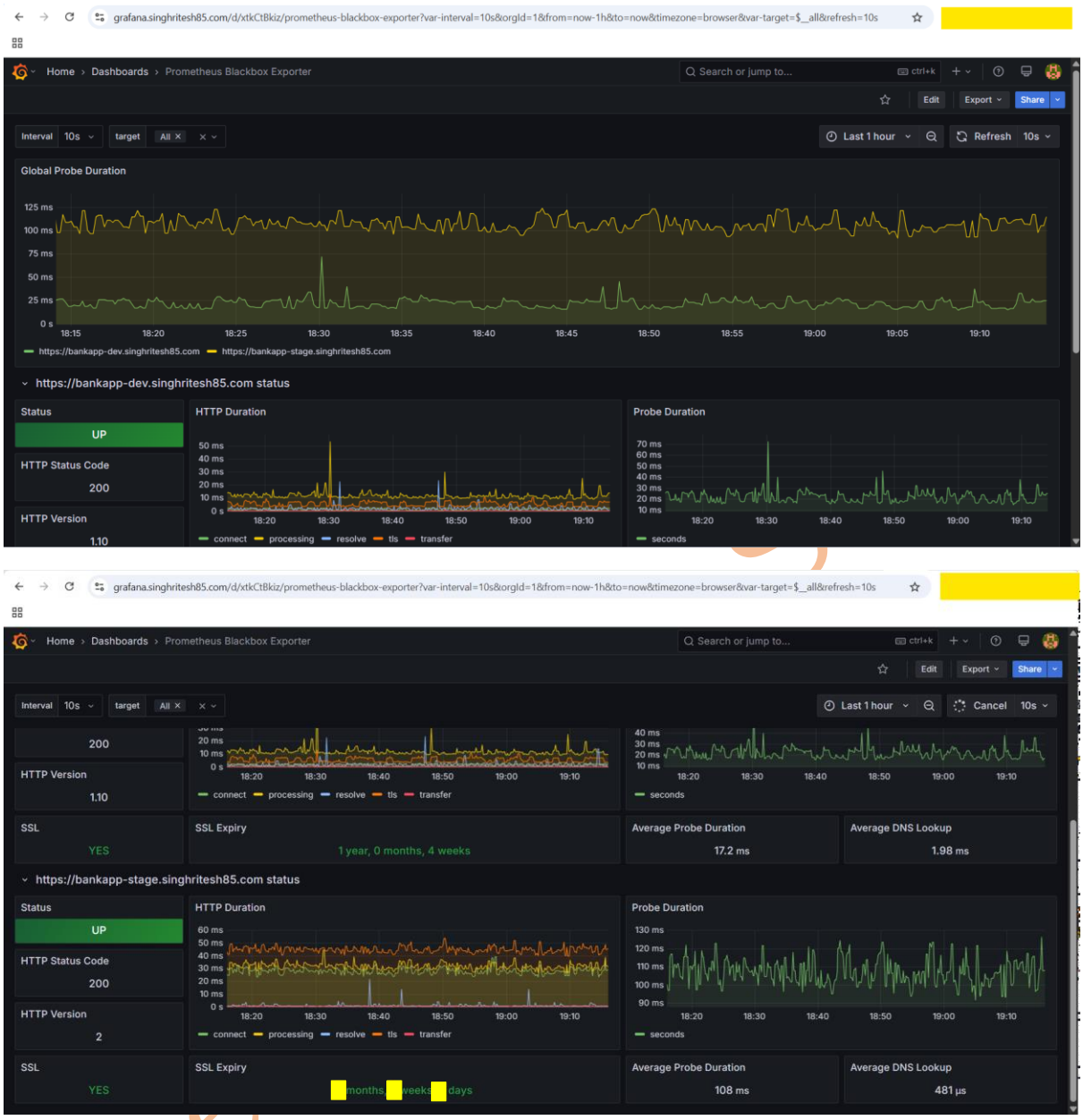
To achieve synthetic monitoring using Prometheus Blackbox Exporter I updated the **/etc/resolv.conf** file for Blackbox Exporter Server as shown in the screenshot attached below. I had used Google's Public DNS Server which is shown in the attached screenshot below.

```
[root@                    ~]# cat /etc/resolv.conf
; generated by /usr/sbin/dhclient-script
search us-east-2.compute.internal
options timeout:2 attempts:5
nameserver 8.8.8.8      #10.10.0.2
```

Finally, I was able to perform the synthetics monitoring on the Bankapp Application URL as shown in the screenshot attached below. Application URL **https://bankapp-dev.singhritesh85.com** and **https://bankapp-stage.singhritesh85.com** had been monitored using blackbox exporter.

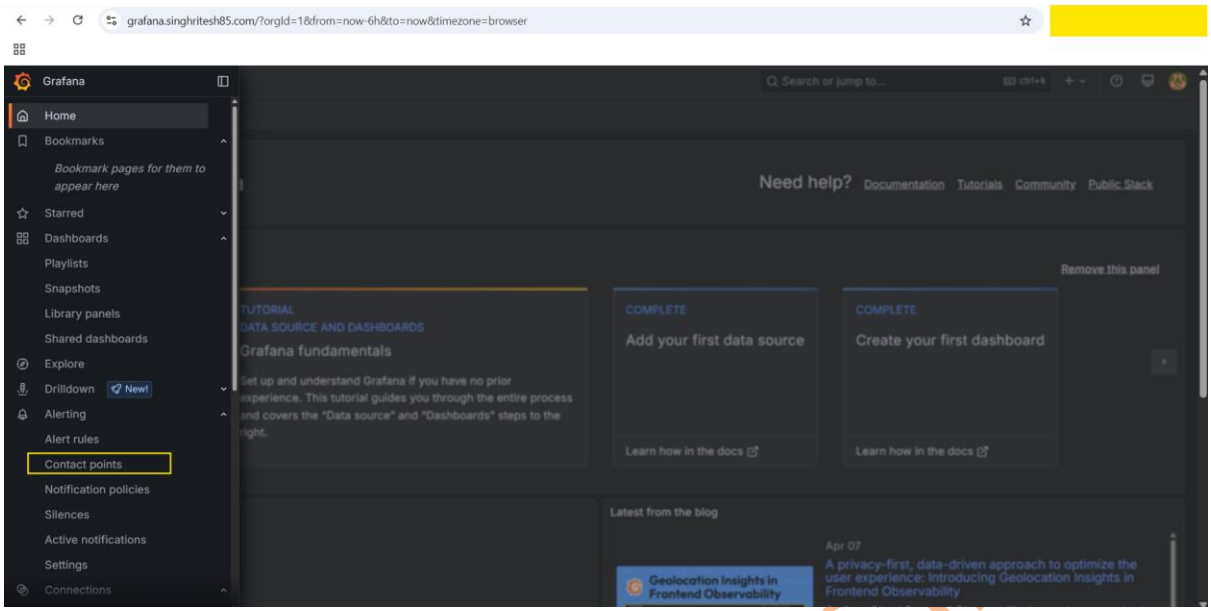
I had installed Blackbox Exporter on a different server and not on the Prometheus Server. **The module name** is monitor_website.yml present of the blackbox exporter server at the path (/opt/blackbox_exporter_linux_amd64/monitor_website.yml). Prometheus blackbox operator is used for endpoint monitoring (Synthetic Monitoring) across the protocol http, https, TCP and ICMP. In this project I am monitoring the Application URL **https://bankapp-dev.singhritesh85.com** and **https://bankapp-stage.singhritesh85.com** with the help of Prometheus Blackbox-Exporter. Prometheus blackbox exporter will send the metrics to Prometheus. For this project Prometheus acts as a DataSource for Grafana and send metrics to Grafana which we can see with the help of Charts and Graphs.

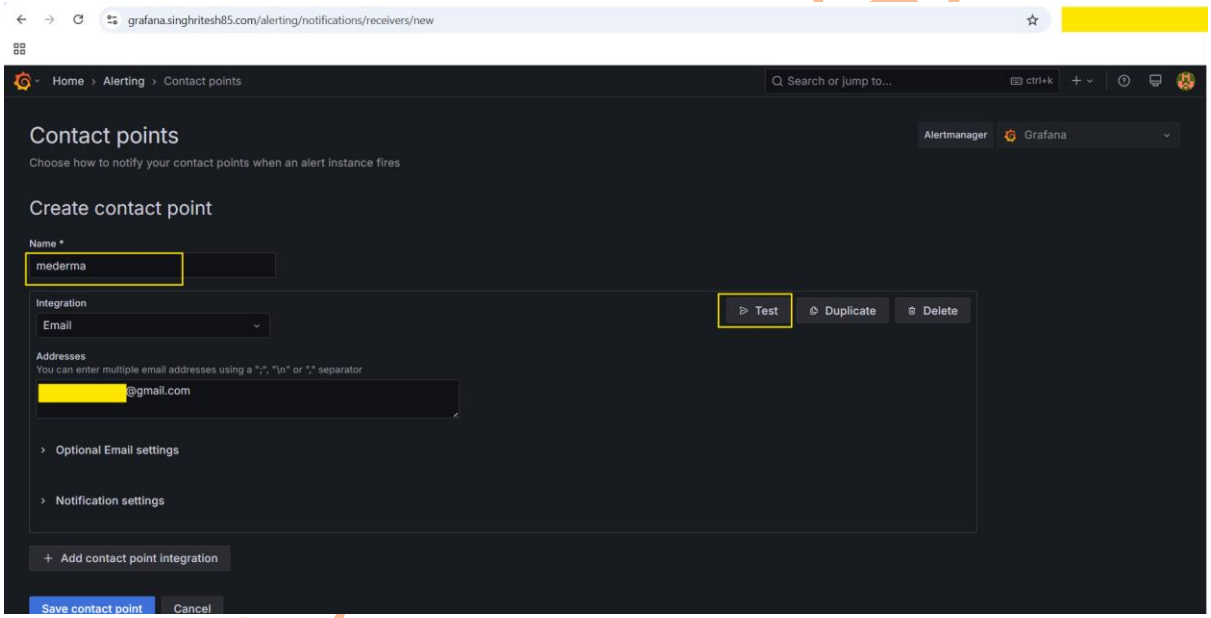
To create the Grafana Dashboard for Application URL Monitoring using blackbox exporter I had used the Grafana ID **7587** and below is the created Dashboard.

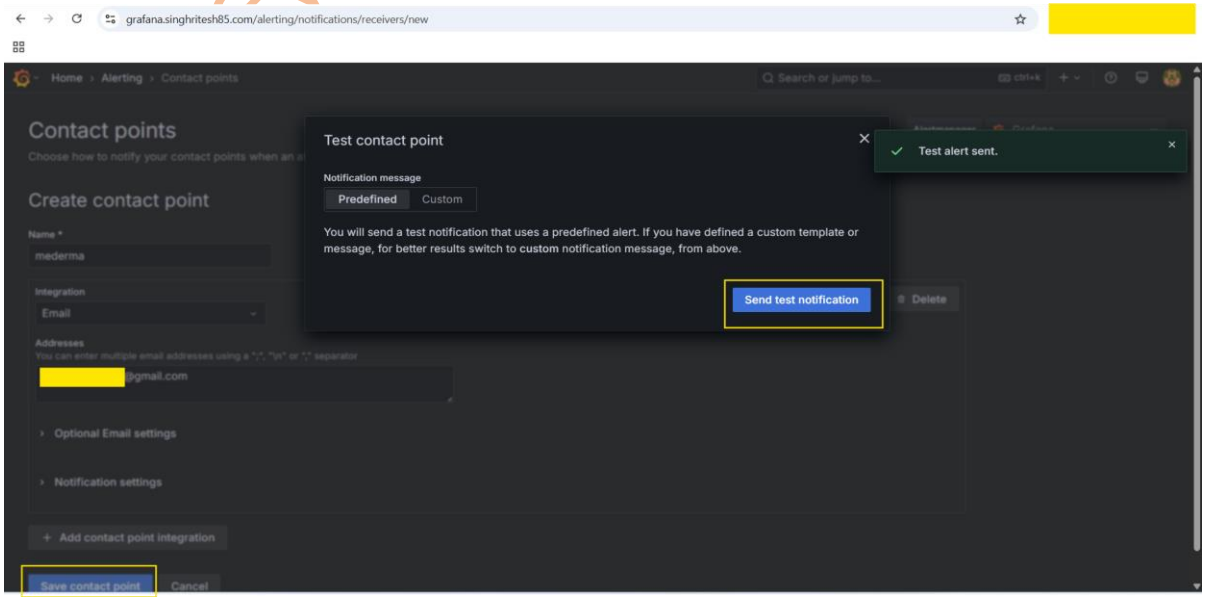


Configuration of Alerts in Grafana

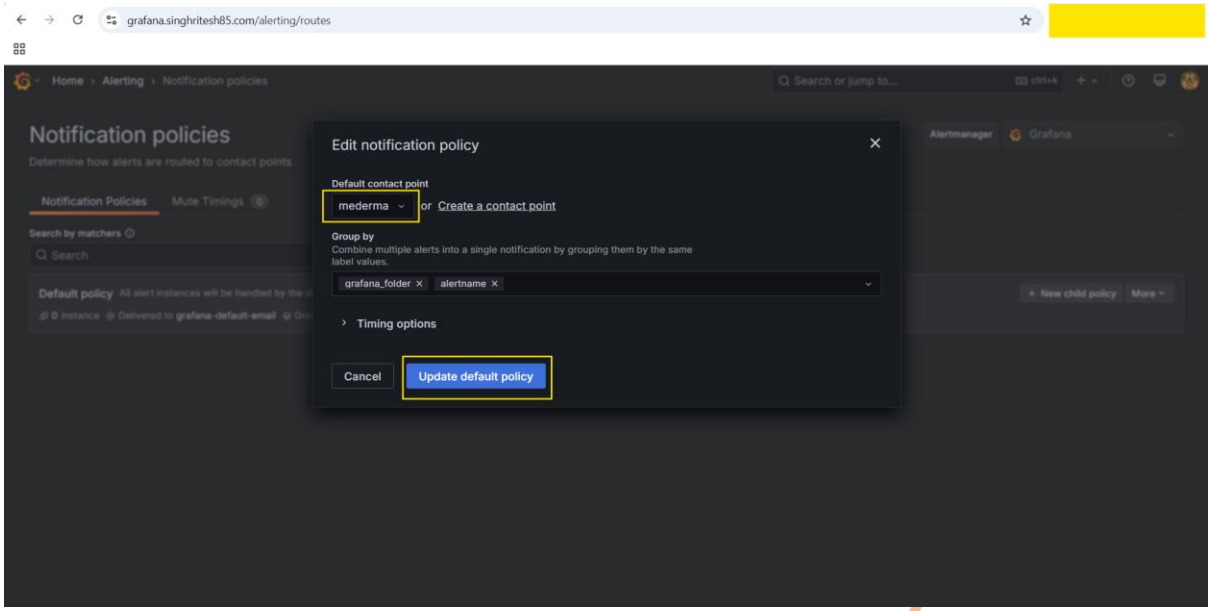
To configure Alerts in Grafana, first I created **contact points** with the Email ID and changed smtp settings in the configuration file **/etc/grafana/grafana.ini** of Grafana which I already discussed above. Here I had configured the contact points in Grafana UI as shown in the screenshot attached below.


grafana.singhritesh85.com/?orgId=1&from=now-6h&to=now&timezone=browser
Grafana
Home
Bookmarks
Bookmark pages for them to appear here
Starred
Dashboards
Playlists
Snapshots
Library panels
Shared dashboards
Explore
Drilldown
New!
Alerting
Alert rules
Contact points
Notification policies
Silences
Active notifications
Settings
Connections
Need help?
Documentation
Tutorials
Community
Public Slack
Remove this panel
Add your first data source
Create your first dashboard
Learn how in the docs
Learn how in the docs
Latest from the blog
Apr 07
A privacy-first, data-driven approach to optimize the user experience: Introducing Geolocation Insights in Frontend Observability

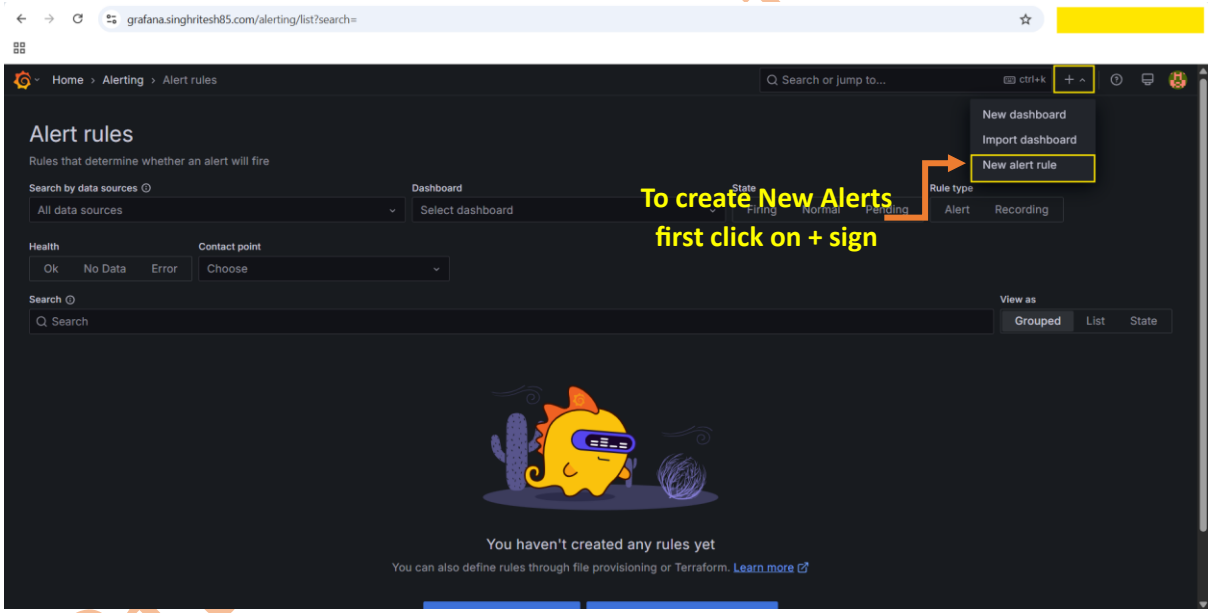

grafana.singhritesh85.com/alerting/notifications/receivers/new
Home
Alerting
Contact points
Contact points
Choose how to notify your contact points when an alert instance fires
Alertmanager
Grafana
Create contact point
Name *
mederma
Integration
Email
Test
Duplicate
Delete
Addresses
You can enter multiple email addresses using a ";", "\n" or "," separator
@gmail.com
Optional Email settings
Notification settings
Add contact point integration
Save contact point
Cancel

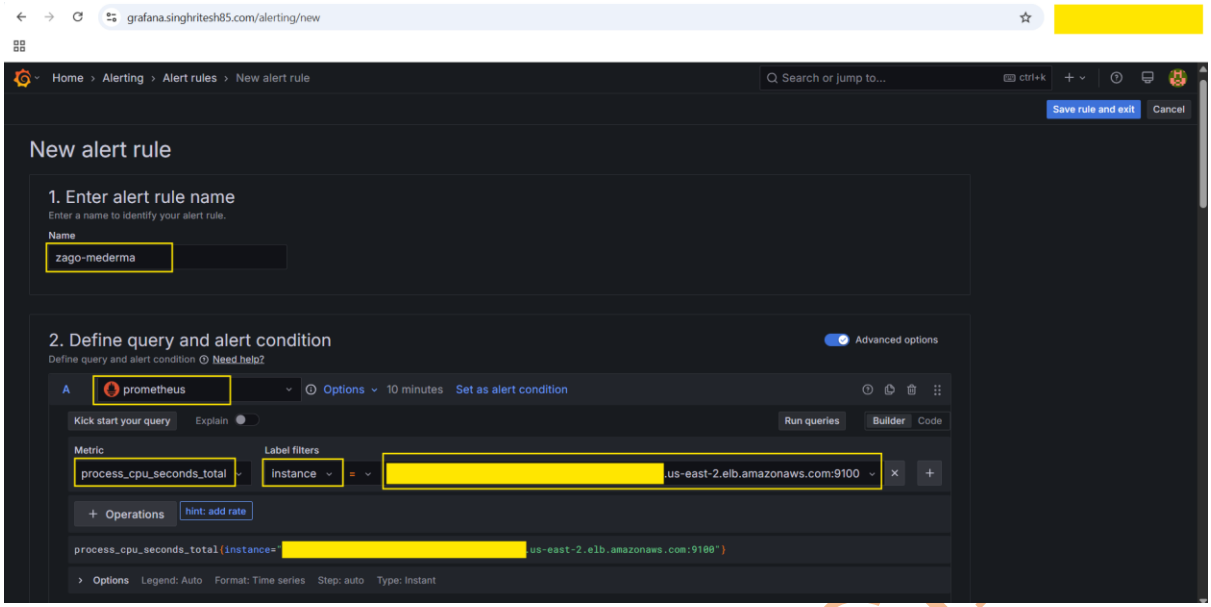

grafana.singhritesh85.com/alerting/notifications/receivers/new
Contact points
Create contact point
Test contact point
Notification message
Predefined
Custom
You will send a test notification that uses a predefined alert. If you have defined a custom template or message, for better results switch to custom notification message, from above.
Send test notification
Test alert sent.
Optional Email settings
Notification settings
Add contact point integration
Save contact point
Cancel

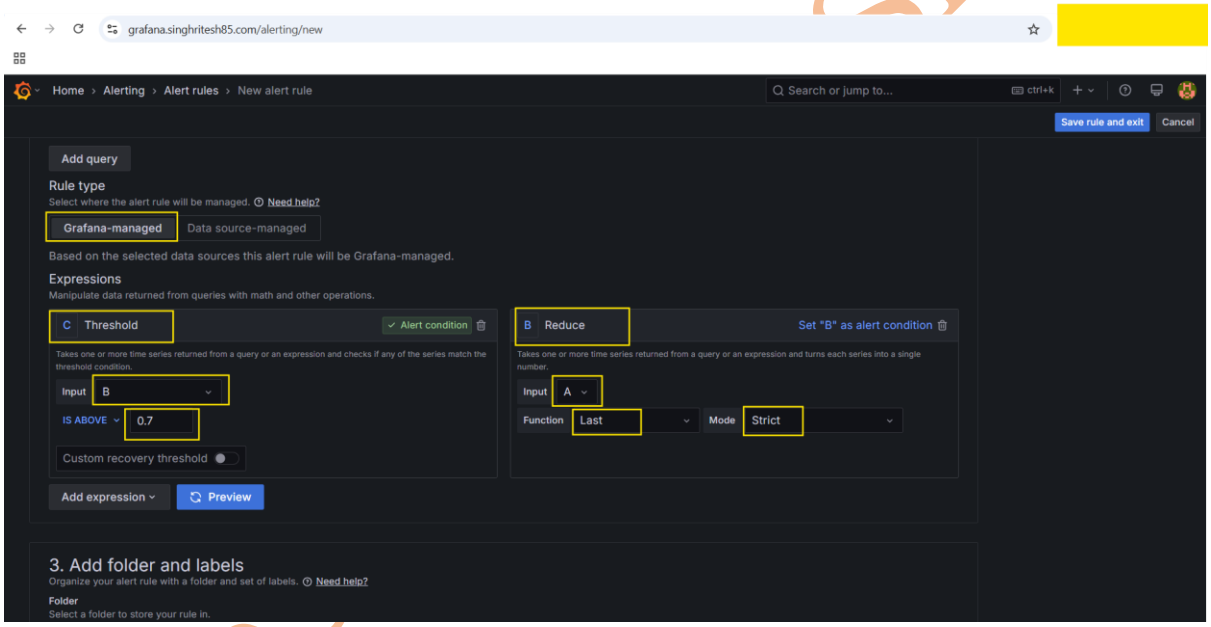
The Default **Notification Policy** had been changed as shown in the screenshot attached below.

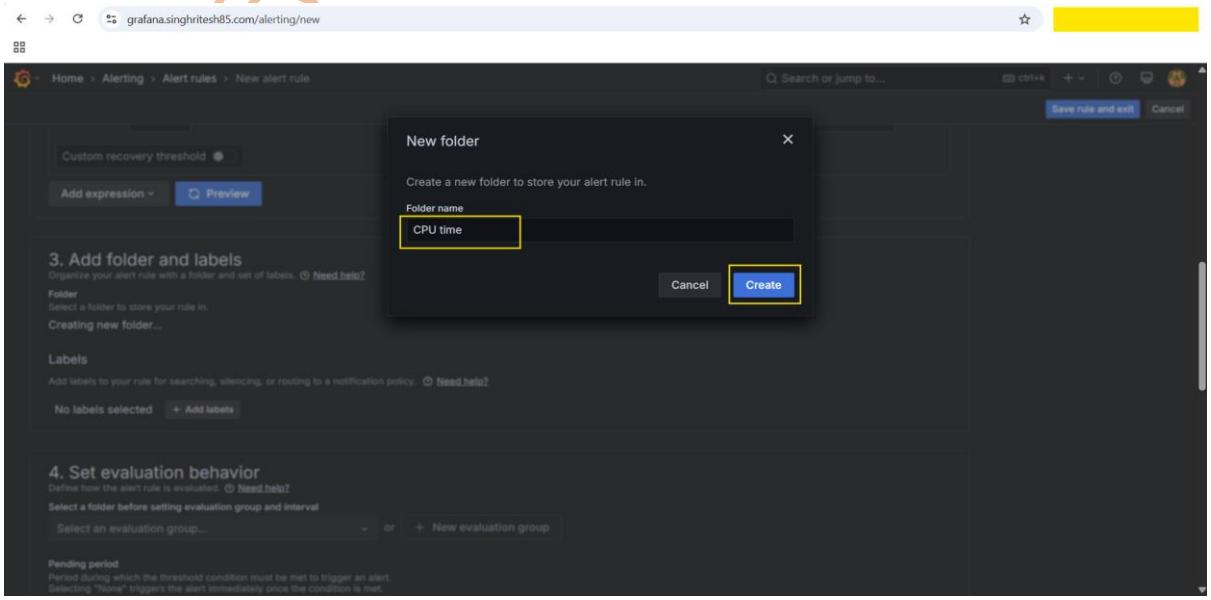


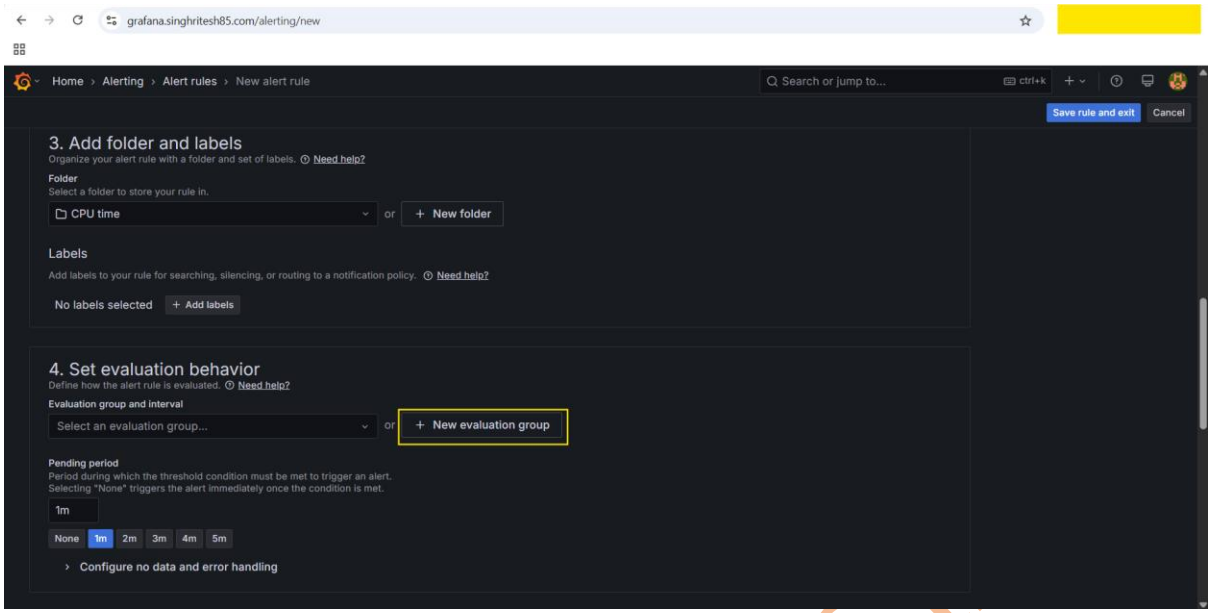
Configure **Alert Rule** as shown in the screenshot attached below.

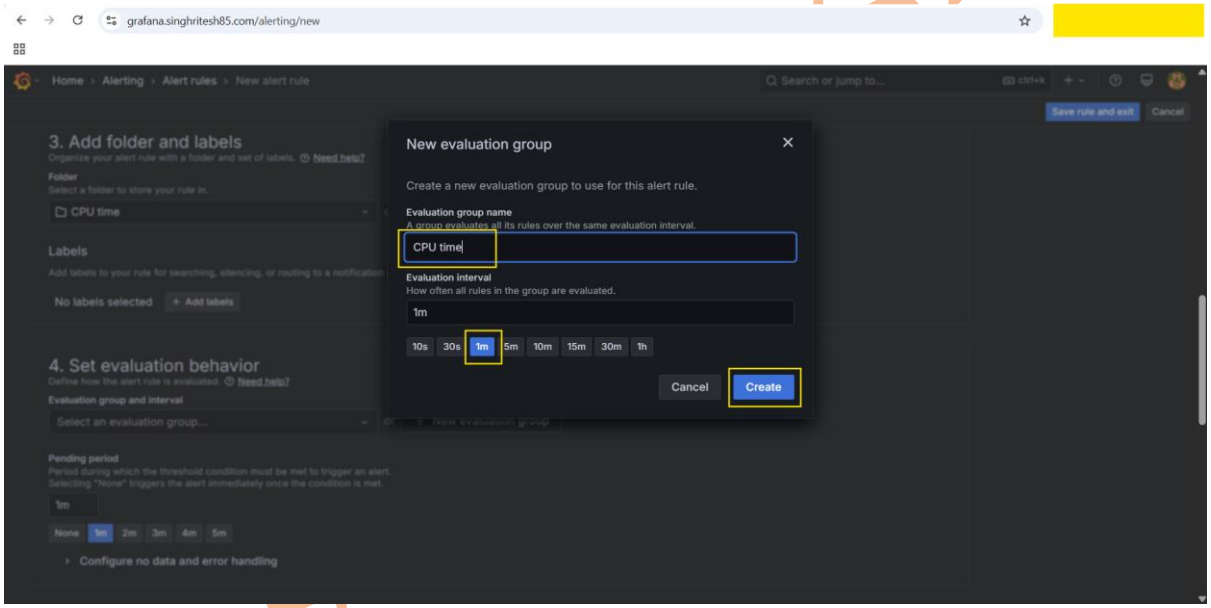


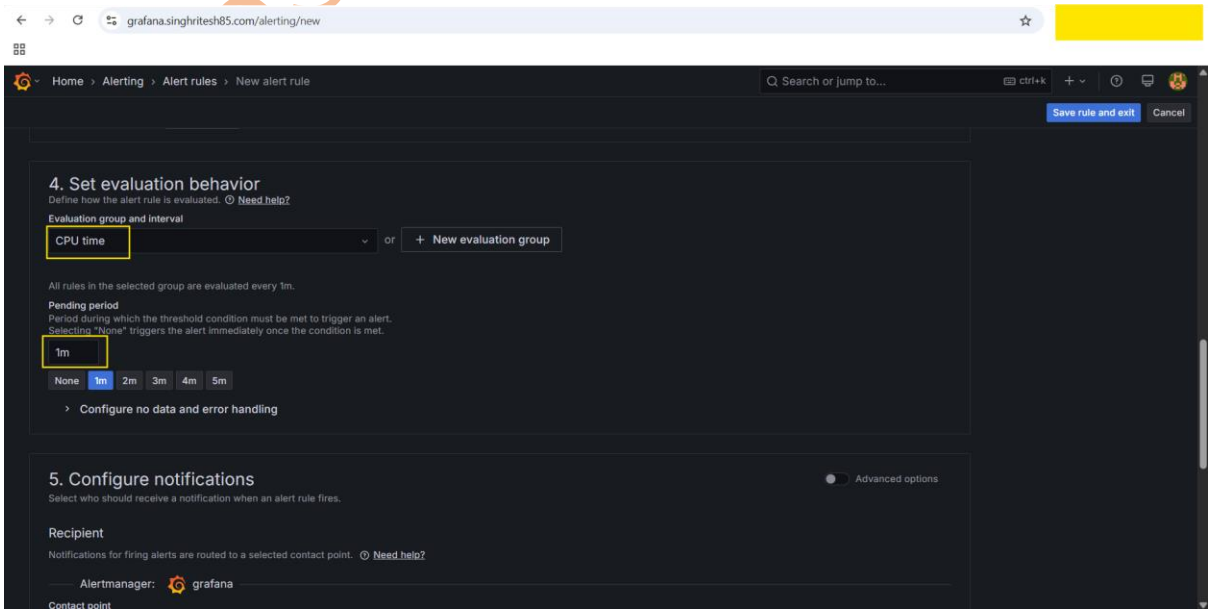

grafana.singhritesh85.com/alerting/new
Home › Alerting › Alert rules › New alert rule
Search or jump to...
Save rule and exit
Cancel
New alert rule
1. Enter alert rule name
Enter a name to identify your alert rule.
Name
zago-mederma
2. Define query and alert condition
Define query and alert condition Need help?
Advanced options
A
prometheus
Options 10 minutes
Set as alert condition
Kick start your query
Explain
Run queries
Builder
Code
Metric
process_cpu_seconds_total
Label filters
instance
us-east-2.elb.amazonaws.com:9100
Operations
hint: add rate
process_cpu_seconds_total{instance="
us-east-2.elb.amazonaws.com:9100"}
Options
Legend: Auto
Format: Time series
Step: auto
Type: Instant

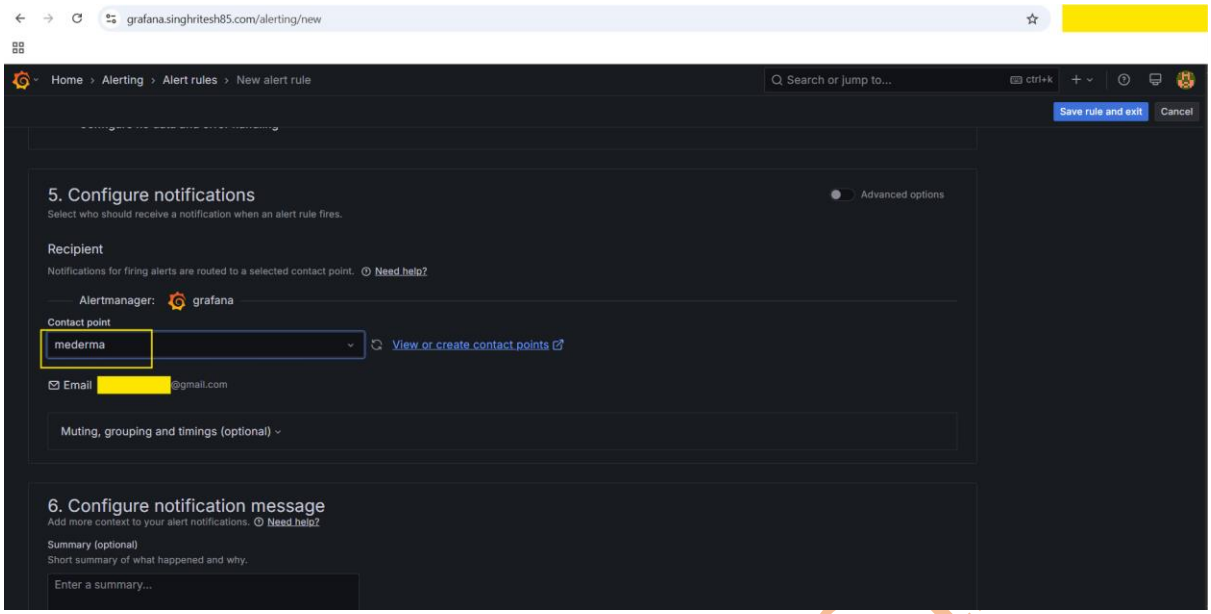

grafana.singhritesh85.com/alerting/new
Home › Alerting › Alert rules › New alert rule
Search or jump to...
Save rule and exit
Cancel
Add query
Rule type
Select where the alert rule will be managed. Need help?
Grafana-managed
Data source-managed
Based on the selected data sources this alert rule will be Grafana-managed.
Expressions
Manipulate data returned from queries with math and other operations.
C Threshold
Alert condition
Takes one or more time series returned from a query or an expression and checks if any of the series match the threshold condition.
Input B
IS ABOVE 0.7
Custom recovery threshold
B Reduce
Set "B" as alert condition
Takes one or more time series returned from a query or an expression and turns each series into a single number.
Input A
Function Last
Mode Strict
Add expression
Preview
3. Add folder and labels
Organize your alert rule with a folder and set of labels. Need help?
Folder
Select a folder to store your rule in.


grafana.singhritesh85.com/alerting/new
New folder
Create a new folder to store your alert rule in.
Folder name
CPU time
Cancel
Create
Custom recovery threshold
Add expression
Preview
3. Add folder and labels
Folder
Select a folder to store your rule in.
Creating new folder...
Labels
No labels selected
Add labels
4. Set evaluation behavior
Select a folder before setting evaluation group and interval
Select an evaluation group...
or
New evaluation group
Pending period

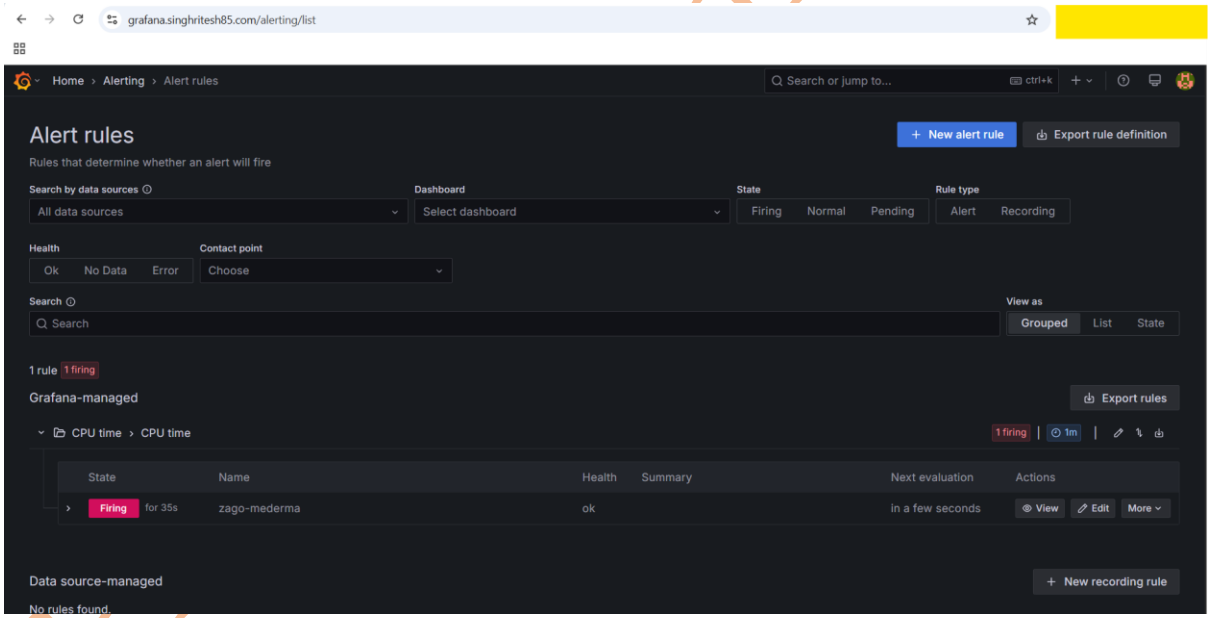

grafana.singhritesh85.com/alerting/new
Home › Alerting › Alert rules › New alert rule
Search or jump to...
Save rule and exit
Cancel
3. Add folder and labels
Organize your alert rule with a folder and set of labels. Need help?
Folder
Select a folder to store your rule in.
CPU time
or
New folder
Labels
Add labels to your rule for searching, silencing, or routing to a notification policy. Need help?
No labels selected
Add labels
4. Set evaluation behavior
Define how the alert rule is evaluated. Need help?
Evaluation group and interval
Select an evaluation group...
or
New evaluation group
Pending period
Period during which the threshold condition must be met to trigger an alert.
Selecting "None" triggers the alert immediately once the condition is met.
1m
None
1m
2m
3m
4m
5m
Configure no data and error handling


grafana.singhritesh85.com/alerting/new
New evaluation group
Create a new evaluation group to use for this alert rule.
Evaluation group name
A group evaluates all its rules over the same evaluation interval.
CPU time
Evaluation interval
How often all rules in the group are evaluated.
1m
10s
30s
1m
5m
10m
15m
30m
1h
Cancel
Create

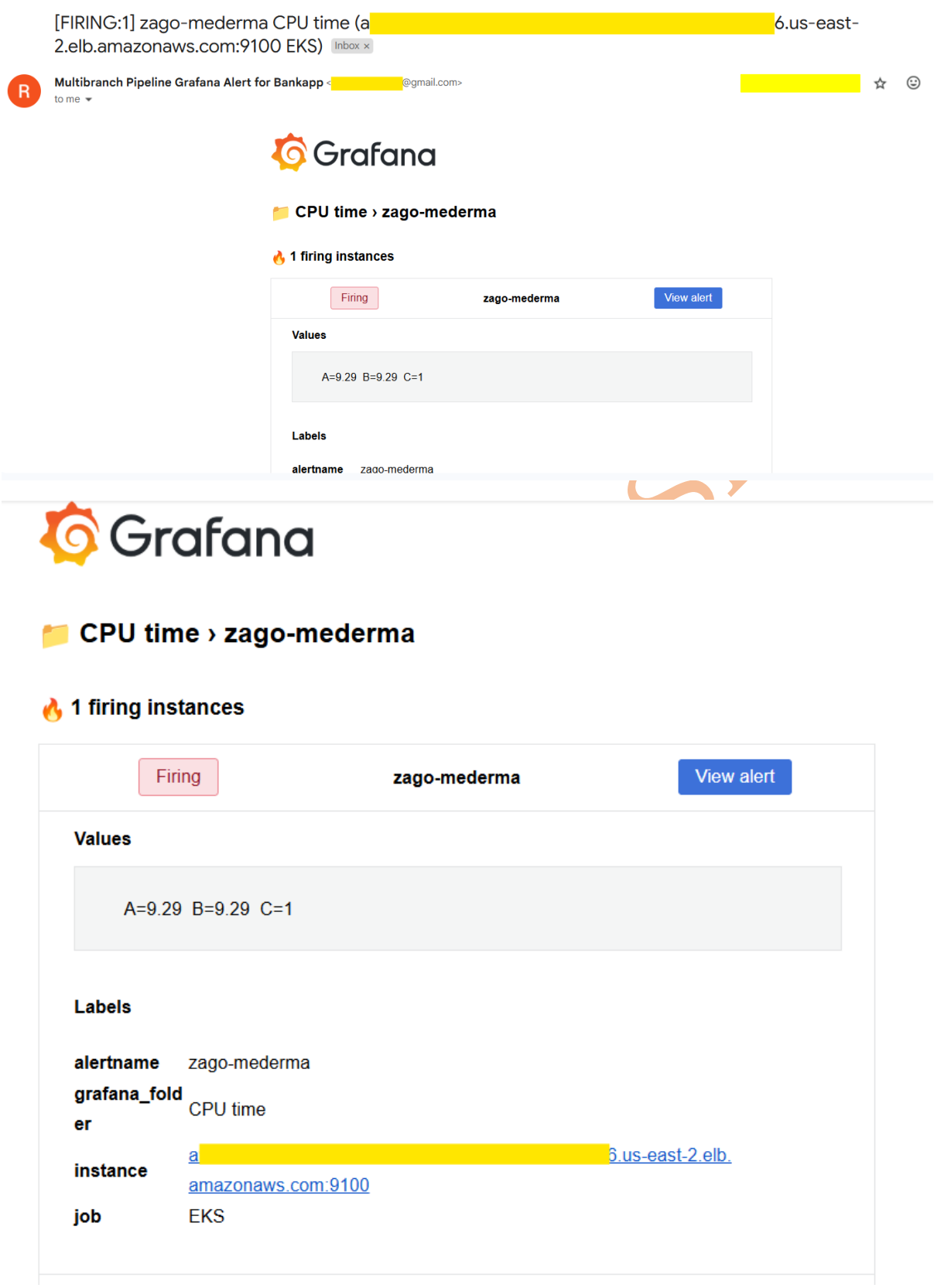

grafana.singhritesh85.com/alerting/new
Home › Alerting › Alert rules › New alert rule
Search or jump to...
Save rule and exit
Cancel
4. Set evaluation behavior
Define how the alert rule is evaluated. Need help?
Evaluation group and interval
CPU time
or
New evaluation group
All rules in the selected group are evaluated every 1m.
Pending period
Period during which the threshold condition must be met to trigger an alert.
Selecting "None" triggers the alert immediately once the condition is met.
1m
None
1m
2m
3m
4m
5m
Configure no data and error handling
5. Configure notifications
Select who should receive a notification when an alert rule fires.
Advanced options
Recipient
Notifications for firing alerts are routed to a selected contact point. Need help?
Alertmanager:
grafana
Contact point



If the Alert Rule is in firing state after condition crosses the threshold condition, then Grafana console screenshot will be showing the same as shown in the screenshot attached below.

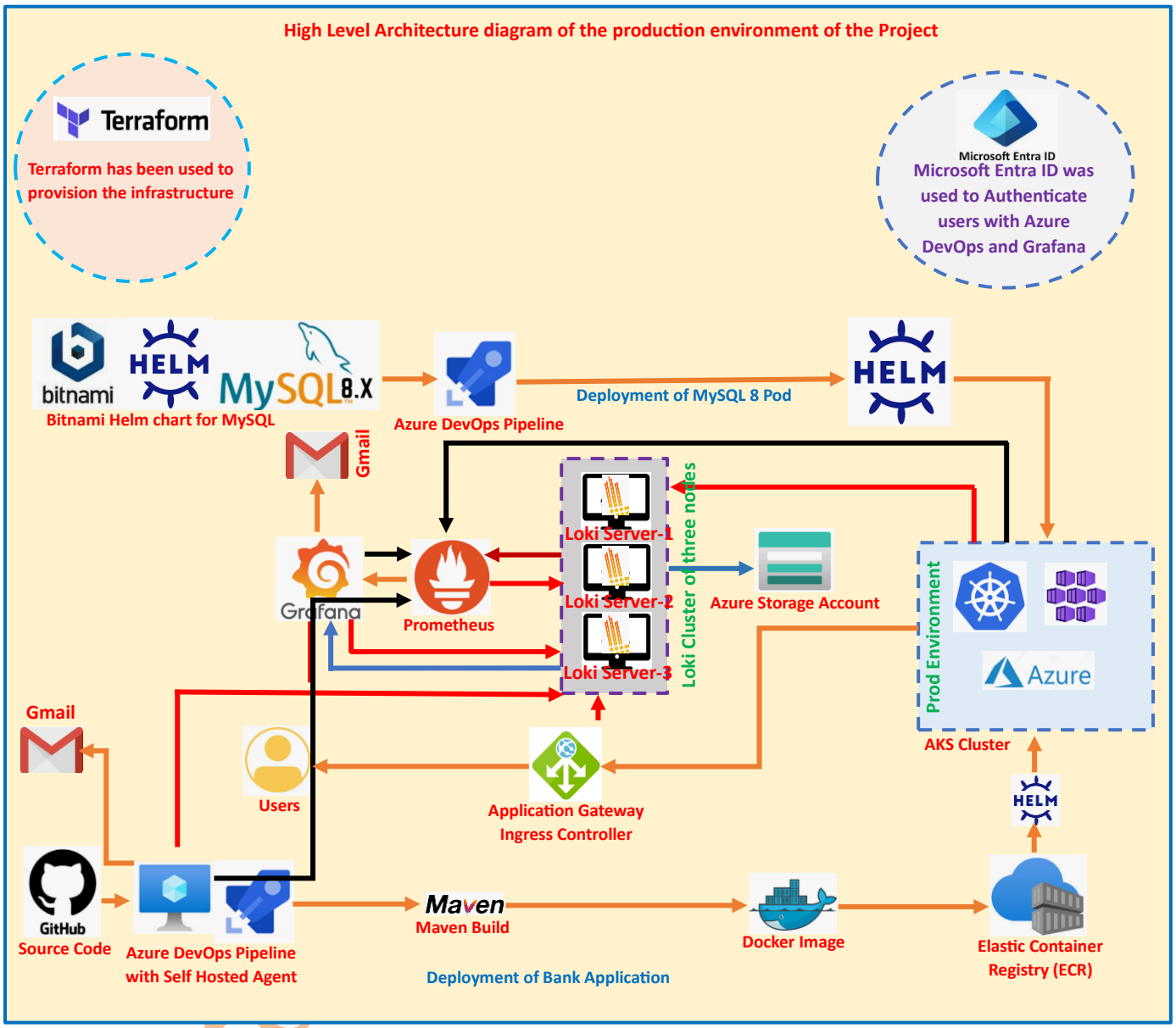


An Email was sent to the Email ID as shown in the screenshot attached below.



In this project to refrain from higher cloud cost for **dev** and **stage** environment I had used the same ECR (Elastic Container Registry) but different tag name for the Docker Images. You can use different ECR for your project in **dev** and **stage** environment.

Module-2: [Production Environment]



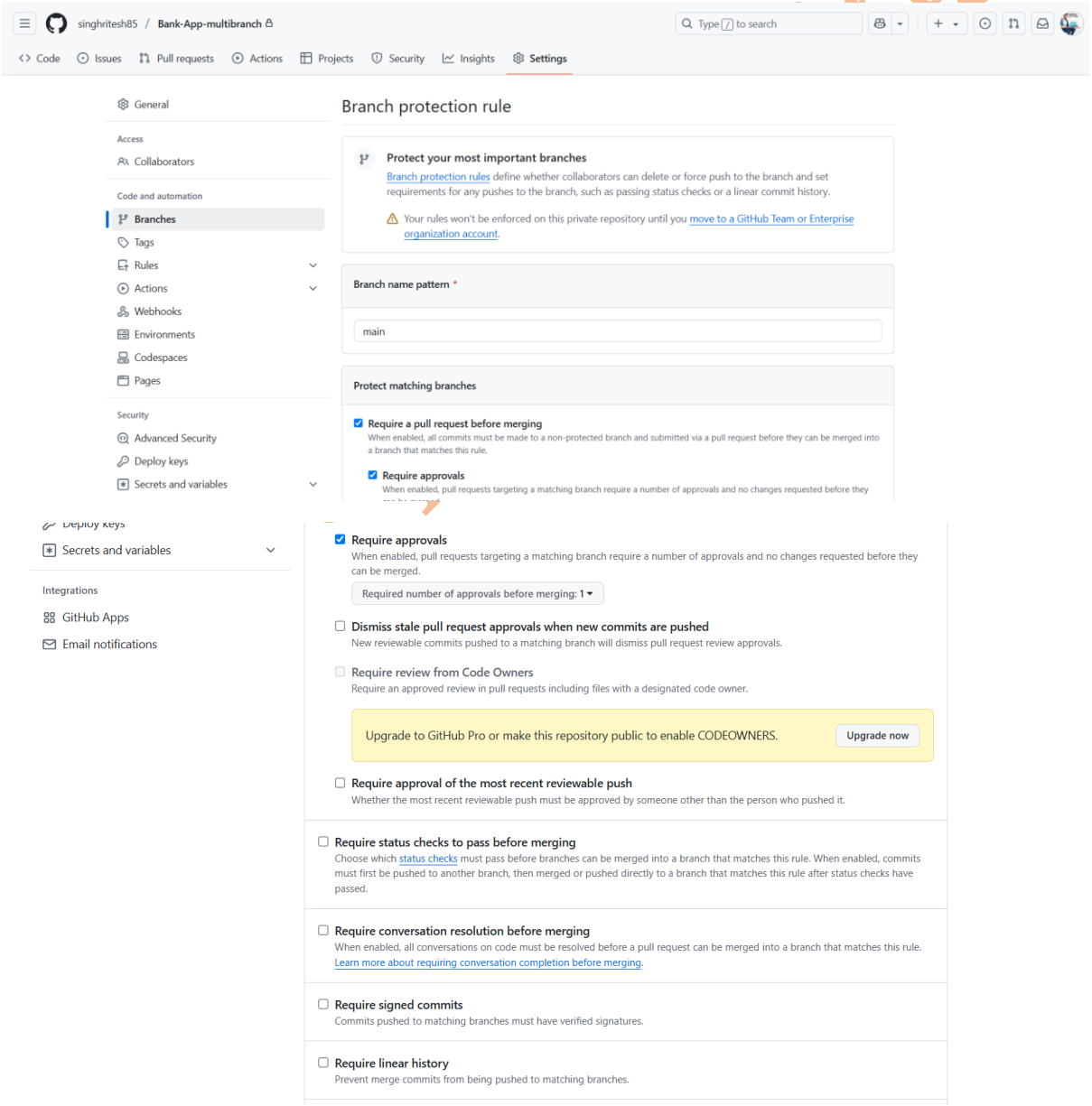
In the above diagram the black arrow (⟶) indicated the metrics exported by node exporter to the prometheus and red arrow (⟶) indicates logs extracted by the promtail to loki.

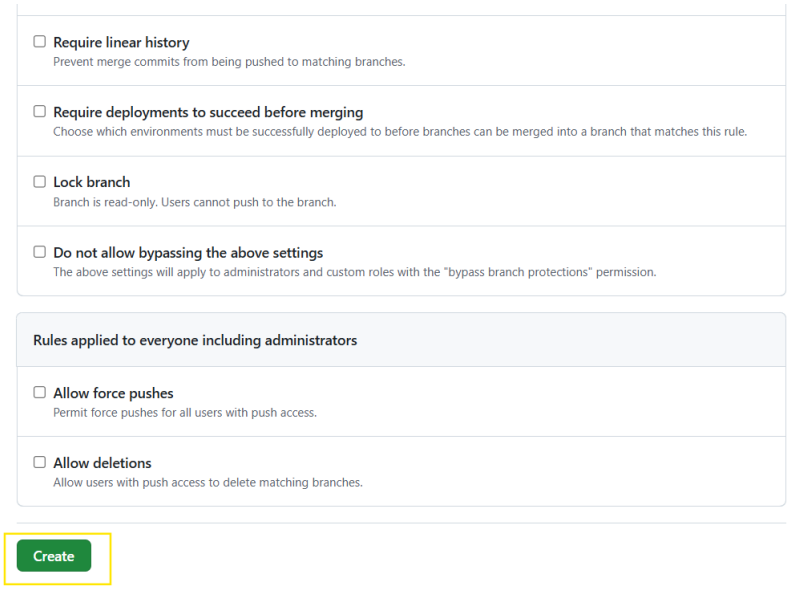
GitHub Branching Strategies

In some of the organisation there are three environments **dev**, **stage** and **prod** and there may be one more branch **pre-prod** which depends on the project and the organisation. Before proceeding further, I discuss here first **GitHub Branch Protection**. In Real time projects Developers are allowed to push their code in dev environment and if it becomes successful after proper testing of the deployed code in dev branch then dev branch will be merged to the stage branch using Pull Request (PR). To approve the pull request of stage environment I would suggest you can keep two approvers, you can keep two approvers as your team-mates. Developers will create a **feature branch** from the **dev branch** then after doing their coding part in **feature branch** they will merge their code into **dev**

branch. For dev branch Pull Request (PR) you can keep one approver who can be your team-mate. Finally, to merge code into the **main branch** from **stage branch** at least three approvals required, the three approvals required including the dignitary member of the organisation because the code merged in this branch will go live. You can create the webhook in Jenkins Pipeline or Azure DevOps Pipeline, so that whenever the code will be merged into the **dev/stage** or **main branch** then it will be deployed automatically. However, I will suggest do not apply webhook in production environment and provide its access to the DevOps Team and DevOps Team should run the appropriate Jenkins Job with required parameters after getting the approval from the required dignitary member of your organisation on **Jira ticket**.

In this project for demonstration purpose, I had applied the GitHub Branch Protection rule in only main branch as shown in the screenshot attached below. No one can push directly to the main branch developer can only merge the stage branch to the main branch after required approval as shown in the screenshot attached below.





For this project to refrain from higher cloud bills I used the Instance Type of t3.medium and Standard_B2s. However, in Organisation you can select the Instance type of m4.4xlarge/t3.2xlarge and vm_size of Standard_DS3_v2/Standard_DS4_v2/Standard_DS5_v2 or some other Instance Type/VM Size depending on your project requirement.

I installed terraform on Alma Linux2 Azure VM and the State file was kept in and state lock had been achieved using the Azure Storage Account container. Before running the terraform script I ran the shell script present in the directory **terraform-bankapp-prod-azure** as shown in the screenshot attached below. This shell script will install the aws cli, kubectl and helm.

```
[root@                terraform-multi-kubernetes-cluster-multicloud]# ./initial-setup.sh
```

Then I logged-out and login again. Then I installed Azure CLI and authenticated and authorize the user as shown in the screenshot attached below.

```
[root@              main]# yum install -y https://packages.microsoft.com/config/rhel/8/packages-microsoft-prod.rpm
[root@                main]# yum install azure-cli -y
[root@              main]# az login
To sign in, use a web browser to open the page                and enter the code          to authenticate.
```

Then you can run the below commands

terraform init -------> initializes a working directory containing configuration files and installs plugins for required providers.

terraform validate --------> verify that terraform configuration file is correct or not

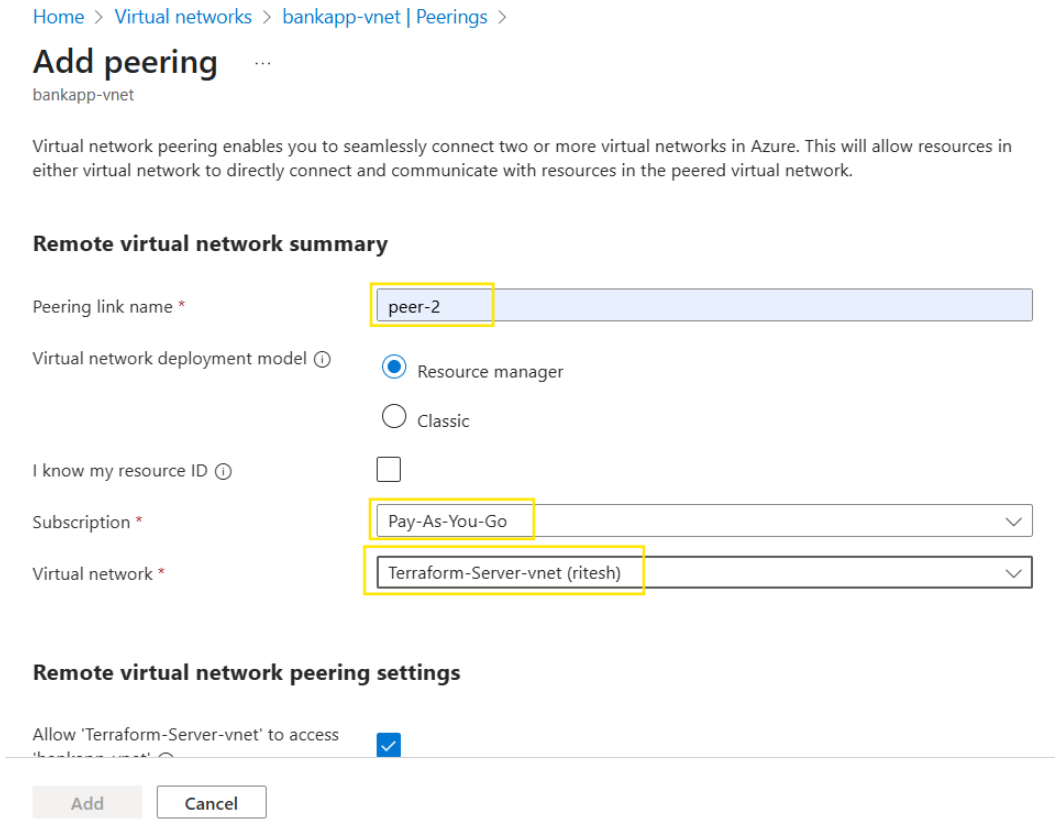
terraform plan ---------> Check which resources are going to be created.

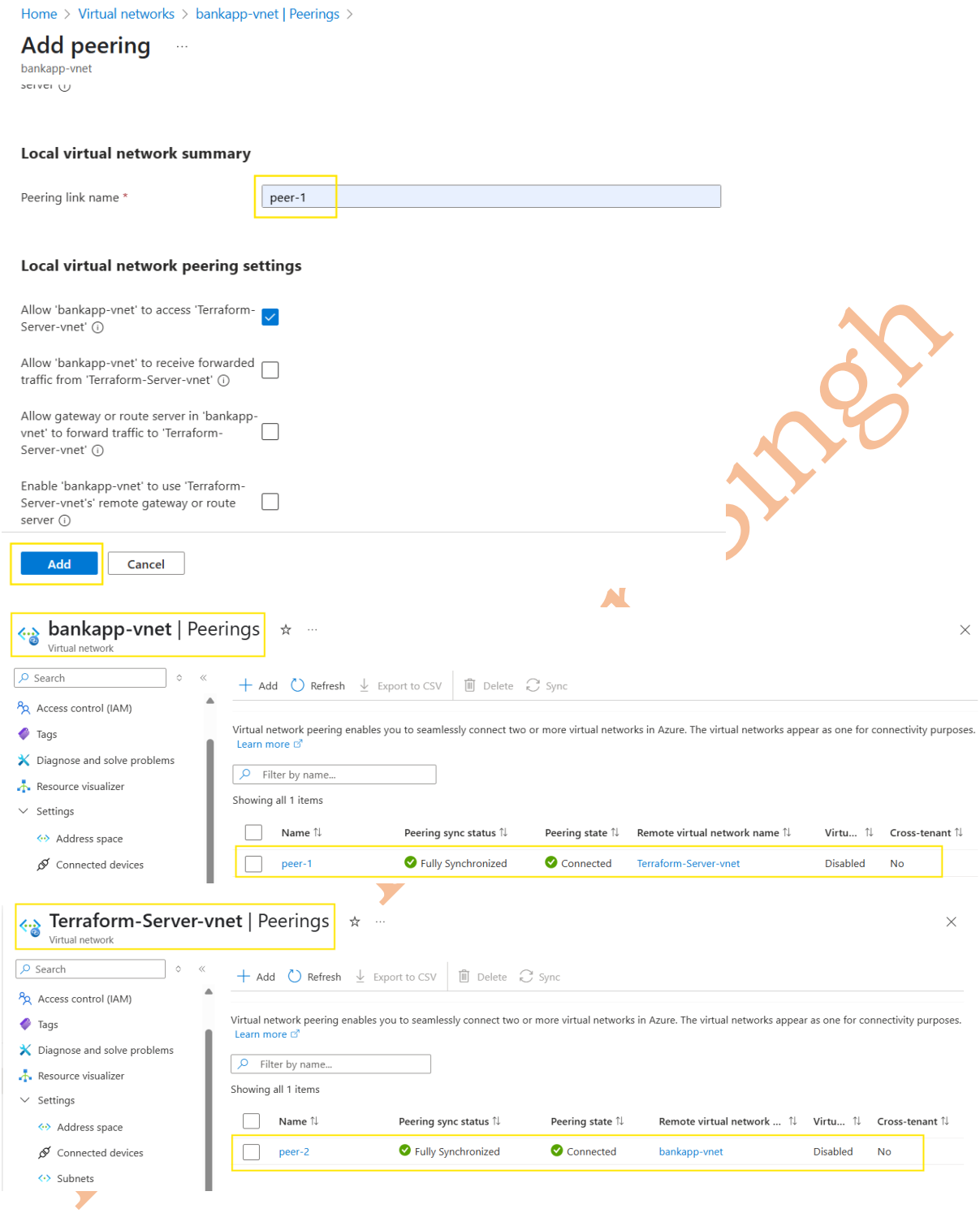
Then you can run the command **terraform apply -auto-approve** -----> Finally, Create the resources.

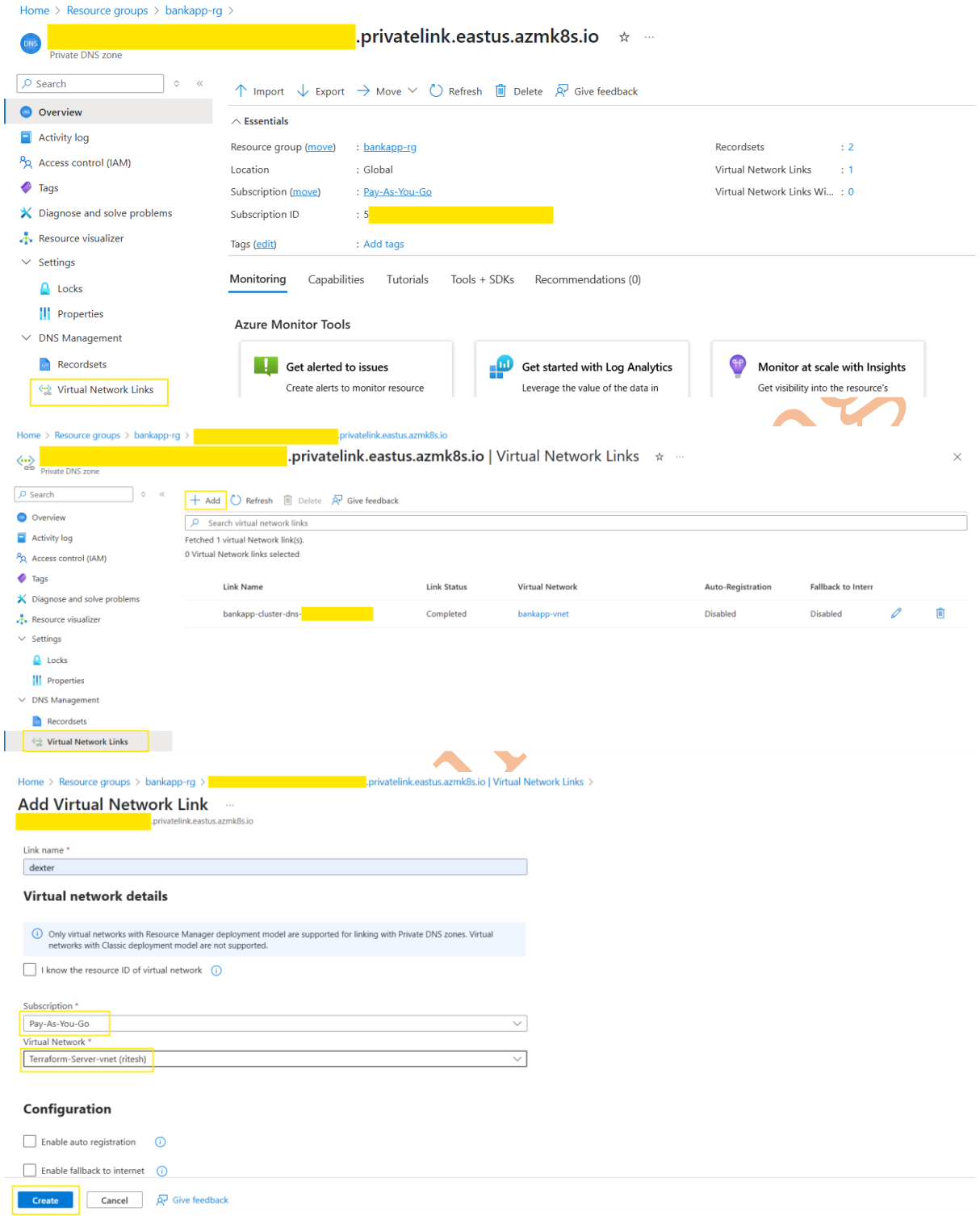
terraform destroy -auto-approve ----------> **Run this command only when you want to destroy all the resources.**

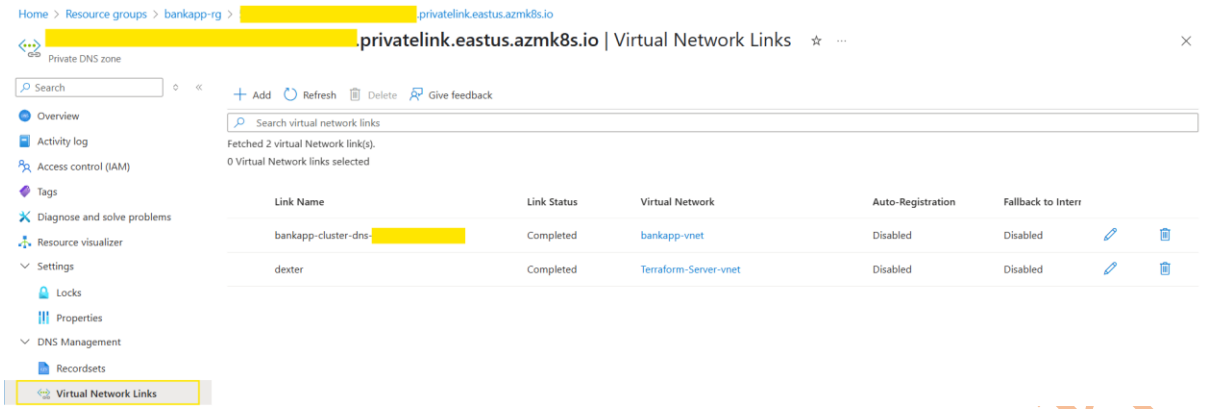
```
module.aks.null_resource.kubectl: Creation complete after 3s [id=                    ]

Apply complete! Resources: 84 added, 0 changed, 0 destroyed.

Outputs:

acr_azure_loki_vm_private_ip = {
  "acr_login_server" = "bankappcontainer24registry.azurecr.io"
  "azure_loki_vm" = [
    "            ",
    "            ",
    "            ",
  ]
}
```

This terraform script took total 20-25 minutes in all to get executed successfully as shown in the screenshot attached above. After that All the resources in Azure had been created. As the created AKS Cluster was Private AKS Cluster (API Server was private) and it was created in the different VNet as compared with the Terraform-Server. If you want to access AKS Cluster from the Terraform-Server using kubectl command then you need to create the VNet Peering between the Terraform-Server VNet and AKS Cluster VNet. Then you also need to create the Virtual Network Link in Private DNS Zone for Private AKS Cluster as shown in the screenshot attached below.




Home > Virtual networks > bankapp-vnet | Peerings >
Add peering
bankapp-vnet
Local virtual network summary
Peering link name *
peer-1
Local virtual network peering settings
Allow 'bankapp-vnet' to access 'Terraform-Server-vnet'
Allow 'bankapp-vnet' to receive forwarded traffic from 'Terraform-Server-vnet'
Allow gateway or route server in 'bankapp-vnet' to forward traffic to 'Terraform-Server-vnet'
Enable 'bankapp-vnet' to use 'Terraform-Server-vnet's' remote gateway or route server
Add
Cancel
bankapp-vnet | Peerings
Virtual network
Search
Access control (IAM)
Tags
Diagnose and solve problems
Resource visualizer
Settings
Address space
Connected devices
Add
Refresh
Export to CSV
Delete
Sync
Virtual network peering enables you to seamlessly connect two or more virtual networks in Azure. The virtual networks appear as one for connectivity purposes.
Learn more
Filter by name...
Showing all 1 items
Name
Peering sync status
Peering state
Remote virtual network name
Virtu...
Cross-tenant
peer-1
Fully Synchronized
Connected
Terraform-Server-vnet
Disabled
No
Terraform-Server-vnet | Peerings
Virtual network
Search
Access control (IAM)
Tags
Diagnose and solve problems
Resource visualizer
Settings
Address space
Connected devices
Subnets
Add
Refresh
Export to CSV
Delete
Sync
Virtual network peering enables you to seamlessly connect two or more virtual networks in Azure. The virtual networks appear as one for connectivity purposes.
Learn more
Filter by name...
Showing all 1 items
Name
Peering sync status
Peering state
Remote virtual network ...
Virtu...
Cross-tenant
peer-2
Fully Synchronized
Connected
bankapp-vnet
Disabled
No


Home > Resource groups > bankapp-rg >
[redacted].privatelink.eastus.azmk8s.io
Private DNS zone
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Resource visualizer
Settings
Locks
Properties
DNS Management
Recordsets
Virtual Network Links
Import
Export
Move
Refresh
Delete
Give feedback
Essentials
Resource group (move) : bankapp-rg
Location : Global
Subscription (move) : Pay-As-You-Go
Subscription ID : 5[redacted]
Tags (edit) : Add tags
Recordsets : 2
Virtual Network Links : 1
Virtual Network Links Wi... : 0
Monitoring
Capabilities
Tutorials
Tools + SDKs
Recommendations (0)
Azure Monitor Tools
Get alerted to issues
Create alerts to monitor resource
Get started with Log Analytics
Leverage the value of the data in
Monitor at scale with Insights
Get visibility into the resource's
Home > Resource groups > bankapp-rg > [redacted].privatelink.eastus.azmk8s.io
[redacted].privatelink.eastus.azmk8s.io | Virtual Network Links
Private DNS zone
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Resource visualizer
Settings
Locks
Properties
DNS Management
Recordsets
Virtual Network Links
Add
Refresh
Delete
Give feedback
Search virtual network links
Fetched 1 virtual Network link(s).
0 Virtual Network links selected
Link Name
Link Status
Virtual Network
Auto-Registration
Fallback to Interr
bankapp-cluster-dns-[redacted]
Completed
bankapp-vnet
Disabled
Disabled
Home > Resource groups > bankapp-rg > [redacted].privatelink.eastus.azmk8s.io | Virtual Network Links >
Add Virtual Network Link
[redacted].privatelink.eastus.azmk8s.io
Link name *
dexter
Virtual network details
Only virtual networks with Resource Manager deployment model are supported for linking with Private DNS zones. Virtual networks with Classic deployment model are not supported.
I know the resource ID of virtual network
Subscription *
Pay-As-You-Go
Virtual Network *
Terraform-Server-vnet (ritesh)
Configuration
Enable auto registration
Enable fallback to internet
Create
Cancel
Give feedback

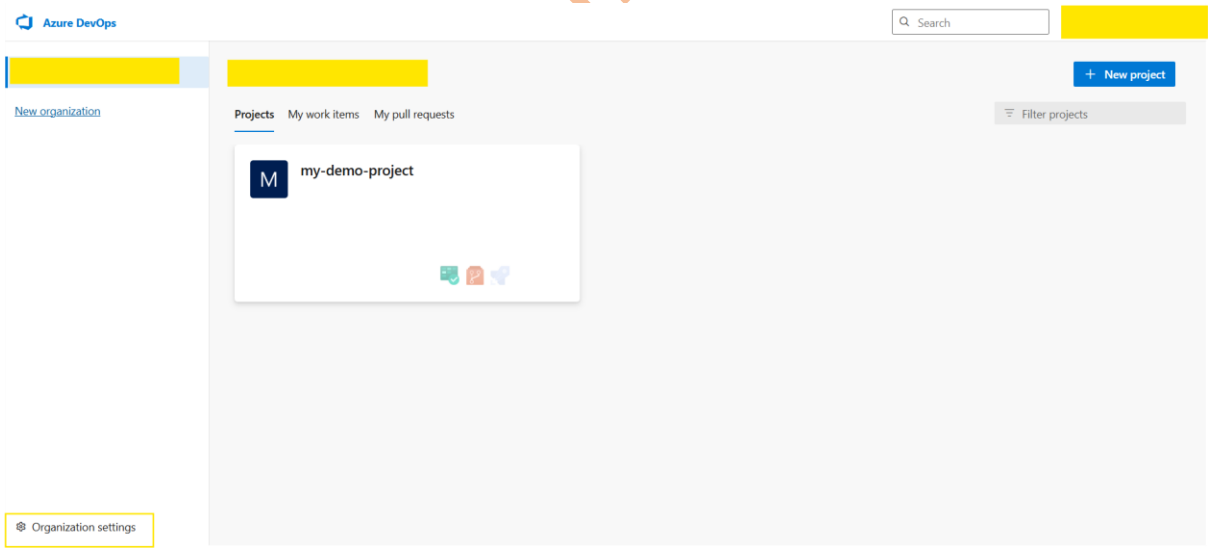


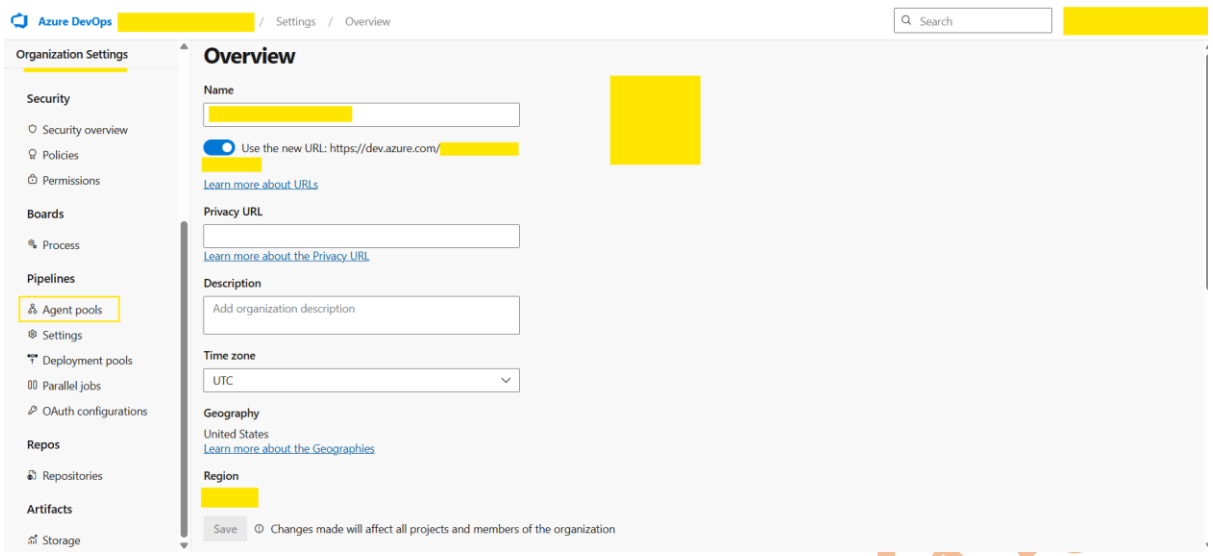

```
[root@Terraform-Server ~]# kubectl get nodes
NAME                                STATUS   ROLES    AGE   VERSION
aks-agentpool-[redacted]-vmss000000   Ready    <none>   52m   v1.30.0
aks-userpool-[redacted]-vmss000000    Ready    <none>   37m   v1.30.0
```

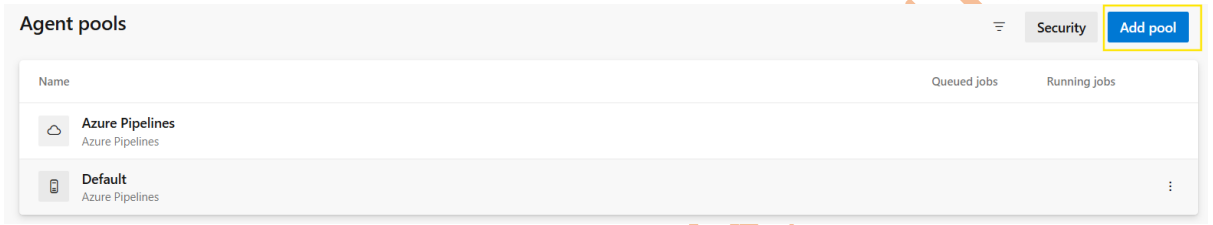
```
[root@Terraform-Server ~]# kubectl get nodes --kubeconfig=/root/.kube/config
NAME                                STATUS   ROLES    AGE   VERSION
aks-agentpool-[redacted]-vmss000000   Ready    <none>   55m   v1.30.0
aks-userpool-[redacted]-vmss000000    Ready    <none>   40m   v1.30.0
```

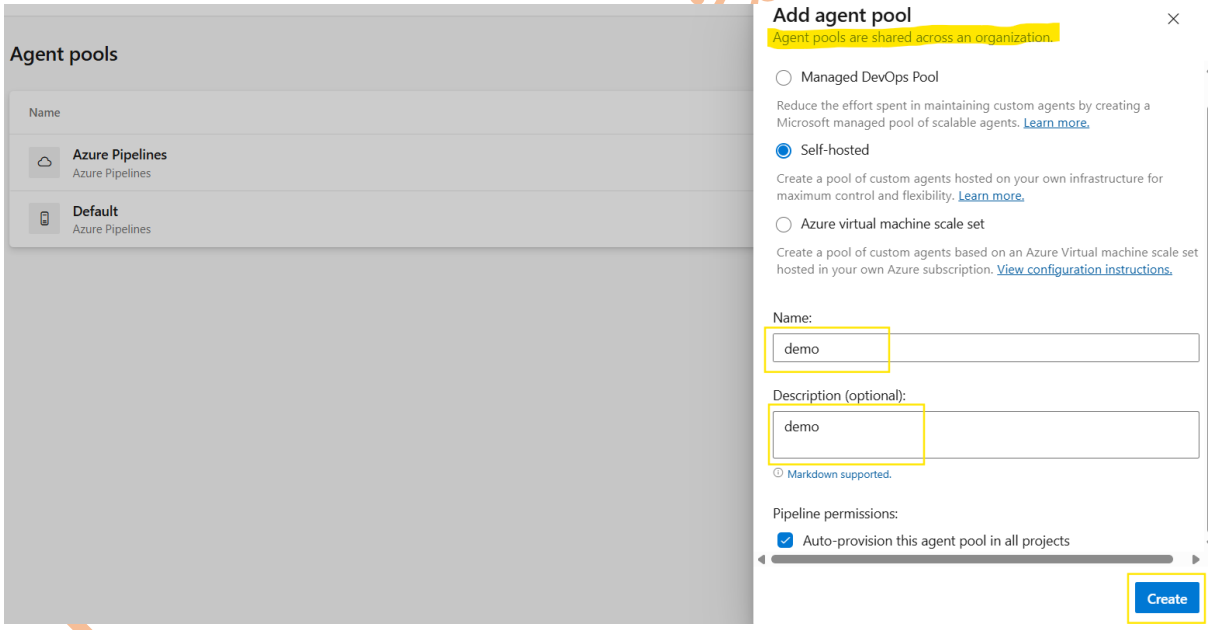
In production for CI/CD pipeline I used Azure DevOps Pipeline. For Azure DevOps Pipeline I had used Self-hosted-Agent and followed the below procedure to install it.

To install the self-hosted agent for Azure DevOps pipeline first I opened the Azure DevOps UI and went to Organisations Settings > Agent Pools > Add Pool.

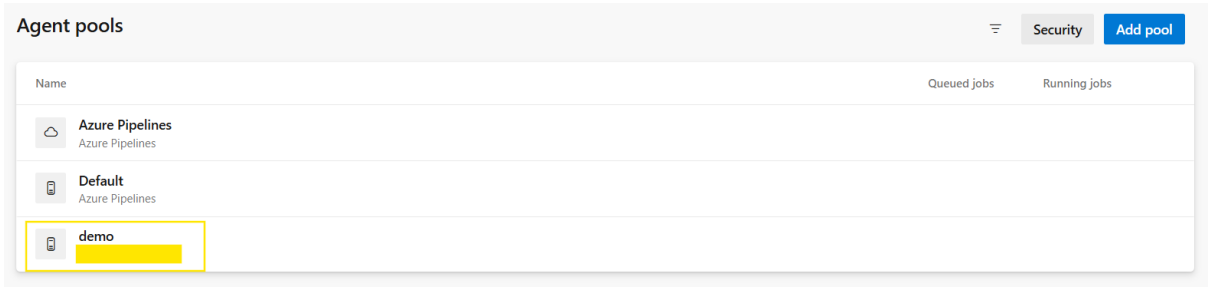




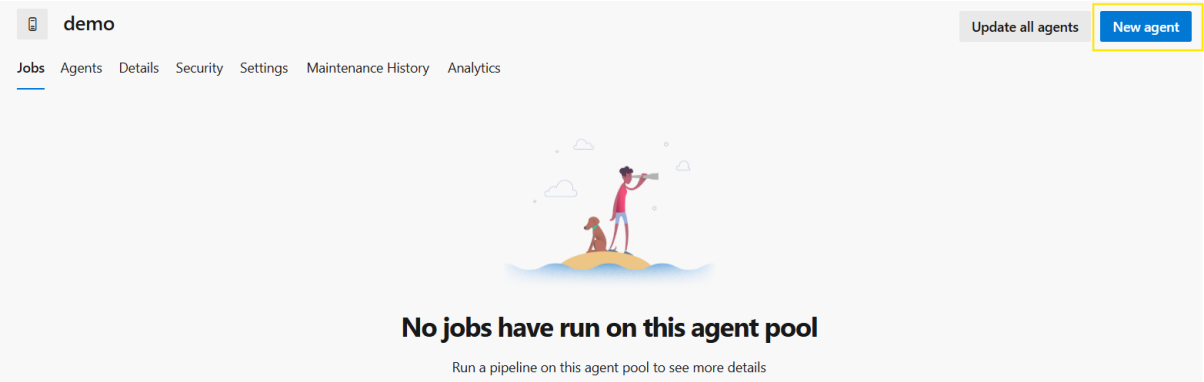




New Agent Pool had been added as shown in the scrrenshot attached below.



Then added the agent as shown in the screenshot attached below.



For Azure DevOps Pipeline I had used Self-hosted-Agent and followed the below procedure to install it.

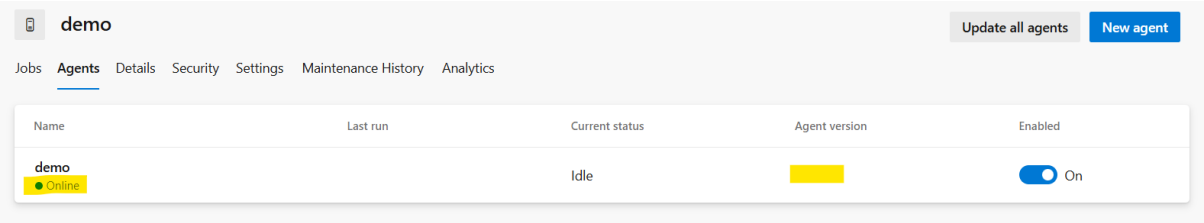
```
[root@devopsagent-vm ~]# cd /opt && mkdir myagent && cd myagent

[root@devopsagent-vm myagent]# wget https://vstsagentpackage.azureedge.net/agent/[redacted]/vsts-agent-linux-x64-[redacted].tar.gz

[root@devopsagent-vm myagent]# tar -xvf vsts-agent-linux-x64-[redacted].tar.gz

[root@devopsagent-vm myagent]# rm -f vsts-agent-linux-x64-[redacted].tar.gz

[root@devopsagent-vm myagent]# ./bin/installdependencies.sh

[demo@devopsagent-vm myagent]$ sudo chown -R demo:demo /opt/myagent/
```

```
[demo@devopsagent-vm myagent]$ ./config.sh

  ___                  ______ _ _
 / _ \                 | ___ (_)        | (_)
/ /_\ \_____ _ _ _ __ ___| |_/ /_ _ __   ___| |_ _ __   ___
|  _  |_  / | | | '__/ _ \  __/| | '_ \ / _ \ | | '_ \ / _ \ __|
| | | |/ /| |_| | | |  __/ |   | | |_) |  __/ | | | | |  __/\__ \
\_| |_/___|\__,_|_|  \___|\_|   |_| .__/ \___|_|_|_| |_|\___||___/
                                   | |
       agent [redacted]            |_|        [redacted]


>> End User License Agreements:

Building sources from a TFVC repository requires accepting the Team Explorer Everywhere End User License Agreement. This step is not required for building sou
rces from Git repositories.

A copy of the Team Explorer Everywhere license agreement can be found at:
  /opt/myagent/license.html

Enter (Y/N) Accept the Team Explorer Everywhere license agreement now? (press enter for N) > Y

>> Connect:

Enter server URL > [redacted]
Enter authentication type (press enter for PAT) >
Enter personal access token > [redacted]
Connecting to server ...

>> Register Agent:

Enter agent pool (press enter for default) > demo
Enter agent name (press enter for devopsagent-vm) > demo
Scanning for tool capabilities.
Connecting to the server.
Successfully added the agent
Testing agent connection.
Enter work folder (press enter for _work) >
2025[redacted]: Settings Saved.
```

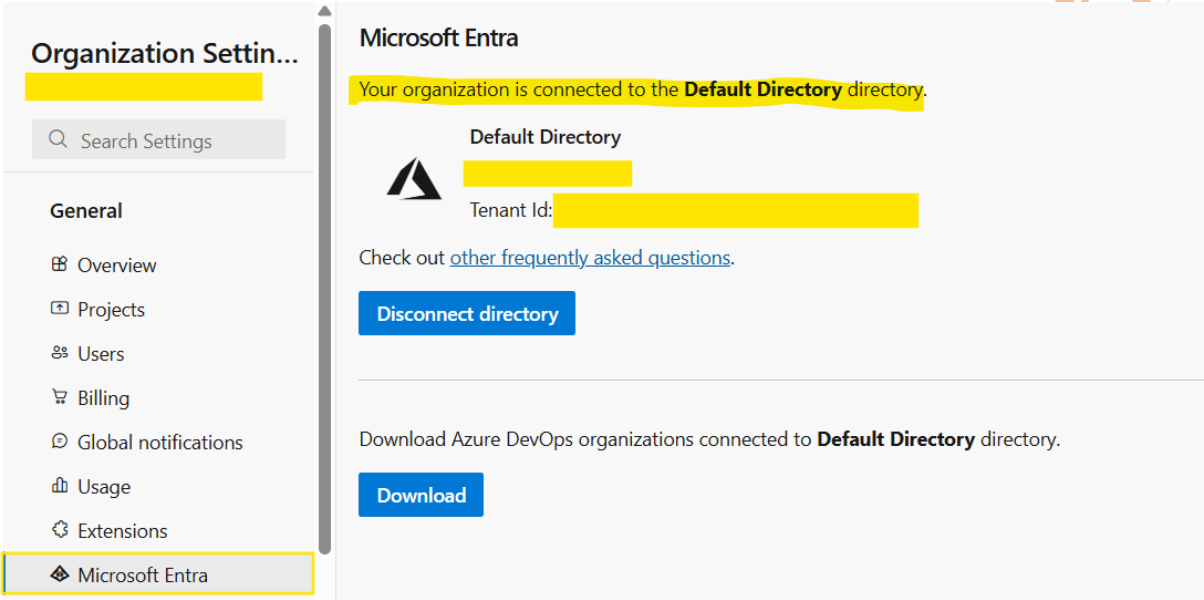
```
[demo@devopsagent-vm myagent]$ ./env.sh
[demo@devopsagent-vm myagent]$
[demo@devopsagent-vm myagent]$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/bin:/opt/sonar-scanner/bin:/opt/apache-maven/bin:/opt/node-v16.0.0/bin:/opt/dependency-check/bin:/usr/local
/bin
```

```
[demo@devopsagent-vm myagent]$ sudo ./svc.sh install

[demo@devopsagent-vm myagent]$ sudo ./svc.sh start
```

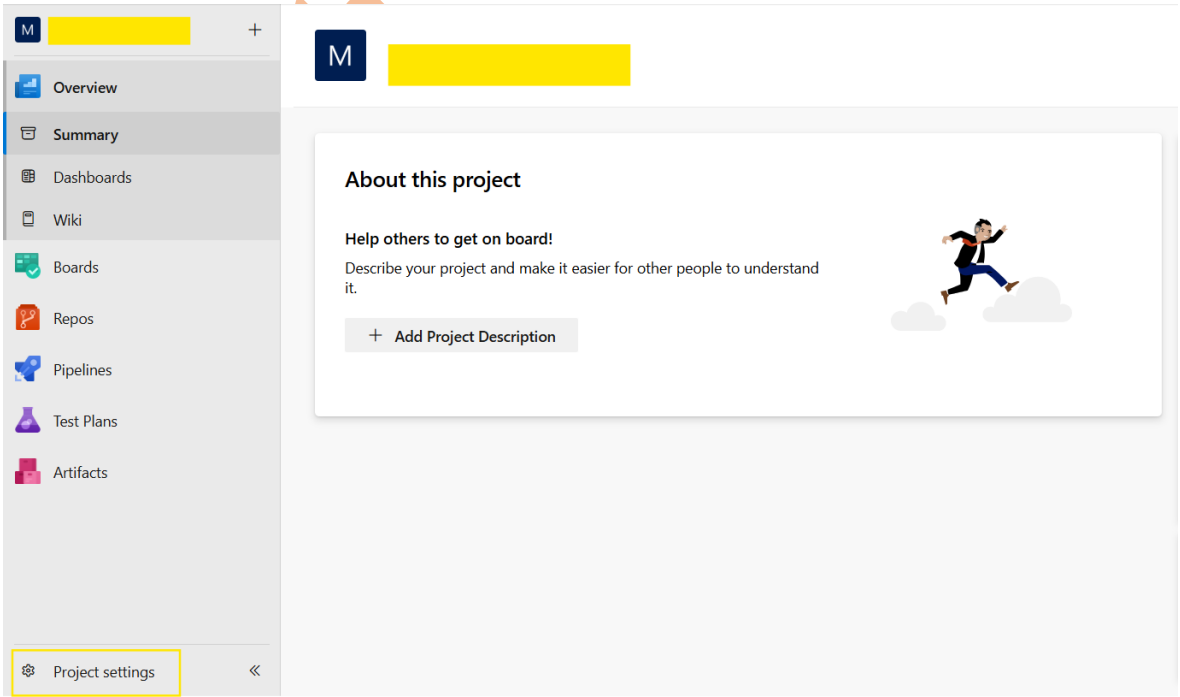
Now the Agent came in online state as shown in the screenshot attached below.

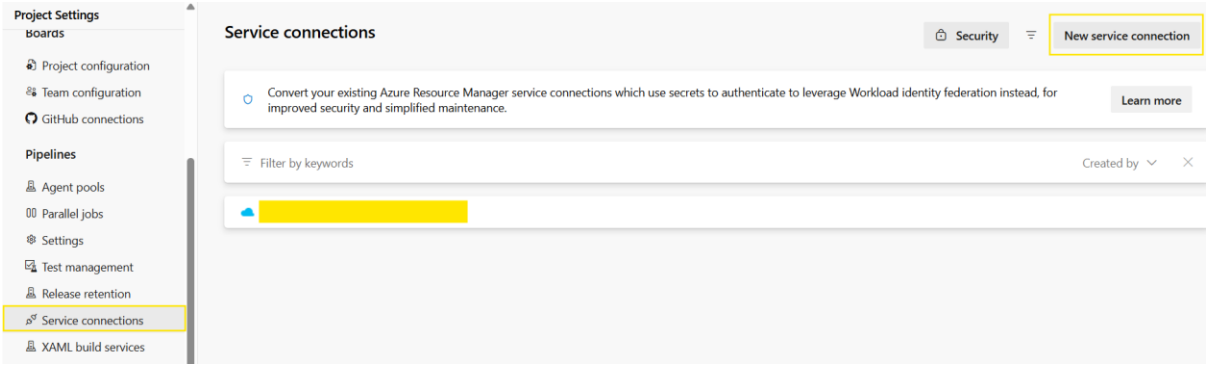


The Azure DevOps account was connected to the Azure Entra ID as shown in the screenshot attached below.

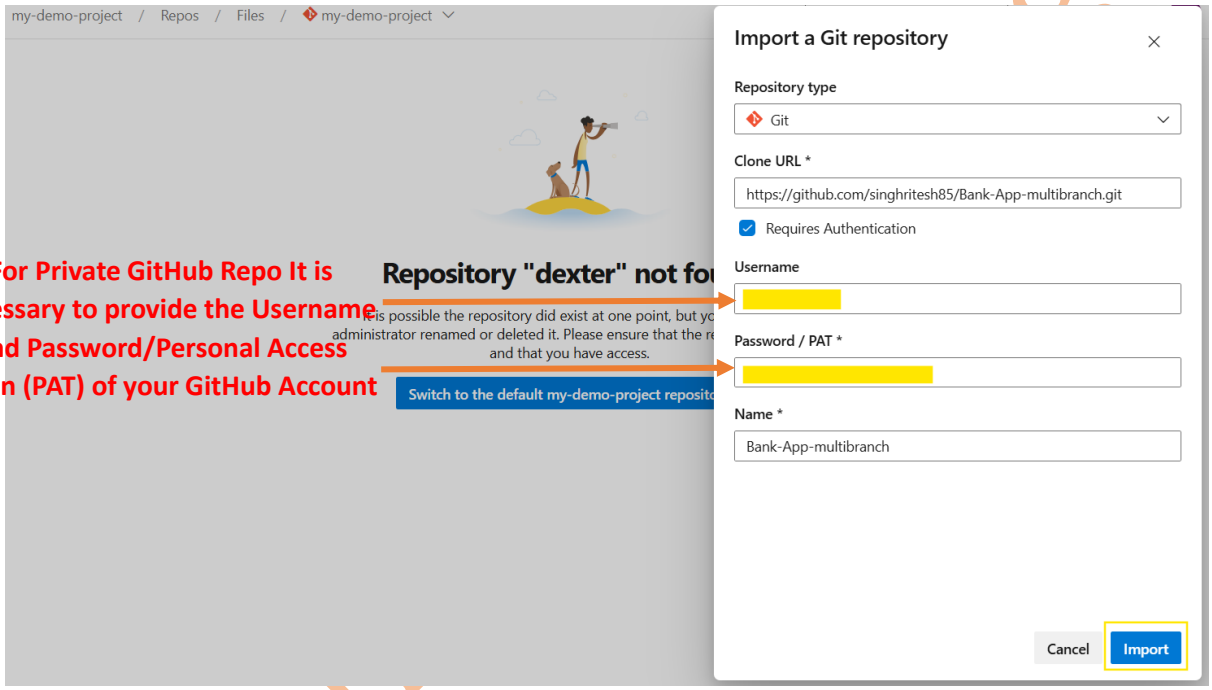


Then I went into the **Azure DevOps project > Project Settings** and created the **Service connections** for GitHub Account, Docker Registry as shown in the screenshot attached blow.



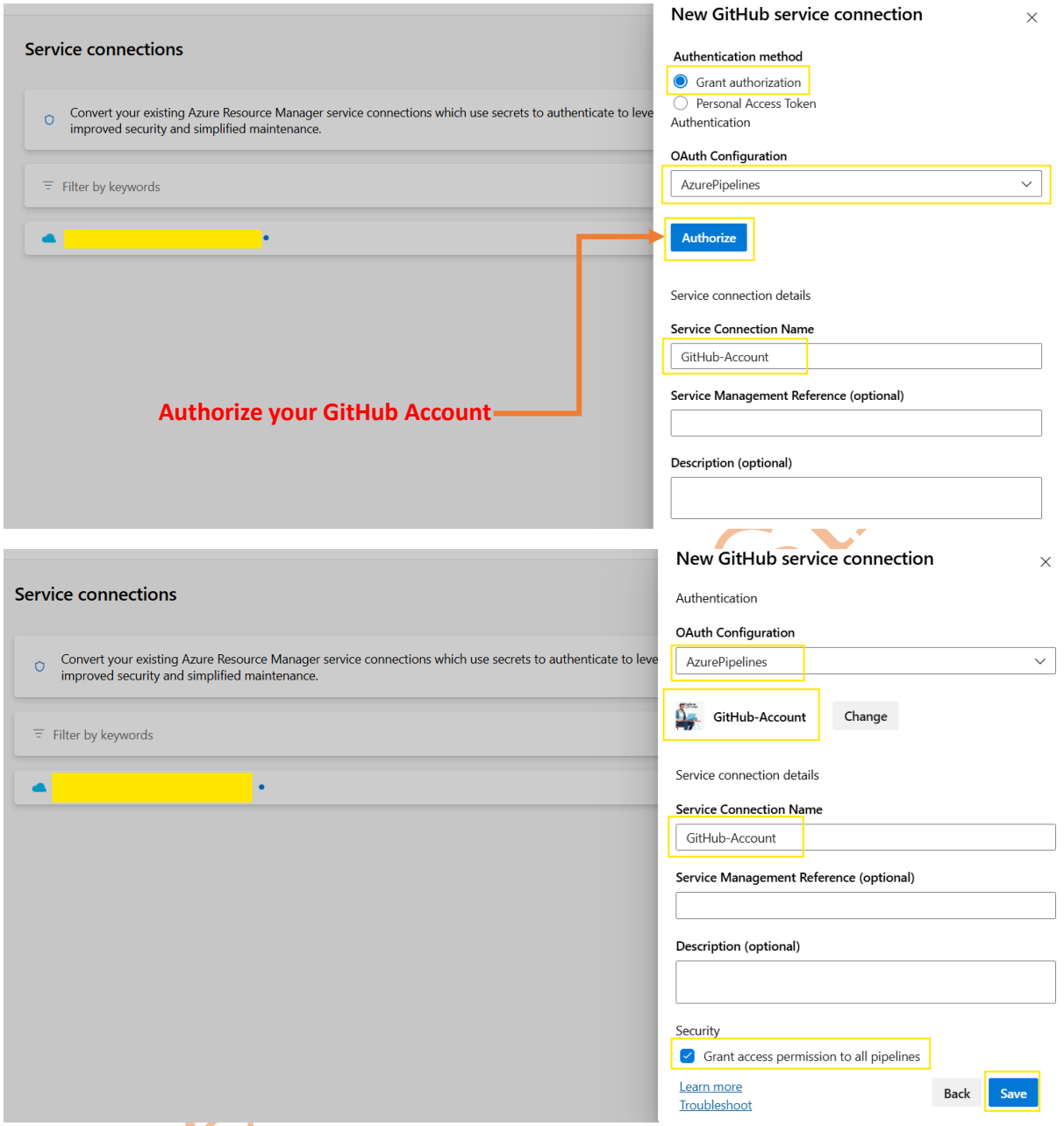


It is possible to Import the GitHub Repo to the Azure DevOps Repo as shown in the screenshot attached below.



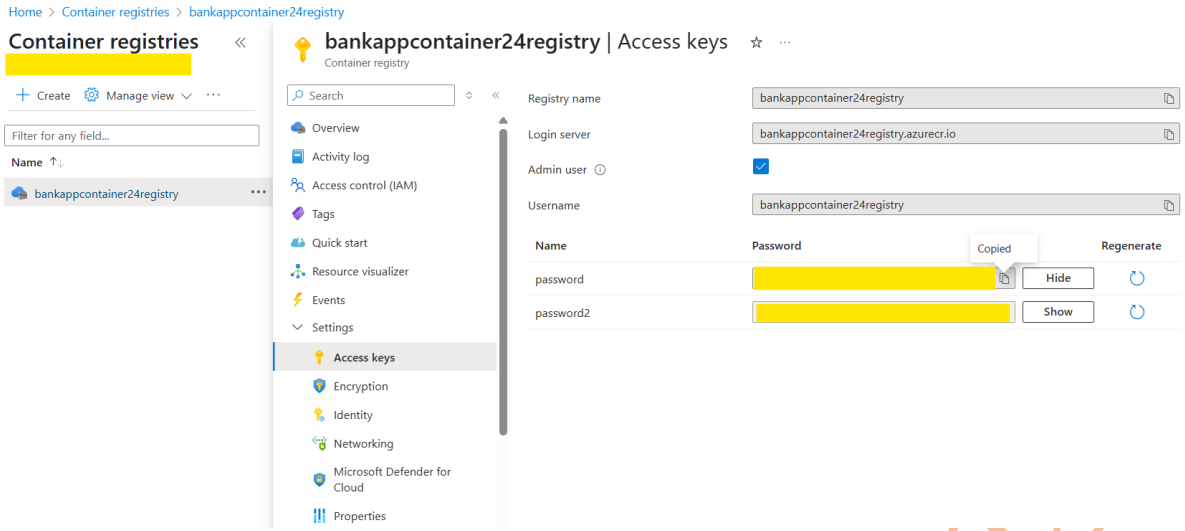
For Private GitHub Repo It is necessary to provide the Username and Password/Personal Access Token (PAT) of your GitHub Account

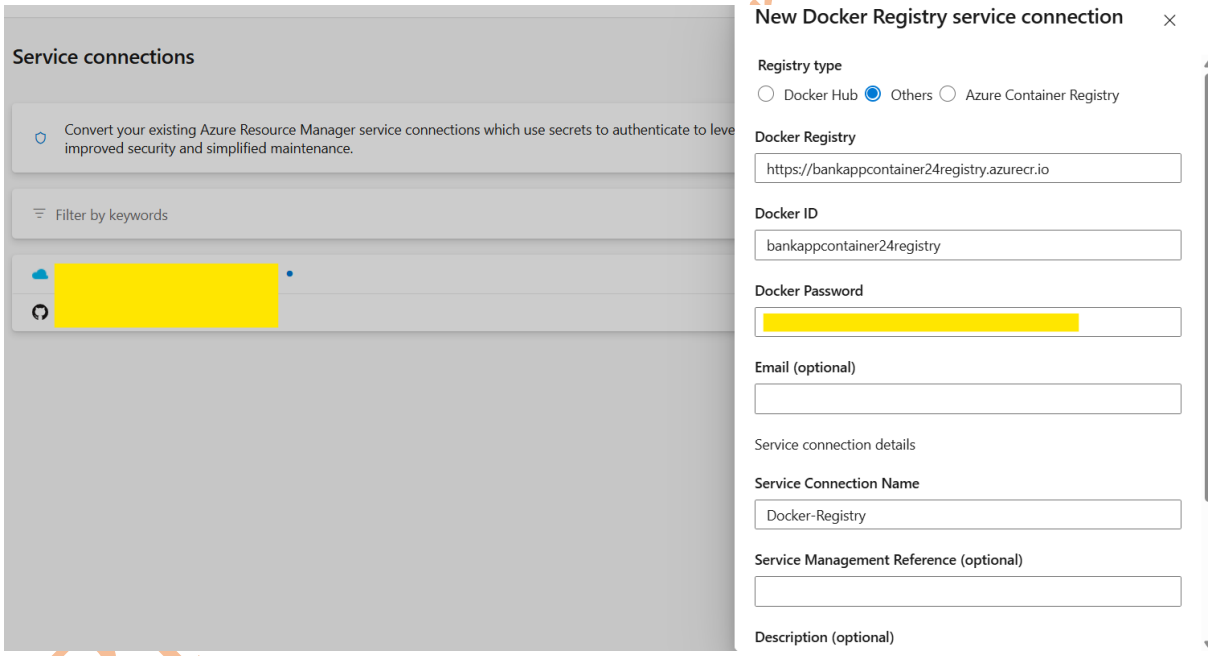
However, for this project I connected GitHub Account with Azure DevOps using the Service Connection as shown in the screenshot attached below.

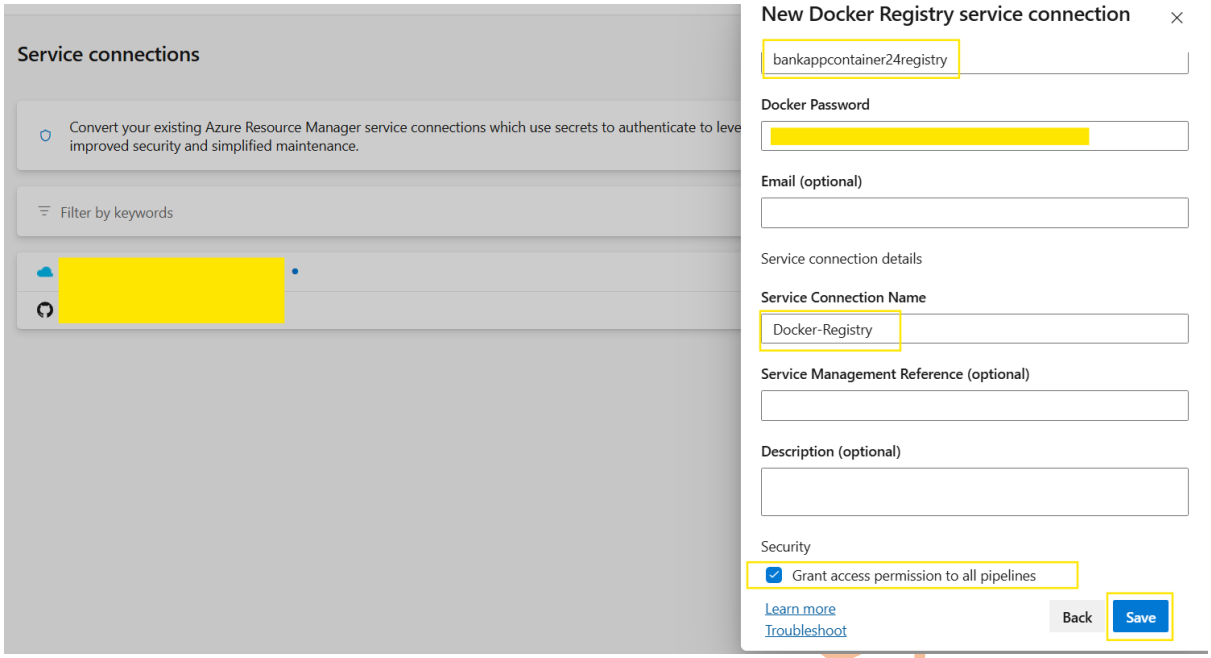


The Azure Container Registry which I had created for this project is as shown in the screenshot attached below.

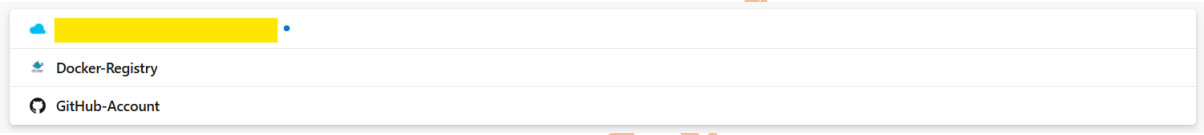
Then I had created the Service Connection for Docker Registry as shown in the screenshot attached below.


Home > Container registries > bankappcontainer24registry
Container registries
Create
Manage view
Filter for any field...
Name
bankappcontainer24registry
bankappcontainer24registry | Access keys
Container registry
Search
Overview
Activity log
Access control (IAM)
Tags
Quick start
Resource visualizer
Events
Settings
Access keys
Encryption
Identity
Networking
Microsoft Defender for Cloud
Properties
Registry name
bankappcontainer24registry
Login server
bankappcontainer24registry.azurecr.io
Admin user
Username
bankappcontainer24registry
Name
Password
Copied
Regenerate
password
Hide
password2
Show


Service connections
Convert your existing Azure Resource Manager service connections which use secrets to authenticate to leve
improved security and simplified maintenance.
Filter by keywords
New Docker Registry service connection
Registry type
Docker Hub
Others
Azure Container Registry
Docker Registry
https://bankappcontainer24registry.azurecr.io
Docker ID
bankappcontainer24registry
Docker Password
Email (optional)
Service connection details
Service Connection Name
Docker-Registry
Service Management Reference (optional)
Description (optional)

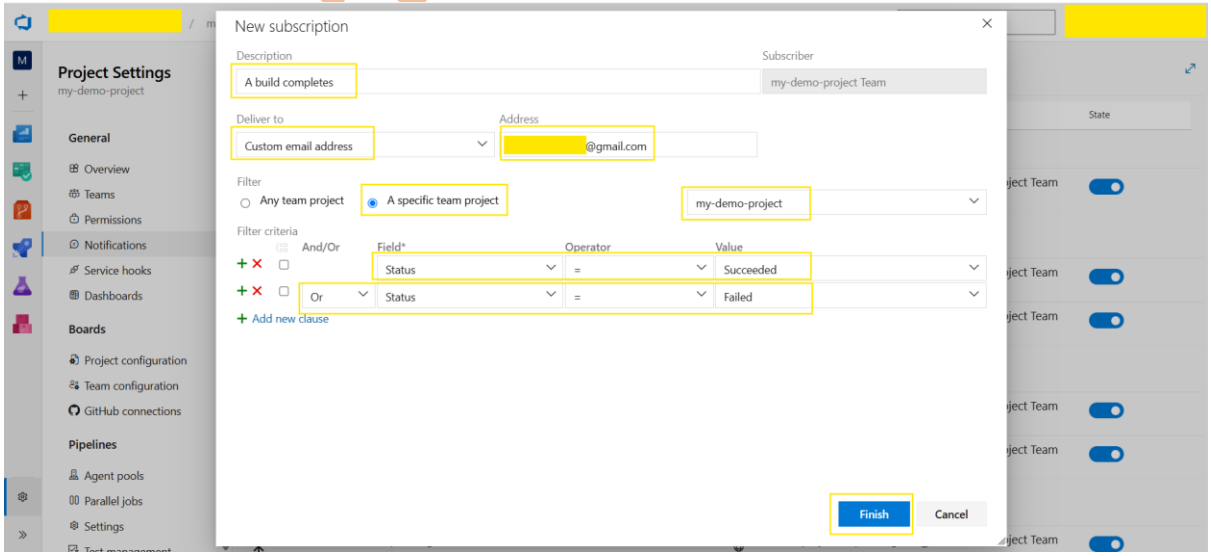


Finaly the service connections had been created as shown in the screenshot attached below.

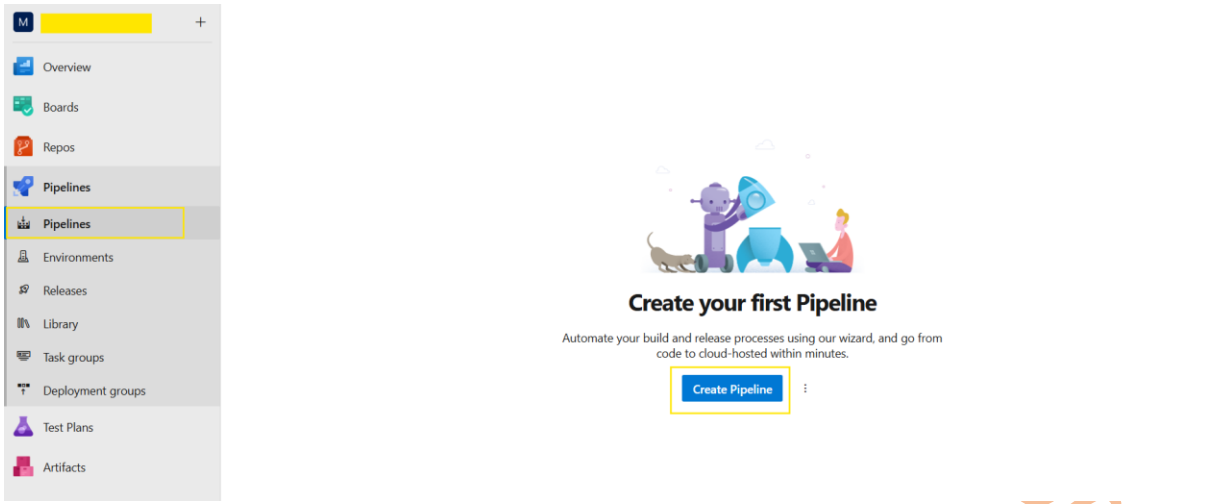


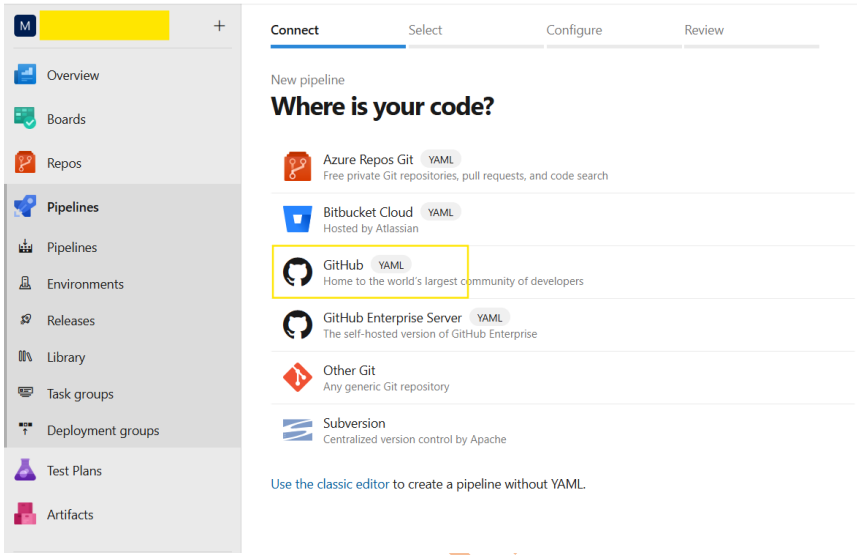
Configuration of Email notification for sending email to group Email Id regarding Success or failure of the Azure DevOps Pipeline

The email notification for sending email for successful or failed pipeline execution to the group email Id had been configured. To configure email notification first go to **Azure DevOps Organisation > Project > Project Settings > Notifications** and do the further configuration as per the screenshot attached below for your reference.

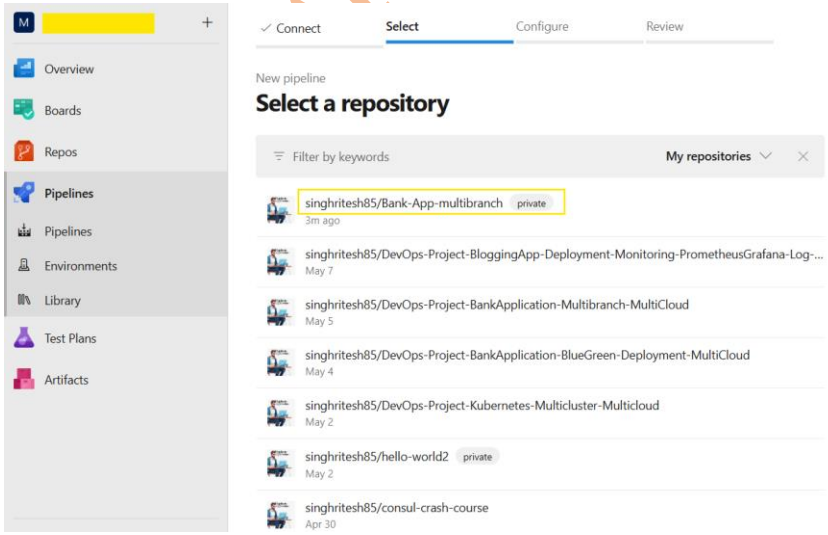


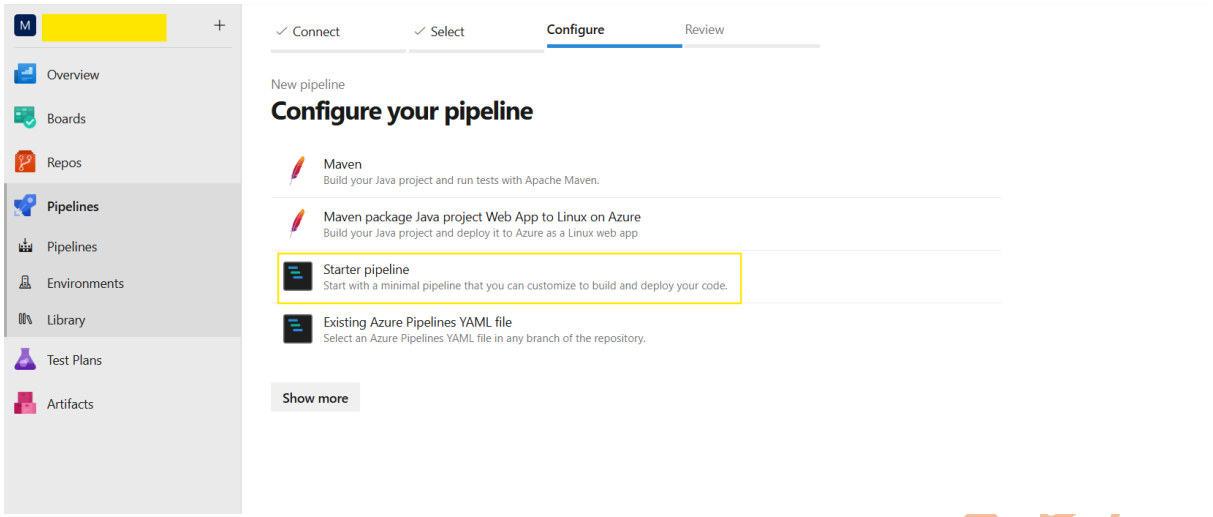
Then I created the Azure DevOps CI/CD pipeline as shown in the screenshot attached below.





At this stage it will ask to authorize your GitHub Account and after authorizing your GitHub Account select the desired GitHub Repo as shown in the screenshot attached below.





If you need to apply webhook in Azure DevOps Pipeline then in **azure-pipelines.yaml** trigger: provide the branch name as shown in the screenshot attached below.

```
trigger:
- main
```

This pipeline triggers for main branch only if you need to apply for other branches then provide the branch names as shown in the screenshot attached below.

```
trigger:
- main
- stage
- dev
```

OpenJDK Docker Image is depricated so in this project eclipse-temurin docker image has been used as a base image in the Dockerfile.

To access Docker Image from the Azure Container Registry during Kubernetes Deployment I created a kubernetes Secrets as shown in the screenshot attached below.

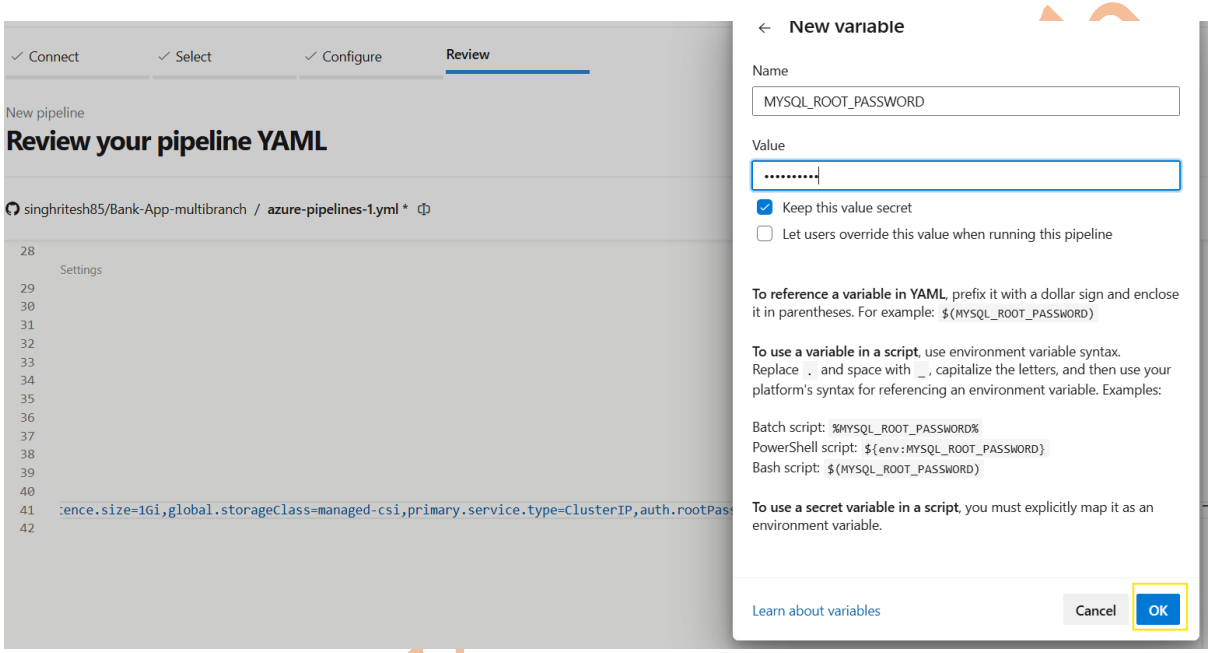
kubectl create secret docker-registry bankapp-auth --docker-server=https://bankappcontainer24registry.azurecr.io --docker-username=bankappcontainer24registry --docker-password=XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX -n bankapp

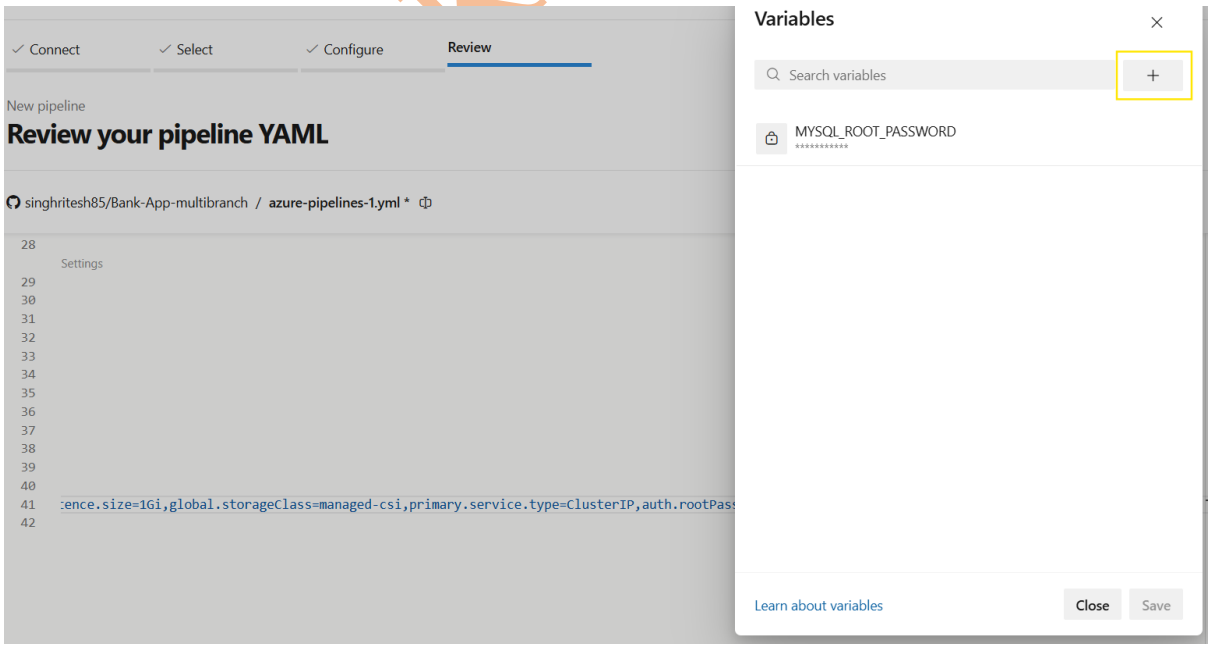
```
[root@                ~]# kubectl create ns bankapp
namespace/bankapp created
[root@                ~]# kubectl create secret docker-registry bankapp-auth --docker-server=https://bankappcontainer24registry.azurecr.io --docker-username=ba
nkappcontainer24registry --docker-password=                                  -n bankapp
secret/bankapp-auth created
```

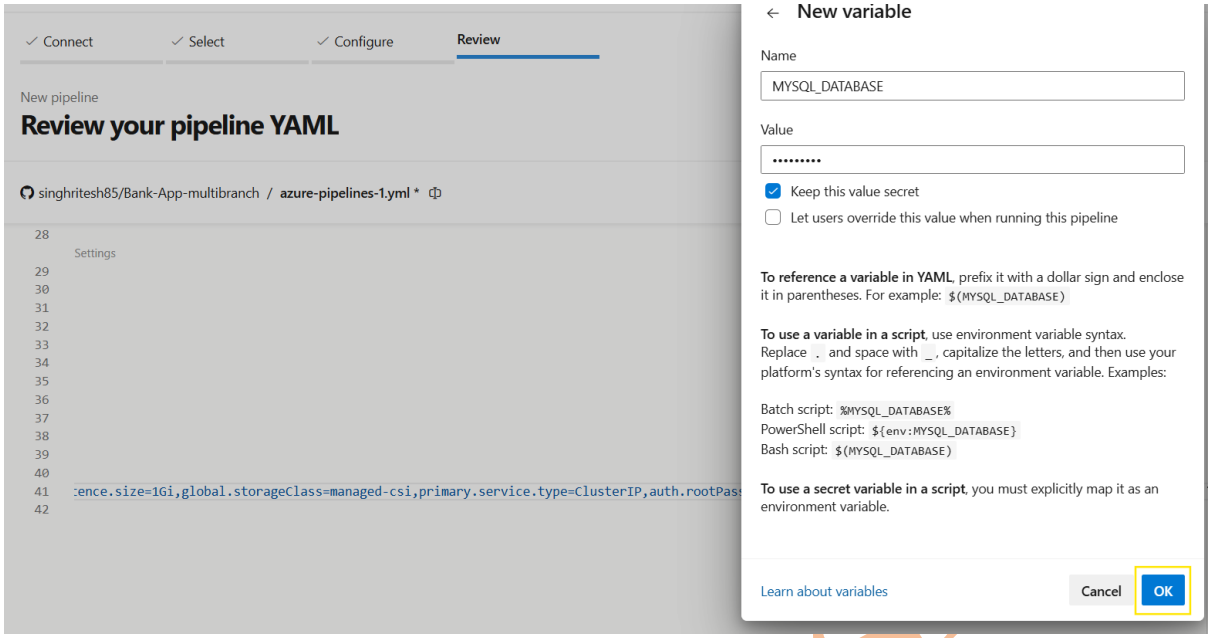
I cloned the GitHub Repo https://github.com/singhritesh85/helm-repo-for-ArgoCD.git on Azure DevOps Agent Server as shown in the screenshot attached below.

```
[demo@devopsagent-vm ~]$ git clone https://github.com/singhritesh85/helm-repo-for-ArgoCD.git
Cloning into 'helm-repo-for-ArgoCD'...
remote: Enumerating objects: 17, done.
remote: Counting objects: 100% (17/17), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 17 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (17/17), 9.65 KiB | 9.65 MiB/s, done.
```

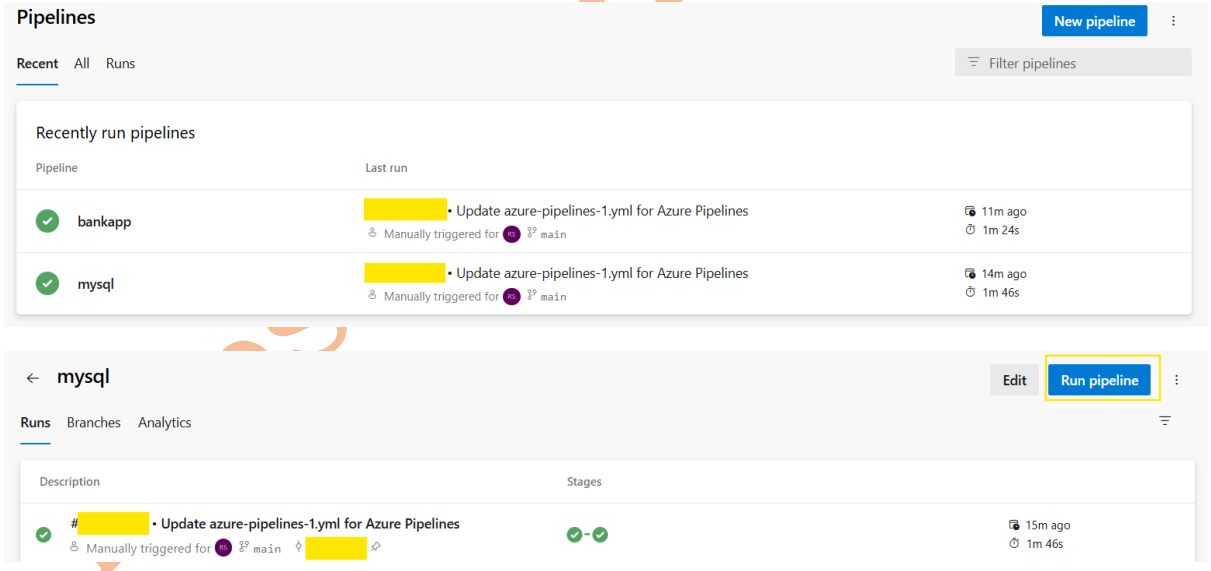
For this project I created two Pipelines in Azure DevOps. It is not a good practice to hardcode the variables in the Azure DevOps Pipeline. To run the Azure DevOps Pipeline for **mysql** I created the variable **MYSQL_ROOT_PASSWORD** and **MYSQL_DATABASE** as shown in the screenshot attached below.

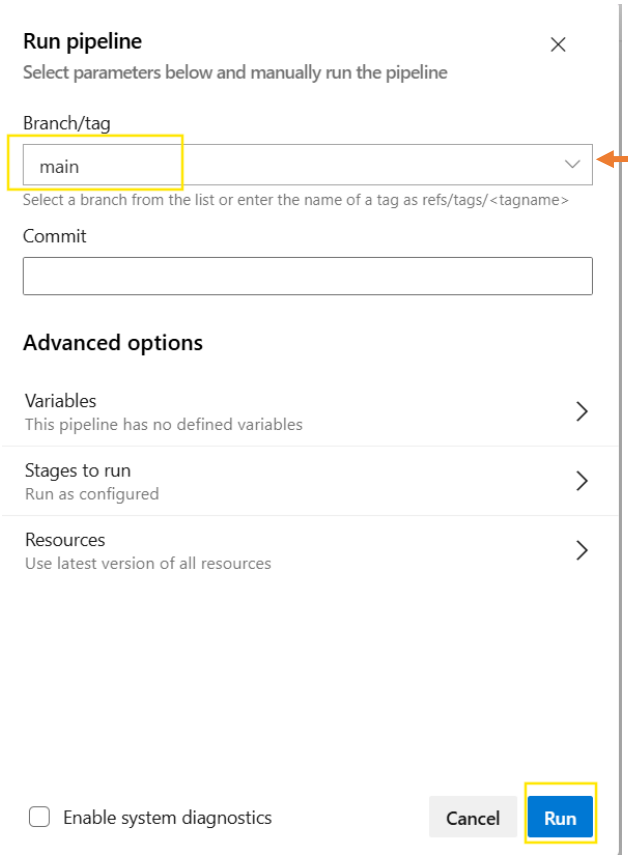




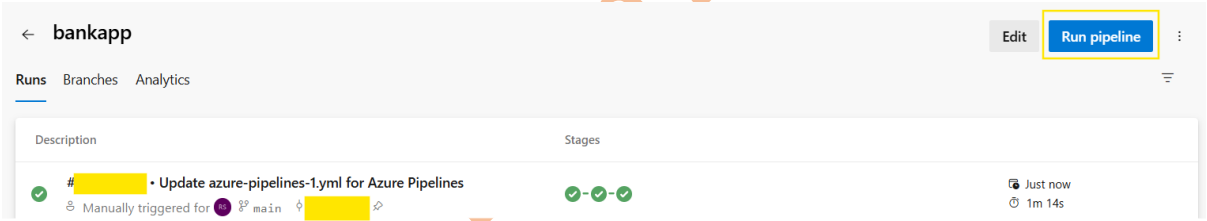


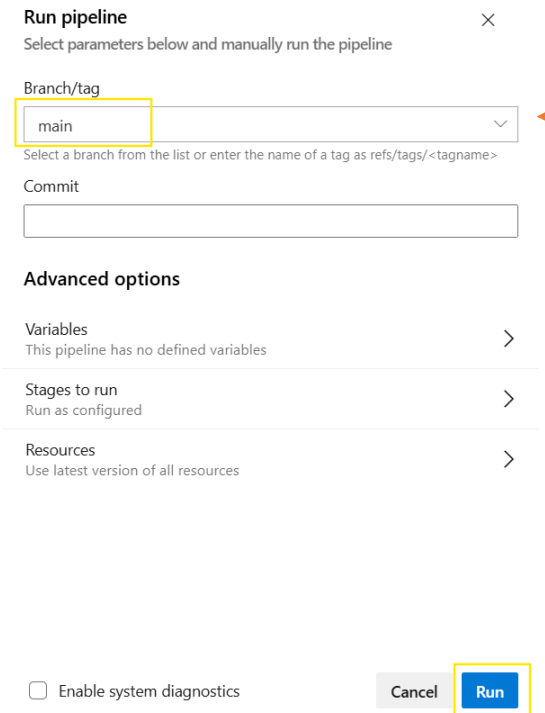
The two Azure DevOps pipelines named as **mysql** and **bankapp** had been created and ran successfully. First run **mysql** pipeline and then **bankapp** pipeline. As I had not applied webhook in production and so the DevOps Team needs to run it manually. If you want to apply the webhook in production then use the method which I used to apply the webhook in production for Azure DevOps Pipeline (with the help of **trigger:**). Below screenshot shows how to run the Azure DevOps Pipeline manually.





Select the branch name





Select the branch name

Finally, the pods had been created as shown in the screenshot attached below.

```
[demo@devopsagent-vm ~]$ kubectl get pods -n bankapp --watch
NAME                          READY   STATUS    RESTARTS   AGE
bankapp-folo-                 1/1     Running   0          100s
^C[demo@devopsagent-vm ~]$
[demo@devopsagent-vm ~]$ kubectl get pods -n mysql --watch
NAME                READY   STATUS    RESTARTS        AGE
mysql-primary-0     1/1     Running   0               10m
mysql-secondary-0   1/1     Running   1 (7m47s ago)   10m
^C[demo@devopsagent-vm ~]$
[demo@devopsagent-vm ~]$
[demo@devopsagent-vm ~]$ kubectl get pvc -n mysql --watch
NAME                    STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
data-mysql-primary-0    Bound    pvc-                                       1Gi        RWO            managed-csi    <unset>                 10m
data-mysql-secondary-0  Bound    pvc-                                       1Gi        RWO            managed-csi    <unset>                 10m
^C[demo@devopsagent-vm ~]$
[demo@devopsagent-vm ~]$ kubectl get svc -n bankapp --watch
NAME           TYPE        CLUSTER-IP    EXTERNAL-IP   PORT(S)   AGE
bankapp-folo   ClusterIP   10.    .29    <none>        80/TCP    2m18s
```

Below screenshot shows the ingress rule which I used to access the Bank Application using the URL.

You can use the Ingress rule which I provided in the GitHub Repo https://github.com/singhritesh85/DevOps-Project-BankApplication-Multibranch-MultiCloud.git with the file name **ingress-rule-prod.yaml**. First you need to create the kubernetes secrets as shown in the screenshot attached below.

kubectl create secret tls ingress-tls --key mykey.key --cert STAR_singhritesh85_com.crt --namespace bankapp --context=bankapp-cluster

```
[root@ ~]# kubectl create secret tls ingress-tls --key mykey.key --cert STAR_singhritesh85_com.crt --namespace bankapp --context=bankapp-cluster
secret/ingress-tls created
```

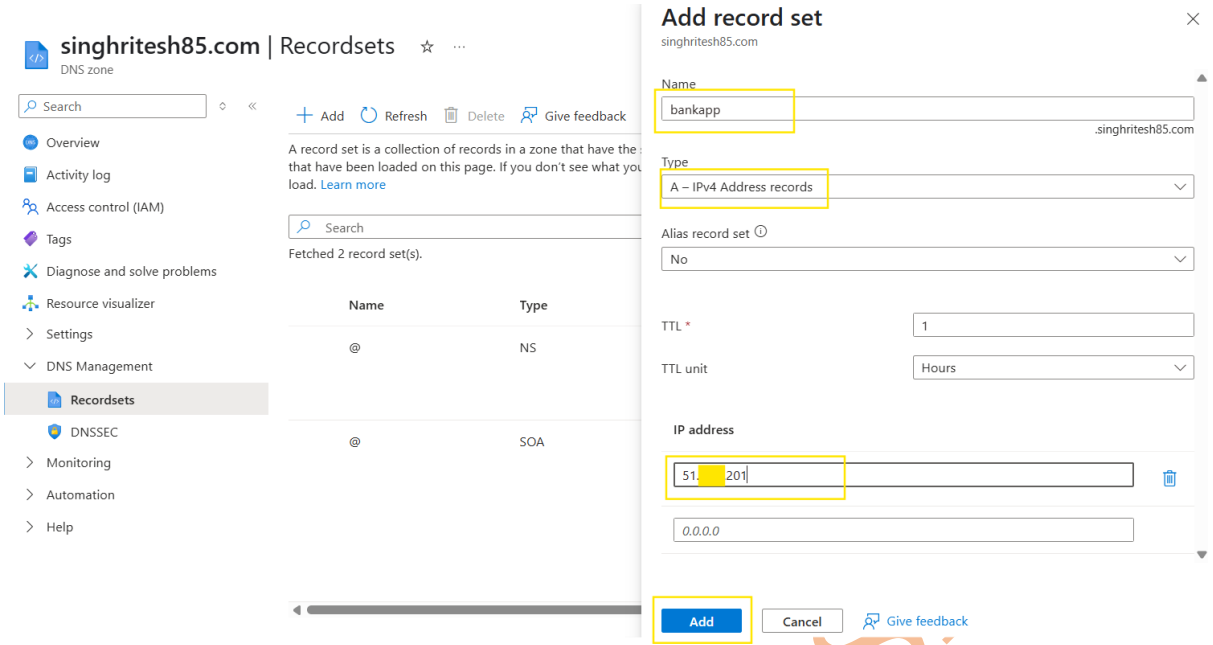
```
[root@ ~]# cat ingress-rule.yaml
# kubectl create secret tls ingress-tls --key mykey.key --cert STAR_singhritesh85_com.crt --namespace bankapp --context=bankapp-cluster
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: bankapp-ingress
  namespace: bankapp
  annotations:
    appgw.ingress.kubernetes.io/ssl-redirect: "true"
spec:
  ingressClassName: azure-application-gateway
  tls:
  - secretName: ingress-tls
  rules:
  - host: bankapp.singhritesh85.com
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: bankapp-folo
            port:
              number: 80
[root@ ~]# kubectl get ing -A --watch
NAMESPACE   NAME              CLASS                       HOSTS                      ADDRESS        PORTS     AGE
bankapp     bankapp-ingress   azure-application-gateway   bankapp.singhritesh85.com   51.    .201   80, 443   17s
```

cat ingress-rule-prod.yaml

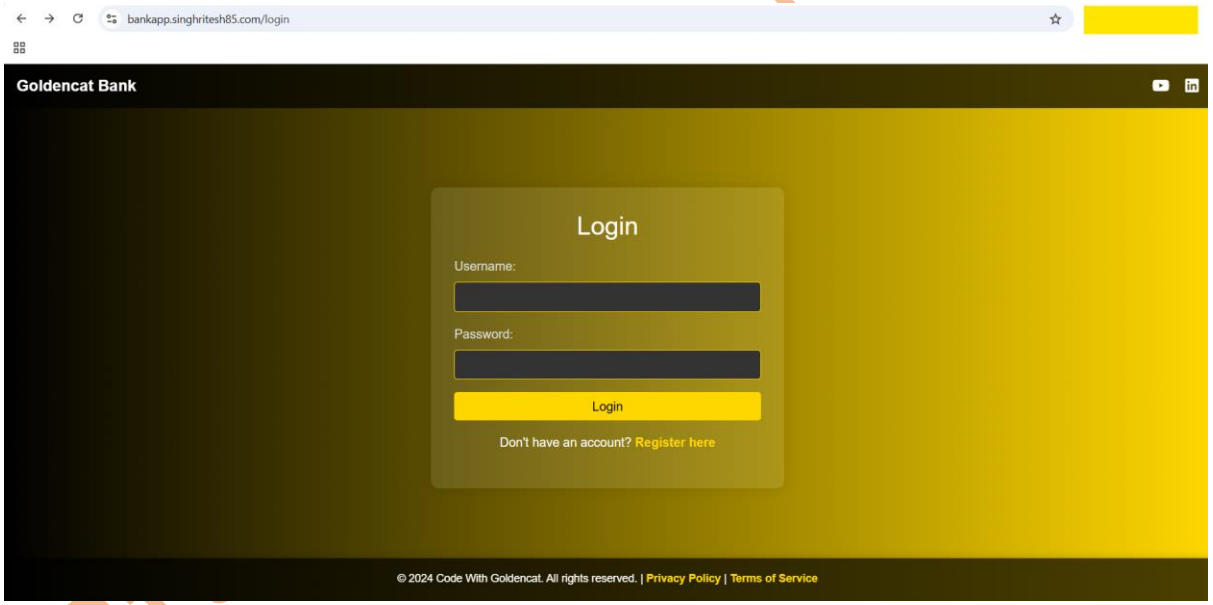
kubectl create secret tls ingress-tls --key mykey.key --cert STAR_singhritesh85_com.crt --namespace bankapp --context=bankapp-cluster

```
---
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
 name: bankapp-ingress
 namespace: bankapp
 annotations:
  appgw.ingress.kubernetes.io/ssl-redirect: "true"
spec:
 ingressClassName: azure-application-gateway
 tls:
 - secretName: ingress-tls
 rules:
 - host: bankapp.singhritesh85.com
  http:
   paths:
   - path: /
    pathType: Prefix
    backend:
     service:
      name: bankapp-folo
      port:
       number: 80
```

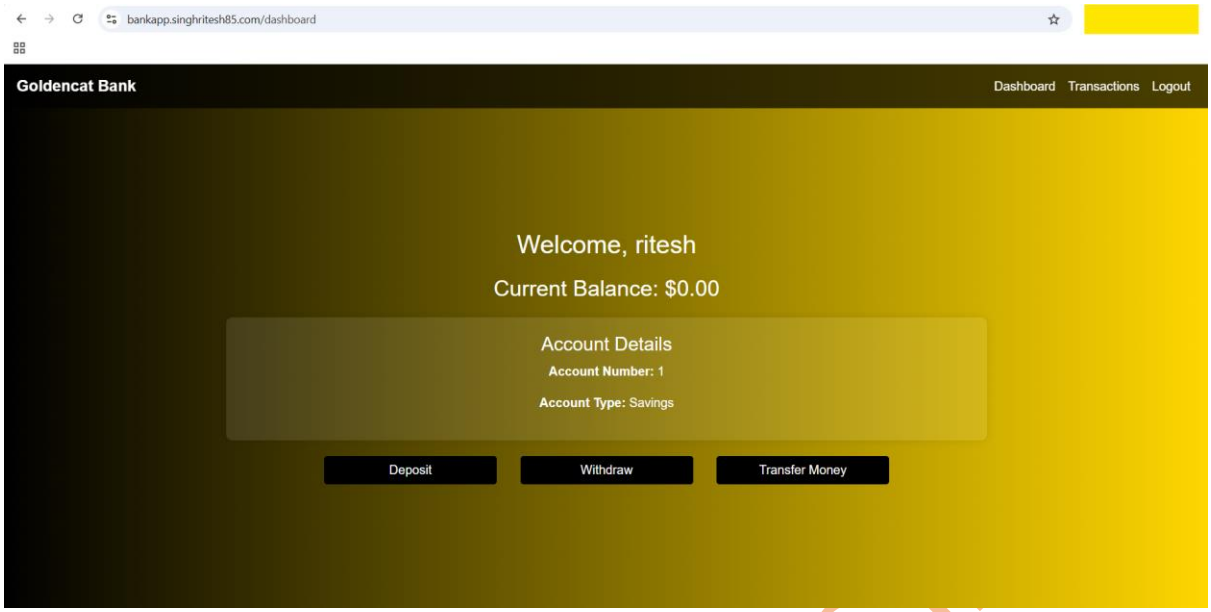
I did the entry for HOST **bankapp.singhritesh85.com** with Public IP Address as shown in the screenshot attached above in the Azure DNS Zone to create the Record Set of A Type.



Finally, I was able to access the Bank Application using the URL as shown in the screenshot attached below.



I registered a user ritesh in the Bank Application as discussed in the non-prod environment and then logged-in with the same user and I had also checked its entry in the MySQL database and found the same as shown in the screenshot attached below.



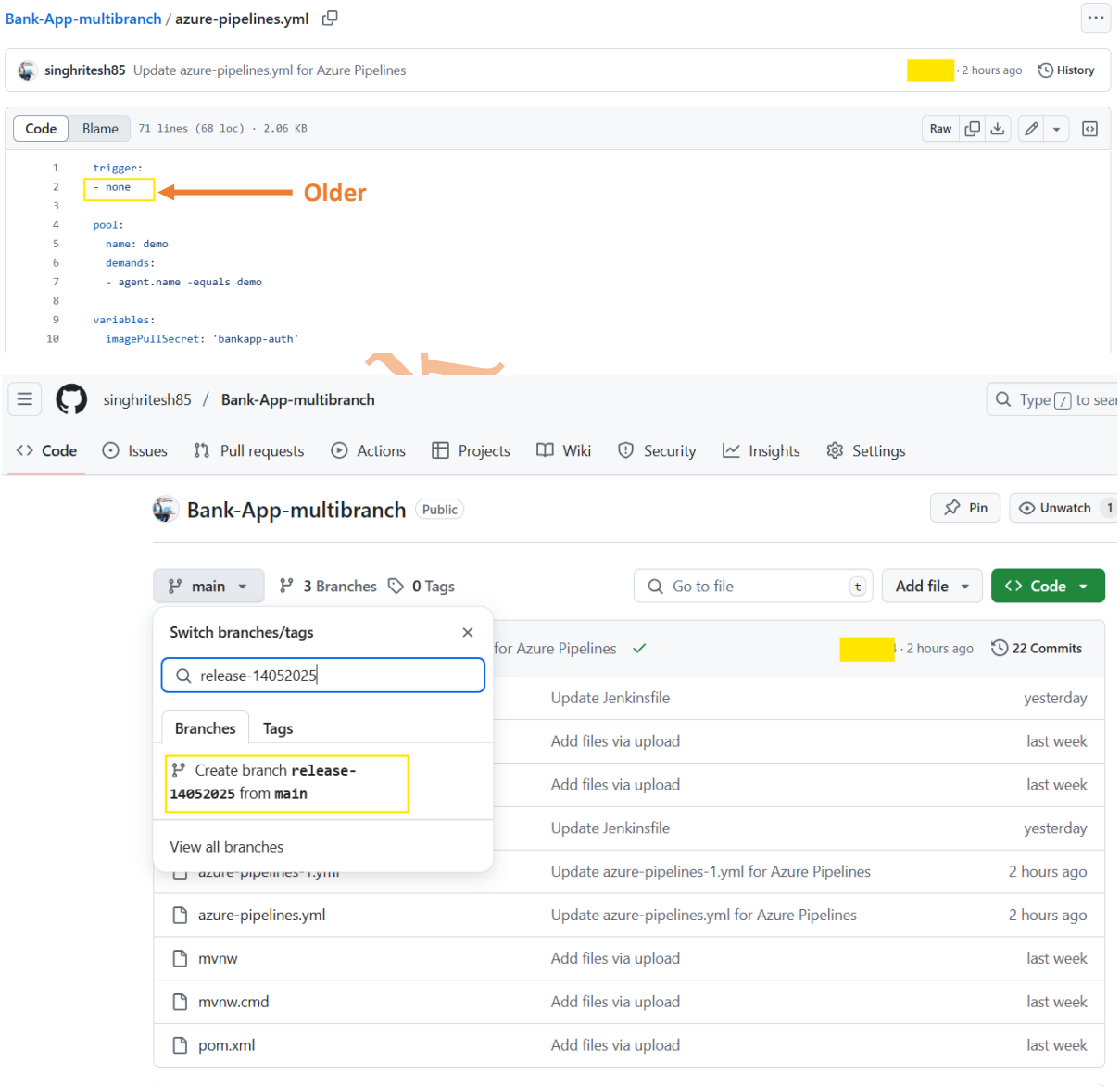

```
[root@                ~]# kubectl exec -it mysql-primary-0 -n mysql /bin/bash
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [POD] -- [COMMAND] instead.
Defaulted container "mysql" out of: mysql, preserve-logs-symlinks (init)
I have no name!@mysql-primary-0:/$ mysql -h localhost -u root --password
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 565
Server version: 8.4.0 Source distribution

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| bankappdb          |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)
```

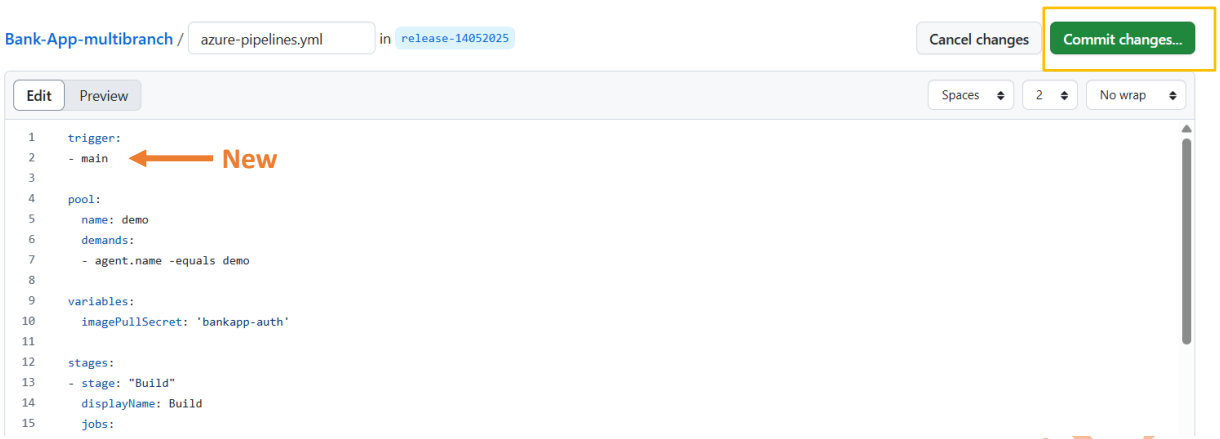
```
mysql> use bankappdb;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+----------------------+
| Tables_in_bankappdb |
+----------------------+
| account              |
| transaction          |
+----------------------+
2 rows in set (0.01 sec)

mysql> select * from account;
+----+---------+--------------------------------------------------------------+----------+
| id | balance | password                                                     | username |
+----+---------+--------------------------------------------------------------+----------+
|  1 |    0.00 |                                                              | ritesh   |
+----+---------+--------------------------------------------------------------+----------+
1 row in set (0.01 sec)
```

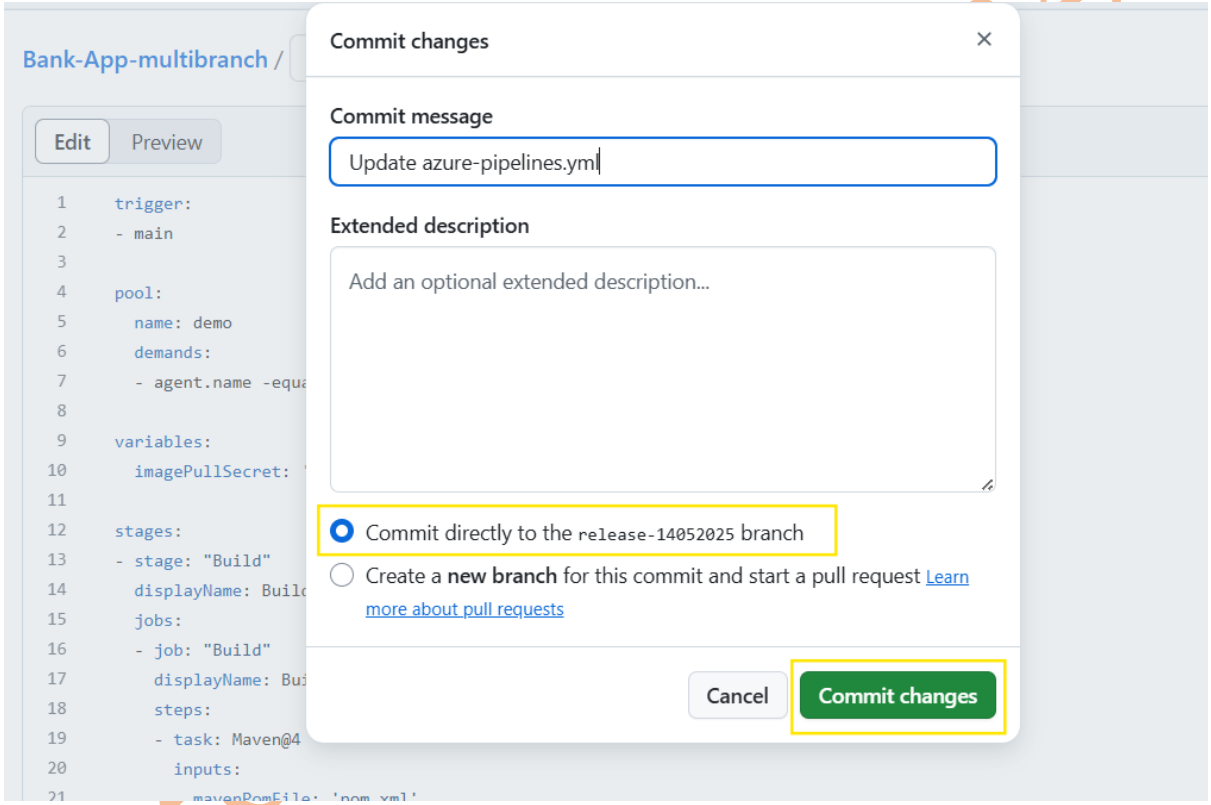
For the below demonstration I created two Azure DevOps Pipeline named as **bankapp** and **mysql**. I had created Azure DevOps Pipeline in such a way that pipeline **mysql** will run manually and pipeline **bankapp** will run with webhook when Pull Request (PR) comes to approver and approver approves the Pull Request (PR).

To run the Azure DevOps Pipeline manually I already discussed above so here I only discuss about **bankapp** pipeline. You can create a feature branch from dev with name feature-14052025 (name with the date extension) and then merge this branch with the stage branch and then to the main branch.

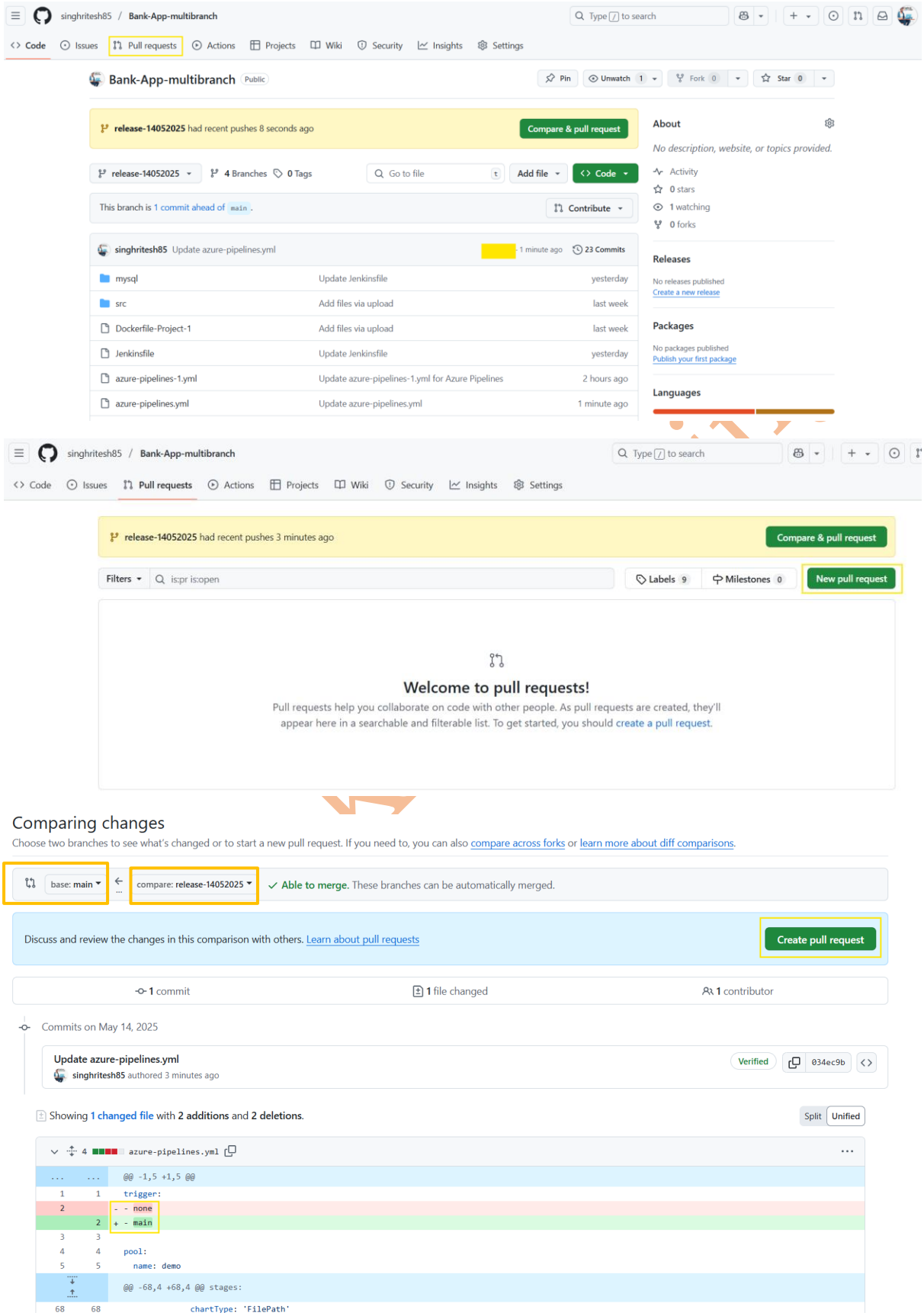
But for this demonstration I will directly create a release branch from main branch itself and then create a PR to merge it with main. In release branch release-14052025 I changed the line trigger: -none to trigger: -main in the file **azure-pipelines.yaml** present in the main branch as shown in the screenshot attached below. It is important to mention here that I kept **pr: none** in the file **azure-pipelines.yaml** and **azure-pipelines-1.yaml** in the **GitHub Repo** https://github.com/singhritesh85/Bank-App-multibranch.git which means whenever a new PR is created in the Repository then Pipeline will not be triggered automatically.

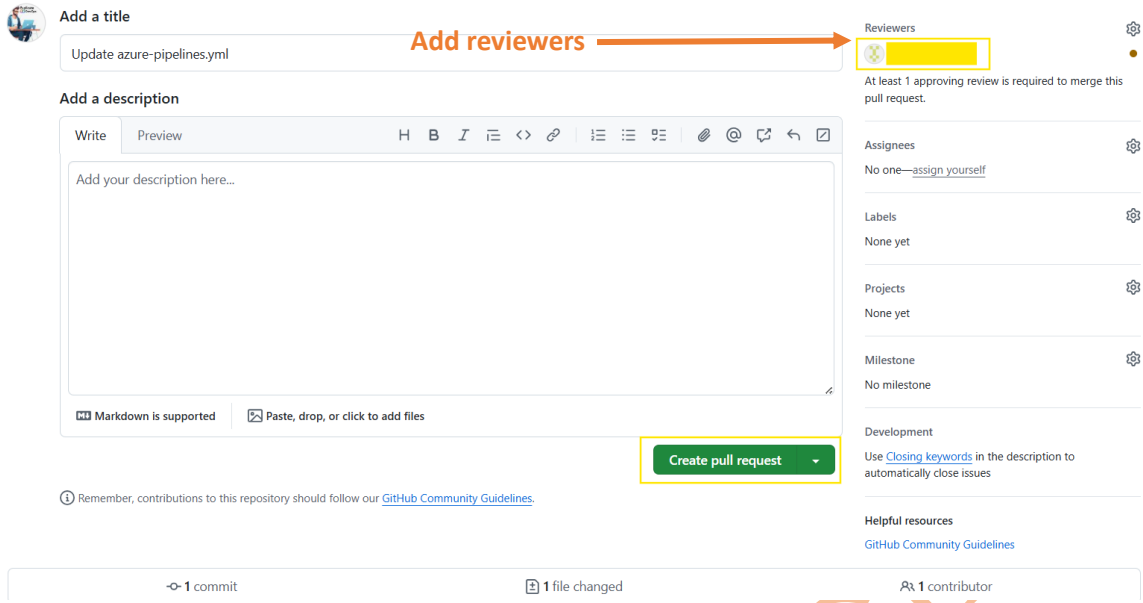




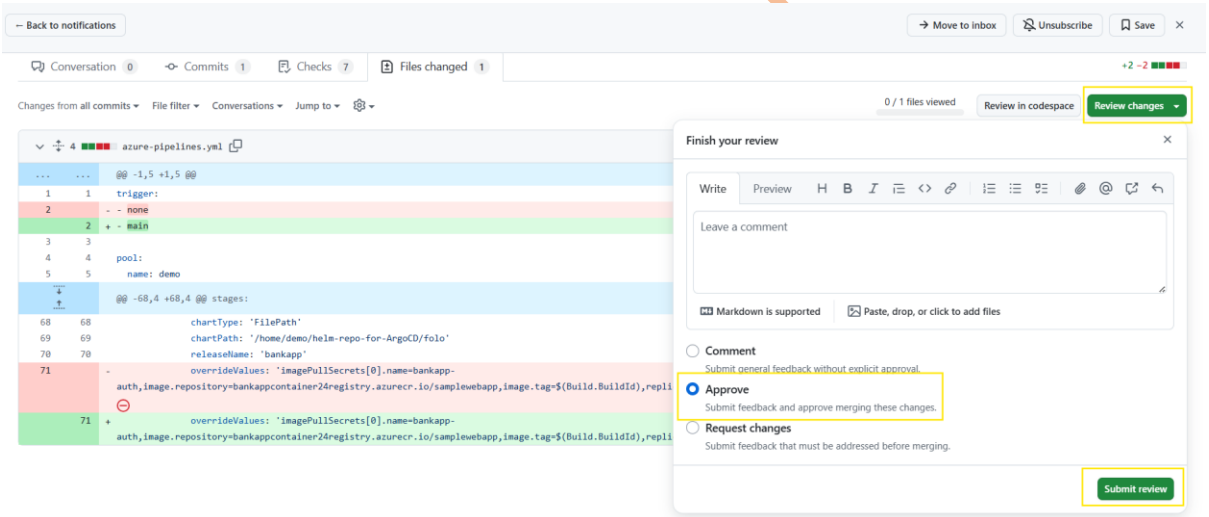


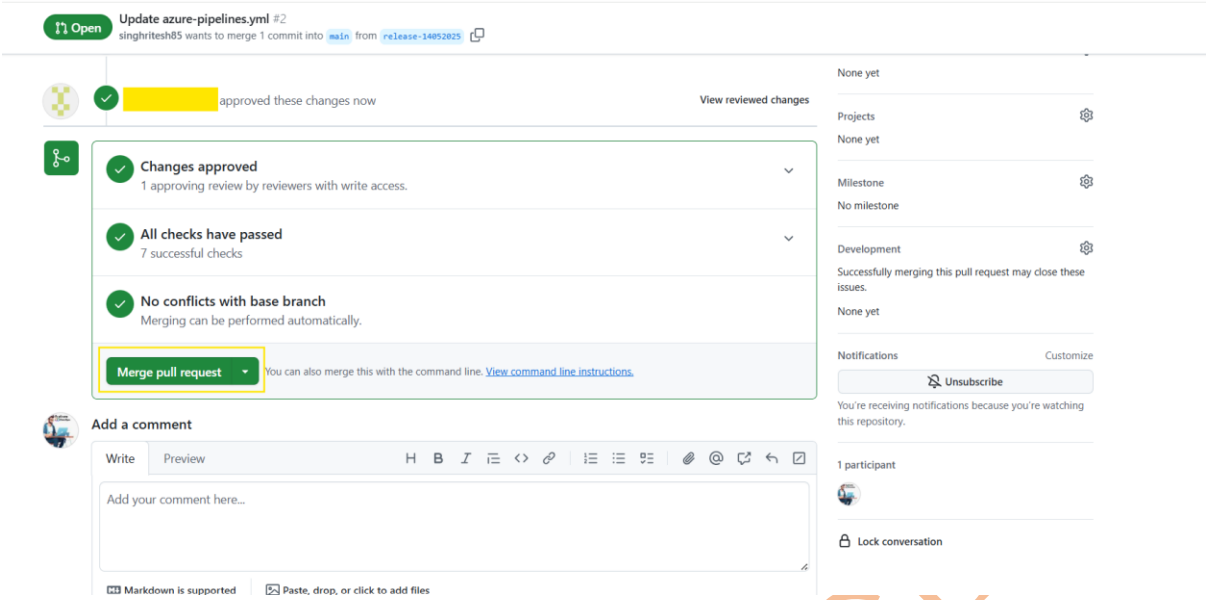
Now to merge the release branch **release-14052023** with the main branch I created a Pull request as shown in the screenshot attached below.


singhritesh85 / Bank-App-multibranch
Type / to search
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
Bank-App-multibranch
Public
Pin
Unwatch 1
Fork 0
Star 0
release-14052025 had recent pushes 8 seconds ago
Compare & pull request
release-14052025
4 Branches
0 Tags
Go to file
Add file
Code
This branch is 1 commit ahead of main.
Contribute
singhritesh85 Update azure-pipelines.yml
1 minute ago
23 Commits
mysql Update Jenkinsfile yesterday
src Add files via upload last week
Dockerfile-Project-1 Add files via upload last week
Jenkinsfile Update Jenkinsfile yesterday
azure-pipelines-1.yml Update azure-pipelines-1.yml for Azure Pipelines 2 hours ago
azure-pipelines.yml Update azure-pipelines.yml 1 minute ago
About
No description, website, or topics provided.
Activity
0 stars
1 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package
Languages
release-14052025 had recent pushes 3 minutes ago
Compare & pull request
Filters
is:pr is:open
Labels 9
Milestones 0
New pull request
Welcome to pull requests!
Pull requests help you collaborate on code with other people. As pull requests are created, they'll appear here in a searchable and filterable list. To get started, you should create a pull request.
Comparing changes
Choose two branches to see what's changed or to start a new pull request. If you need to, you can also compare across forks or learn more about diff comparisons.
base: main
compare: release-14052025
Able to merge. These branches can be automatically merged.
Discuss and review the changes in this comparison with others. Learn about pull requests
Create pull request
1 commit
1 file changed
1 contributor
Commits on May 14, 2025
Update azure-pipelines.yml
singhritesh85 authored 3 minutes ago
Verified
034ec9b
Showing 1 changed file with 2 additions and 2 deletions.
Split
Unified
azure-pipelines.yml
@@ -1,5 +1,5 @@
trigger:
- none
+ main
pool:
name: demo
@@ -68,4 +68,4 @@ stages:
chartType: 'FilePath'



At this stage you need to inform your reviewer, however for this pull request reviewer got notification on their email id. The reviewer will review the same PR and after his/her approval you can merge this pull request as shown in the screenshot attached below.

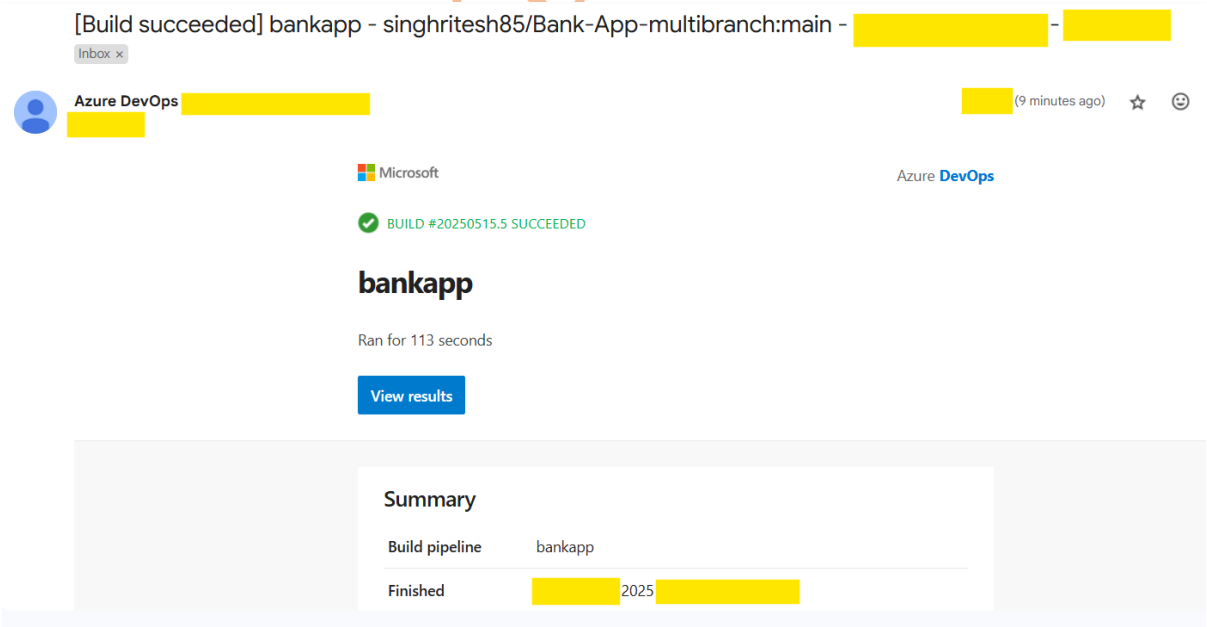


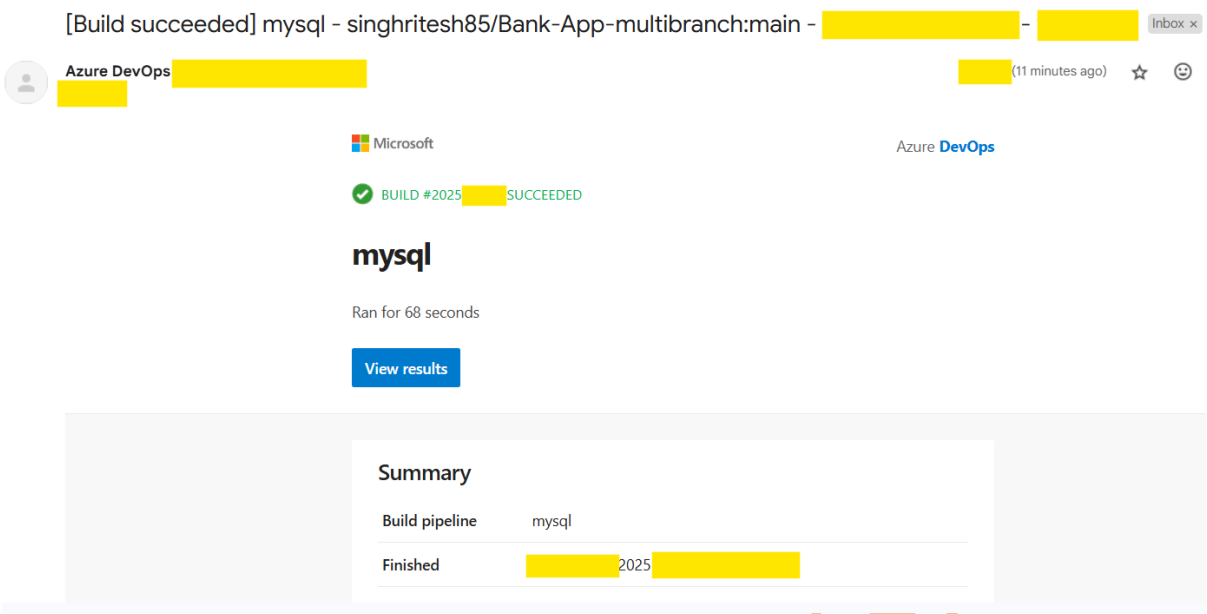


In the **azure-pipelines.yaml** file I changed the trigger: - none to trigger: -main and hence bankapp pipeline had been triggered and application pods had been created as shown in the screenshot attached below.

```
[root@                    ~]# kubectl get pods -n bankapp
NAME                          READY   STATUS    RESTARTS   AGE
bankapp-folo-                 1/1     Running   0          51s
```

After success execution of the Azure DevOps pipelines Email had been triggered to the group Email Id as shown in the screenshot attached below.





I had provided restricted access to the deployment user **demo** using Service Account, Role and Role Binding as shown in the screenshot attached below. The deployment user had all the accesses in the namespaces **bankapp** and **mysql** but does not had access for the entire AKS cluster. That means deployment user **demo** access was restricted to the namespaces **bankapp** and **mysql** in the AKS Cluster.

```
[root@ ~]# kubectl apply -f sa-role-rolebinding-prod.yaml
namespace/bankapp unchanged
serviceaccount/devops-agent-bankapp-aks created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding created
```

```
[root@ ~]# kubectl apply -f secrets-prod.yaml
secret/mysecretname-aks created
```

```
[root@ ~]# kubectl apply -f sa-role-rolebinding-mysql-prod.yaml
namespace/mysql unchanged
serviceaccount/devops-agent-mysql-aks created
role.rbac.authorization.k8s.io/user-role created
rolebinding.rbac.authorization.k8s.io/user-rolebinding created
```

```
[root@ ~]# kubectl apply -f secrets-mysql-prod.yaml
secret/mysecretname-mysql-aks created
```

```
[root@ ~]# kubectl describe secret mysecretname-aks -n bankapp
Name:         mysecretname-aks
Namespace:    bankapp
Labels:       <none>
Annotations:  kubernetes.io/service-account.name: devops-agent-bankapp-aks
              kubernetes.io/service-account.uid:

Type:  kubernetes.io/service-account-token

Data
====
namespace:  7 bytes
token:

ca.crt:     1761 bytes
```

```
[root@                ~]# kubectl describe secret mysecretname-mysql-aks -n mysql
Name:         mysecretname-mysql-aks
Namespace:    mysql
Labels:       <none>
Annotations:  kubernetes.io/service-account.name: devops-agent-mysql-aks
              kubernetes.io/service-account.uid:

Type:  kubernetes.io/service-account-token

Data
====
ca.crt:     1761 bytes
namespace:  5 bytes
token:
```

```
[root@                    ~]# cat sa-role-rolebinding-prod.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: devops-agent-bankapp-aks
  namespace: bankapp
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: bankapp
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: bankapp
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: bankapp
  kind: ServiceAccount
  name: devops-agent-bankapp-aks
```

```
[root@                ~]# cat secrets-prod.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-aks
  namespace: bankapp
  annotations:
    kubernetes.io/service-account.name: devops-agent-bankapp-aks
```

```
[root@                ~]# cat sa-role-rolebinding-mysql-prod.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: devops-agent-mysql-aks
  namespace: mysql
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: mysql
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: mysql
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: mysql
  kind: ServiceAccount
  name: devops-agent-mysql-aks
```

```
[root@                ~]# cat secrets-mysql-prod.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-mysql-aks
  namespace: mysql
  annotations:
    kubernetes.io/service-account.name: devops-agent-mysql-aks
```

```
cat sa-role-rolebinding-prod.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: bankapp
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: devops-agent-bankapp-aks
  namespace: bankapp
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: bankapp
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: bankapp
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: bankapp
  kind: ServiceAccount
  name: devops-agent-bankapp-aks
```

```
cat secrets-prod.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-aks
  namespace: bankapp
  annotations:
    kubernetes.io/service-account.name: devops-agent-bankapp-aks

cat secrets-mysql-prod.yaml
apiVersion: v1
kind: Secret
type: kubernetes.io/service-account-token
metadata:
  name: mysecretname-mysql-aks
  namespace: mysql
  annotations:
    kubernetes.io/service-account.name: devops-agent-mysql-aks
```

```
cat sa-role-rolebinding-mysql-prod.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: mysql
---
apiVersion: v1
kind: ServiceAccount
metadata:
  name: devops-agent-mysql-aks
  namespace: mysql
---
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: user-role
  namespace: mysql
rules:
  - apiGroups: ["*"]
    resources: ["*"]
    verbs: ["*"]
---
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: user-rolebinding
  namespace: mysql
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: user-role
subjects:
- namespace: mysql
  kind: ServiceAccount
  name: devops-agent-mysql-aks
```

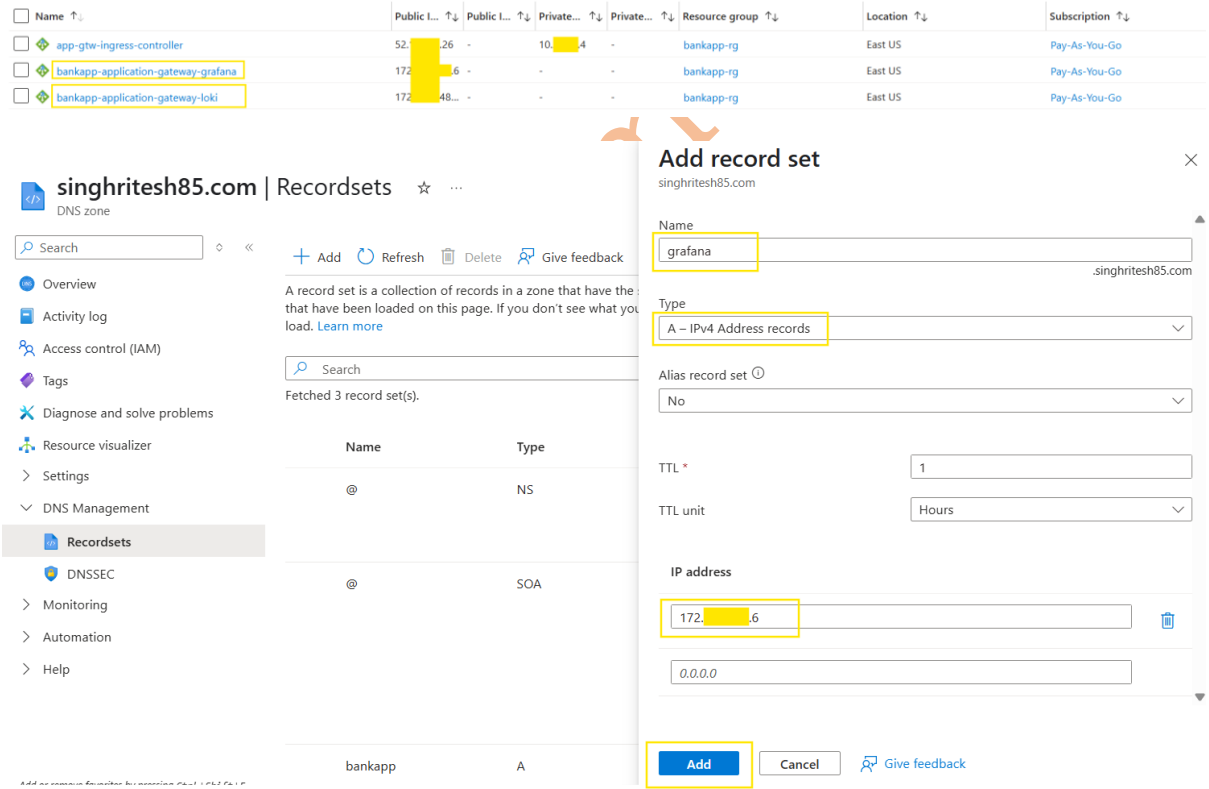
The kubeconfig file which I had shared with the deployment user **demo** is as shown in the screenshot attached below.

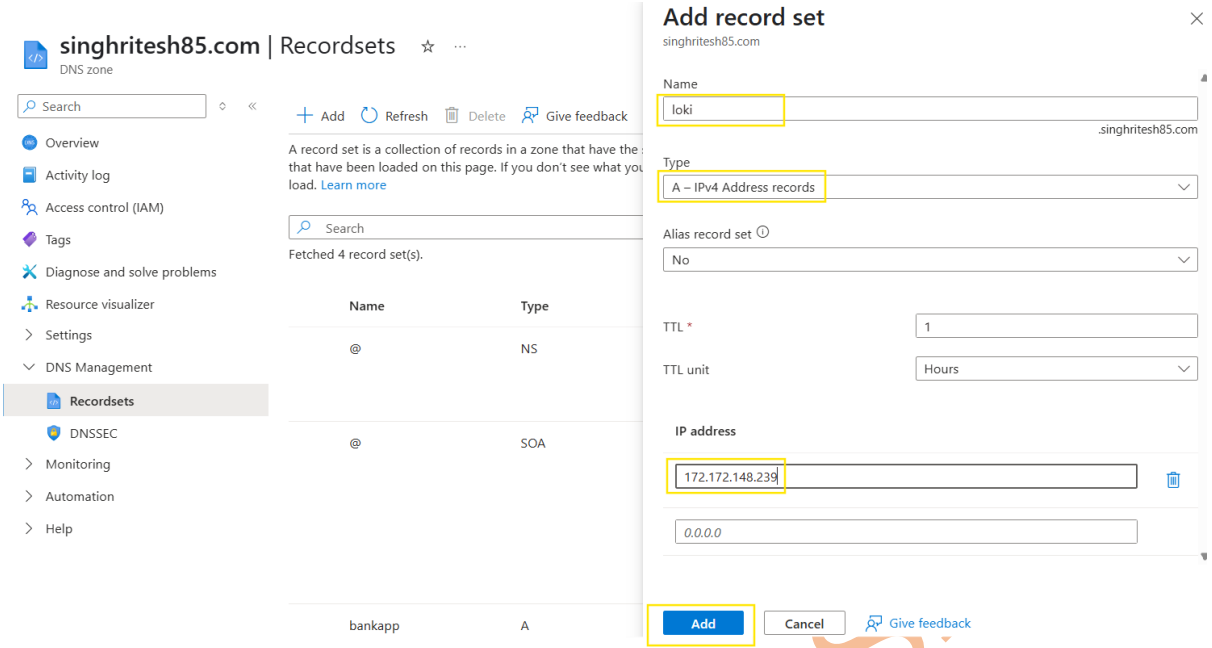
```
cat ~/.kube/config

apiVersion: v1

clusters:

- cluster:

    certificate-authority-data:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

    server: https://bankapp-cluster-dns-
y1drntz2.96d5ebedc80912eb0e8b15cf52a52918.privatelink.eastus.azmk8s.io:443

  name: bankapp-cluster

contexts:

- context:

    cluster: bankapp-cluster

    user: devops-agent-bankapp-aks

  name: devops-agent-bankapp-aks

- context:

    cluster: bankapp-cluster

    user: devops-agent-mysql-aks

  name: devops-agent-mysql-aks

current-context: devops-agent-mysql-aks

kind: Config

preferences: {}

users:

- name: devops-agent-bankapp-aks

  user:

    token:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

- name: devops-agent-mysql-aks

  user:

    token:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

```
[demo@devopsagent-vm ~]$ kubectl config get-contexts
CURRENT   NAME                          CLUSTER           AUTHINFO                  NAMESPACE
          devops-agent-bankapp-aks      bankapp-cluster   devops-agent-bankapp-aks
*         devops-agent-mysql-aks        bankapp-cluster   devops-agent-mysql-aks
[demo@devopsagent-vm ~]$ kubectl get nodes
Error from server (Forbidden): nodes is forbidden: User "system:serviceaccount:mysql:devops-agent-mysql-aks" cannot list resource "nodes" in API group "" at t
he cluster scope
[demo@devopsagent-vm ~]$ kubectl get pods -n mysql
NAME                READY   STATUS    RESTARTS        AGE
mysql-primary-0     1/1     Running   0               5h13m
mysql-secondary-0   1/1     Running   1 (5h11m ago)   5h13m
[demo@devopsagent-vm ~]$ kubectl get pvc -n mysql
NAME                     STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   VOLUMEATTRIBUTESCLASS   AGE
data-mysql-primary-0     Bound    pvc-[redacted]   1Gi        RWO            managed-csi    <unset>                 5h14m
data-mysql-secondary-0   Bound    pvc-[redacted]   1Gi        RWO            managed-csi    <unset>                 5h14m
[demo@devopsagent-vm ~]$ kubectl get pvc -n bankapp
Error from server (Forbidden): persistentvolumeclaims is forbidden: User "system:serviceaccount:mysql:devops-agent-mysql-aks" cannot list resource "persistent
volumeclaims" in API group "" in the namespace "bankapp"
[demo@devopsagent-vm ~]$ kubectl config use-context devops-agent-bankapp-aks
Switched to context "devops-agent-bankapp-aks".
[demo@devopsagent-vm ~]$ kubectl get pods -n bankapp
NAME                          READY   STATUS    RESTARTS   AGE
bankapp-folo-[redacted]   1/1     Running   0          44m
[demo@devopsagent-vm ~]$ kubectl get pods -n mysql
Error from server (Forbidden): pods is forbidden: User "system:serviceaccount:bankapp:devops-agent-bankapp-aks" cannot list resource "pods" in API group "" in
 the namespace "mysql"
```

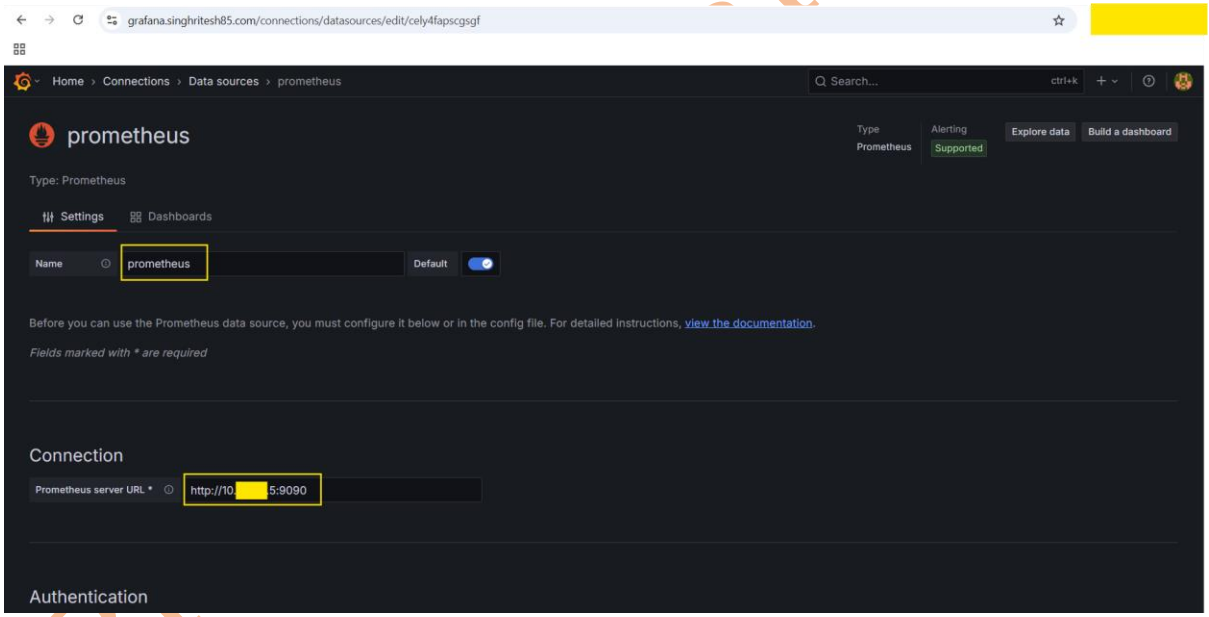
Monitoring using Prometheus and Grafana and Log Aggregation using Loki in Production

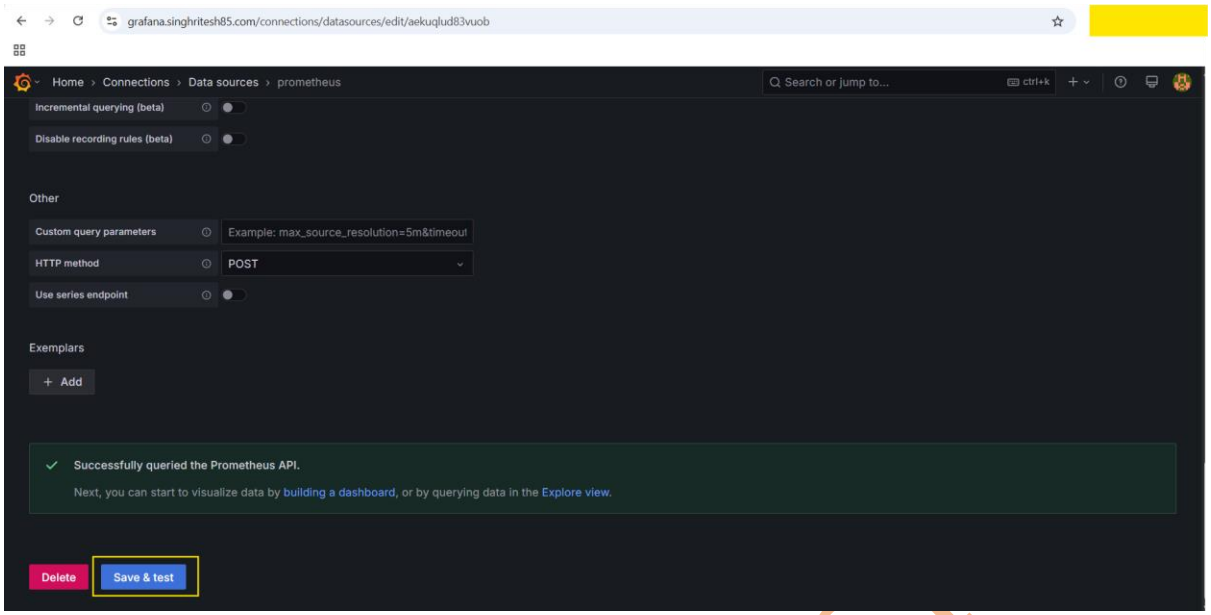
To access Grafana using URL I had created record set of A Type in Azure DNS Zone. I had also created a URL for Loki after creating a Record Set of A Type in Azure DNS Zone as shown in the screenshots attached below. For entry in the Azure DNS Zone to create the Record Set of A Type I obtained the Public IPV4 address from the Azure Application Gateway for Grafana and Loki.

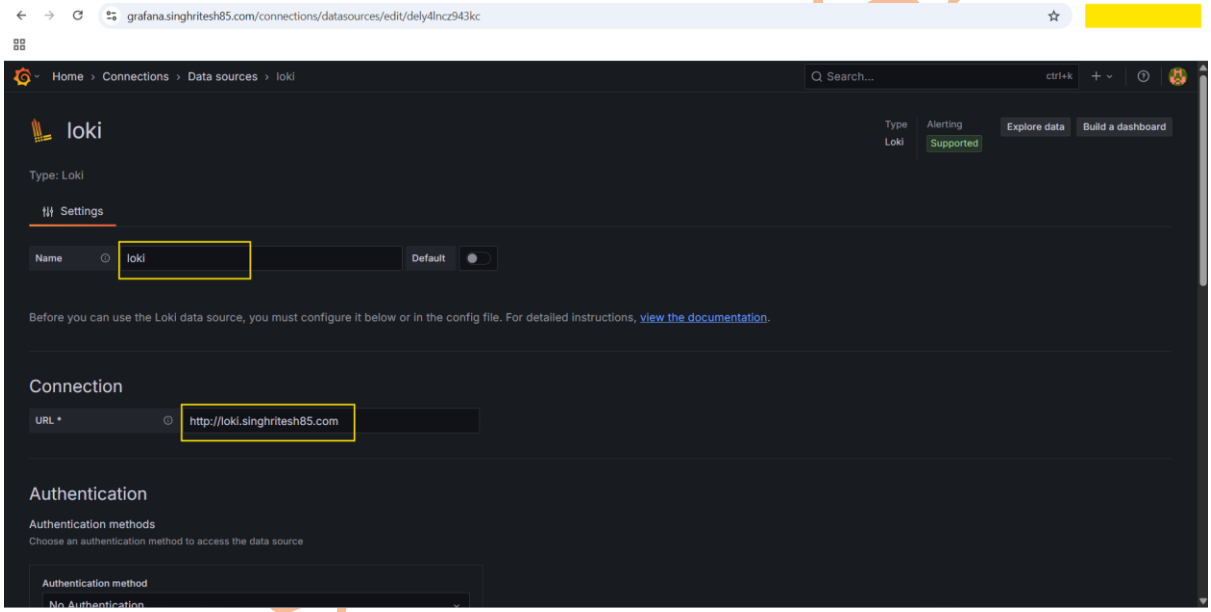




Now, login into the Grafana for the first time and update admin password then login into Grafana and created two data sources each for prometheus and loki as shown in the screenshot attached below.

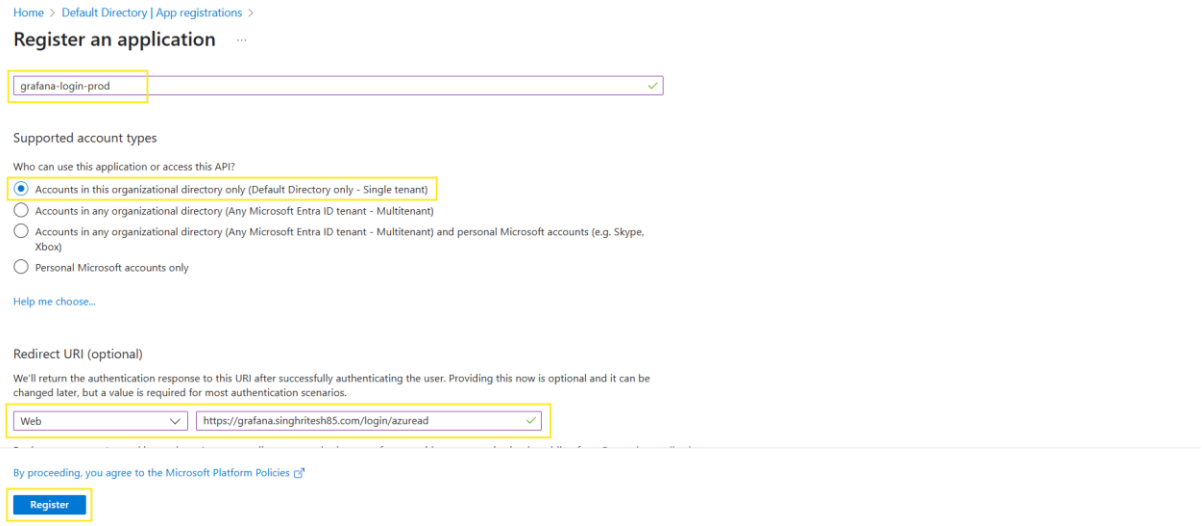




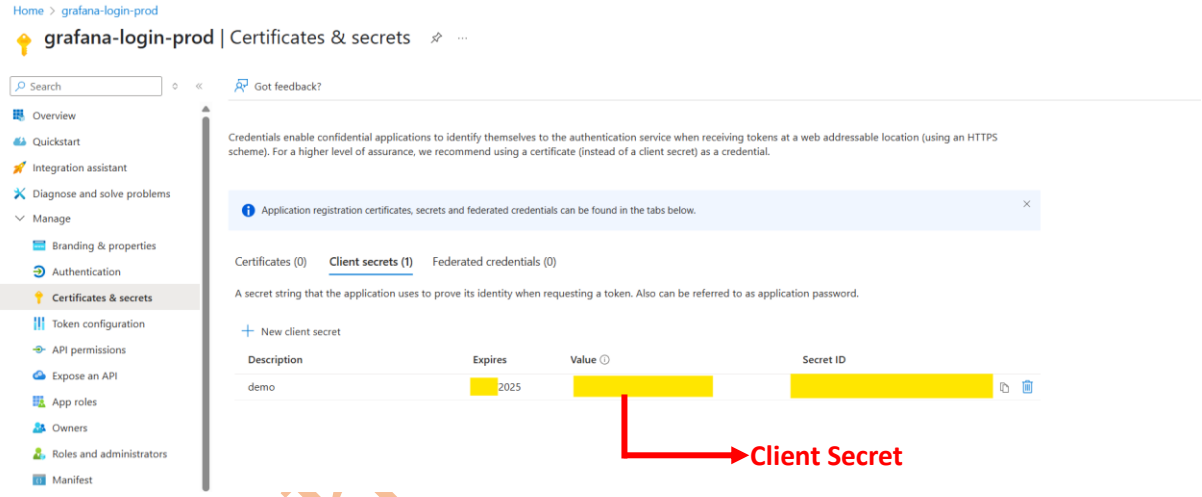


Integration of Azure Entra ID with Grafana

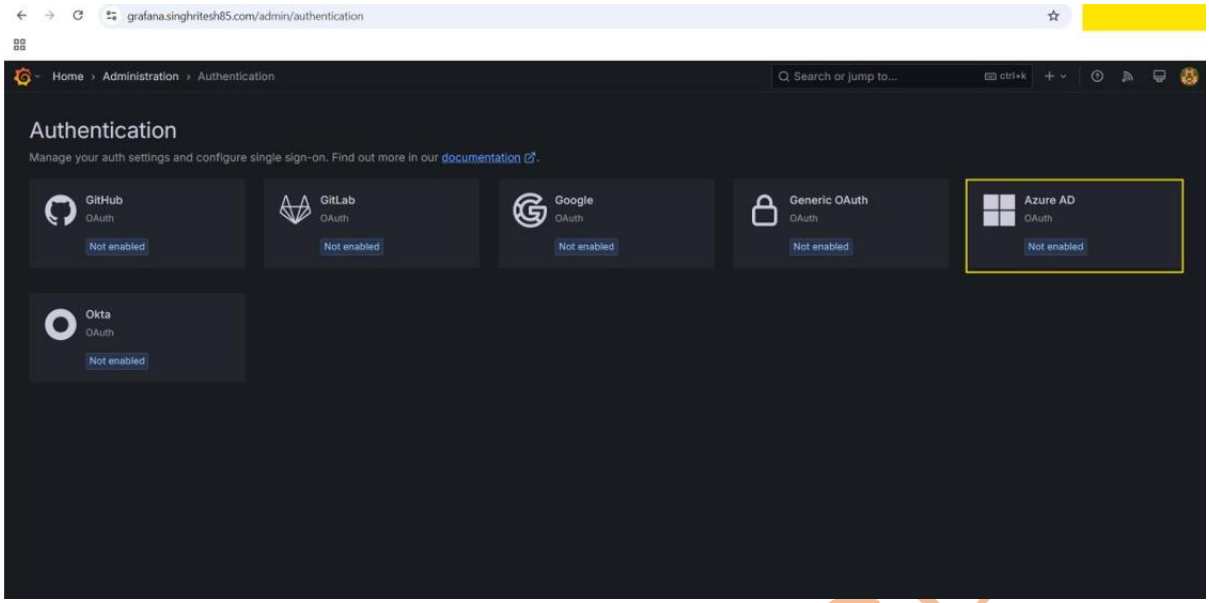
To integrate Azure Entra ID with Grafana I created an Azure Entra ID Service Principal (using APP Registration) as shown in the screenshot attached below. Go to **App Registration > New Registration** and configure a new Application as shown in the screenshot attached below.



Created the Secrets for Registered App Grafana-login in Azure Entra ID as shown in the screenshot attached below.

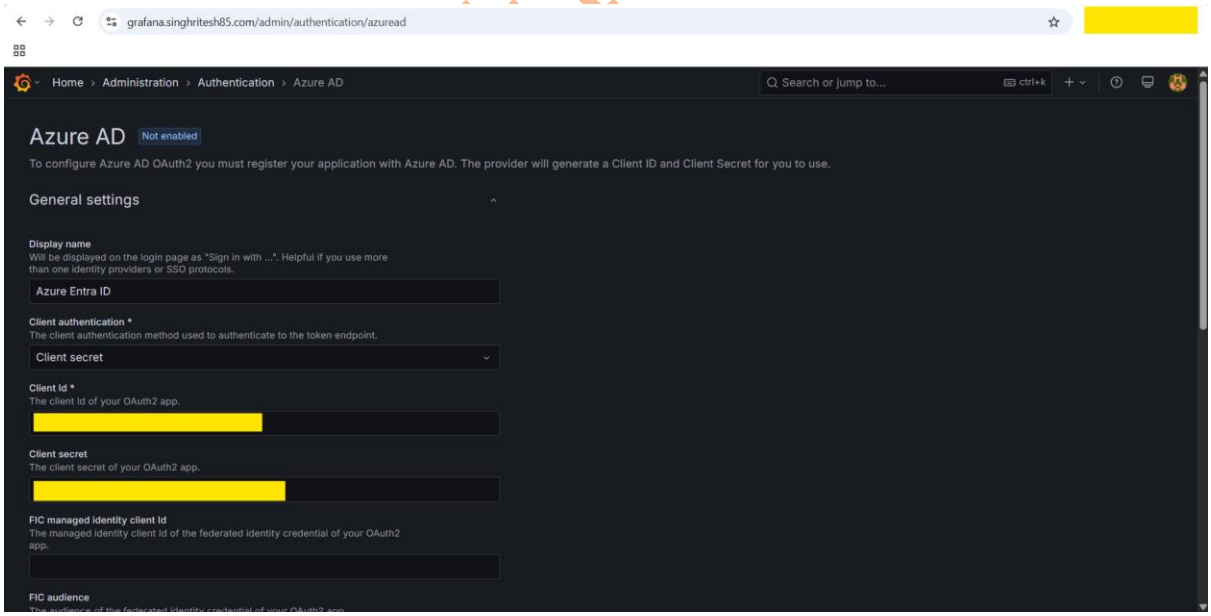


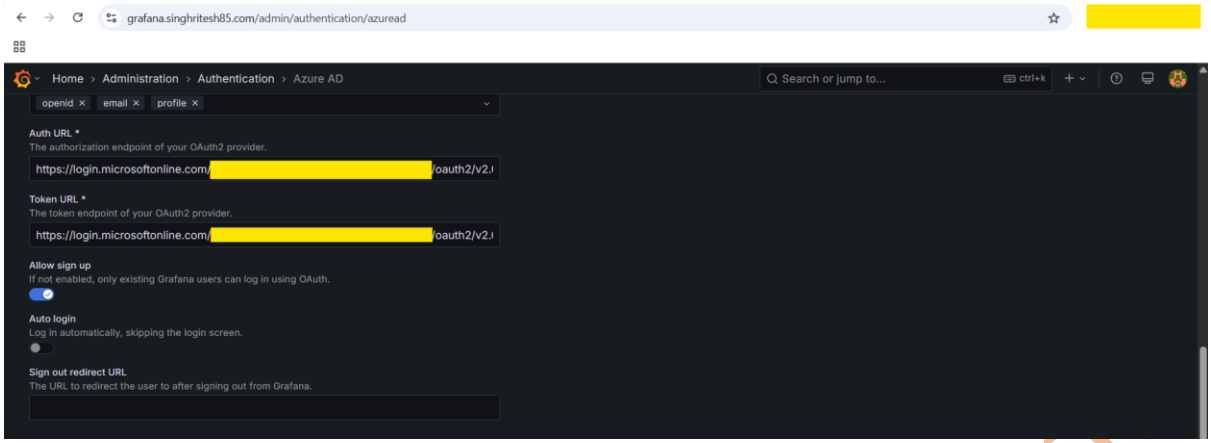
Now, login into the Grafana and update admin password then login into Grafana and Go to Grafana **Home > Administration > Authentication** and **enable the Azure AD OAuth** as shown in the screenshot attached below.

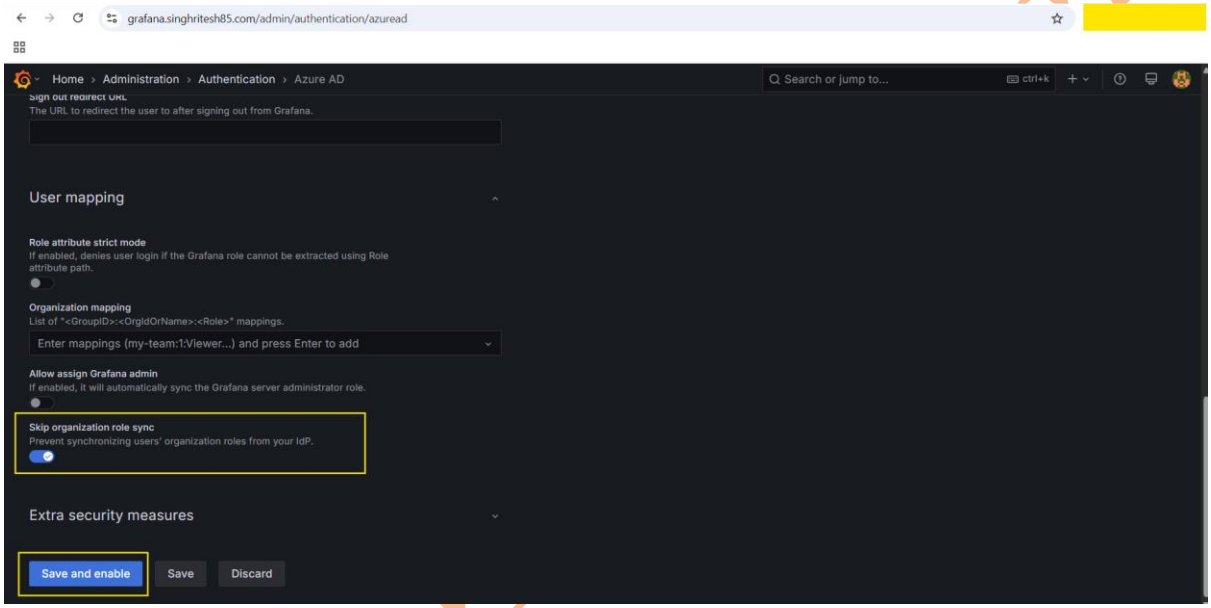


Provide the Client ID, Client Secret, Tenant ID and enable Allow Sign up and enable Skip organization role sync as shown in the screenshot attached below.

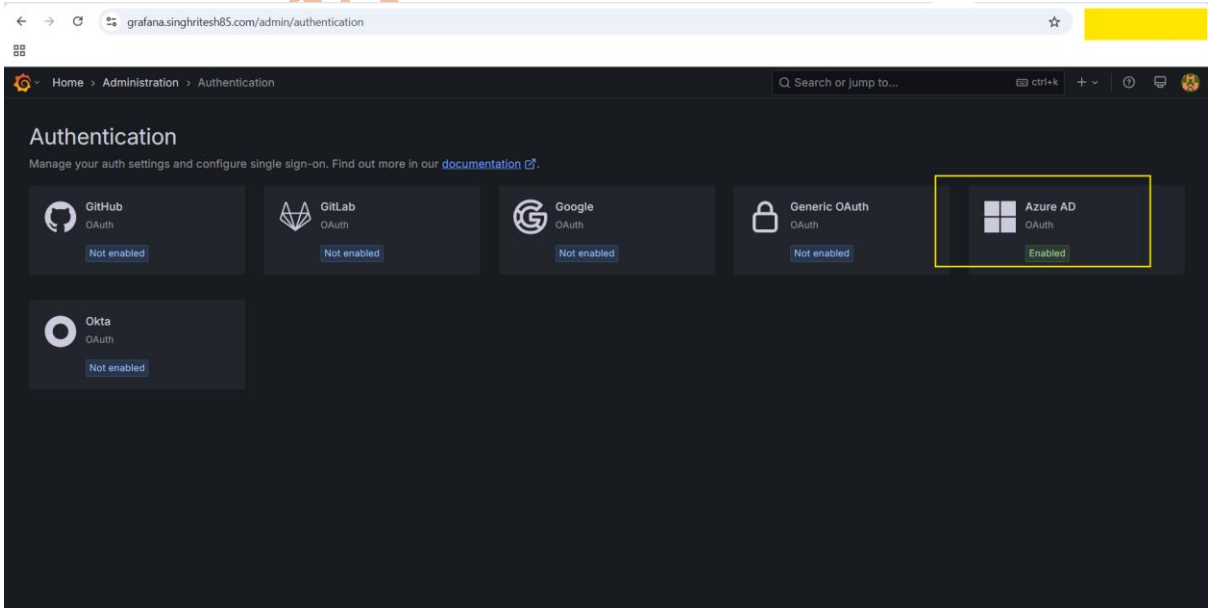
I enabled the **Skip organization role sync** otherwise Grafana will Sync the Azure Entra ID users with Main Org. Role as **Viewer** and Grafana Administrator cannot change it further but if I enabled **Skip organization role sync** then Grafana Administrator can change the viewer Role and can assign another Role as Editor or Admin. In this demonstration I created two users in Azure Entra ID user1 and user2. User1, I had provided as Administrator Role and user2 as default viewer access in Main Organisation (**Main Organisation always have Organisation ID 1**).







Now the Azure AD OAuth is enable as shown in the screenshot attached below.



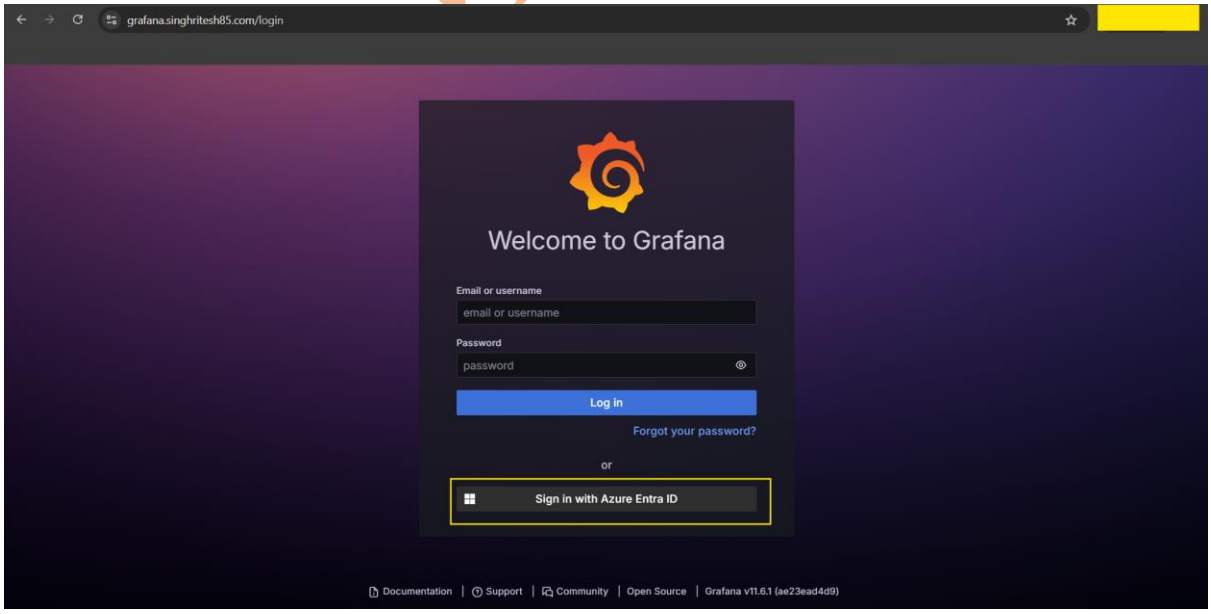
Now go to the Grafana Server and open the file **/etc/grafana/grafana.ini** and edit **root_url** under the [server] then restart the **grafana-server** service as shown in the screenshot attached below.

```
[root@grafana-vm ~]# vim /etc/grafana/grafana.ini
```

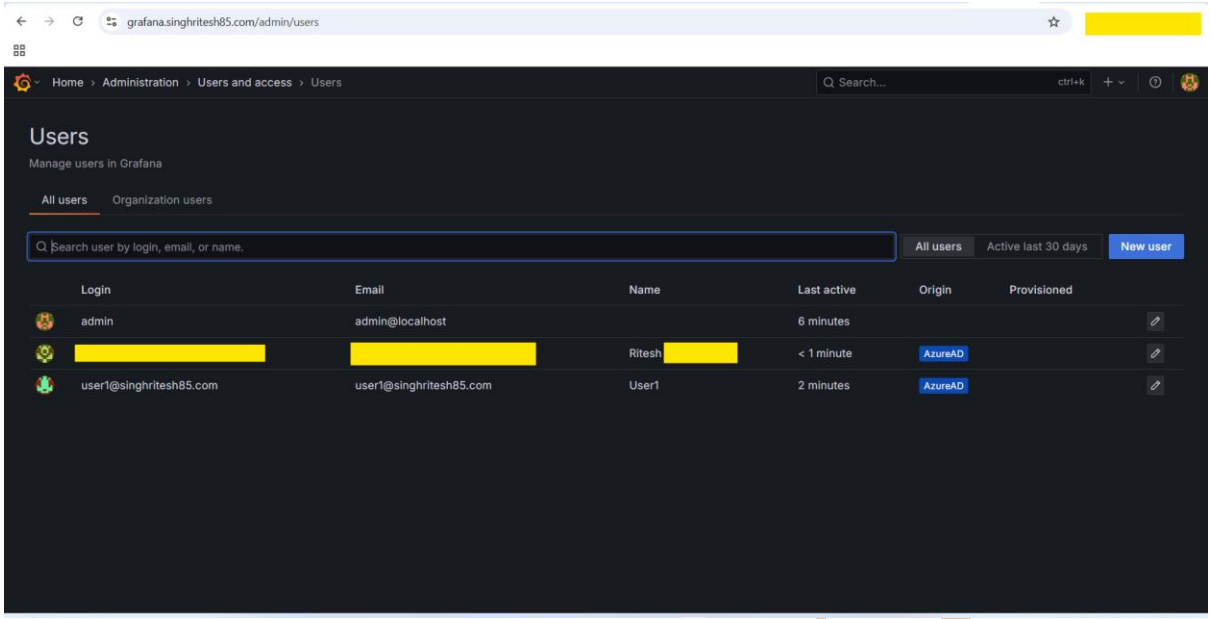
```
#################################### Server ####################################
[server]
# Protocol (http, https, h2, socket)
;protocol = http

# Minimum TLS version allowed. By default, this value is empty. Accepted values are: TLS1.2, TLS1.3. If nothing is set TLS1.2 would be taken
;min_tls_version = ""

# The ip address to bind to, empty will bind to all interfaces
;http_addr =

# The http port to use
;http_port = 3000

# The public facing domain name used to access grafana from a browser
;domain = localhost

# Redirect to correct domain if host header does not match domain
# Prevents DNS rebinding attacks
;enforce_domain = false

# The full public facing url you use in browser, used for redirects and emails
# If you use reverse proxy and sub path specify full url (with sub path)
root_url = https://grafana.singhritesh85.com    #%(protocol)s://%(domain)s:%(http_port)s/

# Serve Grafana from subpath specified in `root_url` setting. By default it is set to `false` for compatibility reasons.
;serve_from_sub_path = false

# Log web requests
;router_logging = false
```

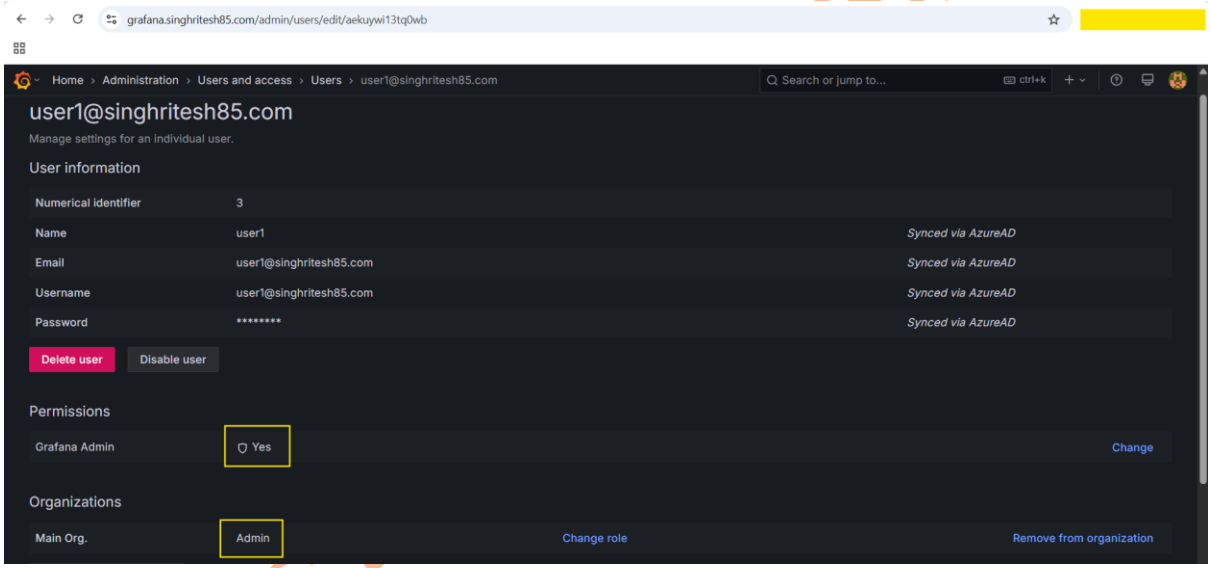
```
[root@grafana-vm ~]# systemctl restart grafana-server
[root@grafana-vm ~]# systemctl status grafana-server
● grafana-server.service - Grafana instance
   Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; vendor preset: disabled)
   Active: active (running) since [redacted] 2025[redacted] UTC; 9s ago
     Docs: http://docs.grafana.org
```

Now, Grafana login dashboard will show the option of sign-in with Azure Entra ID as shown in the screenshot attached below. I logged-in to the Grafana dashboard with the Azure Entra ID user as shown in the screenshot attached below.



For the first time when a user logged-in, they had viewer access which Grafana Administrator can change as per the requirement as shown in the screenshot attached below.





After that user1 will refresh Grafana UI and their Access will be reflected on the UI as well.

Installation of node-exporter and promtail had been done using the helm chart in the AKS Cluster as written below.

helm repo add prometheus-community https://*prometheus-community.github.io/helm-charts*

kubectl create ns node-exporter

helm install my-prometheus-node-exporter prometheus-community/prometheus-node-exporter --version 4.37.1 --set service.type=LoadBalancer -n node-exporter

```
[root@                ~]# helm repo add prometheus-community https://prometheus-community.github.io/helm-charts
"prometheus-community" has been added to your repositories
[root@                ~]# kubectl create ns node-exporter
namespace/node-exporter created
[root@                ~]# helm install my-prometheus-node-exporter prometheus-community/prometheus-node-exporter --version 4.37.1 --set service.type=LoadBala
ncer -n node-exporter

NAME: my-prometheus-node-exporter
LAST DEPLOYED: Thu May 15 09:41:15 2025
NAMESPACE: node-exporter
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
1. Get the application URL by running these commands:
     NOTE: It may take a few minutes for the LoadBalancer IP to be available.
           You can watch the status of by running 'kubectl get svc -w my-prometheus-node-exporter'
  export SERVICE_IP=$(kubectl get svc --namespace node-exporter my-prometheus-node-exporter -o jsonpath='{.status.loadBalancer.ingress[0].ip}')
  echo http://$SERVICE_IP:9100
```

Below screenshot shows the Kubernetes Service which was created for node-exporter using the helm chart.

```
[root@                ~]# kubectl get svc -n node-exporter
NAME                            TYPE           CLUSTER-IP     EXTERNAL-IP       PORT(S)          AGE
my-prometheus-node-exporter     LoadBalancer   10.      .20   128.      .144   9100:30603/TCP   3m32s
```

Then I had updated the prometheus configuration file **/etc/prometheus/prometheus.yml** as shown in the screenshot attached below.

```
- job_name: "AKS-Cluster"
  static_configs:
    - targets: ["128.      .144:9100"]
```

```
[root@prometheus-vm ~]# systemctl restart prometheus.service
[root@prometheus-vm ~]# systemctl status prometheus.service
● prometheus.service - Prometheus
   Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; vendor preset: disabled)
   Active: active (running) since Thu 2025                UTC; 6s ago
```

I installed the promtail using the helm chart as written below. First, I cloned the helm chart present in GitHub Repo.

git clone https://github.com/singhritesh85/helm-chart-promtail.git

After cloning helm chart from GitHub, I updated the values.yaml file of promtail helm chart with Loki Servers Private IP Addresses as shown in the screenshot attached below.

kubectl create ns promtail && helm upgrade --install promtail ./helm-chart-promtail -f ./helm-chart-promtail/values.yaml -n promtail

kubectl get pods -n promtail --watch

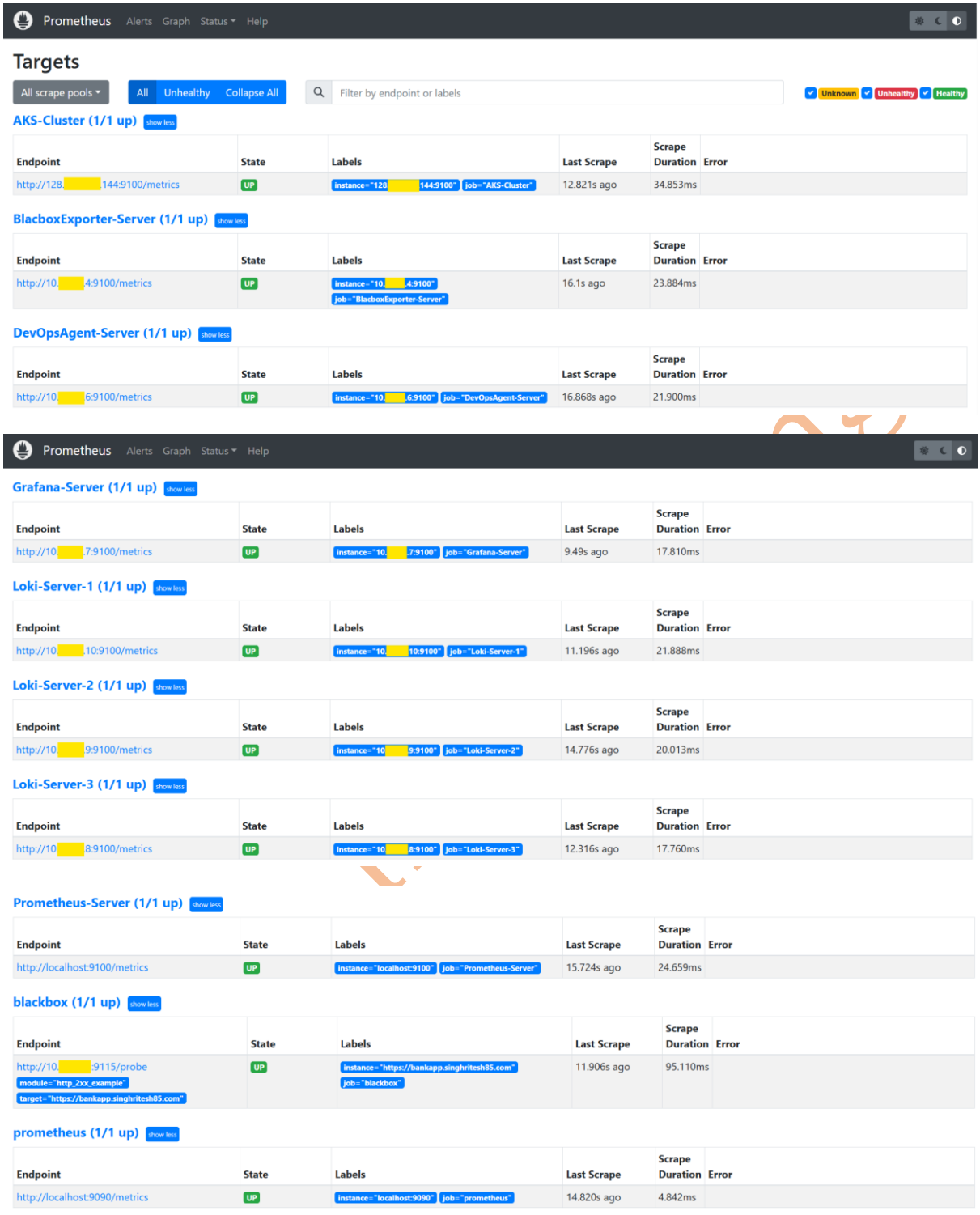
```
logLevel: info
# -- The log format of the Promtail server
# Must be reference in `config.file` to configure `server.log_format`
# Valid formats: `logfmt, json`
# See default config in `values.yaml`
logFormat: logfmt
# -- The port of the Promtail server
# Must be reference in `config.file` to configure `server.http_listen_port`
# See default config in `values.yaml`
serverPort: 3101
# -- The config of clients of the Promtail server
# Must be reference in `config.file` to configure `clients`
# @default -- See `values.yaml`
clients:
  - url: http://10.[redacted].10:3100/loki/api/v1/push
  - url: http://10.[redacted].9:3100/loki/api/v1/push
  - url: http://10.[redacted].8:3100/loki/api/v1/push

# -- Configures where Promtail will save it's positions file, to resume reading after restarts.
# Must be referenced in `config.file` to configure `positions`
positions:
  filename: /run/promtail/positions.yaml
# -- The config to enable tracing
enableTracing: false
```

```
[root@[redacted] ~]# git clone https://github.com/singhritesh85/helm-chart-promtail.git
Cloning into 'helm-chart-promtail'...
remote: Enumerating objects: 35, done.
remote: Counting objects: 100% (35/35), done.
remote: Compressing objects: 100% (34/34), done.
remote: Total 35 (delta 7), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (35/35), 24.57 KiB | 8.19 MiB/s, done.
Resolving deltas: 100% (7/7), done.
[root@[redacted] ~]# vim helm-chart-promtail/
.git/          Chart.yaml     README.md      README.md.gotmpl  ci/            templates/     values.yaml
[root@[redacted] ~]# vim helm-chart-promtail/values.yaml
[root@[redacted] ~]# kubectl create ns promtail && helm upgrade --install promtail ./helm-chart-promtail -f ./helm-chart-promtail/values.yaml -n promtai
l
namespace/promtail created
Release "promtail" does not exist. Installing it now.
NAME: promtail
LAST DEPLOYED: [redacted] 2025
NAMESPACE: promtail
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
**********************************************************************
 Welcome to Grafana Promtail
 Chart version: 6.16.6
 Promtail version: 3.0.0
**********************************************************************

Verify the application is working by running these commands:
* kubectl --namespace promtail port-forward daemonset/promtail 3101
* curl http://127.0.0.1:3101/metrics
```

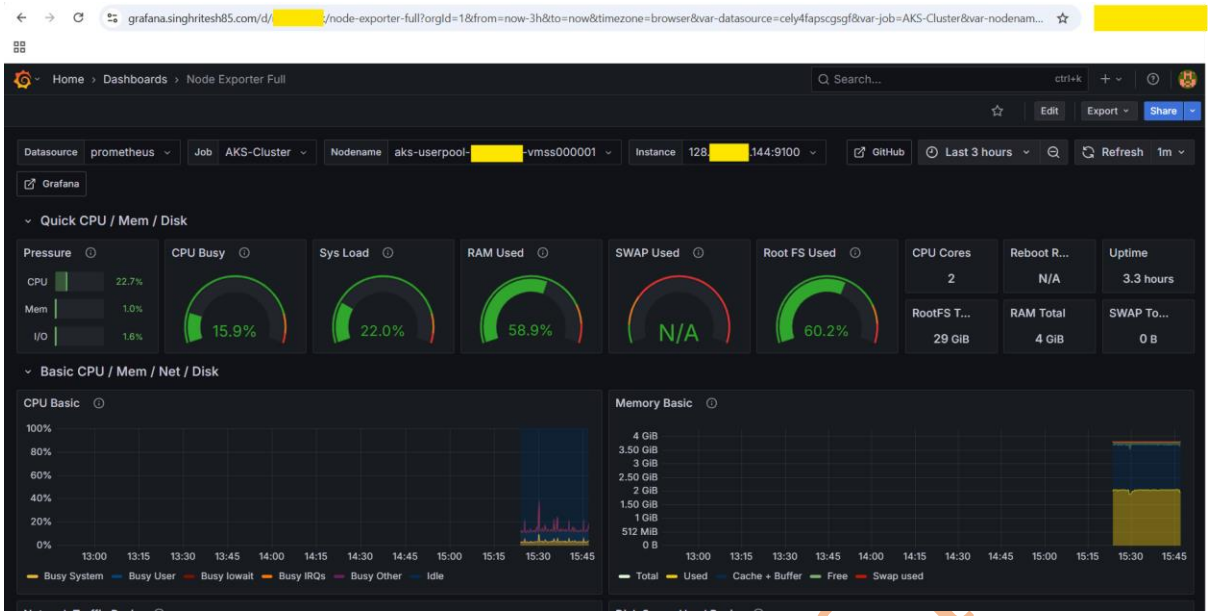
```
[root@[redacted] ~]# kubectl get pods -n promtail --watch
NAME                READY   STATUS    RESTARTS   AGE
promtail-[redacted] 1/1     Running   0          2m23s
promtail-[redacted] 1/1     Running   0          2m23s
promtail-[redacted] 1/1     Running   0          2m23s
```

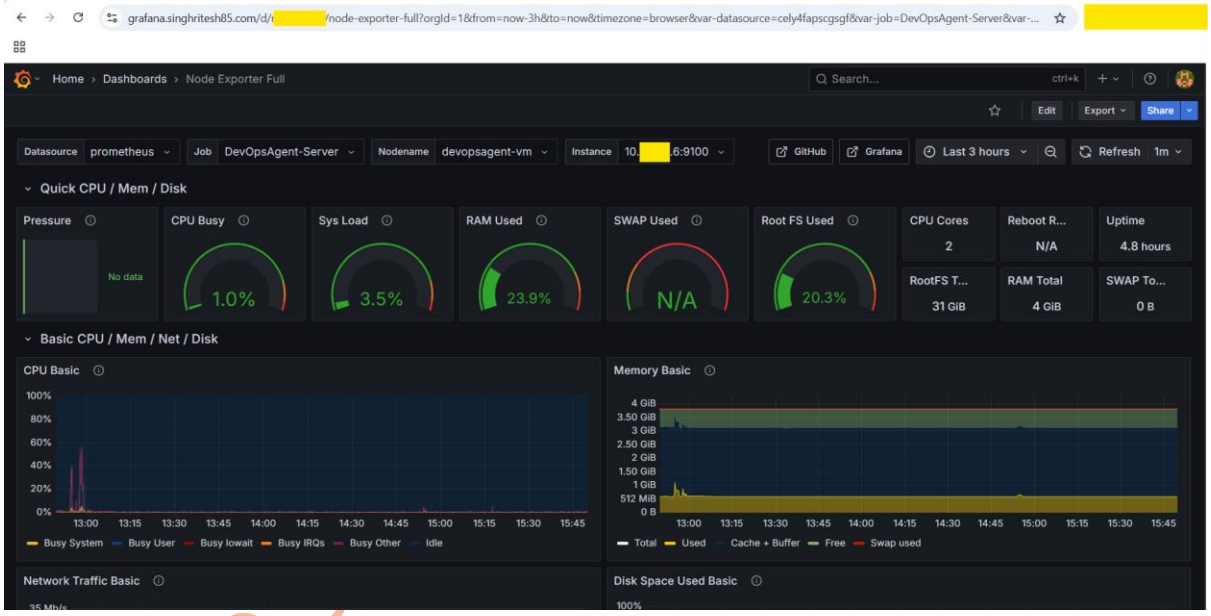
For Monitoring Tool I had used Prometheus and Grafana. I had already integrated Prometheus and Loki as a data source for Grafana which was already discussed above.

Here I checked the prometheus console and I found all the Targets was UP as shown in the screenshot attached below.

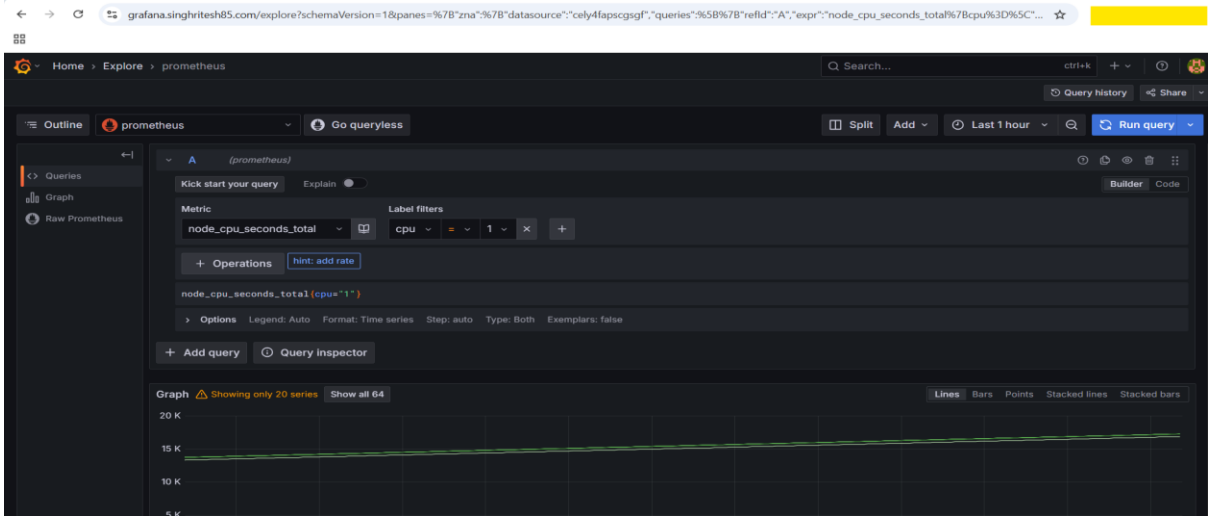


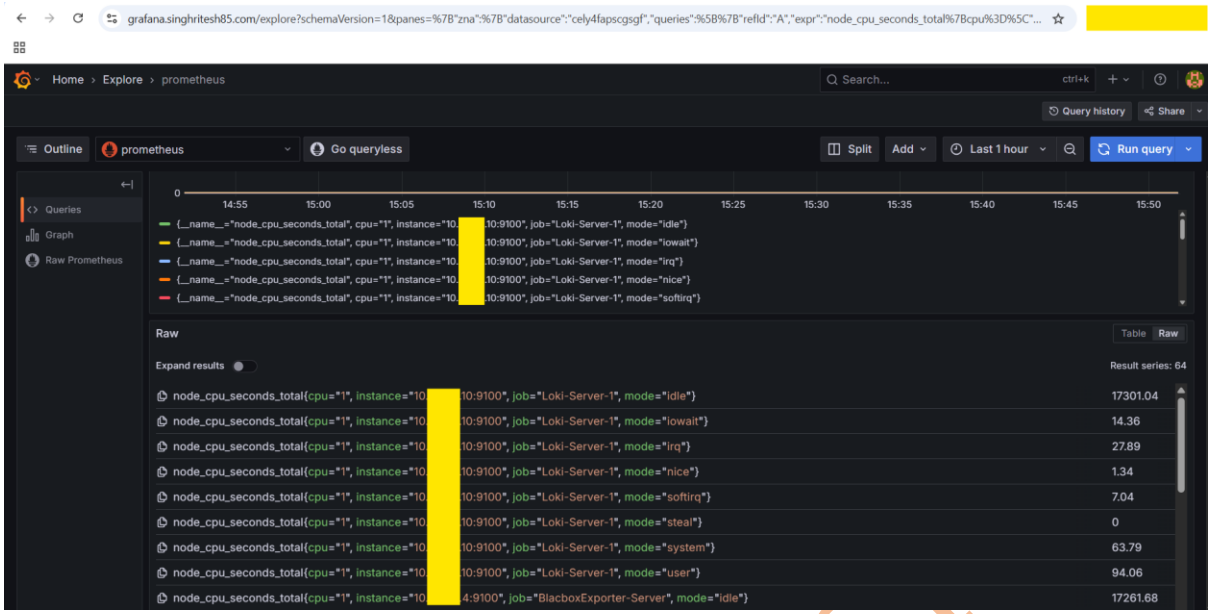
For Monitoring all the Servers and AKS Cluster health using the Node Exporter I used Grafana ID **1860**.



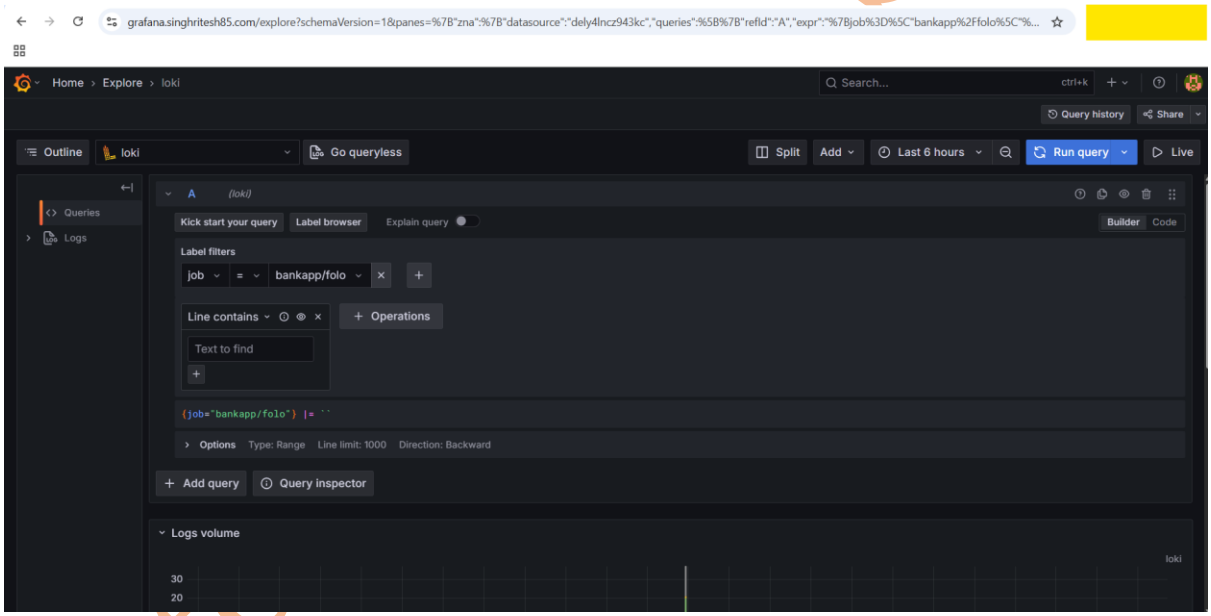


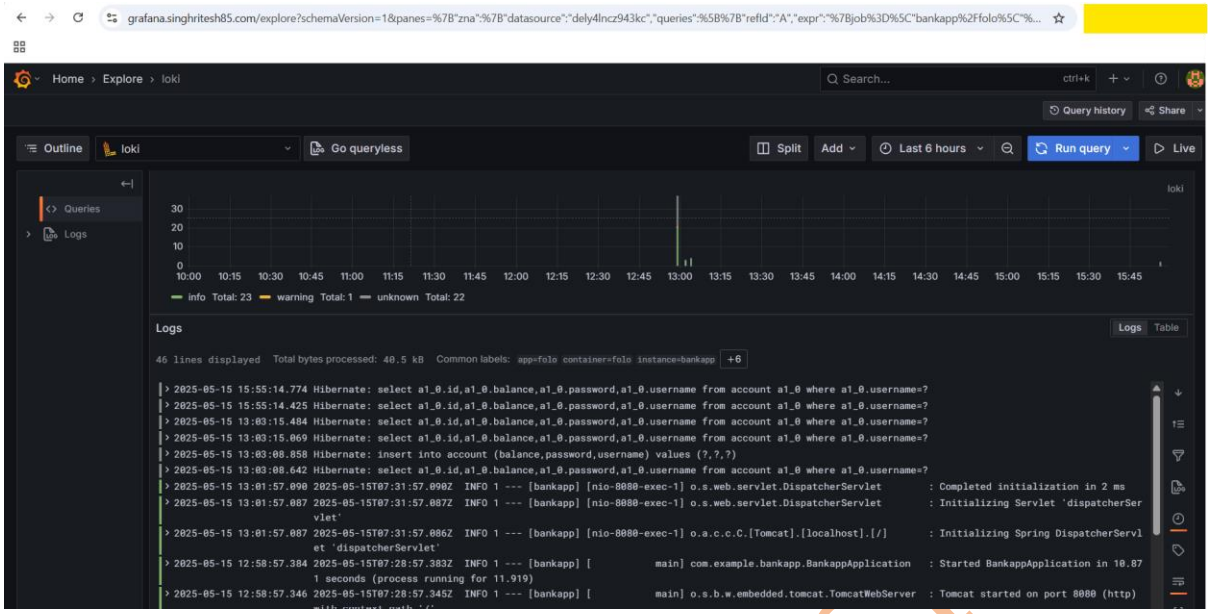
Grafana Metrics for SonarQube I started exploring as shown in the screenshot attached below.





Logs using Loki through Grafana I started exploring as shown in the screenshot attached below.

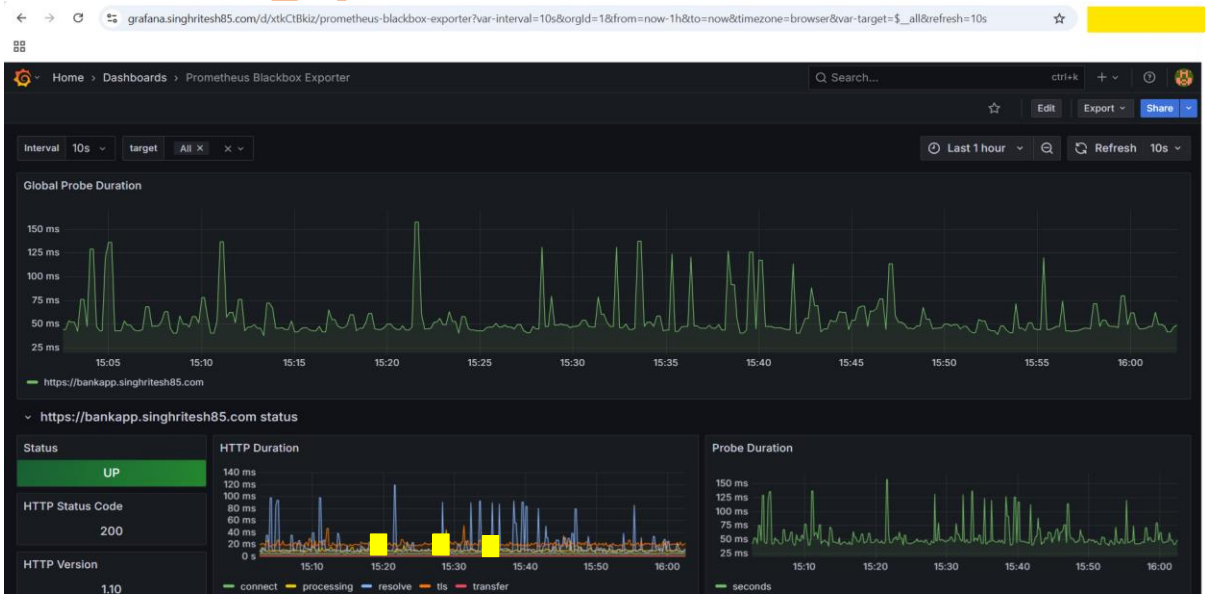


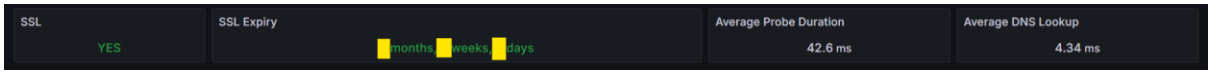


Finally, I was able to perform the synthetic monitoring on the Bankapp Application URL as shown in the screenshot attached below. Application URL **https://bankapp.singhritesh85.com** had been monitored using blackbox exporter.

I had installed Blackbox Exporter on a different server and not on the Prometheus Server. **The module name** is monitor_website.yml present of the blackbox exporter server at the path (/opt/blackbox_exporter_linux_amd64/monitor_website.yml). Prometheus blackbox operator is used for endpoint monitoring (Synthetic Monitoring) across the protocol http, https, TCP, and ICMP. In this project I am monitoring the Application URL **https://bankapp.singhritesh85.com** with the help of Prometheus Blackbox-Exporter. Prometheus blackbox exporter will send the metrics to Prometheus. For this project Prometheus acts as a DataSource for Grafana and send metrics to Grafana which we can see with the help of Charts and Graphs.

To create the Grafana Dashboard for Application URL Monitoring using blackbox exporter I had used the Grafana ID **7587** and below is the created Dashboard.

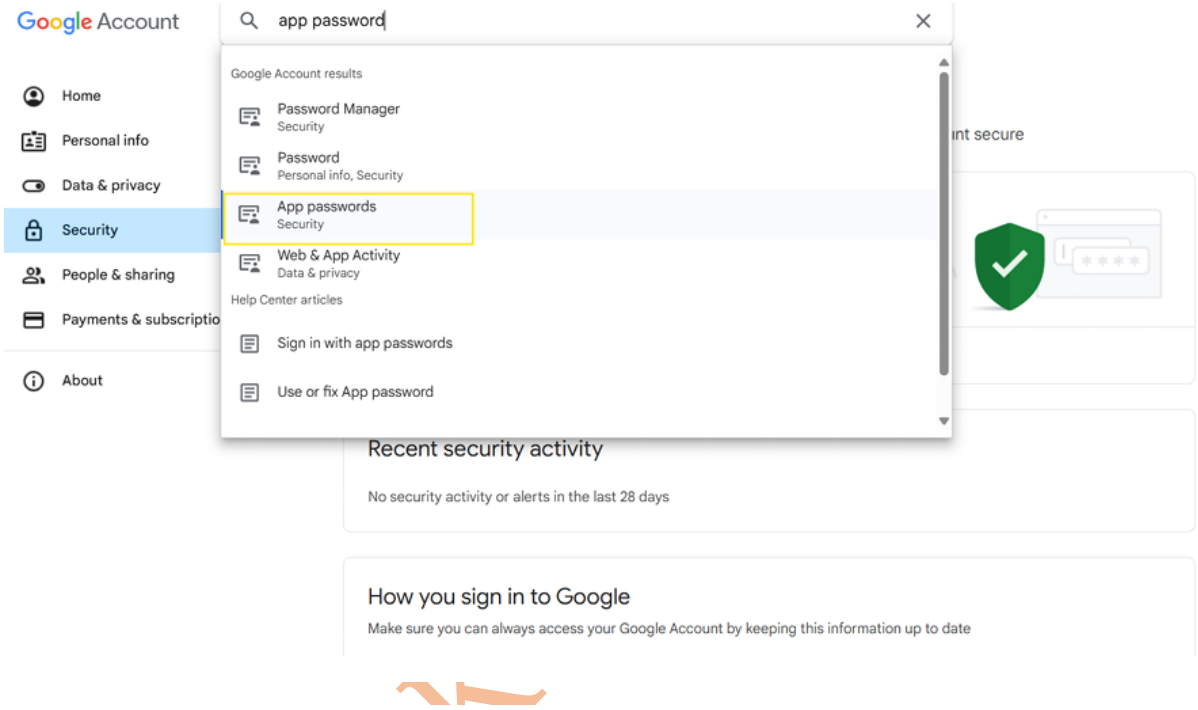


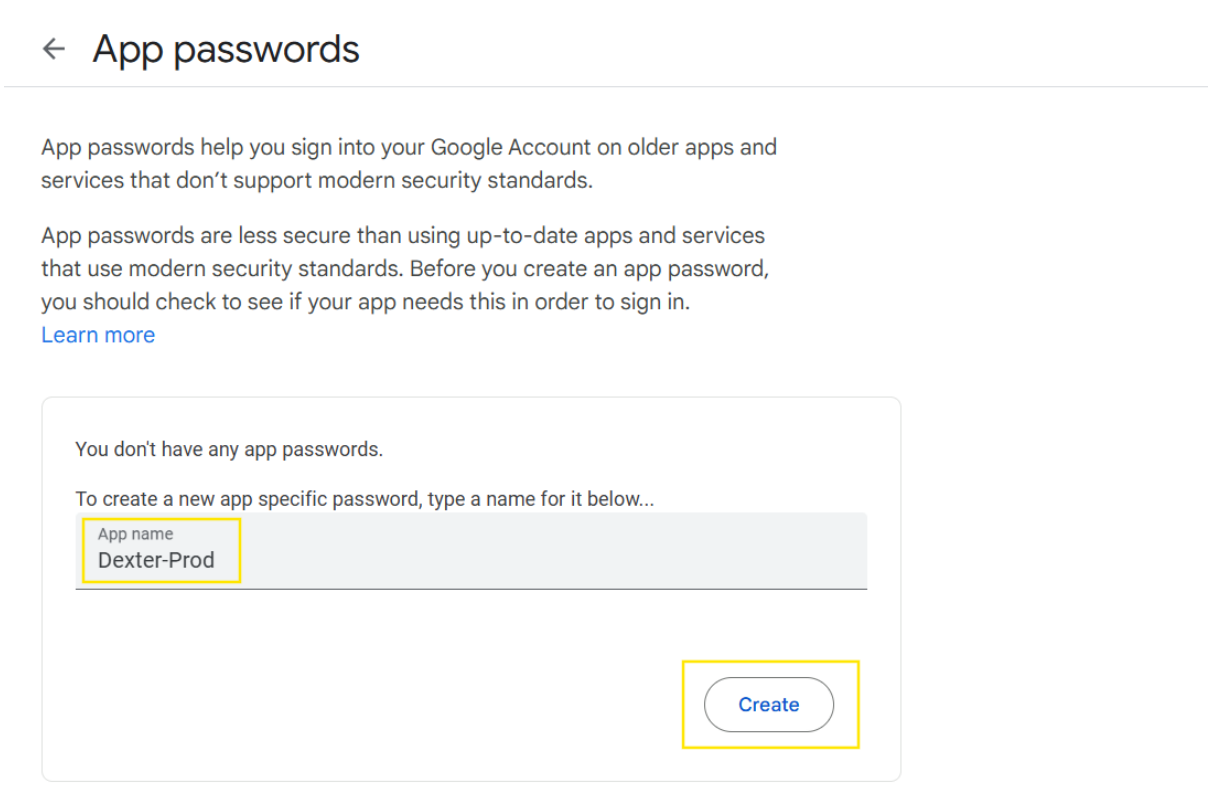


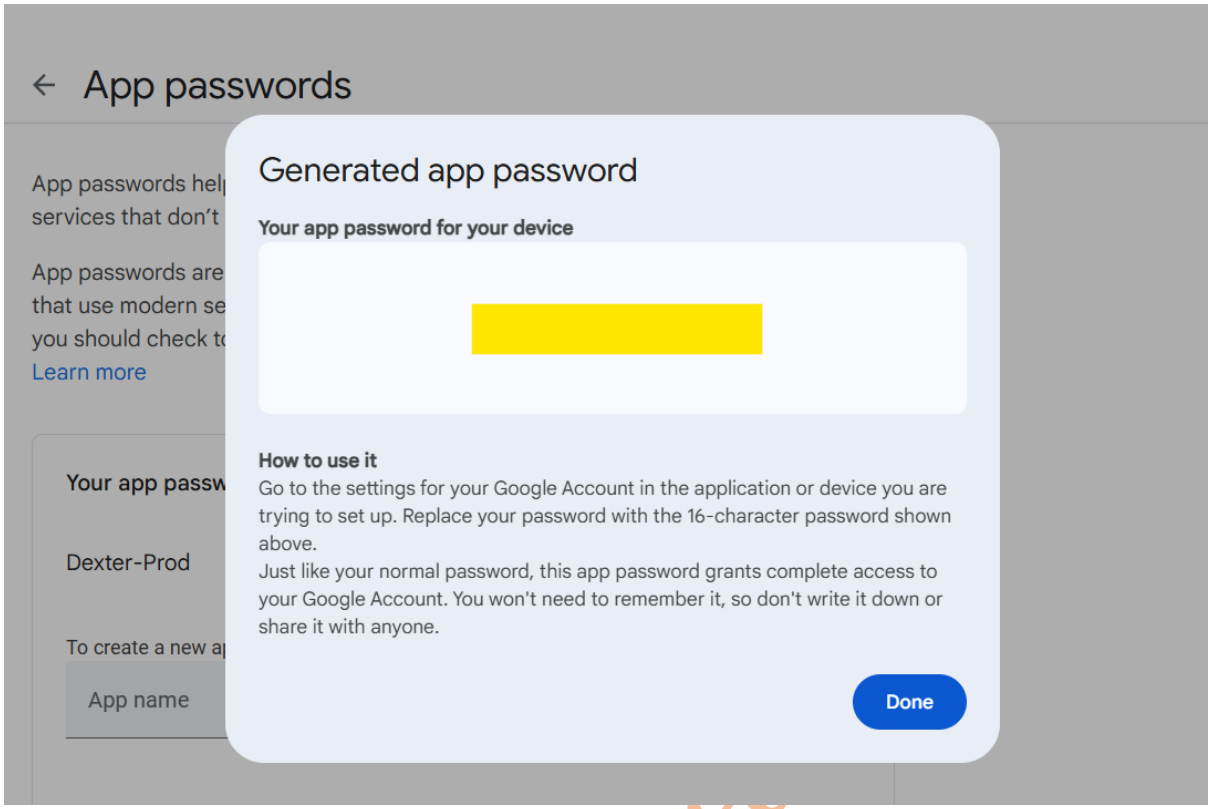
Configuration of Email to Send notification on Group Email-ID using Grafana

To configure Gmail to send notification to group Email ID I should have App Password for my Gmail account as shown in the screenshot attached below.

Go to your **Gmail Account > Manage your Google Account > Security** and then search for **app password** and click on **App Passwords** as shown in the screenshot attached below.







I had deleted this App Password after completion of this project, this App password does not exist anymore. You can use your own Gmail Account's App Passwords.

It is Possible to Send notification to group Email ID using Jenkins and Grafana through Amazon Simple Email Services (Amazon SES). To know more you can refer the project present in my GitHub Repo https://github.com/singhritesh85/DevOps-Project-2tier-WebApp-Deployment.git. However, for this project I used Gmail App Passwords as explained above.

Now, configuration of email to Send notification on Group Email-ID using Jenkins and Grafana is as discussed below.

Configuration of Alerts in Grafana

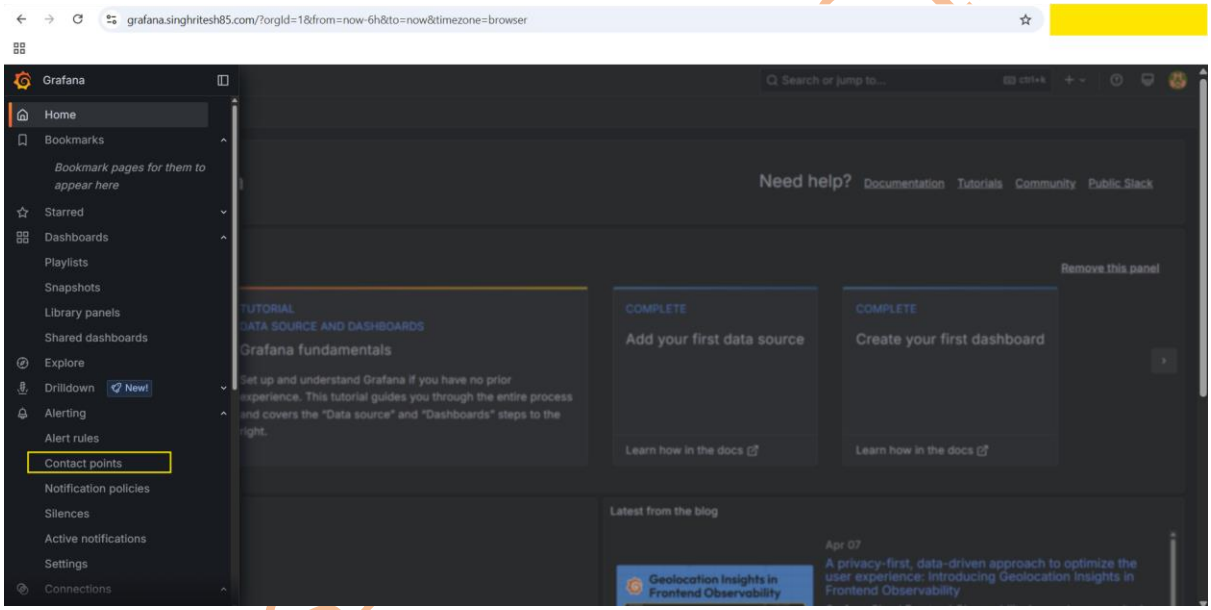
To configure Alerts in Grafana, first I created **contact points** with the Email ID and changed smtp settings in the configuration file **/etc/grafana/grafana.ini** of Grafana which I discussed below.

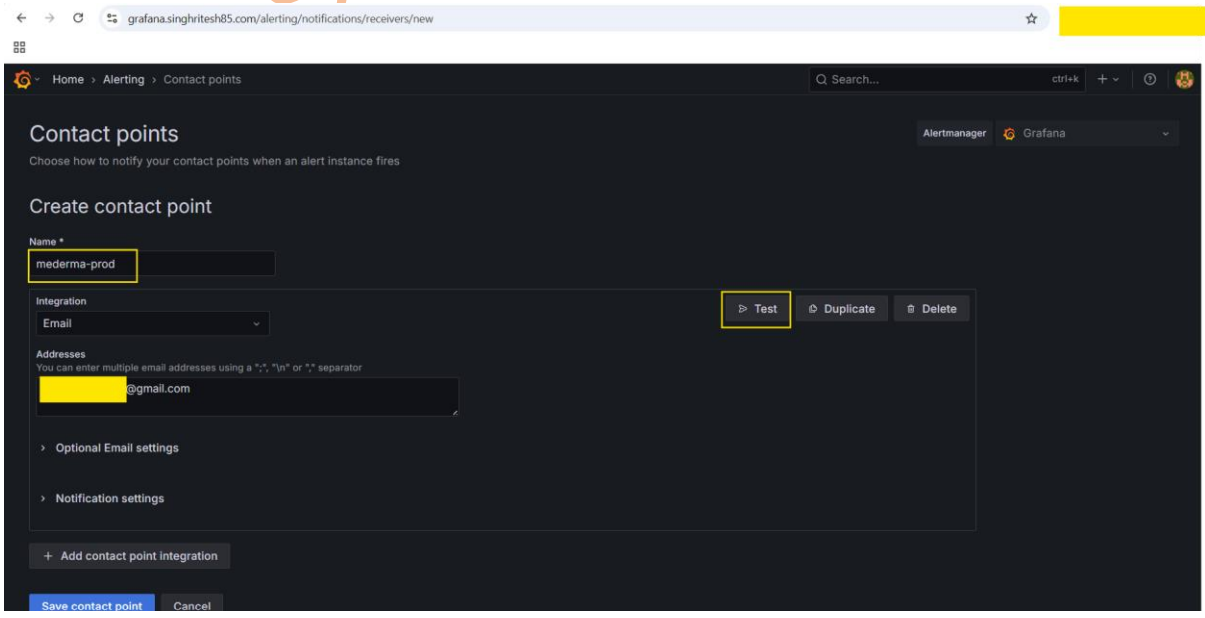
```
################################### SMTP / Emailing ##########################
[smtp]
enabled = true
host = smtp.gmail.com:587
user = 
# If the password contains # or ; you have to wrap it with triple quotes. Ex """#password;"""
password = 
;cert_file =
;key_file =
skip_verify = true
from_address = 
from_name = Grafana Alert for Azure DevOps Pipeline for Production BankApp
# EHLO identity in SMTP dialog (defaults to instance_name)
;ehlo_identity = dashboard.example.com
# SMTP startTLS policy (defaults to 'OpportunisticStartTLS')
;startTLS_policy = NoStartTLS
# Enable trace propagation in e-mail headers, using the 'traceparent', 'tracestate' and (optionally) 'baggage' fields (defaults to false)
;enable_tracing = false
```

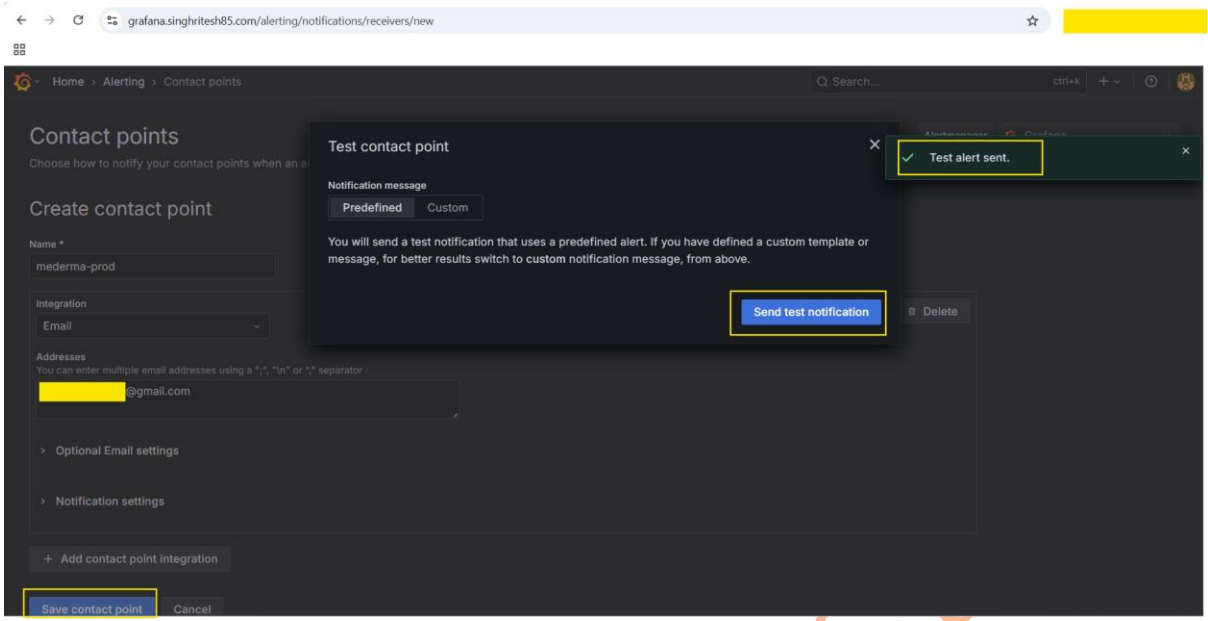
Then restarted the **grafana-server** service as shown in the screenshot attached below.

```
[root@grafana-vm ~]# systemctl restart grafana-server
[root@grafana-vm ~]# systemctl status grafana-server
● grafana-server.service - Grafana instance
   Loaded: loaded (/usr/lib/systemd/system/grafana-server.service; enabled; vendor preset: disabled)
```

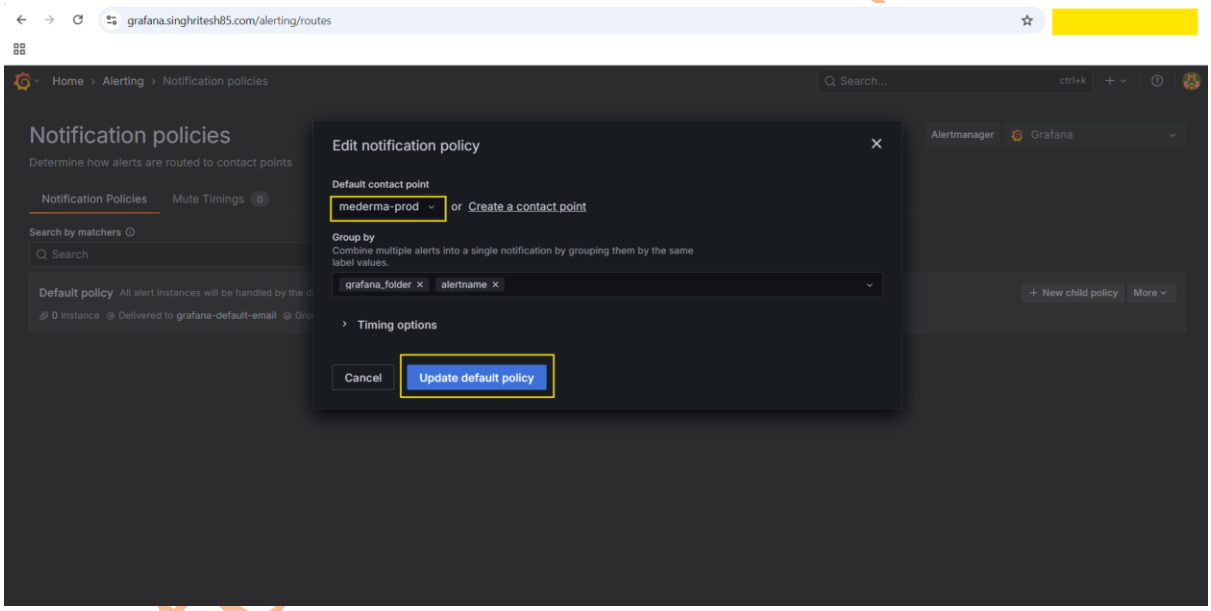
Here I had configured the contact points in Grafana UI as shown in the screenshot attached below.



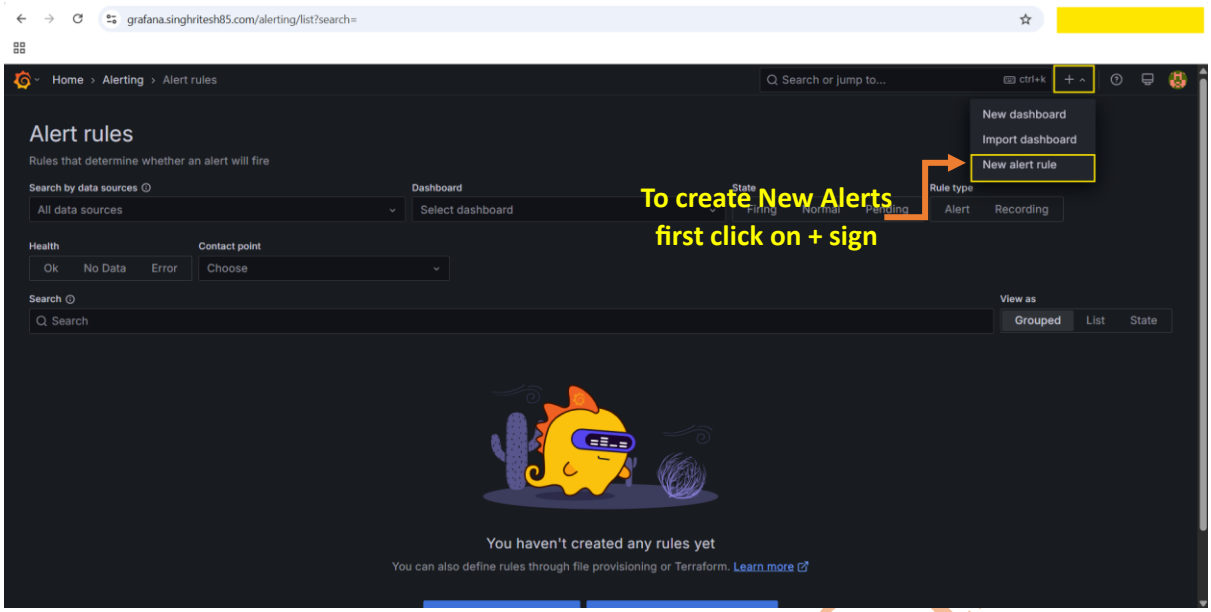


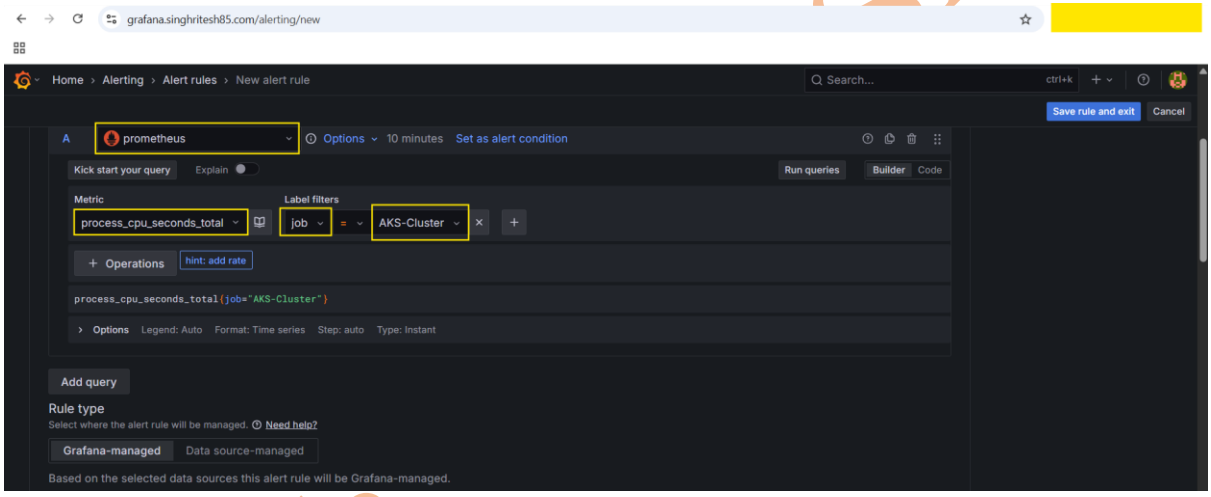


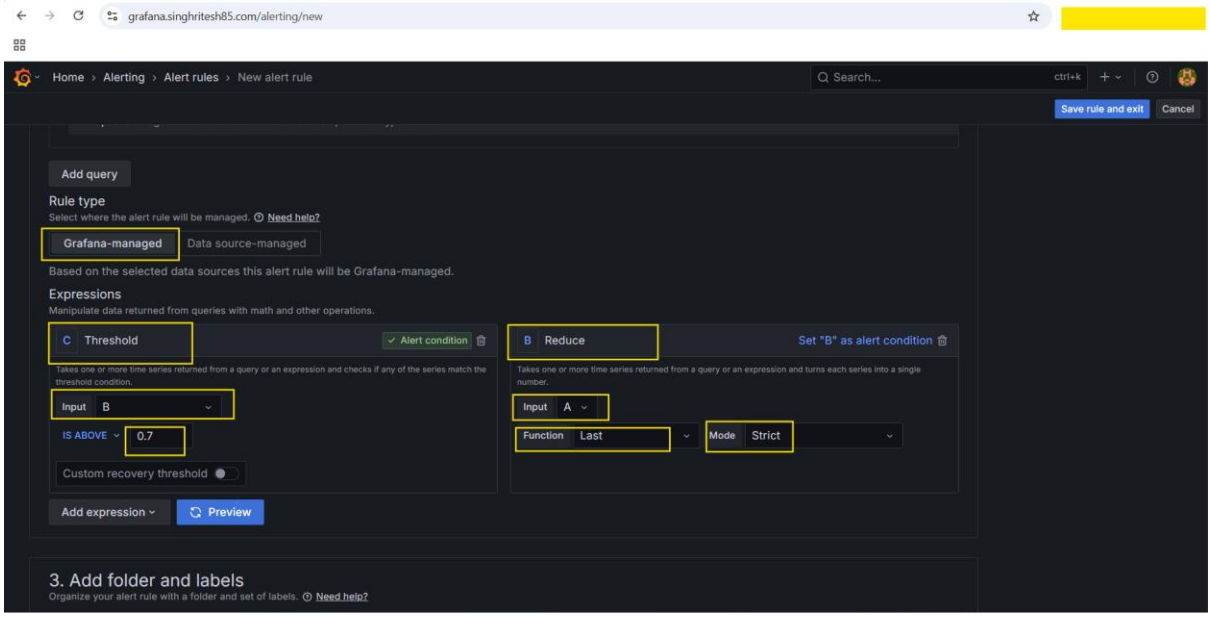
The Default **Notification Policy** had been changed as shown in the screenshot attached below.

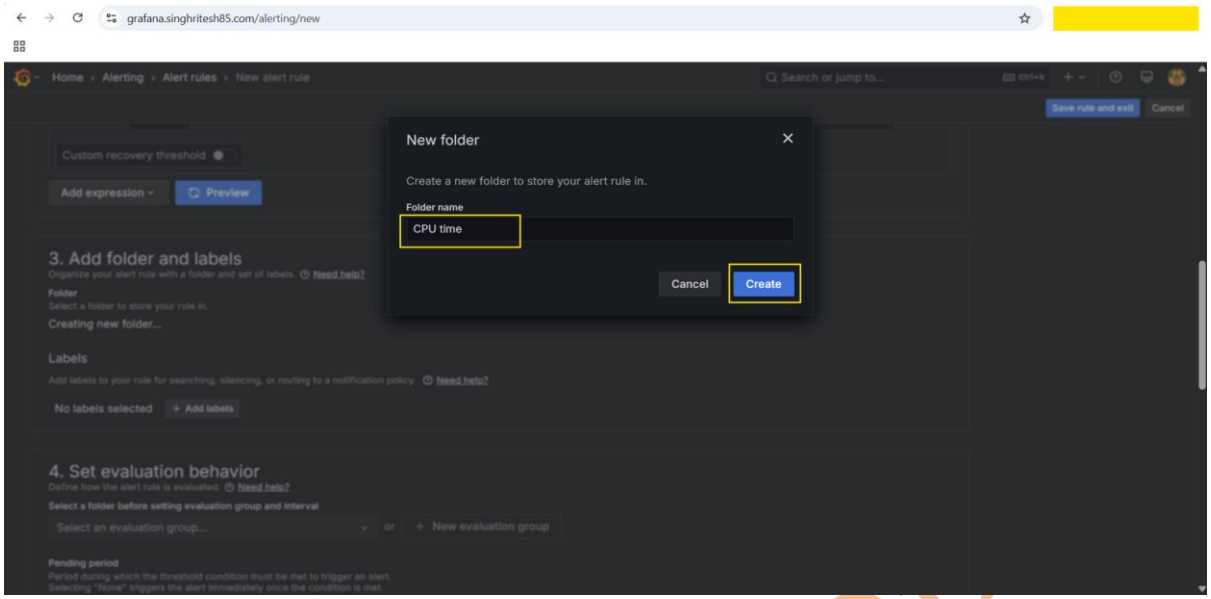


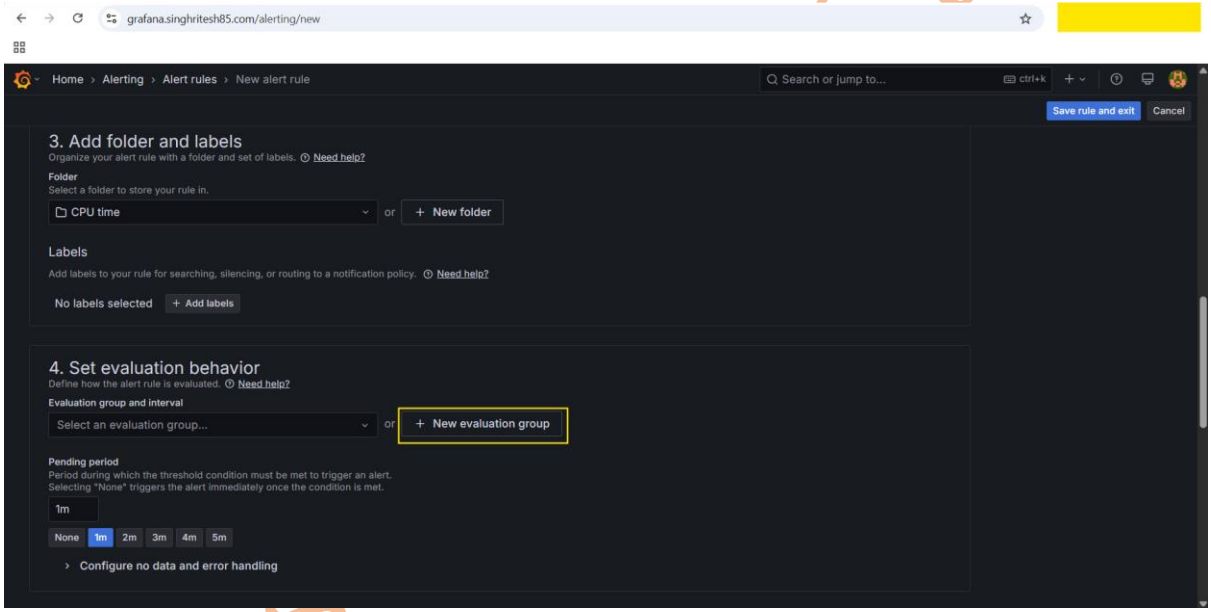
Configure **Alert Rule** as shown in the screenshot attached below.

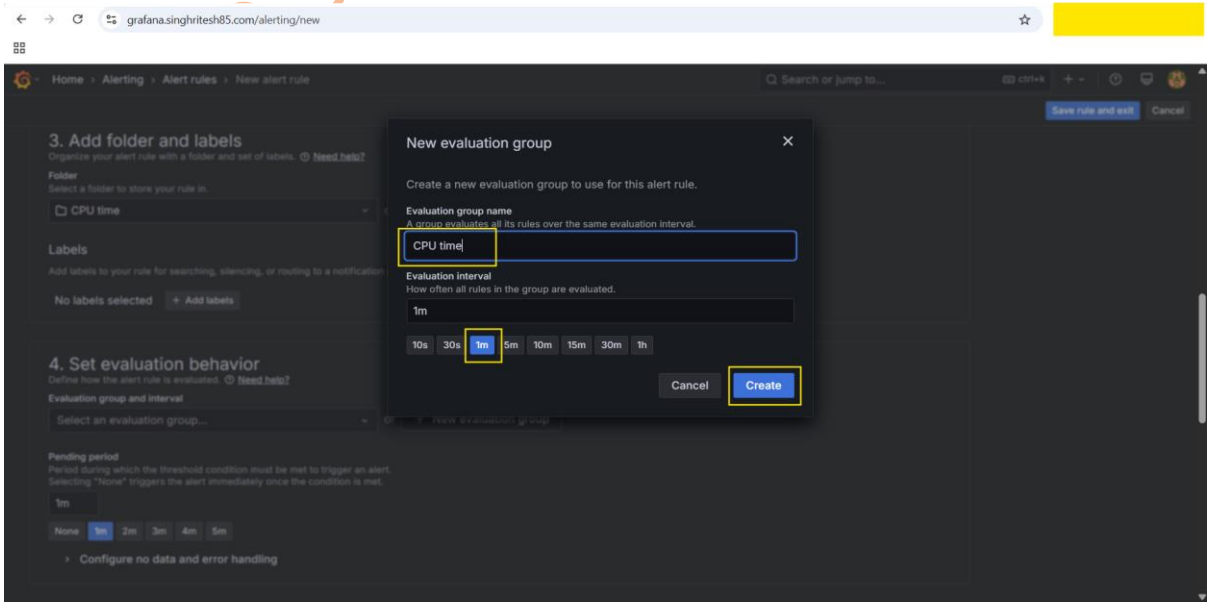

grafana.singhritesh85.com/alerting/list?search=
Home › Alerting › Alert rules
Search or jump to...
ctrl+k
Alert rules
Rules that determine whether an alert will fire
New dashboard
Import dashboard
New alert rule
To create New Alerts
first click on + sign
Search by data sources
All data sources
Dashboard
Select dashboard
State
Firing
Normal
Pending
Rule type
Alert
Recording
Health
Ok
No Data
Error
Contact point
Choose
Search
Search
View as
Grouped
List
State
You haven't created any rules yet
You can also define rules through file provisioning or Terraform. Learn more

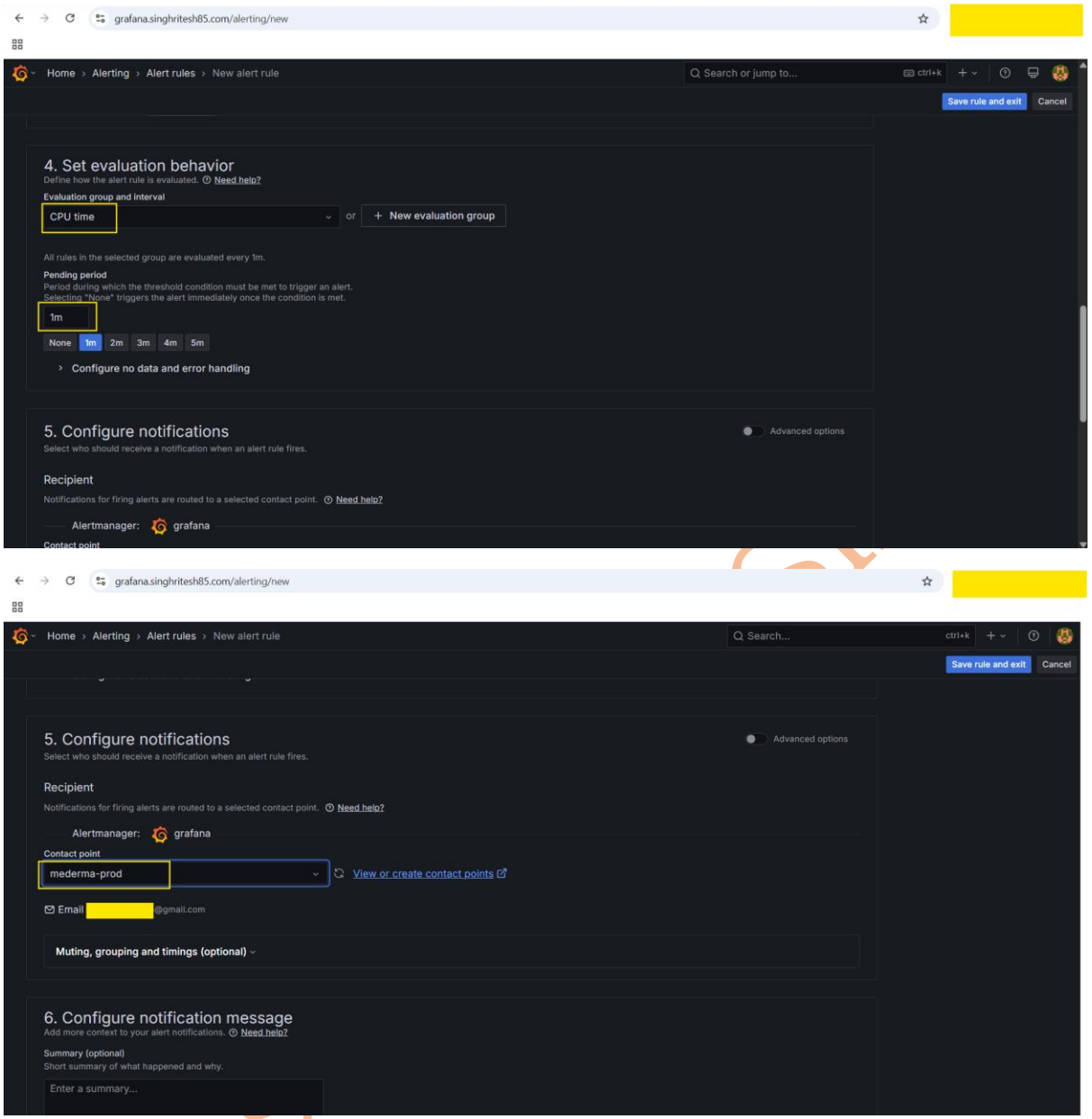

grafana.singhritesh85.com/alerting/new
Home › Alerting › Alert rules › New alert rule
Search...
ctrl+k
Save rule and exit
Cancel
A
prometheus
Options
10 minutes
Set as alert condition
Kick start your query
Explain
Run queries
Builder
Code
Metric
Label filters
process_cpu_seconds_total
job
=
AKS-Cluster
Operations
Hint: add rate
process_cpu_seconds_total{job="AKS-Cluster"}
Options
Legend: Auto
Format: Time series
Step: auto
Type: Instant
Add query
Rule type
Select where the alert rule will be managed.
Need help?
Grafana-managed
Data source-managed
Based on the selected data sources this alert rule will be Grafana-managed.


grafana.singhritesh85.com/alerting/new
Home › Alerting › Alert rules › New alert rule
Search...
ctrl+k
Save rule and exit
Cancel
Add query
Rule type
Select where the alert rule will be managed.
Need help?
Grafana-managed
Data source-managed
Based on the selected data sources this alert rule will be Grafana-managed.
Expressions
Manipulate data returned from queries with math and other operations.
C
Threshold
Alert condition
B
Reduce
Set "B" as alert condition
Input
B
IS ABOVE
0.7
Input
A
Function
Last
Mode
Strict
Custom recovery threshold
Add expression
Preview
3. Add folder and labels
Organize your alert rule with a folder and set of labels.
Need help?

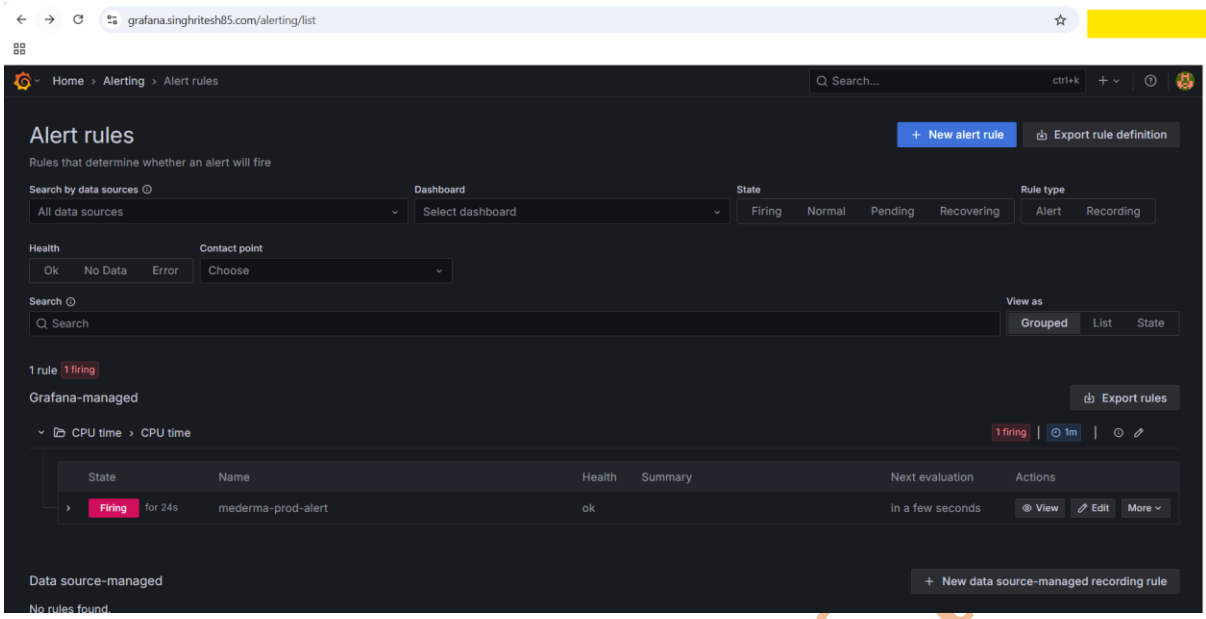

grafana.singhritesh85.com/alerting/new
Home › Alerting › Alert rules › New alert rule
Search or jump to...
Save rule and exit
Cancel
Custom recovery threshold
Add expression
Preview
New folder
Create a new folder to store your alert rule in.
Folder name
CPU time
Cancel
Create
3. Add folder and labels
Organize your alert rule with a folder and set of labels. Need help?
Folder
Select a folder to store your rule in.
Creating new folder...
Labels
Add labels to your rule for searching, silencing, or routing to a notification policy. Need help?
No labels selected
+ Add labels
4. Set evaluation behavior
Define how the alert rule is evaluated. Need help?
Select a folder before setting evaluation group and interval
Select an evaluation group...
or
+ New evaluation group
Pending period
Period during which the threshold condition must be met to trigger an alert.
Selecting "None" triggers the alert immediately once the condition is met.


grafana.singhritesh85.com/alerting/new
Home › Alerting › Alert rules › New alert rule
Search or jump to...
Save rule and exit
Cancel
3. Add folder and labels
Organize your alert rule with a folder and set of labels. Need help?
Folder
Select a folder to store your rule in.
CPU time
or
+ New folder
Labels
Add labels to your rule for searching, silencing, or routing to a notification policy. Need help?
No labels selected
+ Add labels
4. Set evaluation behavior
Define how the alert rule is evaluated. Need help?
Evaluation group and interval
Select an evaluation group...
or
+ New evaluation group
Pending period
Period during which the threshold condition must be met to trigger an alert.
Selecting "None" triggers the alert immediately once the condition is met.
1m
None
1m
2m
3m
4m
5m
Configure no data and error handling

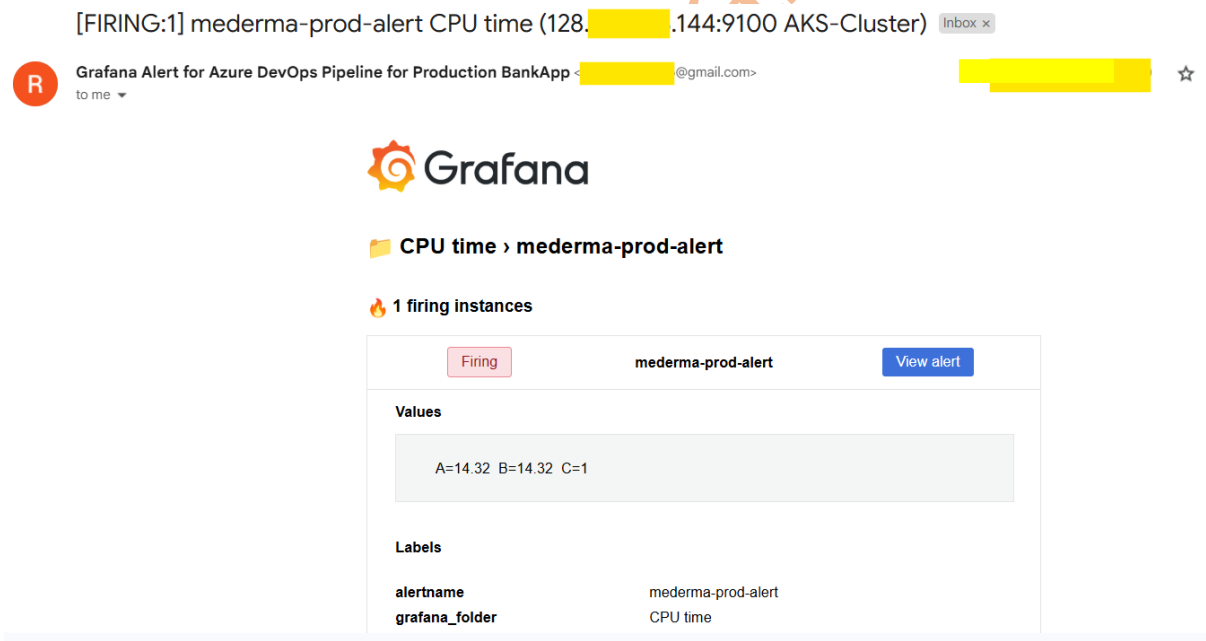

grafana.singhritesh85.com/alerting/new
Home › Alerting › Alert rules › New alert rule
Search or jump to...
Save rule and exit
Cancel
3. Add folder and labels
New evaluation group
Create a new evaluation group to use for this alert rule.
Evaluation group name
A group evaluates all its rules over the same evaluation interval.
CPU time
Evaluation interval
How often all rules in the group are evaluated.
1m
10s
30s
1m
5m
10m
15m
30m
1h
Cancel
Create
Labels
No labels selected
+ Add labels
4. Set evaluation behavior
Evaluation group and interval
Pending period
1m
None
1m
2m
3m
4m
5m
Configure no data and error handling

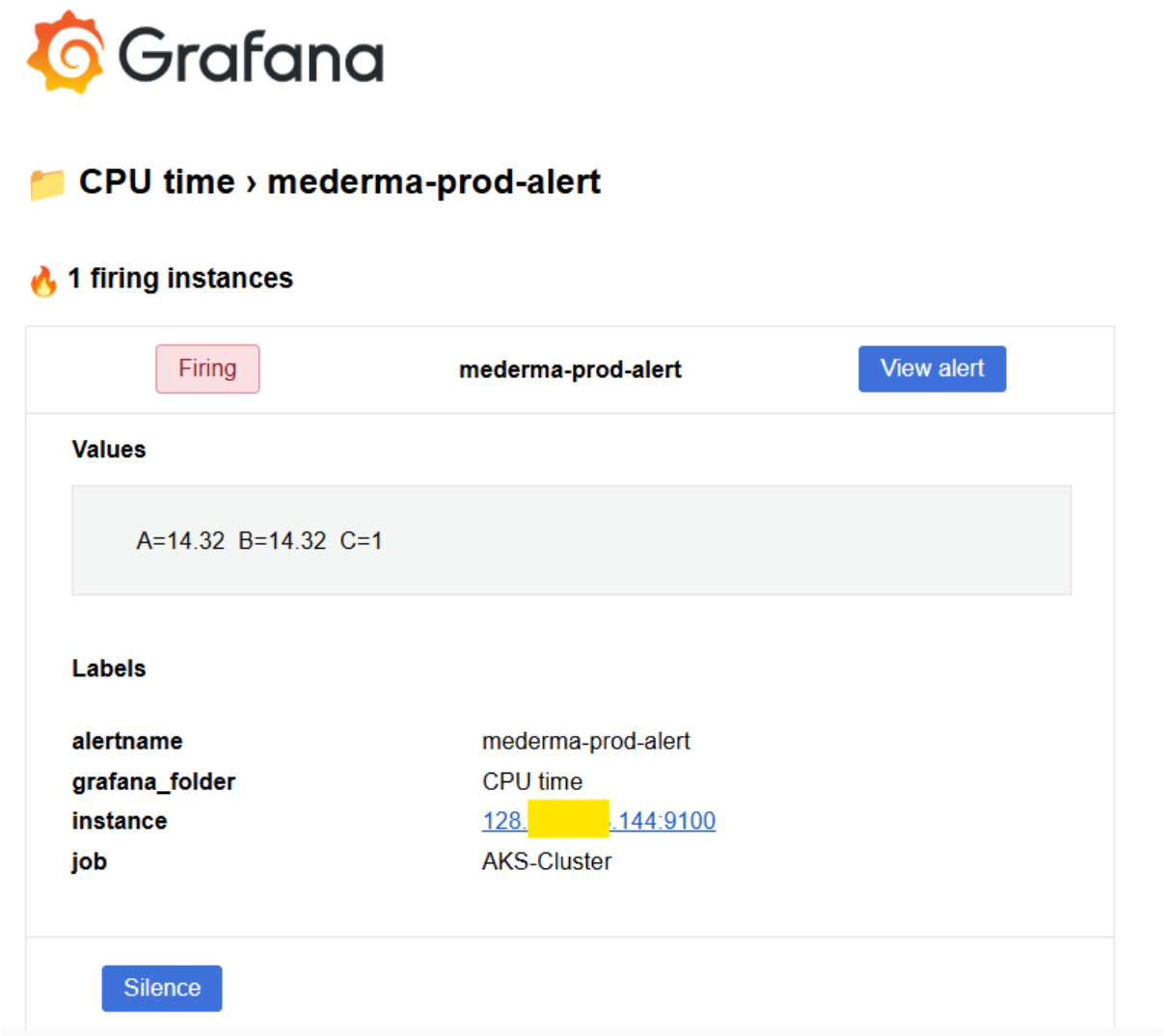


If the Alert Rule is in firing state after condition crosses the threshold condition, then Grafana console screenshot will be showing the same as shown in the screenshot attached below.



An Email was sent to the Email ID as shown in the screenshot attached below.




Grafana
CPU time › mederma-prod-alert
1 firing instances
Firing
mederma-prod-alert
View alert
Values
A=14.32 B=14.32 C=1
Labels
alertname
mederma-prod-alert
grafana_folder
CPU time
instance
128. .144:9100
job
AKS-Cluster
Silence

Source Code: - https://github.com/singhritesh85/Bank-App-multibranch.git

GitHub Repo: - https://github.com/singhritesh85/DevOps-Project-BankApplication-Multibranch-MultiCloud.git

Helm Chart: - https://github.com/singhritesh85/helm-repo-for-ArgoCD.git

https://github.com/singhritesh85/helm-repo-for-bitnami.git

Terraform Script: - https://github.com/singhritesh85/DevOps-Project-BankApplication-Multibranch-MultiCloud.git

Reference: - https://github.com/Goldencat98/Bank-App.git