Azure Devops:

1.What is difference b/w Deployment group and Environments?

They both have the same theory in actual deployment. But since Deployment group jobs are not yet supported in YAML. In this case, the Environments comes up.

2.What is difference b/w build pipeline and release pipeline?

A Build Pipeline is used to generate Artifacts out of Source Code. A Release Pipeline consumes the Artifacts and conducts follow-up actions within a multi-staging system

3.How to configure approval in azure devops?

In your Azure DevOps project, go to the resource (for example, environment) that needs to be protected.

Navigate to Approvals and Checks for the resource.

Control option give to time out 1 day, 2 days any time give it and apply.

4.what is b/w Microsoft-hosted agents and Self-hosted agents?

Ans: Microsoft-hosted agents If your pipelines are in Azure Pipelines, then you’ve got a convenient option to run your jobs using a Microsoft-hosted agent. With Microsoft-hosted agents, maintenance and upgrades are taken care of for you. Each time you run a pipeline, you get a fresh virtual machine for each job in the pipeline. The virtual machine is discarded after one job (which means any change that a job makes to the virtual machine file system, such as checking out code,will be unavailable to the next job). Microsoft-hosted agents can run jobs directly on the VM or in a container. Azure Pipelines provides a predefined agent pool named Azure Pipelines with Microsoft-hosted agents.

Self-hosted agents an agent that you set up and manage on your own to run jobs is a self-hosted agent. You can use self-hosted agents in Azure Pipelines or Azure DevOps Server, formerly named Team Foundation Server (TFS). Self-hosted agents give you more control to install dependent software needed for your builds and deployments. Also, machine-level caches and configuration persist from run to run, which can boost speed.

How to enable ci/CD?

Build pipeline –trigger –enable the Contiune intergration Release pipeline– release –contiune deployment trigger.

What is Azure Artifacts?

What Is Deployment groups?

A deployment group is a logical set of deployment target machines that have agents installed on each one. Deployment groups represent the physical environments; for example, “Dev”, “Test”, or “Production” environment

In effect, a deployment group is just another grouping of agents, much like an agent pool.

Deployment groups are only available with Classic release pipelines and are different from deployment jobs

Deployment groups:

A deployment group is a logical set of deployment target machines that have agents installed on each one. Deployment groups represent the physical environments; for example, “Dev”, “Test”, “UAT”, and “Production”. In effect, a deployment group is just another grouping of agents, much like an agent pool.

Environments:

Environment represents a collection of resources such as namespaces within Kubernetes clusters, Azure Web Apps, virtual machines, databases, which can be targeted by deployments from a pipeline.

What is azure Artifacts?

Azure Artifacts enables developers to share their code efficiently and manage all their packages from one place. With Azure Artifacts, developers can publish packages to their feeds and share it within the same team, across organizations, and even publicly.

Developers can also consume packages from different feeds and public registries such as NuGet.org or npmjs.com. Azure Artifacts supports multiple package types such as NuGet, npm, Python, Maven, and Universal Packages.

Need to create new feed in artifacts and create new pipeline for using need feed like Nuget–pack and push. then run pipeline along with feed once it get success add to feed to our pipeline.

Just open build pipeline add task like nuget and added feed then run pipeline artifacts will store that place.

2nd Way:

Through visula studio create new project– dependecy package source-browse some xyz feed 3rd way:

and through run power shell will particaller command will configure we use it.

What is Azure Key Vault secrets?

Azure Key Vault enables developers to securely store and manage secrets such as API keys, credentials or certificates. Azure Key Vault service supports two types of containers: vaults and managed HSM (hardware security module) pools. Vaults support storing software and HSM-backed keys, secrets, and certificates, while managed HSM pools only support HSM-backed keys.

Need to create azure key-vault in azure portal and configure with azure pipeline

Goto key-valut navigation bar and create key-vault.

How to create Azure key vaults?

Release pipeline —agent –> tasks–Display name–>select azure subscription–>select key-vault–>secrets filter

Tasks:

https://learn.microsoft.com/en-us/azure/devops/pipelines/release/azure-key-vault?view=azure-devops&tabs=yaml

1.Key vault tasks

2.Powershell tasks

3.Copy file to

https://learn.microsoft.com/en-us/azure/devops/pipelines/release/deploy-using-approvals?view=azure-devops

21-06-2023:

https://www.azuredevopslabs.com/labs/vstsextend/azurekeyvault/

22-02-2023:

<https://learn.microsoft.com/en-us/azure/devops/organizations/security/access-levels?view=azure-devops>

Access level:

1. Basic
2. Basic +Test plan
3. Stakeholder

Azure Repos:

Azure DevOps services access

Admin----Devops person and Manager

Contributor ----developers

Reader--- developers

23-06-2023:

Service connection:

Project setting –Create service connection –

Scope level

Subscription

Subscription –name

Resource Group name

Service connection –name will click on save button

Then service will be established

Service principal establishing

Azure Key-vault

Azure Service principal

Azure Manage identity

Azure active directory

Azure devops to ACR connection

Azure Devops Key-vault

Dot net application—build pipeline

.Net core

MSbuild

Replace token

Copyfile

Publish build artifacts

Release pipeline.

Developer

Function apps

Web apps

Logic apps

APIM

PR approves

Build pipeline

.Net core

MSbuild

Replace token—optional

Copyfile

Publish build artifacts

Release pipeline

Function apps-- Azure function app

Azure key vault

Web apps ----- Azure app service deploy

Logic Apps: ARM Template deployment

Azure App service Deploy

Azure key vault

APIM: ARM template deployment

SQL/ mysql/ Oracle DB connection

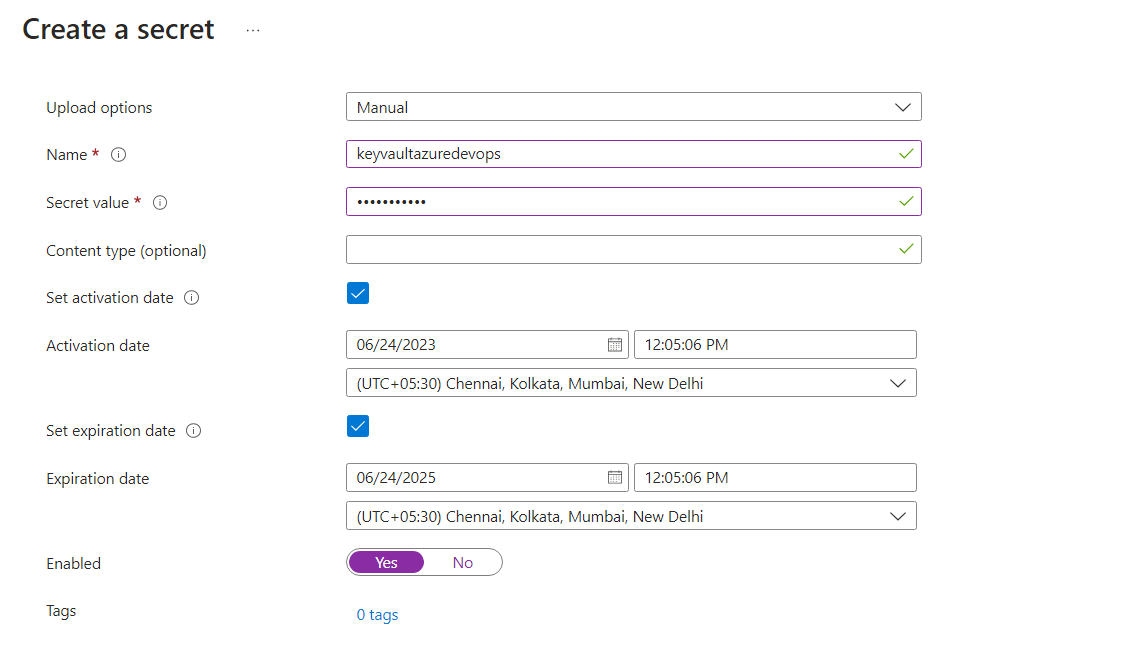
<https://www.nitorinfotech.com/blog/databases-using-azure-devops>

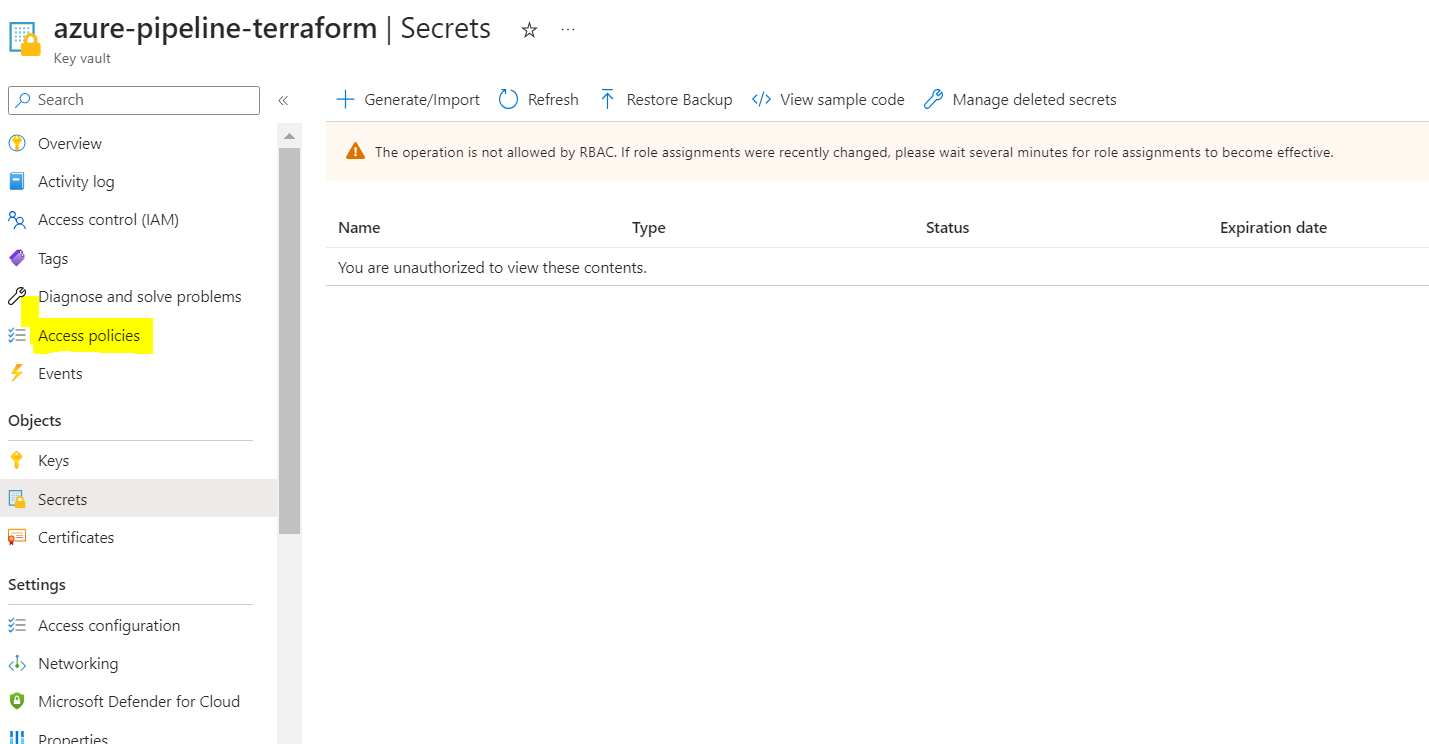
Azure key vault

<https://medium.com/geekculture/how-to-use-key-vault-secrets-in-azure-pipelines-658198f2eea6>

24-06-2023:

Key vault created in portal then create secretes



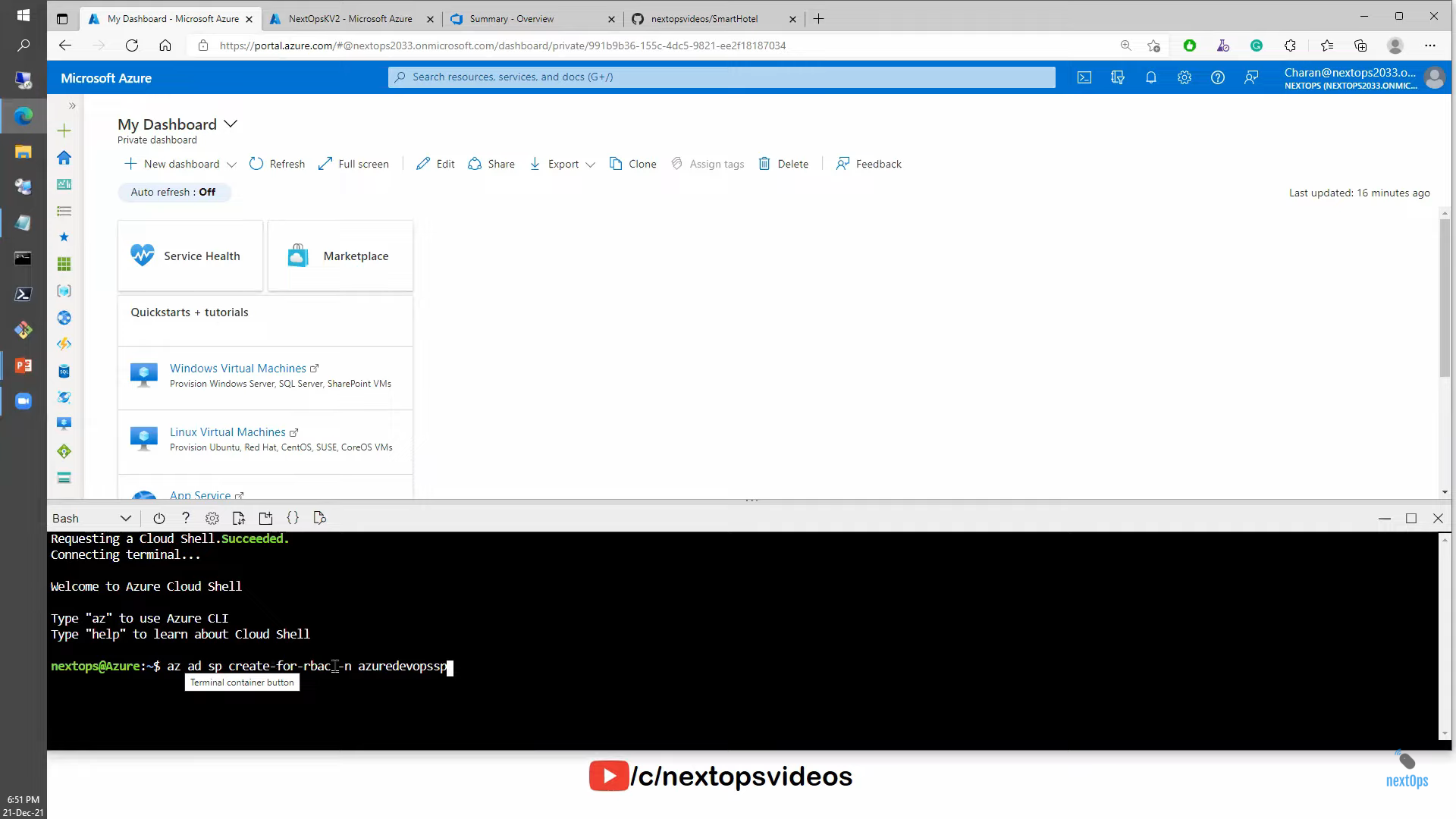


From here to accessing to azure devops in pipeline expose and secure secrets

Then will give access to azuredevops—goto access polices

Manage identity

Service principal



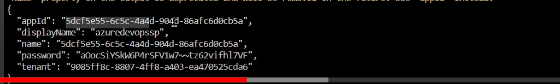
Power shell command :

Az ad sp create-for-rbac -n azuredevopssp

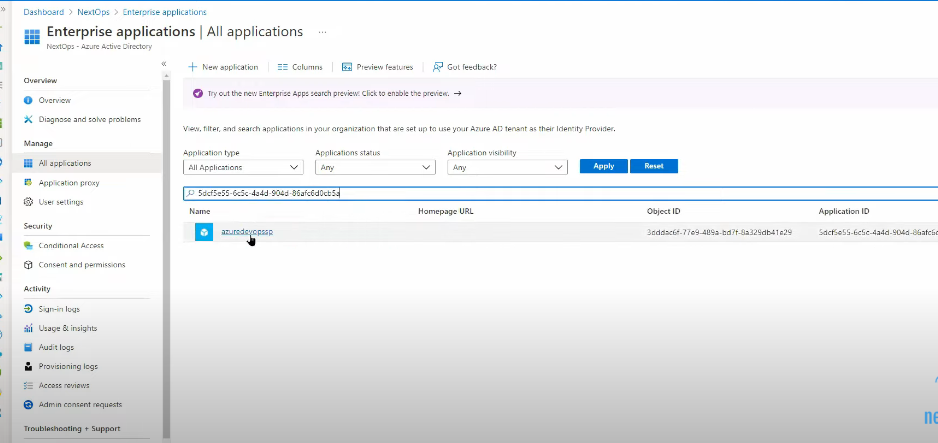
**Service principals** can be used for automated processes like scripts, CI/CD pipelines, and other automation scenarios.

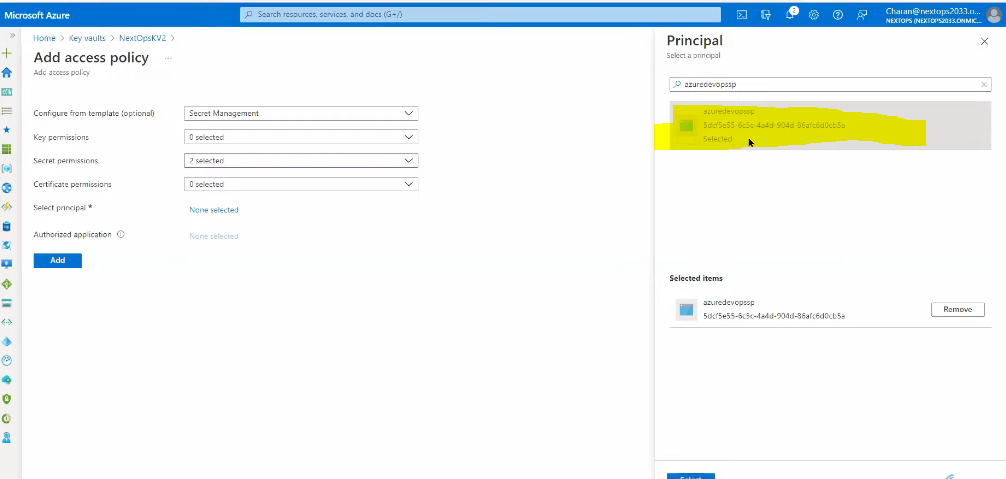
The main difference between the two is that Managed Identity is tied to a specific Azure resource while Service Principal is a standalone identity

This will create and getting



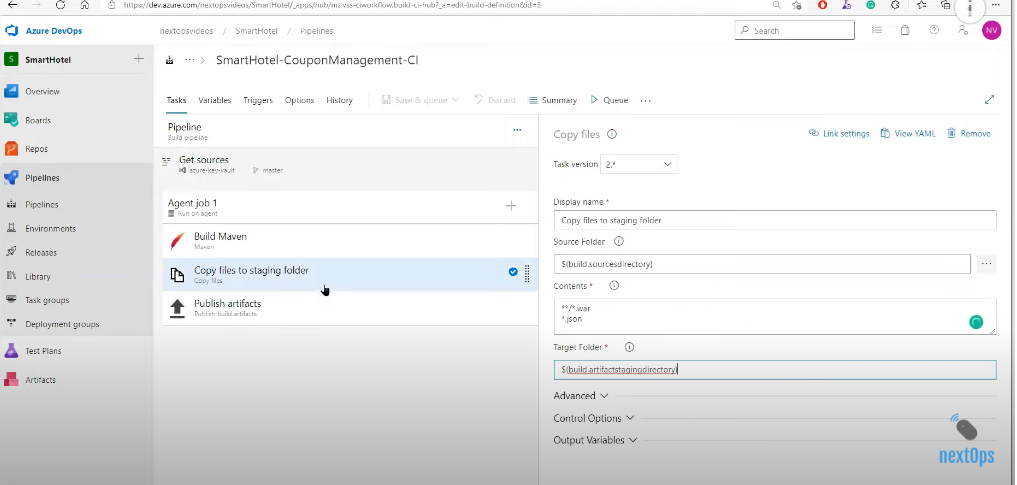
Where can stored this values in

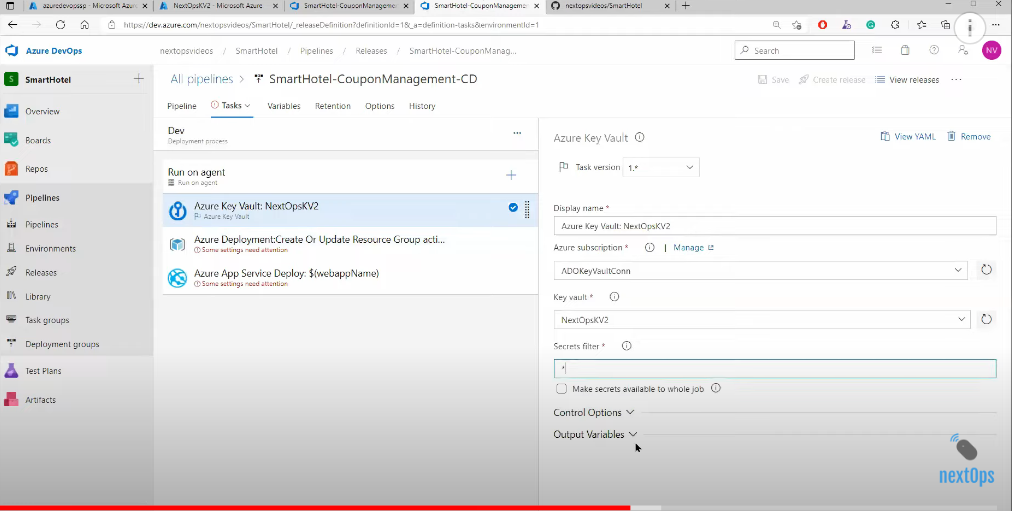




Just add then infra creation done.

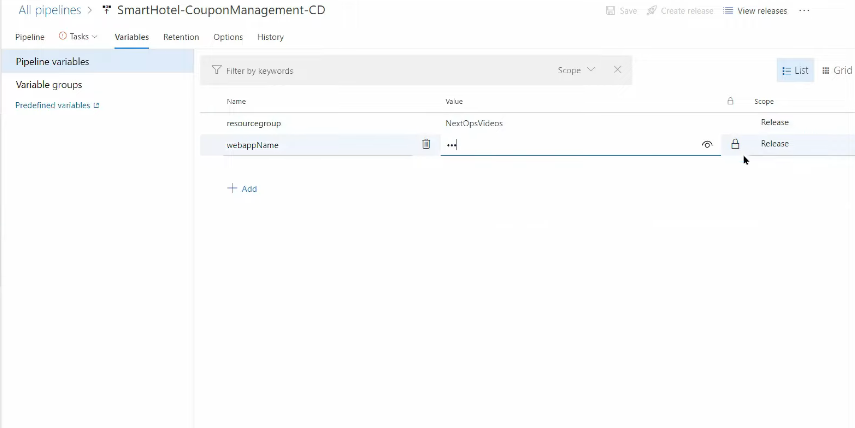
Pipeline creation

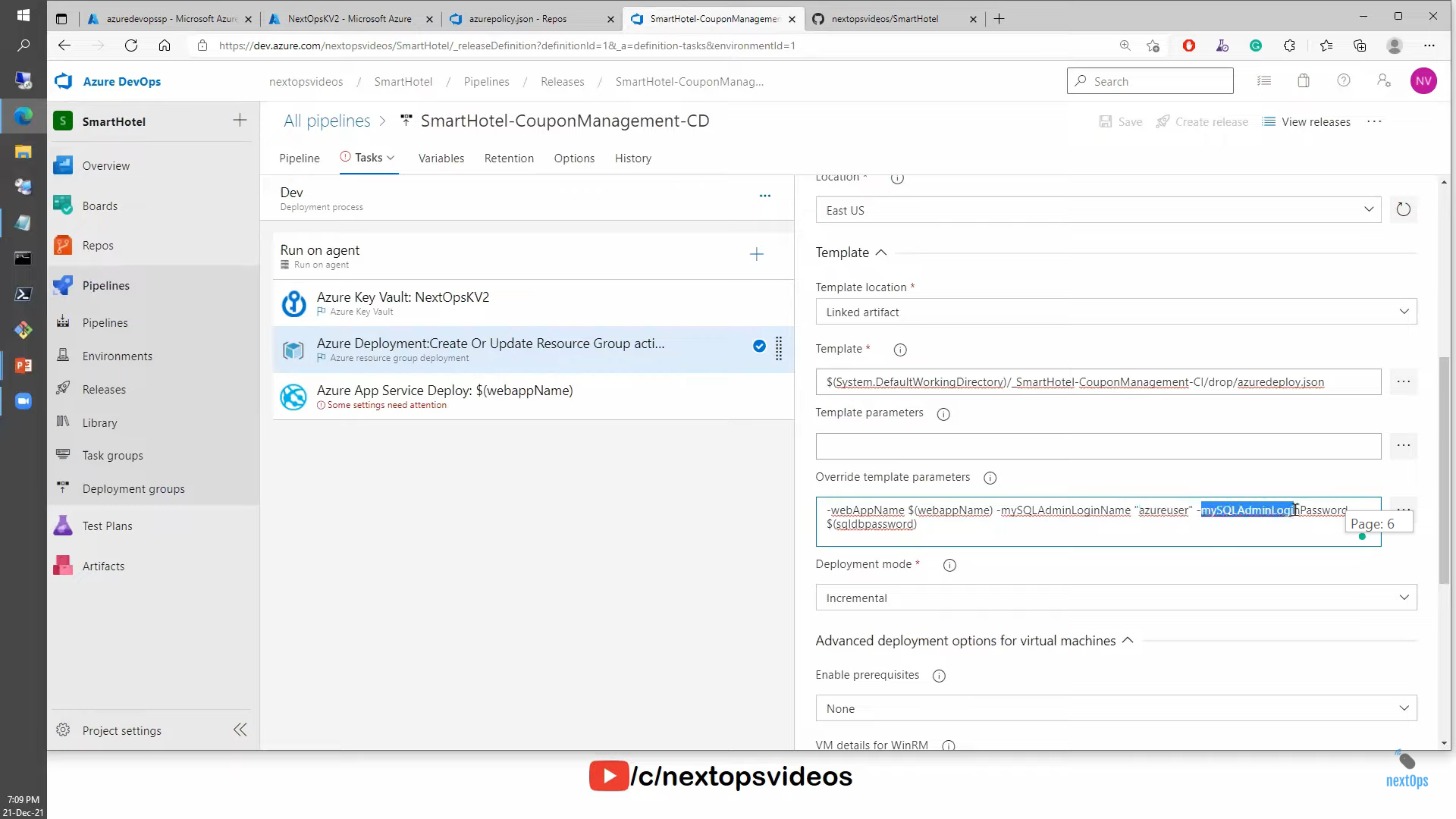




Pipeline Variable----

Variable groups





Web app deploy ---web app host url--

Mysql server deploy then user name and password.

What is Azure key vault?

Centralized secrets

Which application need to access and policy

Key

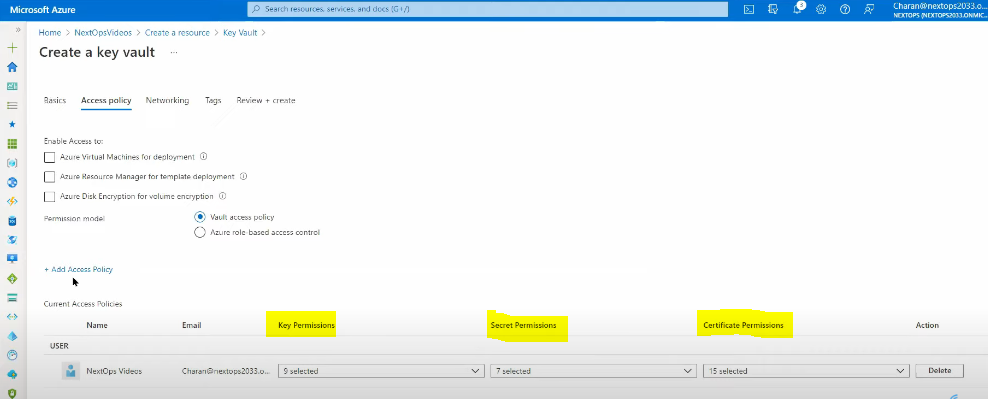
Certificates

Secrets

Soft- delete

If key vault want see and access rbac

Add policy ---



Setting

Key if connect to Linxu machine---will use key

Secret –just create secret like password. GENERATED URI

Access policy give one user

But we want give machine

27-06-2023:

**Azure DevOps End 2 End Project with AKS, ACR and Azure SQL**

Project Setup:

1. Prepare source code repository
2. Prepare infrastructure
3. Create build pipeline
4. Create release pipeline
5. Enable CI/CD
6. Run the pipeline
7. Verify the deployment

Let switch to azure portal

1.Creating resource group and virtual network

2.Create AKS cluster, while creating AKS integration ACR

At time will creating AKS and ACR.

3.Creating SQL Database, create new database along with networking and authentication

**Azure Devops Setup**

1.Azure Devops project preparation

2.Organization level access

Kubernetes extension

Replace token

No paralasim in public project.

Classic editor—empty job.

Build Pipeline

1. Replace token--
2. Replace token—
3. Docker compose
4. Docker compose
5. Docker compose
6. Docker file and docker compose –multiple -container run in docker compose
7. Copy file to
8. Publish build artificats

Variable and variable groups

ACR

SqlServer

Sqluser

Sql password

3. Docker compose -Azure subscription azure portal to azure devops like service connection

Project name

Action –like docker run image

Action -like build service image

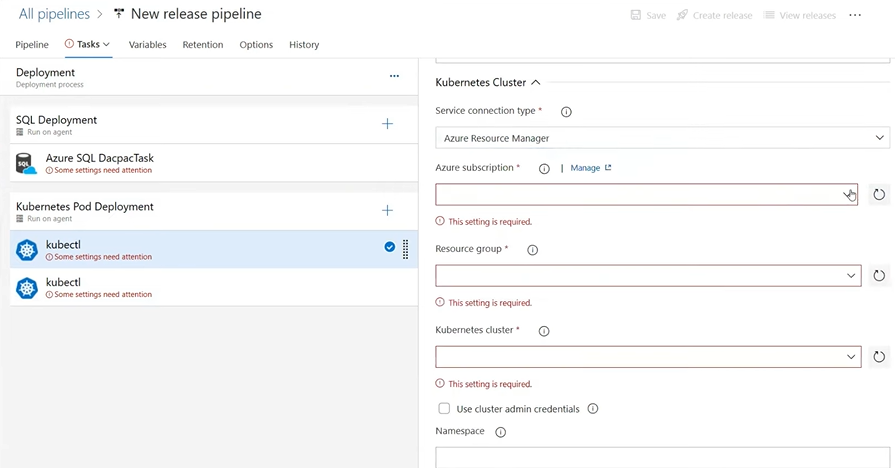
Action –like push service image

Action -like log service image.

All tasks save and run pipeline.

Release pipeline

Empty job:



Agent Job: SQL Deployment

Task :dacpack

Agent job: Kubenretes Deployment

Task1: Kubctl ->Apply command🡪 path

Task 2:Kubctl:🡪Kubctl set🡪path

Variable 🡪

Sqlserver

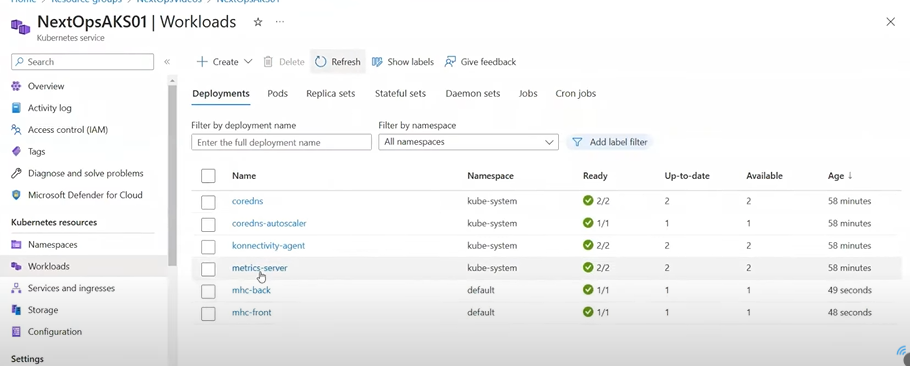
ACR

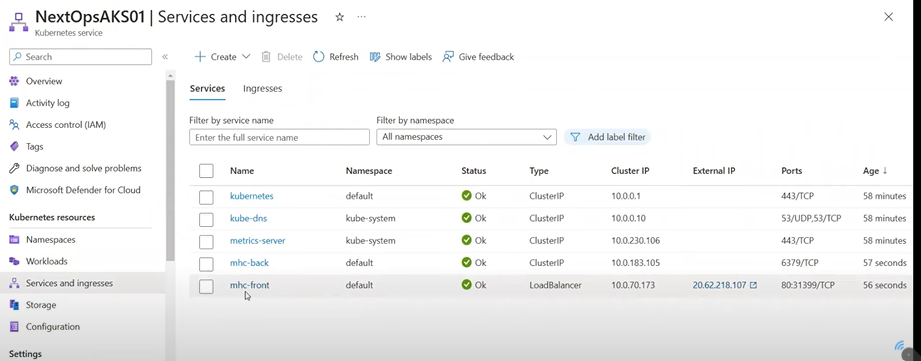
Sqluser

Sqlpassword

Then release pipeline will deploy and log just click logs we can see the logs.

Will see repositories files in the portal





Load balancer service like IP Address access to external.