**Implementação de IaC com Template ARM**

**KeyVault**

**20/03/2020/ Versão 1.0**

Sumário

[Tabela de versionamento 3](#_Toc36030605)

[Introdução 4](#_Toc36030606)

[Criação visual via Portal 4](#_Toc36030607)

[Utilização do Pipeline do Azure Devops via ARM Templates 6](#_Toc36030608)

[Pré-requisitos (Dependências) 6](#_Toc36030609)

[Variáveis 6](#_Toc36030610)

[Scripts JSONs - REPOS 6](#_Toc36030611)

[Criação PipeLine 7](#_Toc36030612)

[Variáveis 7](#_Toc36030613)

[Tasks 8](#_Toc36030614)

# Tabela de versionamento

|  |  |  |  |
| --- | --- | --- | --- |
| Versão | Data | Autor | Descrição |
| 0.1 | 12/03/2020 | Daniel Huanca | Inseridos Prints e informação coletada. |
| 1.0 | 20/03/2020 | Daniel Huanca | Estruturada documentação seguindo Template definido. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introdução

Documentação de implementação do componente KeyVault utilizando ARM Templates do Azure no Pipeline do Azure Devops.

Com o intuito de provisionar a infraestrutura necessária para migração das aplicações On-premises para a Cloud Azure de maneira automatizada. ( CI / CD)

# 

# Criação visual via Portal

Seguem telas coletadas na interface gráfica do Portal contendo todos os campos necessários para criação do serviço no Azure.

Componentes básicos para provisionamento do Serviço.

**- Key vault name**

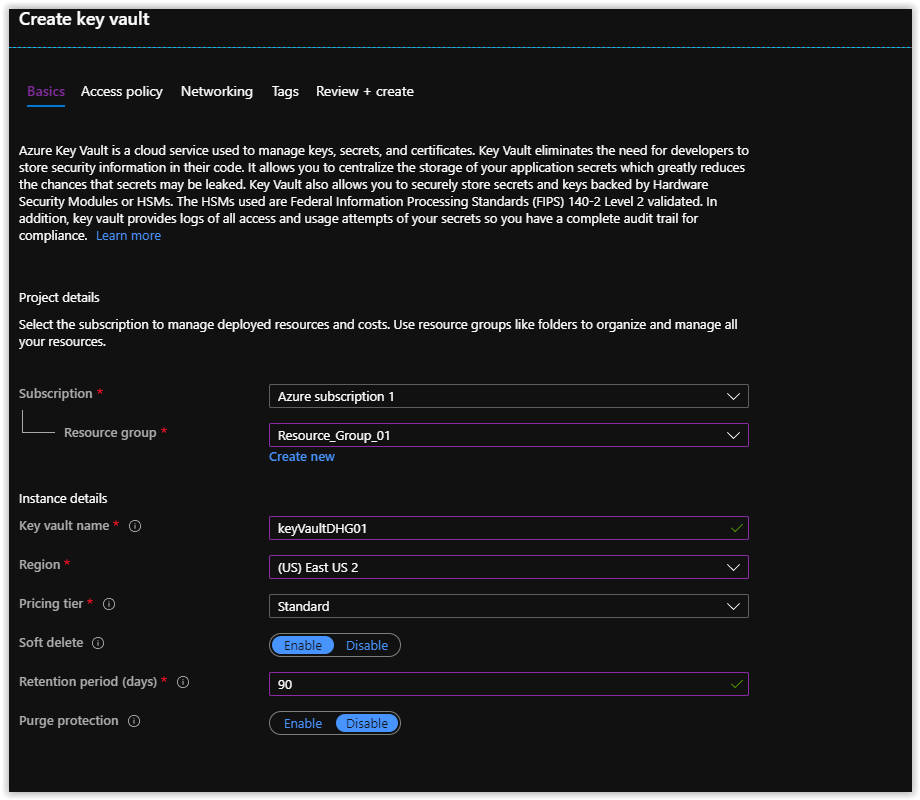
**- Region**

**- Pricing tier**

**- Soft delete**

**- Retention period**

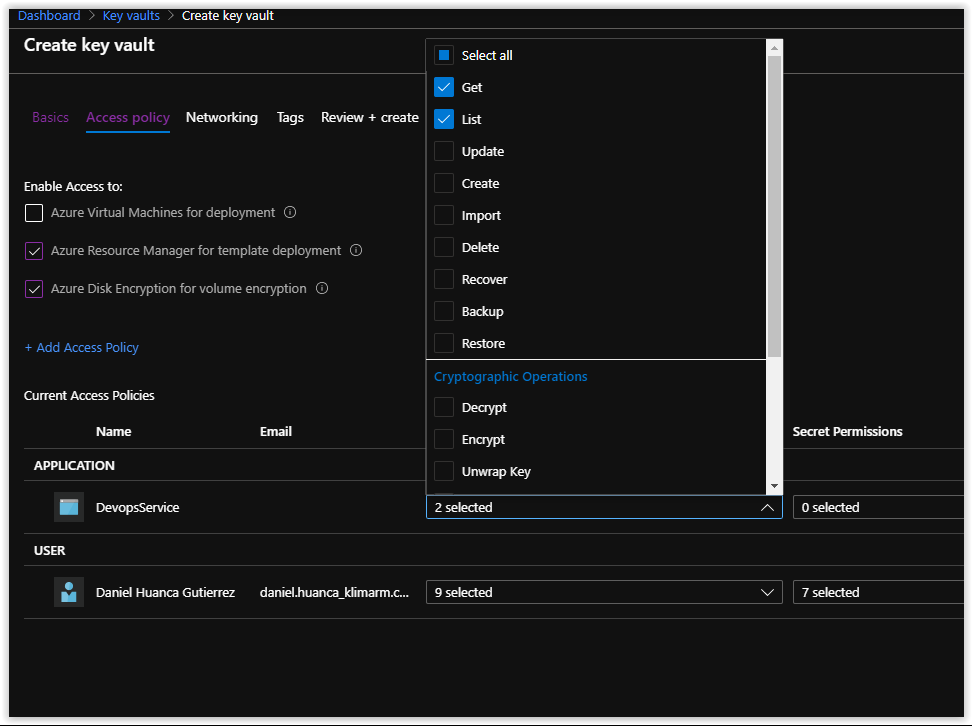
**- Purge Protection**



Configuração de Acess policy.

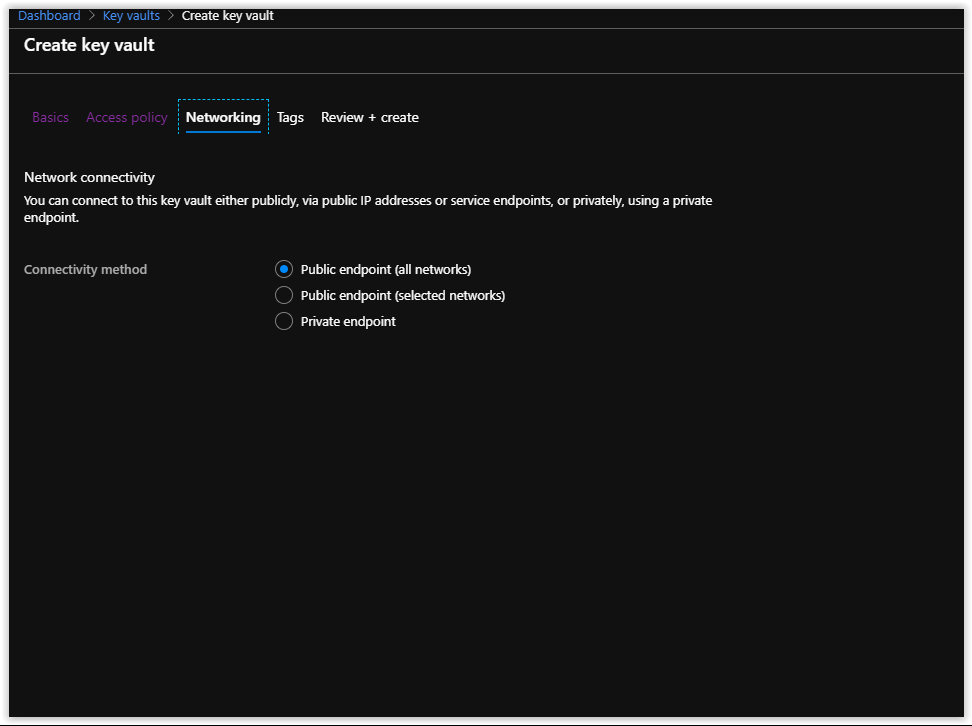
Importante pois disso depende a comunicação do Azure Devops com o Key Vault.

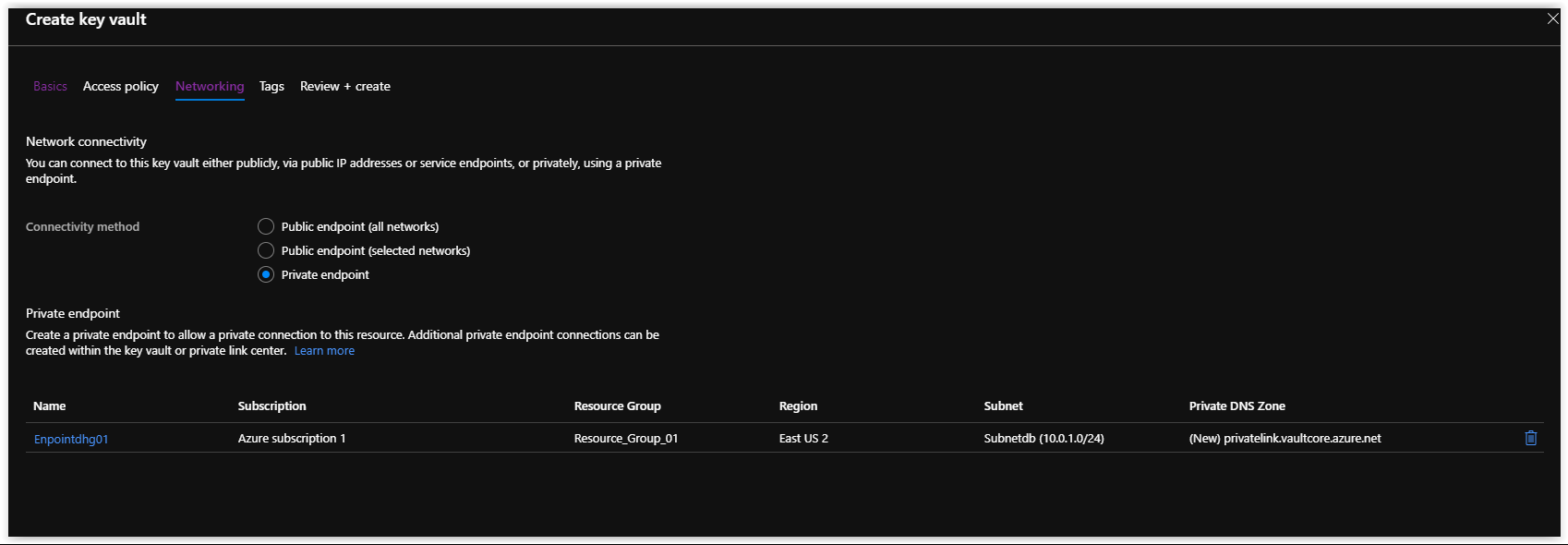
Adicionar uma Policy para aplicação de DevopsService contendo as permissões de acesso ao Sercrets (Get e List)



Networking

Configurações de conexão de acesso ao KeyVault, nesse caso definito Private endpoint e adicionado IP do analista que fará a gestão dos Secrets.





# Utilização do Pipeline do Azure Devops via ARM Templates

Descrição de pré requisitos, configuração de variáveis necessárias e explicação dos Scripts JSONs.

## Pré-requisitos (Dependências)

O(s) pré-requisito(s) da aplicação é(são) o(s) seguinte(s):

**- Resource Group**

**- Service Connection (Entre Azure e Azure Devops)**

**- Service Principals**

Sem a prévia criação do(s) componente(s) acima não é possível o deploy do serviço.

## Variáveis

Particularidades do serviço para correta implementação via PipeLine.

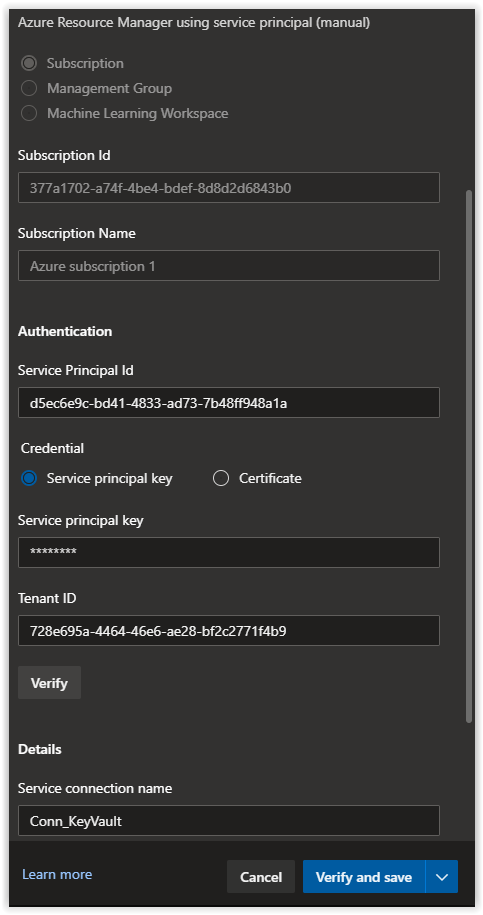
No caso do Key Vault, existem algumas particularidades antes de realizar o Release (CD) do KeyVault.

É necessário a configuração de Um Service Principals no Azure para que o Azure Devops se conecte ao Azure.



Com o Service Principals, anotar o Application ID, Key e Tenant ID.

E criar uma Service Connection no Azure Devops.



## Scripts JSONs - REPOS

Componentes:

**- Template:**

O Script utilizado para coleta genérica e que foi utilizado para testes.

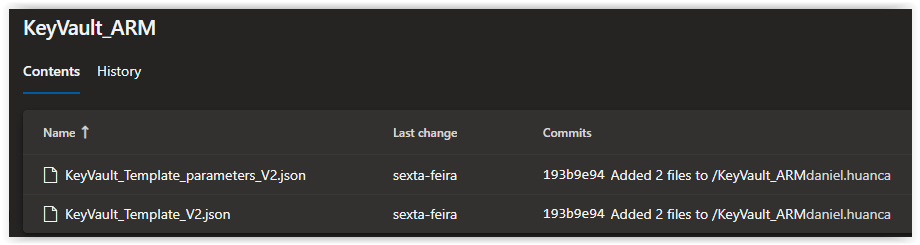
- KeyVault\_Template.json

**- Parameters**

JSON de Parâmetros utilizado para testes. O qual deve ser utilizado como base para eventuais customizações ou levantamento de variáveis.

- KeyVault \_Template\_Paramenters.json

Devem estar corretamente no REPOS para serem utilizados como artefato do Pipeline:



## Criação PipeLine

Na pipeline da aplicação a ser migrada, criar uma task separada e antes dos pacotes de aplicação para provisionamento da Infraestrutura como código. (IaC)

Cada aplicação possui as suas particularidades que devem ser levantadas no planejamento da migração para definir os componentes a serem provisionados.

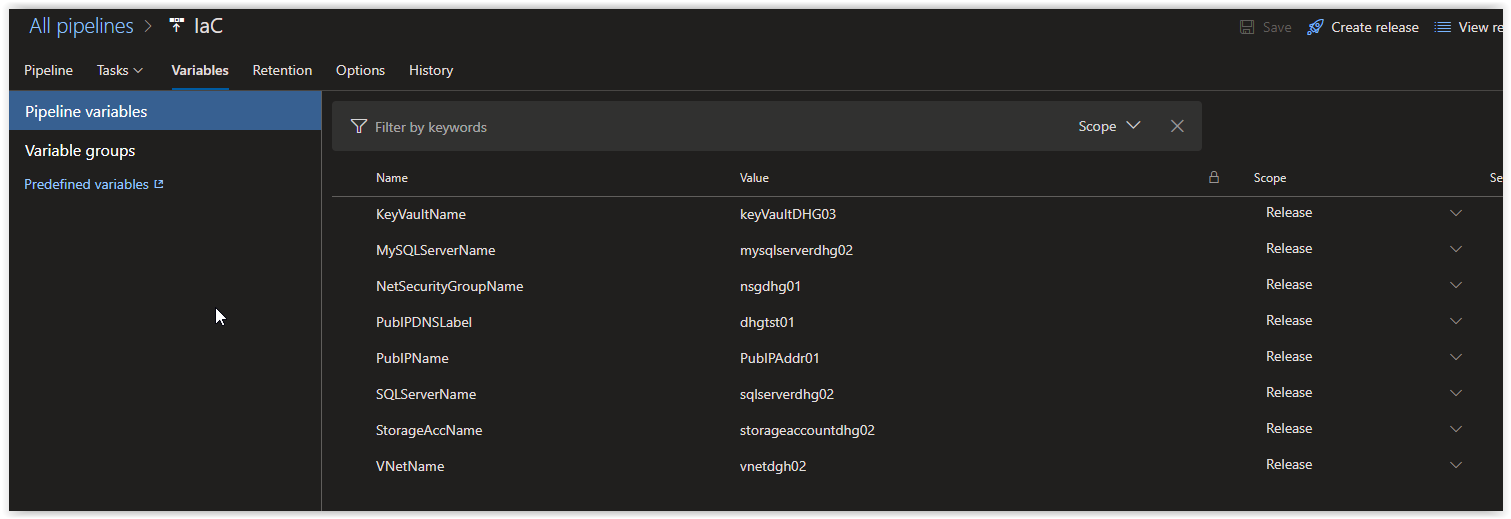
### Variáveis

Cadastrar na Pipeline previamente as variáveis que serão utilizadas pelo Deploy da infra, seguindo nomenclatura pré definida.

Variáveis utilizadas pelo Serviço MySQL Server

**- KeyVaultName**

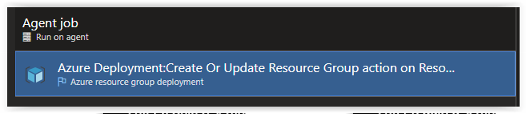
Conforme evidência abaixo as variáveis de cada aplicação deverão ser cadastradas nas variáveis do Pipeline.



### Tasks

Para Deploy da infra para o KeyVault são necessárias as Tasks abaixo:

- Tipo de Task – Azure resource group deployment



**Campos a serem utilizados:**

**- Subscription:** Subscrição aonde será realizado o deploy da infra.

**- Resource Group:** Resource group aonde será feito o deploy da infra.

**- Location:** Região onde será realizado o deploy da infra.

**- Template:** Local no REPOS onde se encontra o Template.

**- Parameters:** Local no REPOs onde se encontra o arquivo de Parameters.

**- Override template parameters:** Campo utilizado para sobrescrever parâmetros descritos no arquivo de parâmetros, ideal realizar a alteração nesse campo, evitando alterações no template/parameters que serão utilizados em outras aplicações.

Deve ser utilizado no seguinte formato:

**-PARAMETRO VALOR**

Onde **PARAMETRO** está ligado ao existente no arquivo JSON de parâmetros como abaixo:



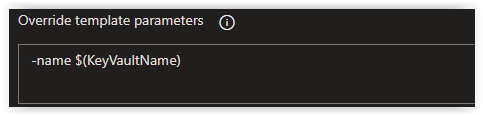
E **VALOR** o nome estabelecido por convenção de nomenclatura ou configuração.

**Parâmetros que podem (Conforme necessidade) ser alterados do SQL Server:**

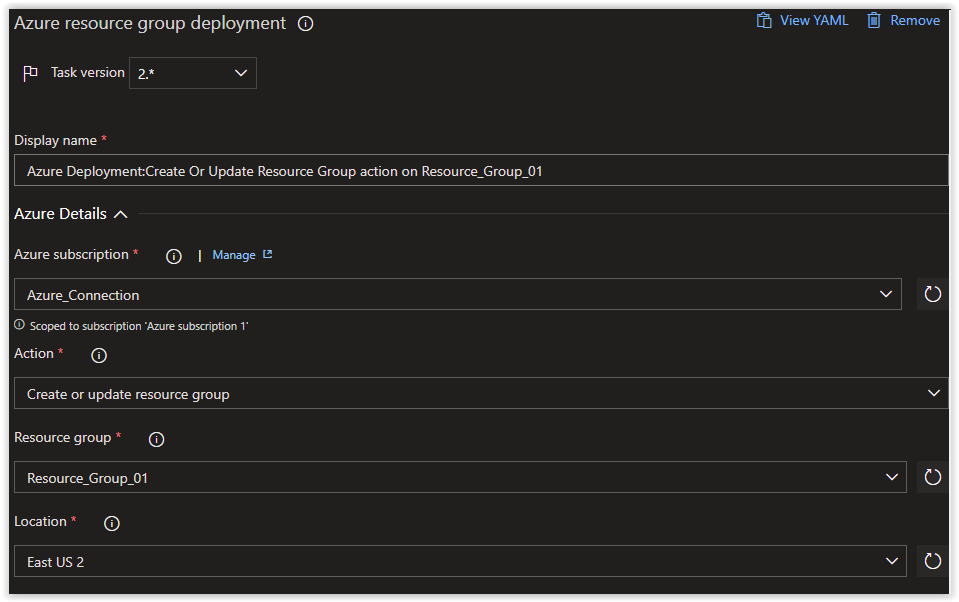
- Location

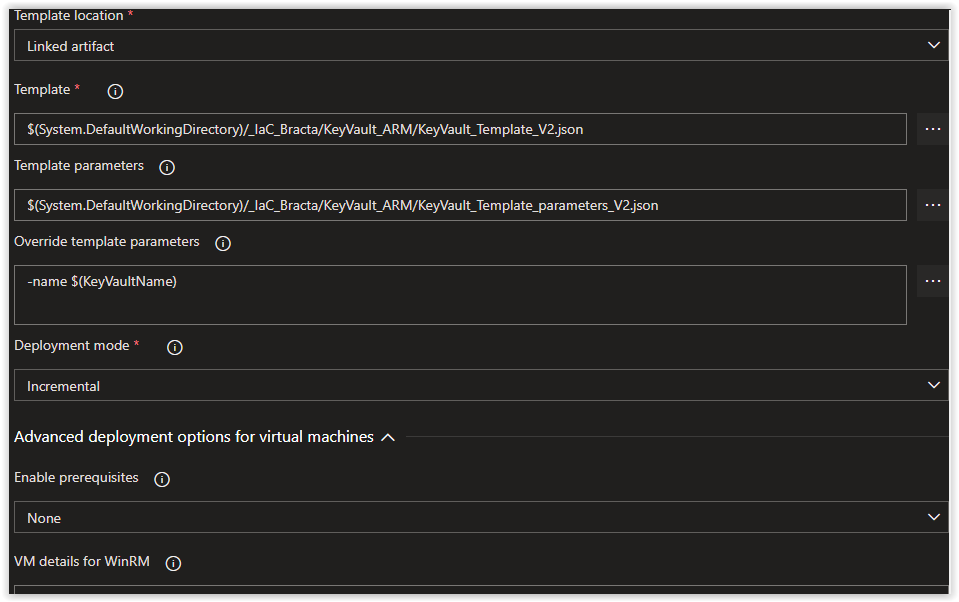
O(s) único(s) obrigatório(s) seria(m), **name** que no exemplo é(são) trazido(s) das variáveis cadastrada anteriormente estando no formato **$(VARIAVEL\_CADASTRADA)**.

Dessa maneira ao executar o provisionamento é utilizado o nome de instancia e login de administrador solicitados.



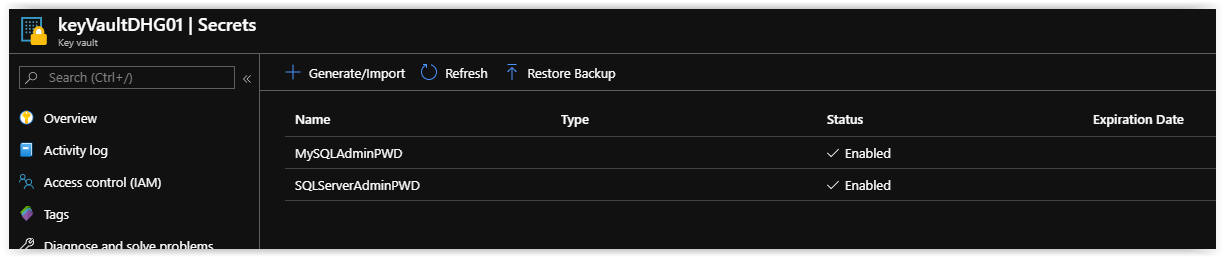
**Tela de config da task:**



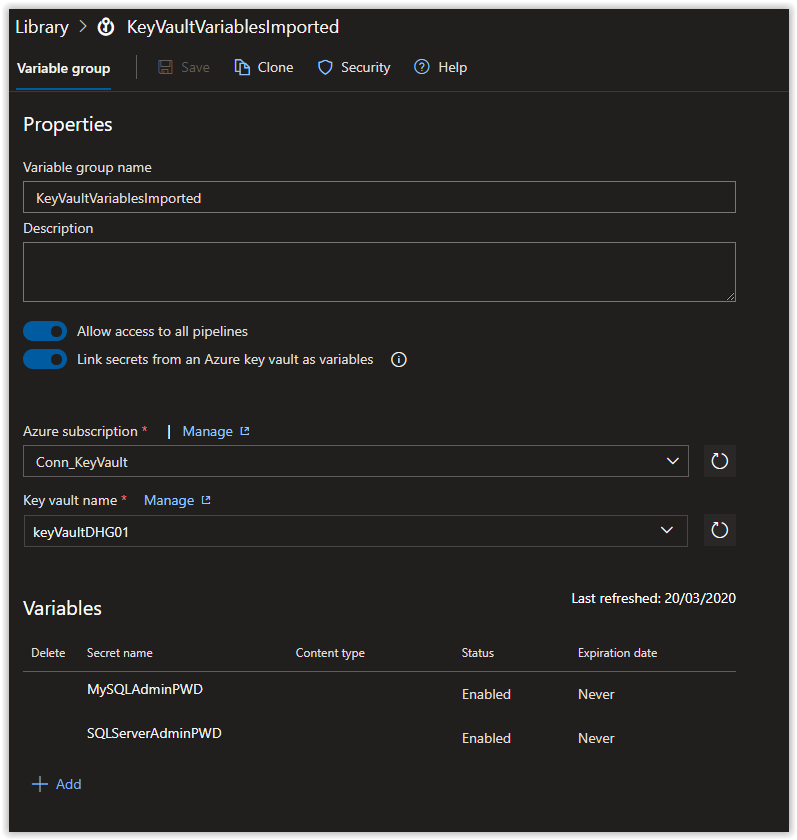


Após a configuração da task como descrito acima é possível provisionar um KeyVault para armazenamento de senhas de outros serviços e aplicações.

A partir desse ponto, é possível registar o Secrets a serem utilizados:



Que devem ser importados em um Variable Group no Azure Devops.



Já estando apto a utilizar as credenciais nas Tasks de outras aplicações ou serviços.