ZABBIX 2022 Conference BRAZIL

02 a 05 de junho de 2022 | São Paulo - SP

ZABBIX 2022 Conference BRAZIL

Secrets via HashiCorp Vault no Zabbix: como funciona?

ZABBIX 2022 Conference BRAZIL

Secrets via HashiCorp Vault no Zabbix: como funciona?

Conceitos

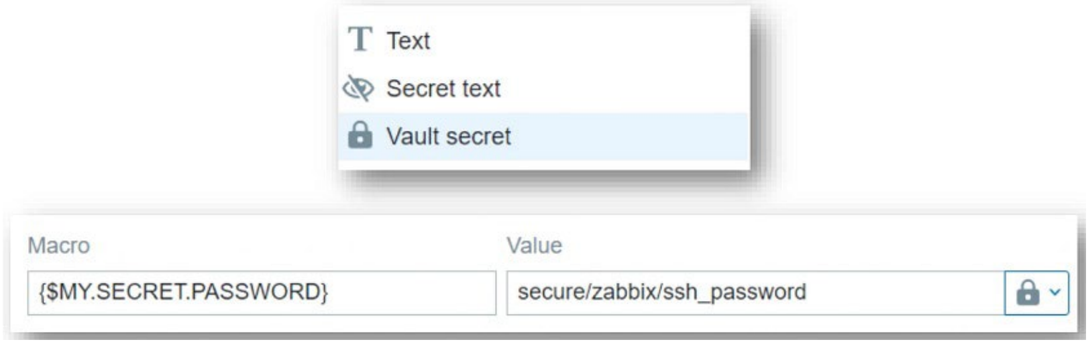
Bio

Patrícia Ladislau

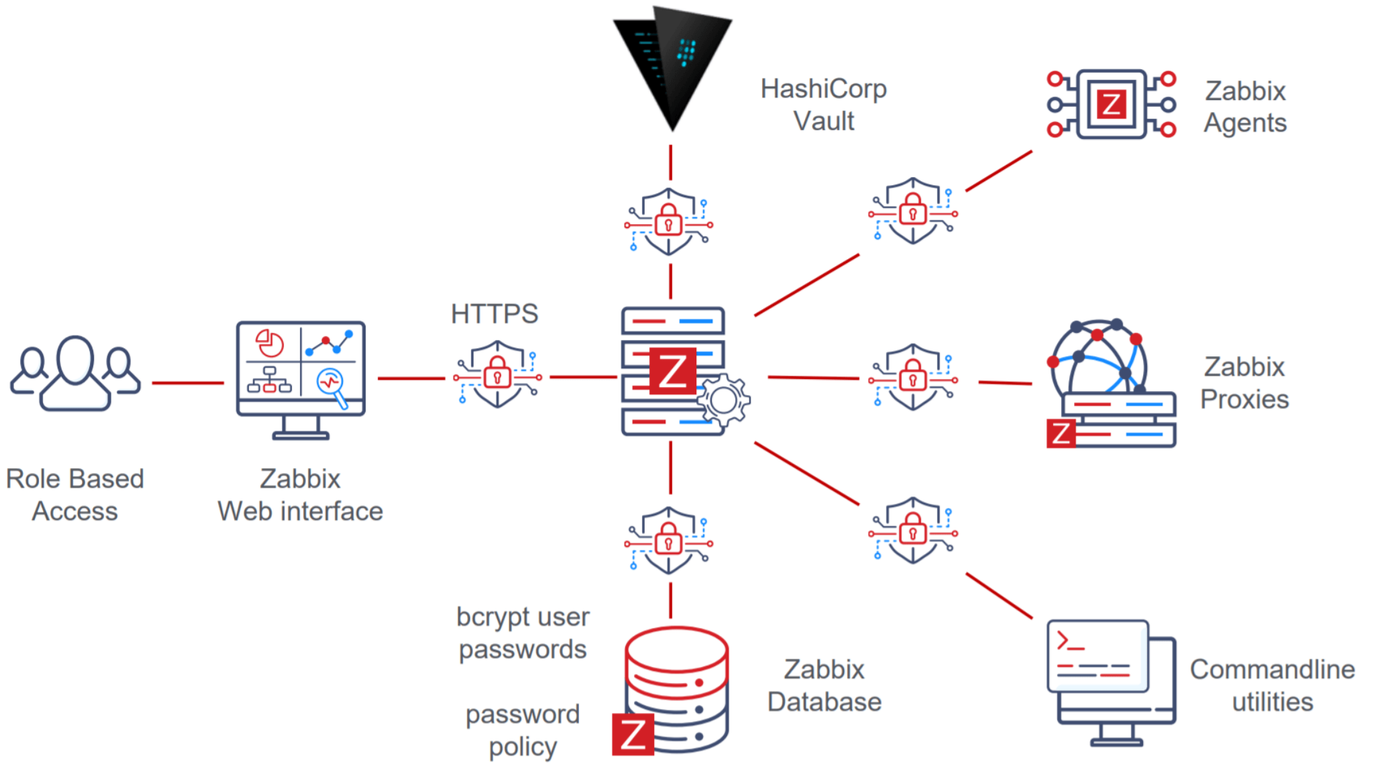
ZCS | ZCP | ZCE
Analista | SRE Unirede
+9 anos de atuação na área de T.I.
Líder Zabbix GirlZ (Comunidade Zabbix-BR)

Informações sensíveis no Zabbix

```
### Option: DBUser
#       Database user.
#
# Mandatory: no
# Default:
# DBUser=

#DBUser=zabbix

### Option: DBPassword
#       Database password.
#       Comment this line if no password is used.
#
# Mandatory: no
# Default:
# DBPassword=
```



Segurança entre componentes



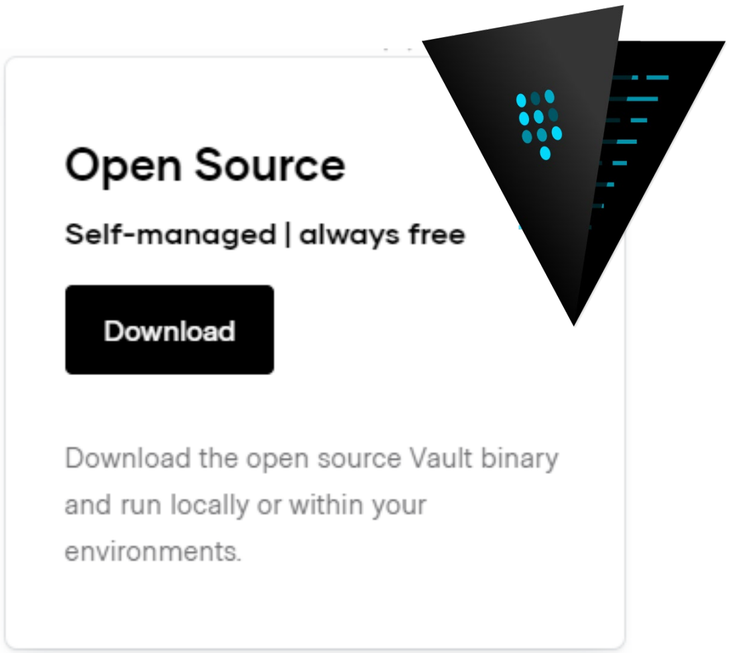
Fonte da imagem: [Securing Zabbix 6.0 LTS by Kārlis Saliņš / Zabbix Summit Online 2021 - Zabbix Blog]

O que é o HashiCorp Vault?

Armazena e gerencia o acesso às informações sensíveis de aplicações, sistemas e infraestrutura

OSS, como o Zabbix

Centralize e proteja os segredos
Controle o acesso
Tenha o registro detalhado para auditoria



Fonte da imagem: https://www.vaultproject.io/

Secrets?

Compreende qualquer coisa que você queira controlar rigorosamente o acesso

Informações sensíveis (chaves de API, senhas, certificados

Exemplos: Macros de Usuário
Credenciais de Banco de Dados

Uma camada de segurança com **TLS** deve ser adicionada

Informação armazenada **fora da base de dados do Zabbix**

Fonte da imagem: https://www.zabbix.com

ZABBIX 2022 Conference BRAZIL

Preparando o HashiCorp Vault

Pré-requisitos

Possuir um certificado para o HC Vault e configurá-lo no .hcl que contém as configurações
Uma dica do tipo “salva-tempo”: *vault –autocomplete-install*
O vault inicia selado

Inicializando o Vault

vault status & vault operator init

```
[root@vault ~]# vault status
Key                Value
---                -----
Seal Type          shamir
Initialized        false
Sealed             true
Total Shares       0
Threshold          0
Unseal Progress    0/0
Unseal Nonce       n/a
Version            1.6.0
Storage Type       file
HA Enabled         false
```

```
[root@vault ~]# vault operator init
Unseal Key 1: +H0g84rtUnx+CGeggswK2SqtAbDjeoCrhqoC+wsp1dri
Unseal Key 2: ibdnpnt8mCH7gqpQ9tApT2Sg1WZNTDqrR+4C+Wv+m9sD
Unseal Key 3: WdND9lo+pzgKO72gnU4WhBhadBwFAHAkbjETd7k+Qe83
Unseal Key 4: yp+2nilPLFOpsbrdCIXSh1TfTUMlE2cMqSUYNl6FgzPq
Unseal Key 5: Huz2CD7yHkMNzpfVQjYoKrKgVKnmg3wCNl10AjV4KYiD

Initial Root Token: s.NjnOZ1aGnBw4NSDxpQD69qEo

Vault initialized with 5 key shares and a key threshold of 3. Please securely
distribute the key shares printed above. When the Vault is re-sealed,
restarted, or stopped, you must supply at least 3 of these keys to unseal it
before it can start servicing requests.

Vault does not store the generated master key. Without at least 3 key to
reconstruct the master key, Vault will remain permanently sealed!

It is possible to generate new unseal keys, provided you have a quorum of
existing unseal keys shares. See "vault operator rekey" for more information.
```

Unseal Vault

vault operator unseal

```
[root@vault ~]# vault operator unseal
Unseal Key (will be hidden):
Key                Value
---                -----
Seal Type          shamir
Initialized        true
Sealed             true
Total Shares       5
Threshold          3
Unseal Progress    1/3
Unseal Nonce       996ecffc-c6ff-5cdf-a6e3-b1cc2e171265
Version            1.6.0
Storage Type       file
HA Enabled         false
```

```
[root@vault ~]# vault operator unseal
Unseal Key (will be hidden):
Key                Value
---                -----
Seal Type          shamir
Initialized        true
Sealed             false
Total Shares       5
Threshold          3
Version            1.6.0
Storage Type       file
Cluster Name       vault-cluster-0434122e
Cluster ID         19d03346-60c3-f128-212f-1c5c1996b301
HA Enabled         false
```

Unseal Vault

vault operator unseal

```
[root@vault ~]# vault operator unseal
Unseal Key (will be hidden):
Key                Value
---                -----
Seal Type          shamir
Initialized        true
Sealed             true
Total Shares       5
Threshold          3
Unseal Progress    1/3
Unseal Nonce       996ecffc-c6ff-5cdf-a6e3-b1cc2e171265
Version            1.6.0
Storage Type       file
HA Enabled         false
```

```
[root@vault ~]# vault operator unseal
Unseal Key (will be hidden):
Key                Value
---                -----
Seal Type          shamir
Initialized        true
Sealed             false
Total Shares       5
Threshold          3
Version            1.6.0
Storage Type       file
Cluster Name       vault-cluster-0434122e
Cluster ID         19d03346-60c3-f128-212f-1c5c1996b301
HA Enabled         false
```

Status e Login

vault status & vault login

```
[root@vault ~]# vault status
Key             Value
---             -----
Seal Type       shamir
Initialized     true
Sealed          false
Total Shares    5
Threshold       3
Version         1.6.0
Storage Type    file
Cluster Name    vault-cluster-0434122e
Cluster ID      19d03346-60c3-f128-212f-1c5c1996b301
HA Enabled      false
```

```
[root@vault ~]# vault login
Token (will be hidden):
Success! You are now authenticated. The token information displayed below
is already stored in the token helper. You do NOT need to run "vault login"
again. Future Vault requests will automatically use this token.

Key                  Value
---                  -----
token                s.NjnOZ1aGnBw4NSDxpQD69qEo
token_accessor       DsOzOibHl608pJ5vggeEZNjY
token_duration       ∞
token_renewable      false
token_policies       ["root"]
identity_policies    []
policies             ["root"]
```

Criando e listando o path zabbix

vault secrets enable –path=zabbix/ kv-v2

```
[root@vault ~]# vault secrets enable -path=zabbix/ kv-v2
Success! Enabled the kv-v2 secrets engine at: zabbix/
```

vault kv list zabbix

```
[root@vault ~]# vault kv list zabbix
Keys
----
database
macros
```

Inserindo as secrets

vault kv put

```
[root@vault ~]# vault kv put zabbix/macros username=zabbixmon password=zabbixmon
Key                 Value
---                 -----
created_time        2022-05-29T15:39:43.297103971Z
deletion_time       n/a
destroyed           false
version             1
```

```
[root@vault ~]# vault kv put zabbix/database username=zabbix password=zabbix
Key                 Value
---                 -----
created_time        2022-05-29T15:24:58.384992035Z
deletion_time       n/a
destroyed           false
version             1
```

Consultando as secrets

vault kv get

```
[root@vault ~]# vault kv get zabbix/macros
====== Metadata ======
Key              Value
---              -----
created_time     2022-05-29T15:39:43.297103971Z
deletion_time    n/a
destroyed        false
version          1

====== Data ======
Key         Value
---         -----
password    zabbixmon
username    zabbixmon
```

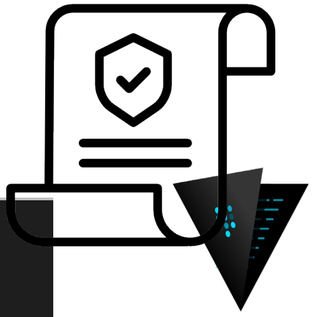
```
[root@vault ~]# vault kv get zabbix/database
====== Metadata ======
Key              Value
---              -----
created_time     2022-05-29T15:24:58.3849920352
deletion_time    n/a
destroyed        false
version          1

====== Data ======
Key         Value
---         -----
password    zabbix
username    zabbix
```

Vault Policies

Controle de acesso às secrets

Formato HCL

Negativa de acesso por padrão



```
path "zabbix/data/database"
{
capabilities = [ "list" , "read" ]
}
path "zabbix/data/macros"
{
capabilities = [ "list" , "read" ]
}
```

Server e Frontend policies - Criando

zabbix-frontend policy

/etc/vault.d/zabbix-frontend-policy.hcl

```
path "zabbix/data/database"
{
capabilities = [ "list" , "read" ]
}
```

zabbix-server policy

/etc/vault.d/zabbix-server-policy.hcl

```
path "zabbix/data/database"
{
capabilities = [ "list" , "read" ]
}
path "zabbix/data/macros"
{
capabilities = [ "list" , "read" ]
}
```

Aplicando as policies

vault policy write zabbix-frontend /etc/vault.d/zabbix-frontend-policy.hcl

```
[root@vault ~]# vault policy write zabbix-frontend /etc/vault.d/zabbix-frontend-policy.hcl
Success! Uploaded policy: zabbix-frontend
```

vault policy write zabbix-server /etc/vault.d/zabbix-server-policy.hcl

```
[root@vault ~]# vault policy write zabbix-server /etc/vault.d/zabbix-server-policy.hcl
Success! Uploaded policy: zabbix-server
```

Obtendo os tokens

vault token create -policy=zabbix-frontend

```
[root@vault ~]# vault token create -policy=zabbix-frontend
Key                  Value
---                  -----
token                s.5ThODYwXkwRVP7gW5q0Rligb
token_accessor       mUPk1QVdBeeZFyhNXYwqKzVW
token_duration       768h
token_renewable      true
token_policies       ["default" "zabbix-frontend"]
identity_policies    []
policies             ["default" "zabbix-frontend"]
```

vault token create -policy=zabbix-server

```
[root@vault ~]# vault token create -policy=zabbix-server
Key                  Value
---                  -----
token                s.7EGJA9lVhmo1Ht012PKnqfuv
token_accessor       dob9U8V5SWiqz7yU4j5fvu7M
token_duration       768h
token_renewable      true
token_policies       ["default" "zabbix-server"]
identity_policies    []
policies             ["default" "zabbix-server"]
```

Importante!

Pode-se utilizar ZBX_DATA_CACHE_TTL para controlar a frequência de atualização/invalidação dos dados em cache.
As secrets ficam no configuration cache do Zabbix, e os valores são obtidos toda vez que o Zabbix atualiza sua configuração.
É possível atualizar os valores das secrets no vault com o comando:

secrets_reload

ZABBIX 2022 Conference BRAZIL

Configurando o Zabbix Server & Frontend

zabbix_server.conf

Desabilite a configuração usual:

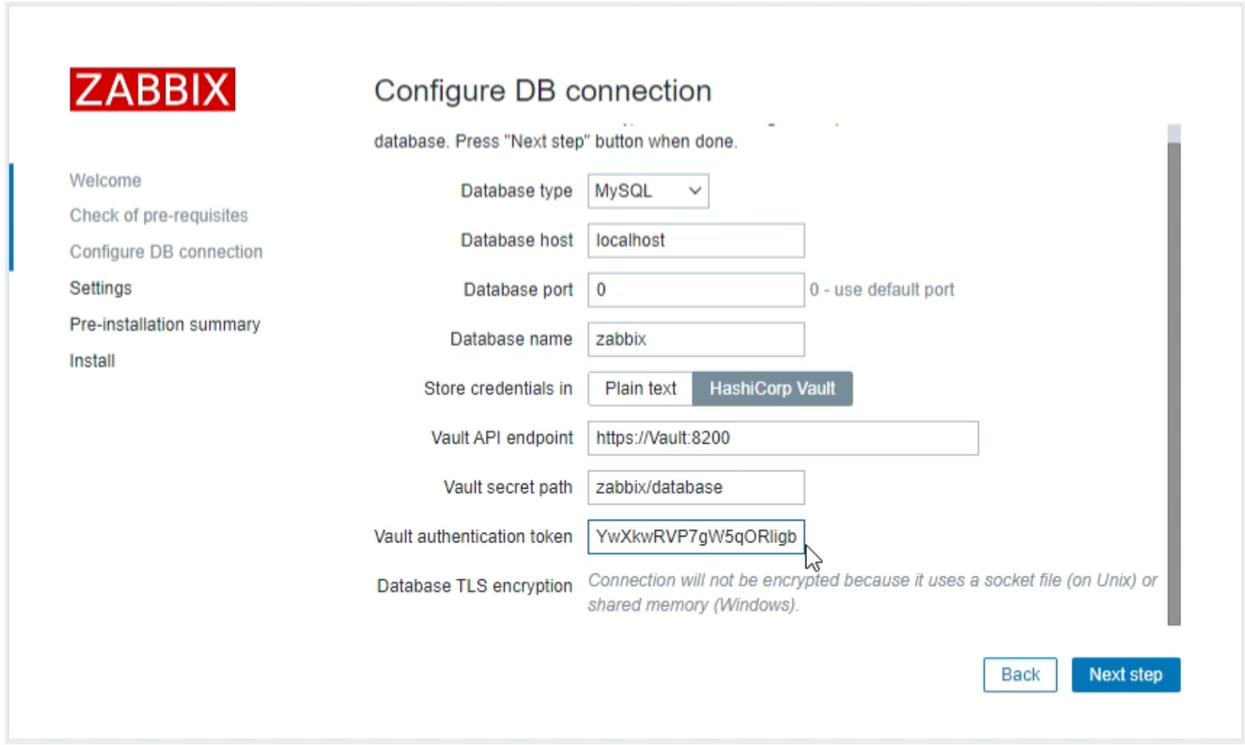
```
### Option: DBUser
#       Database user.
#
# Mandatory: no
# Default:
# DBUser=

#DBUser=zabbix

### Option: DBPassword
#       Database password.
#       Comment this line if no password is used.
#
# Mandatory: no
# Default:
# DBPassword=
```

E habilite as opções do Vault:

```
### Option: VaultToken
#       Vault authentication token that should have been generated exclusively for Zabbix server with read only permission
#       to paths specified in Vault macros and read only permission to path specified in optional VaultDBPath
#       configuration parameter.
#       It is an error if VaultToken and VAULT_TOKEN environment variable are defined at the same time.
#
# Mandatory: no
# Default
VaultToken=s.7EGJA9lVhmo1Ht012PKnqfuv

### Option: VaultURL
#       Vault server HTTP[S] URL. System-wide CA certificates directory will be used if SSLCALocation is not specified.
#
# Mandatory: no
# Default:
VaultURL=https://Vault:8200

### Option: VaultDBPath
#       Vault path from where credentials for database will be retrieved by keys 'password' and 'username'.
#       Example: secret/zabbix/database
#       This option can only be used if DBUser and DBPassword are not specified.
#
# Mandatory: no
# Default:
VaultDBPath=zabbix/database
```

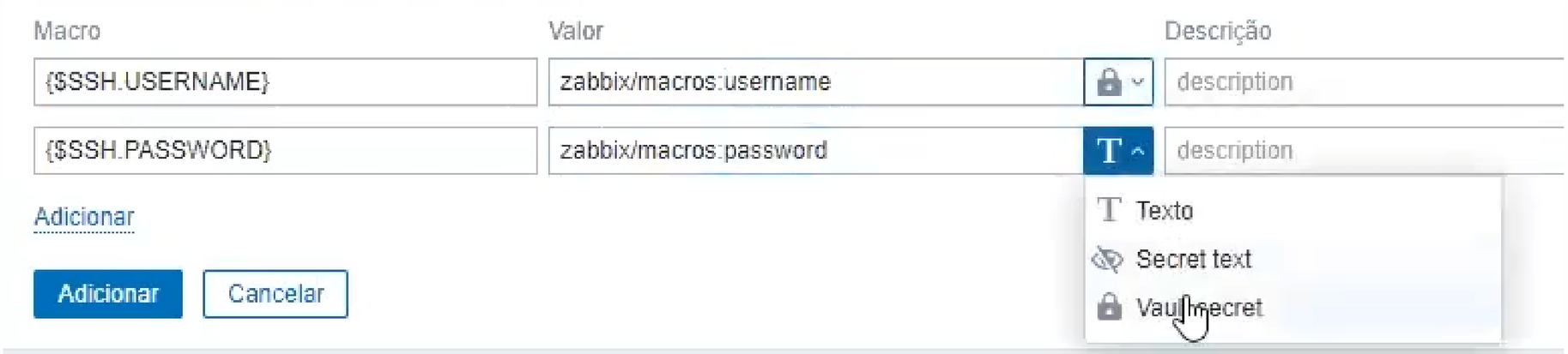
Frontend | Macros

De “Store Credentials in” até “Vault authentication token”, todas as informações que devem ser fornecidas serão provenientes da configuração do Vault e dos arquivos de configuração do Zabbix



Frontend | Macros

Utilize os segredos obtidos no Vault



Outras opções de vaults no roadmap

CyberArk – prevista para o release v6.2

Repositório GitHub

ZABBIX 2022 Conference BRAZIL

OBRIGADA