**Aws, database, user administration and policies**

Enable AWS secret engine :

vault secrets enable -path=aws aws

vault write aws/config/root \

access\_key=ACCESS\_KEY\_ID \

secret\_key=SECRET\_ACCESS\_KEY

vault write aws/roles/ec2role \

credential\_type=iam\_user \

policy\_document=-<<EOF

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Action": "ec2:\*",

"Resource": "\*"

}

]

}

EOF

vault write aws/roles/s3role \

credential\_type=iam\_user \

policy\_document=-<<EOF

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Action": "s3:\*",

"Resource": "\*"

}

]

} EOF

vault read aws/creds/ec2role

output :

Key Value

--- -----

lease\_id aws/creds/ec2role/e75QUrah9HuVm3c5KUBJaKg9

lease\_duration 1800

lease\_renewable true

access\_key AKIA4OJTSXOYK5G5ODEH

secret\_key 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

security\_token <nil>

vault read aws/creds/s3role

output :

Key Value

--- -----

lease\_id aws/creds/s3role/e75QUrah9HuVm3c5KUBJaKg9

lease\_duration 1800

lease\_renewable true

access\_key AKIA4OJTSXOYK5G5ODEH

secret\_key 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

security\_token <nil>

Database secrets engine

vault secrets enable database

**POSTGRES:**

**configure postgres**

vault write database/config/postgresql \

plugin\_name=postgresql-database-plugin \

connection\_url="postgresql://{{username}}:{{password}}@54.255.248.129:54 32/postgres?sslmode=disable" \

allowed\_roles=readonly \

username="root" \

password="rootpassword"

**create role ,**

## vault write database/roles/readonly \

## db\_name=postgresql \

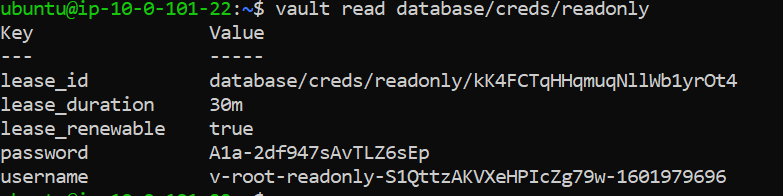
## creation\_statements="CREATE ROLE \"{{name}}\" WITH LOGIN PASSWORD '{{password}}' VALID UNTIL '{{expiration}}'; \

## GRANT SELECT ON ALL TABLES IN SCHEMA public TO \"{{name}}\";" \

## default\_ttl="1h" \

## max\_ttl="2h"

vault read database/creds/readonly



**SQL :**

**Create role :**

vault write database/roles/my-sql-role \

db\_name=my-mysql-database \

creation\_statements="CREATE USER '{{name}}'@'%' IDENTIFIED BY '{{password}}';GRANT SELECT ON \*.\* TO '{{name}}'@'%';" \

default\_ttl="1h" \

max\_ttl="1h"

**configure sql :**

vault write database/config/my-mysql-database \

plugin\_name=mysql-database-plugin \

connection\_url="{{username}}:{{password}}@tcp(54.255.248.129:3306)/" \

allowed\_roles="my-sql-role" \

username="root" \

password="Welcome123"

vault read database/creds/my-sql-role

**output:**

Key Value

--- -----

lease\_id database/creds/my-sql-role/aGYp7UIjCqrc3rdRBWbUvwdP

lease\_duration 1h

lease\_renewable true

password A1a-kaJNJ0FiTd2BcahZ

username v-root-my-role-XARNzWzG7o6bfev2t

**ACCESS RESTRICTIONS:**

**Create Policies :**

**Apple policy to have ec2 restricted :**

# Allow tokens to look up their own properties

path "auth/token/lookup-self" {

capabilities = ["read"]

}

# Allow tokens to renew themselves

path "auth/token/renew-self" {

capabilities = ["update"]

}

# Allow tokens to revoke themselves

path "auth/token/revoke-self" {

capabilities = ["update"]

}

# Allow a token to look up its own capabilities on a path

path "sys/capabilities-self" {

capabilities = ["update"]

}

# Allow a token to look up its own entity by id or name

path "identity/entity/\*" {

capabilities = ["list"]

}

path "identity/entity/id/5de92794-4421-603a-b5db-16f1b791cda7" {

capabilities = ["read","list","update","create","delete"]

}

# Allow a token to look up its own entity by id or name

path "sys/auth" {

capabilities = ["list","read"]

}

# Allow a token to renew a lease via lease\_id in the request body; old path for

# old clients, new path for newer

path "sys/renew" {

capabilities = ["update"]

}

path "sys/leases/renew" {

capabilities = ["update"]

}

# Allow looking up lease properties. This requires knowing the lease ID ahead

# of time and does not divulge any sensitive information.

path "sys/leases/lookup" {

capabilities = ["update"]

}

path "aws/\*" {

capabilities = ["list"]

}

path "aws/roles/ec2role"{

capabilities = ["list","delete","create","update","read"]

}

path "aws/creds/ec2role" {

capabilities = ["read","list","update","create","delete"]

}

**Orange policy to have restricted s3role:**

path "aws/\*" {

capabilities = ["list"]

}

path "aws/roles/s3role" {

capabilities = ["list","read","update","delete","create"]

}

path "aws/creds/s3role" {

capabilities = ["list","read","update","create"]

}

# Allow a token to look up its own capabilities on a path

path "sys/capabilities-self" {

capabilities = ["update"]

}

# Allow a token to look up its own entity by id or name

path "identity/entity/\*" {

capabilities = ["list"]

}

path "identity/entity/id/9bcd5c15-c304-398d-bd72-d95fbc34bac9" {

capabilities = ["read","list","update","create","delete"]

}

path "sys/policies/acl"

{

capabilities = ["list"]

}

**Sql policy to have restricted access to database role –** my-sql-role :

# Allow a token to look up its own capabilities on a path

path "sys/capabilities-self" {

capabilities = ["update"]

}

path "database/\*" {

capabilities = ["list"]

}

path "database/roles/my-sql-role" {

capabilities = ["create", "read", "update", "delete", "list"]

}

path "database/creds/my-sql-role" {

capabilities = ["create", "read", "update", "delete", "list"]

}

Curl commands for aws :

**Configure root :**

Vi payload.json

{

"access\_key": "AKIA4OJTSXOYKQC7CPMB",

"secret\_key": "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",

"region": "us-east-1"

}

curl \

--header "X-Vault-Token: s.9NuiX2lkeyVB8YhfC7ct3GDS" \

--request POST \

--data @payload.json \

http:// 1c05353a-default-vault-c083-1304294467.ap-southeast-1.elb.amazonaws.com/v1/aws/config/root

curl --header "X-Vault-Token: s.9NuiX2lkeyVB8YhfC7ct3GDS" http://1c05353a-default-vault-c083-1304294467.ap-southeast-1.elb.amazonaws.com/v1/aws/config/root

{"request\_id":"d26f0a93-eea0-b718-2cfc-58cd41cec4b8","lease\_id":"","renewable":false,"lease\_duration":0,"data":{"access\_key":"AKIA4OJTSXOYKQC7CPMB","iam\_endpoint":"","max\_retries":-1,"region":"us-east-1","sts\_endpoint":""},"wrap\_info":null,"warnings":null,"auth":null}

**Configure lease :**

Leasepayload.json:

{

"lease": "30m",

"lease\_max": "1h"

}

curl \

--header "X-Vault-Token: s.9NuiX2lkeyVB8YhfC7ct3GDS" \

--request POST \

--data @leasepayload.json \

http://1c05353a-default-vault-c083-1304294467.ap-southeast-1.elb.amazonaws.com/v1/aws/config/lease

curl --header "X-Vault-Token: s.9NuiX2lkeyVB8YhfC7ct3GDS" http://1c05353a-default-vault-c083-1304294467.ap-southeast-1.elb.amazonaws.com/v1/aws/

config/lease

{"request\_id":"999ff9ed-dddf-39a3-b8b4-cd24ef9777da","lease\_id":"","renewable":false,"lease\_duration":0,"data":{"lease":"30m0s","lease\_max":"1h0m0s"},"wrap\_info":null,"warnings":null,"auth":null}

**Create/update role :**

Rolepayload.json:

{

"credential\_type": “assumed role”

"role\_arns”: “arn:aws:iam::052578464564:policy/eksRead”

}

curl \

--header "X-Vault-Token: s.9NuiX2lkeyVB8YhfC7ct3GDS" \

--request POST \

--data @rolepayload.json \

http://1c05353a-default-vault-c083-1304294467.ap-southeast-1.elb.amazonaws.com/v1/aws/roles/eksaccess

curl \

--header "X-Vault-Token: s.9NuiX2lkeyVB8YhfC7ct3GDS" \

http://1c05353a-default-vault-c083-1304294467.ap-southeast-1.elb.amazonaws.com/v1/aws/roles/eksaccess

{"request\_id":"28839470-1559-671d-af25-54d309c3e1ba","lease\_id":"","renewable":false,"lease\_duration":0,"data":{"credential\_type":"iam\_user","default\_sts\_ttl":0,"iam\_groups":null,"max\_sts\_ttl":0,"permissions\_boundary\_arn":"","policy\_arns":["arn:aws:iam::052578464564:policy/eksRead"],"policy\_document":"","role\_arns":null,"user\_path":""},"wrap\_info":null,"warnings":null,"auth":null}

curl \

--header "X-Vault-Token: s.9NuiX2lkeyVB8YhfC7ct3GDS" \

--request LIST \

http://1c05353a-default-vault-c083-1304294467.ap-southeast-1.elb.amazonaws.com/v1/aws/roles

{"request\_id":"10d578d1-609e-ae1b-2c05-bac1f2cefe77","lease\_id":"","renewable":false,"lease\_duration":0,"data":{"keys":["ec2role","eksaccess","mys3role","s3role"]},"wrap\_info":null,"warnings":null,"auth":null}

curl \

--header "X-Vault-Token: s.9NuiX2lkeyVB8YhfC7ct3GDS" \

--request DELETE \

http://1c05353a-default-vault-c083-1304294467.ap-southeast-1.elb.amazonaws.com/v1/aws/roles

**generate credentials:**

curl --header "X-Vault-Token: s.9NuiX2lkeyVB8YhfC7ct3GDS" http://1c05353a-default-vault-c083-1304294467.ap-southeast-1.elb.amazonaws.com/v1/aws/creds/ec2role

{"request\_id":"449a7bd6-916e-fbef-6bb9-12ef57e704cb","lease\_id":"aws/creds/ec2role/k6kCQkLonEaXnGgN3gdF6gSh","renewable":true,"lease\_duration":1800,"data":{"access\_key":"AKIA4OJTSXOYBWJHU24P","secret\_key":"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","security\_token":null},"wrap\_info":null,"warnings":null,"auth":null}

**LINK** : <https://www.vaultproject.io/api-docs/secret/aws>

**DATABASE LINKS FOR CURL :**

<https://www.vaultproject.io/api-docs/secret/databases/postgresql>

<https://www.vaultproject.io/api-docs/secret/databases/mysql-maria>