Objectives:

- Setting up a HashiCorp server on a workstation in development mode

Installing a vault:

Add the HashiCorp GPG key:

1. From the home directory (~/), execute:

```
$curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key add -
```

Expected output:

```
/home/ubuntu $curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo apt-key ad
OK
/home/ubuntu $
```

2. Add the official HashiCorp Linux repository, execute:

$ sudo apt-add-repository "deb [arch=amd64]

https://apt.releases.hashicorp.com $(lsb_release -cs) main"

NOTE: this command is all on one line

Expected output:

```
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease [101 kB]
Get:4 https://apt.releases.hashicorp.com focal InRelease [4419 B]
Get:5 http://security.ubuntu.com/ubuntu focal-security InRelease [109 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [947 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [760 k
Get:8 https://apt.releases.hashicorp.com focal/main amd64 Packages [22.2 kB]
Get:9 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [623 kB]
Fetched 2680 kB in 1s (1791 kB/s)
Reading package lists... Done
```

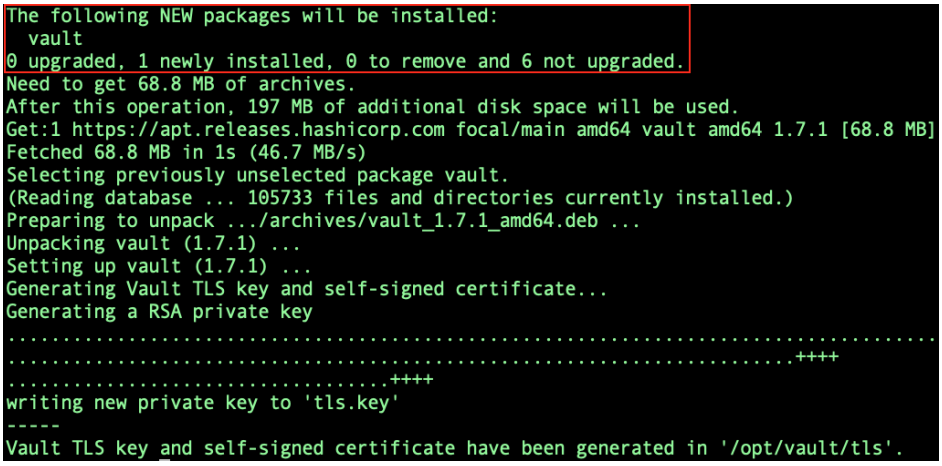
Updating and installing a vault:

3. Execute:

   $ sudo apt-get update && sudo apt-get upgrade -y

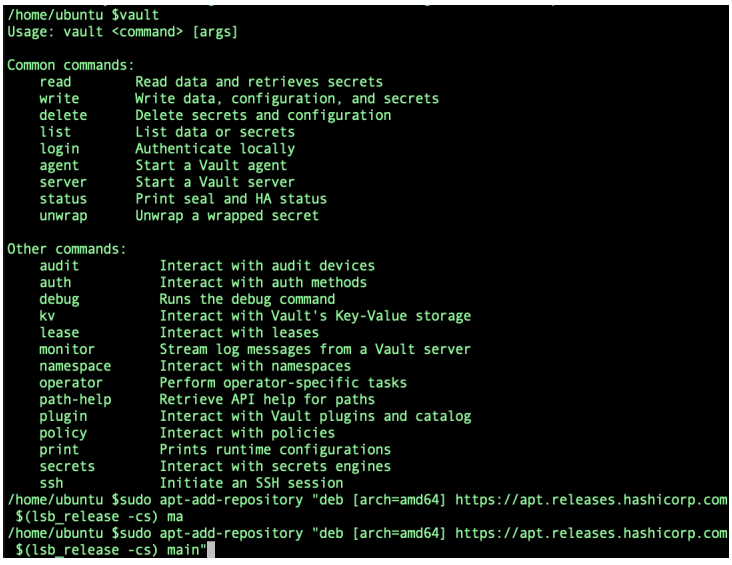
   $ sudo apt-get install -y vault

   Expected output:



Verify the installation of the vault package on a VM

4. Execute: `$vault`



Starting the server in dev mode:

5. Execute:

```
$vault server -dev -dev-listen-address="0.0.0.0:8200"
```

Expected output:

```
==> Vault server started! Log data will stream in below:
```

```
WARNING! dev mode is enabled! In this mode, Vault runs entirely in-memory
and starts unsealed with a single unseal key. The root token is already
authenticated to the CLI, so you can immediately begin using Vault.

You may need to set the following environment variable:

    $ export VAULT_ADDR='http://0.0.0.0:8200'

The unseal key and root token are displayed below in case you want to
seal/unseal the Vault or re-authenticate.

Unseal Key: X94cHEblIfMI5TbBzAuPVBfn7SyJv469M9x708n7DGk=
Root Token: s.Oof32bYNKaZC8SUa1UmKlWof

Development mode should NOT be used in production installations!
```

Note: **do not end** the vault server or **close** the terminal window to ensure the server continues to run for the next steps.

Note the Root Token in the output

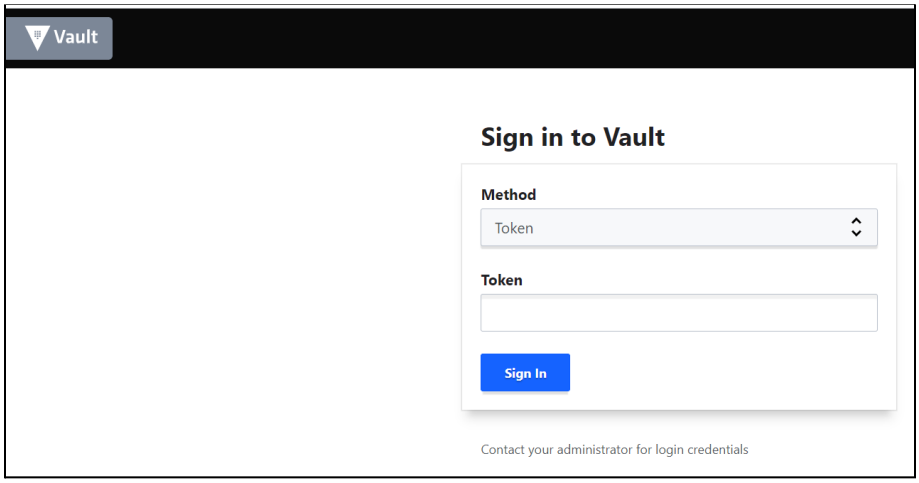
Quick testing:

6. Open a web browser to: IP address of the VM (xx.xx.xxx.xx) on port 8200.

   Example: http://xx.xx.xxx.xx:8200

   NOTE: use http **not https** for the IP address, since we have not enable port 443

7. A sign in window to the vault should be displaying :



8. Use the **Root Token** to Sign In, which is still visible in the terminal window running the vault server. With a successful sign in, the following page will display:

NOTE: Don't stop the vault server, open a new terminal tab (Ctrl + t), and sign into the VM in the new tab in order to follow the next steps via command line.

9. Change **~/.profile** to add three entries at **the end of the file** and update path:

```
# if running bash
if [ -n "$BASH_VERSION" ]; then
    # include .bashrc if it exists
    if [ -f "$HOME/.bashrc" ]; then
        . "$HOME/.bashrc"
    fi
fi

# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/bin" ] ; then
    PATH="$HOME/bin:$PATH"
fi

# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/.local/bin" ] ; then
    PATH="$HOME/.local/bin:$PATH"
fi

export VAULT_ADDR='http://0.0.0.0:8200'
export VAULT_API_ADDR='https://0.0.0.0:8200'
export VAULT_URL='https://0.0.0.0:8200'
```

10. **Execute:** `$source .profile`

    Validate whether the .profile file was updated correctly and is returning the expected value for the VAULT_URL variable:

11. **Execute:** `$echo $VAULT_URL`

```
~ $echo $VAULT_URL
https://0.0.0.0:8200
```

Notify your instructor that you are done with the lab

END OF LAB