Mapping out Active Directory on Ubuntu with BloodHound CE and PlumHound, Cybersecurity Pen Testing Tools

Step 1: Install Ubuntu Desktop on a VM
I am running on version 24.04 of Ubuntu Desktop

Step 2: Installing Docker in Command Line

 # Add Docker's official GPG key:

- sudo apt-get update && sudo apt-get upgrade
- sudo apt-get install ca-certificates curl
- sudo install -m 0755 -d /etc/apt/keyrings
- sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
- sudo chmod a+r /etc/apt/keyrings/docker.asc

 # Add the repository to Apt sources:

- echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
- sudo apt-get update

 # Install Latest Version of Docker

- sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

 # Start Docker

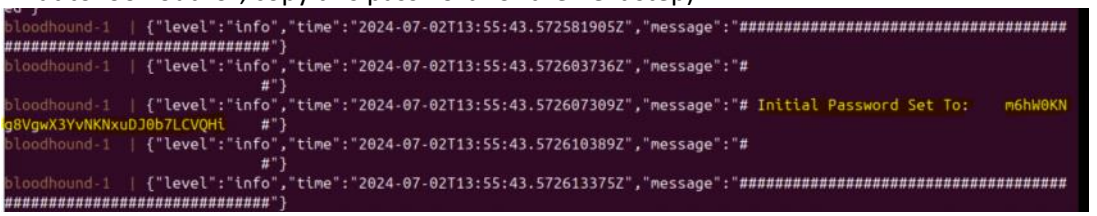
- sudo service docker start

 # Test if docker is working

- sudo docker run hello-world

Step 3: Install BloodHound CE Using Docker Compose

- sudo su
- curl -L https://ghst.ly/getbhce | docker compose -f - up
  (after running the docker compose look out for the initial password, there is a picture below for what to look out for, copy this password for the next step)



This will automatically set up and start the neo4j console, postgres and bloodhound in individual containers

 # Exit out of super user

- exit
- exit

 # Check if Containers are running

- sudo docker ps

After the containers are up and running, navigate to http://localhost:8080 (bloodhound)
Login with username: admin
Login with the initial password found in the output of the docker compose, you will then be prompted to change your password, make sure to remember it.

If you want to access the neo4j console navigate to http://localhost:7474 and login using Username: neo4j Password: bloodhoundcommunityedition (you will need this password when you run plumhound)

 # Set up Docker Containers to start up on reboot, first make sure the containers are running and find the container names (if for some reason the containers are not running, use the docker start container_name **command)**

- sudo docker ps -a (-a shows all containers, even ones that are not running)

```
lhadmin@LHPLHT01:~$ sudo docker ps -a
CONTAINER ID   IMAGE                          COMMAND                  CREATED          STATUS                    PORTS
                                                                NAMES
6ef9ed302258   specterops/bloodhound:latest   "/bloodhound -config…"   29 minutes ago   Up 4 minutes              127.0.0
.1:8080->8080/tcp                                               lhadmin-bloodhound-1
11e0d35a5bfe   postgres:13.2                  "docker-entrypoint.s…"   29 minutes ago   Up 5 minutes (healthy)    5432/tc
p                                                               lhadmin-app-db-1
4187dea697fc   neo4j:4.4                      "tini -g -- /startup…"   29 minutes ago   Up 5 minutes (healthy)    127.0.0
.1:7474->7474/tcp, 7473/tcp, 127.0.0.1:7687->7687/tcp           lhadmin-graph-db-1
lhadmin@LHPLHT01:~$
```

 # After you get container names for bloodhound, neo4j, and postgres enter the following commands

- sudo docker update --restart unless-stopped lhadmin-bloodhound-1
- sudo docker update --restart unless-stopped lhadmin-app-db-1
- sudo docker update --restart unless-stopped lhadmin-graph-db-1
- sudo docker update --restart=on-failure lhadmin-bloodhound-1
- sudo docker update --restart=on-failure lhadmin-app-db-1
- sudo docker update --restart=on-failure lhadmin-graph-db-1

Step 4: Install BloodHound.Py Ingestion Tool (Basically SharpHound for Linux, this program scans the AD which you then upload to bloodhound to map out your AD)

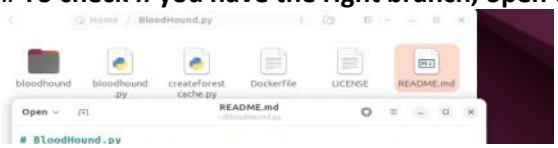
 # If you do not have pip installed, install it

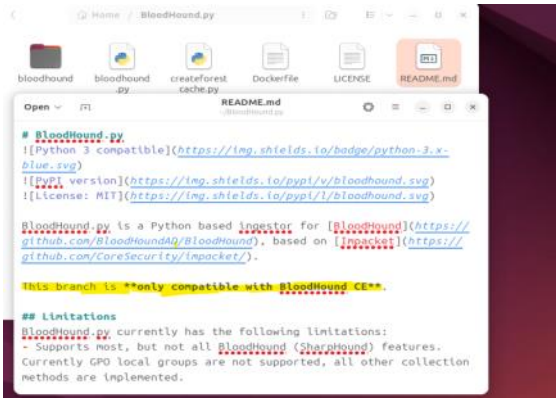
- sudo apt install python3-pip

 # Git clone from github, make sure to download the CE branch which only works with bloodhound CE

- git clone --branch bloodhound-ce https://github.com/dirkjanm/BloodHound.py.git

 # To check if you have the right branch, open the readme in the BloodHound.py folder





 # Navigate to the BloodHound.py folder

- cd BloodHound.py

 # Install Requirements

- pip install . (if that does not work run **pip install . --break-system-packages**)

Step 5: Install PlumHound

 # Navigate back to ~ directory

- cd .. OR cd ~

 # git clone of PlumHound

- git clone https://github.com/PlumHound/PlumHound.git

 # Navigate to PlumHound directory

- cd PlumHound

 # Install Requirements

- sudo pip3 install -r requirements.txt (if that does not work run **sudo pip3 install -r requirements.txt --break-system-package**s)

Step 6: Run BloodHound.py over the AD to collect info to input into BloodHound CE

 # Make a new folder for collecting info

- mkdir test

#Run BloodHound.py in new folder (You can use any user account to run this scan, just know it will likely set off alerts)

- cd test
- bloodhound-python -c all -d libertyhardware.com -u "Username" -p "Password" --zip

 # -c is the collection method, -d is the domain, -u is username, -p is password, and --zip puts everything into a zip folder.

 # after scanning navigate to the test folder and unzip the folder that was just created, if you look inside there should be .json files

Step 7: Upload Information into BloodHound CE

 # Navigate to http://localhost:8080 (login with admin and the password you changed earlier)
 # Click the settings icon and head to ADMINISTRATION
 # In the FILE INGEST area, upload the .json files from the unzipped folder you just scanned using BloodHound.py
 # After the files are uploaded you can go to the EXPLORE tab and navigate through your mapped out AD!

Step 8: Running PlumHound against BloodHound

 # Navigate to the PlumHound folder
- cd PlumHound

 # Run PlumHound with neo4j console password
- python3 PlumHound.py -x tasks/default.tasks -p bloodhoundcommunityedition

 # This scan will be thrown into the reports folder which is located in the PlumHound folder
 # Navigate to the reports folder and open index.html
 # You can either do it manually or through the command line
 # To do it thorough the command line navigate to PlumHoud/reports and type this command
- firefox index.html (if you don't have firefox you can simply install it with apt install firefox)

 # From here you should have all the information on your Active Directory provided by BloodHound and PlumHound through the BloodHound console (http://localhost:8080) and the index.html file