Threat Hunting Home-Lab Using Velociraptor

By GOKUL KRISHNA K

Objective

Install and configure Velociraptor on a Linux-based server to create a centralized endpoint monitoring and forensic analysis platform capable of collecting system artifacts, performing live response operations, and enabling proactive threat-hunting activities. This includes setting up the Velociraptor server and client components, managing endpoint check-ins, configuring artifact collections, and analyzing acquired data to identify suspicious behaviors, security anomalies, and indicators of compromise across the environment

Tools used

.Velociraptor Server & Client
.Ubuntu server
.Ubuntu Desktop
.Virtual Box

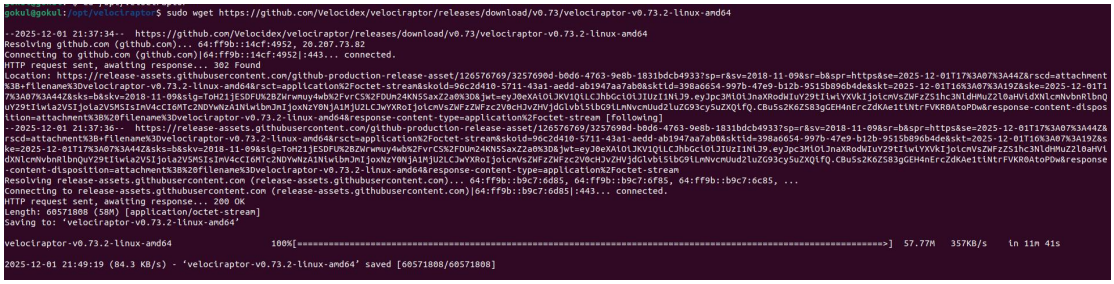
Create a Directory for Velociraptor

```
gokul@gokul:~$ sudo mkdir -p /opt/velociraptor
gokul@gokul:~$ cd /opt/velociraptor
```

Command: sudo mkdir -p /opt/velociraptor
cd /opt/velociraptor

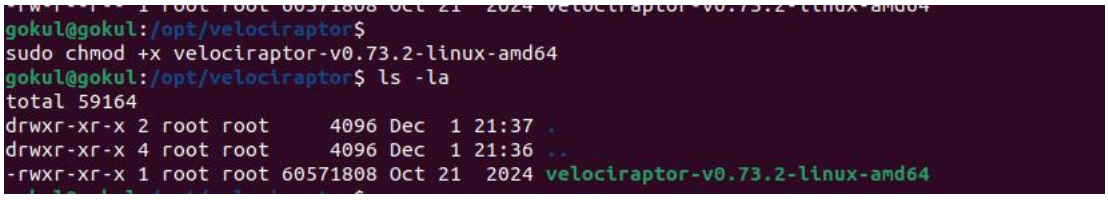
Download Velociraptor

I downloaded the velociraptor binary from github



Command: sudo wget
https://github.com/Velocidex/velociraptor/releases/download/v0.73/velociraptor-v0.73.2-linux-amd64

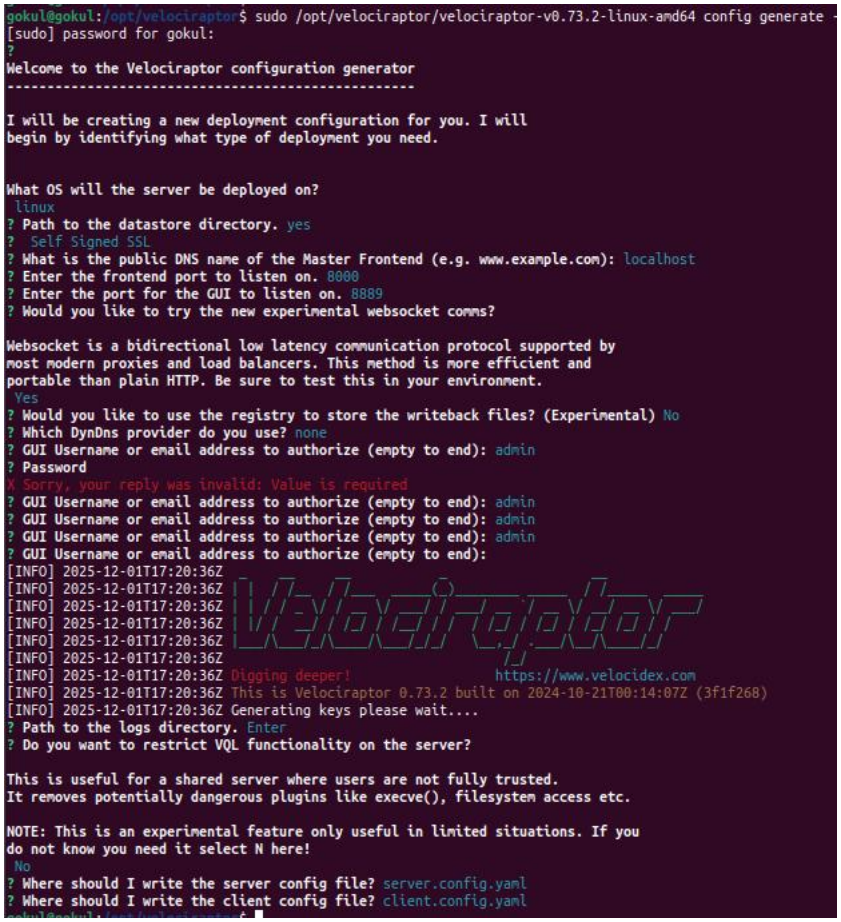
I rename and make the binary executable:



This is for give the executable permission to the file for execute

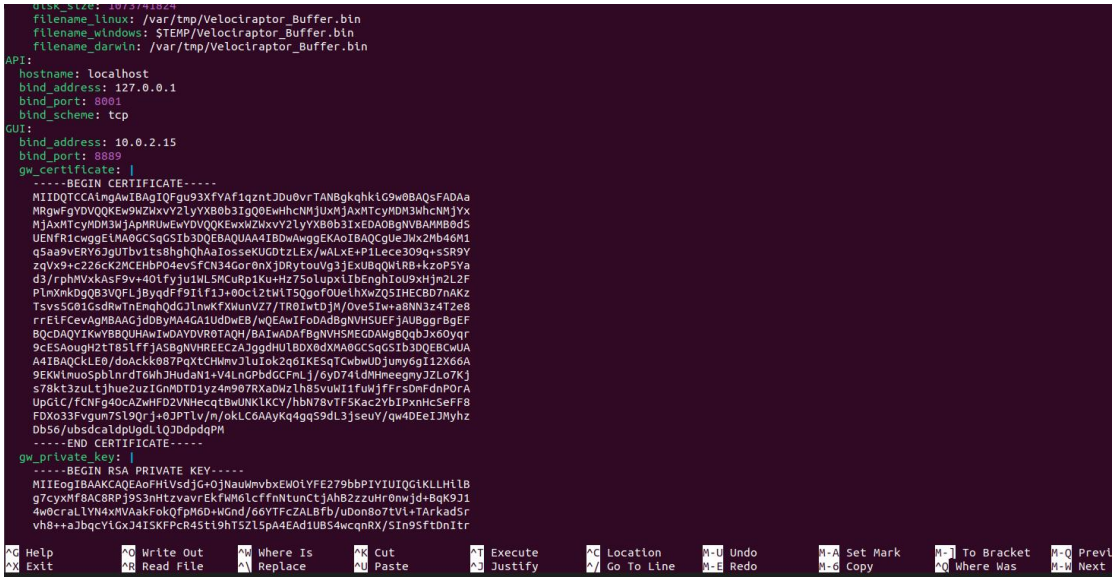
Command: sudo chmod +x velociraptor-v0.73.2-linux-amd64

I configure Velociraptor Using Interactive Mode :



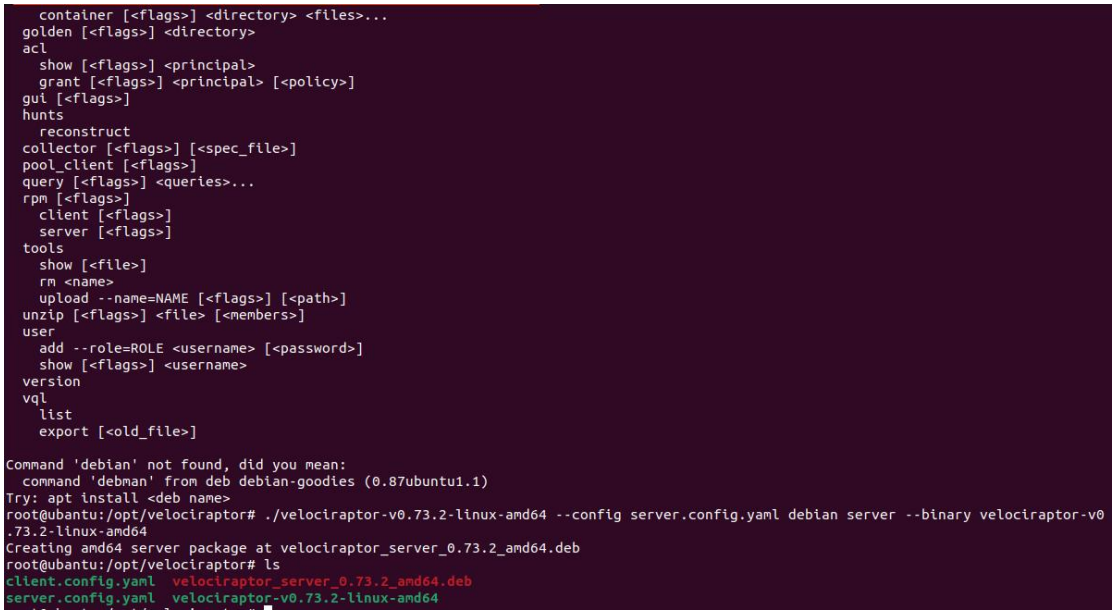
I use Velociraptor's interactive configuration mode because it automatically guides us through all the required setup steps, ensures correct server–client settings, and prevents configuration mistakes.

Update Server Config file



I open Server Config file with the help of nano command and changed the bind address to my velo server ip

I Created debian file for Velociraptor server and Install



I create and install a Velociraptor .deb file to properly install the server as a system-managed service on Linux, ensuring easier deployment, updates, and long-term stability.

Command: ./velociraptor-v0.73.2-linux-amd64 --config server.config.yaml debian server --binary velociraptor-v0.73.2-linux-amd64

I Extract and Install

```
root@ubantu:/opt/velociraptor# dpkg -i velociraptor_server_0.73.2_amd64.deb
Selecting previously unselected package velociraptor-server.
(Reading database ... 202631 files and directories currently installed.)
Preparing to unpack velociraptor_server_0.73.2_amd64.deb ...
Unpacking velociraptor-server (0.73.2) ...
Setting up velociraptor-server (0.73.2) ...
Adding group `velociraptor' (GID 137) ...
Done.
Adding system user `velociraptor' (UID 129) ...
Adding new user `velociraptor' (UID 129) with group `velociraptor' ...
Not creating home directory `/etc/velociraptor/'.
Created symlink /etc/systemd/system/multi-user.target.wants/velociraptor_server.service → /etc/systemd/system/velociraptor_server.se
rvice.
root@ubantu:/opt/velociraptor#
```

Extracting and installing Velociraptor unpacks the program files, places them into correct system directories, applies proper permissions, and enables Velociraptor to run reliably as a managed Linux service

Command: dpkg -i velociraptor-v0.73.2-linux-amd64.deb

Checking the Validity

```
root@ubantu:/opt/velociraptor# systemctl status velociraptor_server.service
● velociraptor_server.service - Velociraptor server
     Loaded: loaded (/etc/systemd/system/velociraptor_server.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2025-12-03 19:59:06 IST; 2min 41s ago
   Main PID: 3838 (velociraptor.bi)
      Tasks: 14 (limit: 2262)
     Memory: 48.3M
        CPU: 13.163s
     CGroup: /system.slice/velociraptor_server.service
             ├─3838 /usr/local/bin/velociraptor.bin --config /etc/velociraptor/server.config.yaml frontend
             └─3844 /usr/local/bin/velociraptor.bin --config /etc/velociraptor/server.config.yaml frontend

Dec 03 19:59:06 ubantu systemd[1]: Started Velociraptor server.
root@ubantu:/opt/velociraptor#
```

I run this command to check whether the Velociraptor server service is running correctly, to view its current status, and to identify any errors or issues during startup.

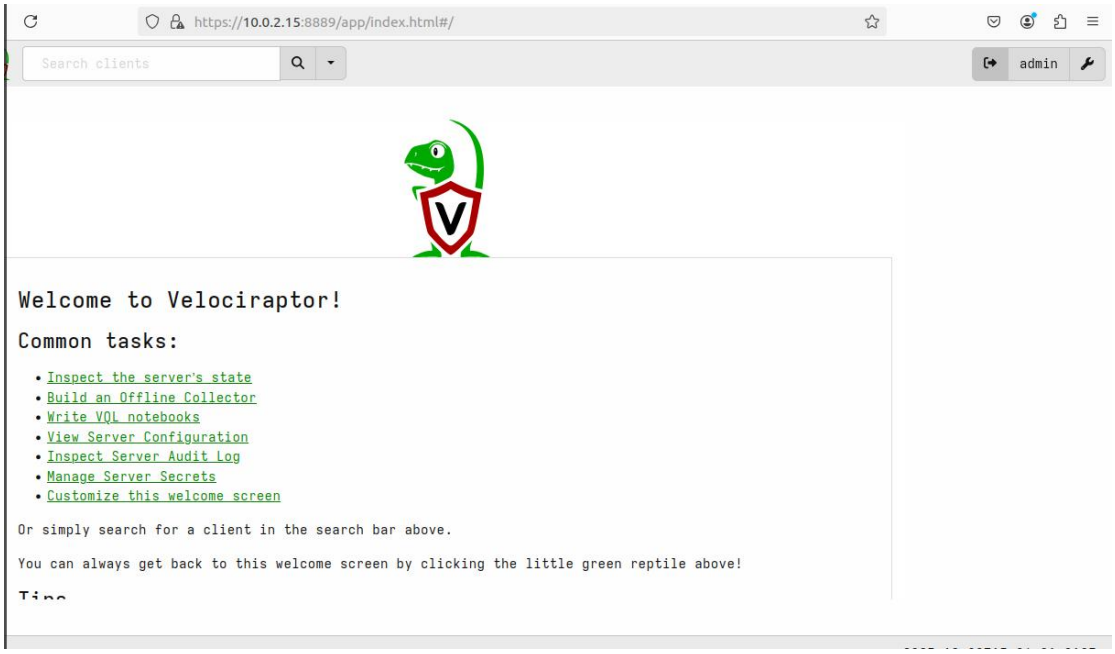
Command: systemctl status velociraptor_server.service

Open firewall ports

```
root@ubantu:/opt/velociraptor# ufw allow 8000/tcp
Rules updated
Rules updated (v6)
root@ubantu:/opt/velociraptor# ufw allow 8889/tcp
Rules updated
Rules updated (v6)
root@ubantu:/opt/velociraptor#
```

open firewall ports to allow communication between the Velociraptor server and its endpoint clients, ensuring they can connect, send telemetry, and receive instructions.

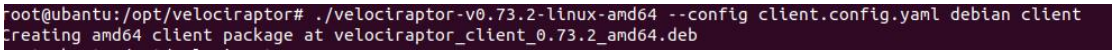
Command: ufw allow port number



So I had set the server of velociraptor.next I am going to configure client

Velociraptor Ubuntu Client Set up

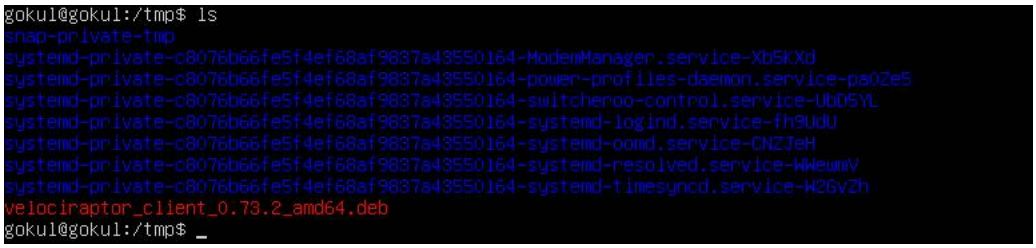
In this stage i created client debian file on Velociraptor server. The same binary file will be used for both client and server.



Command: ./velociraptor-v0.73.2-linux-amd64 --config client.config.yaml debian client

Next I transfered this file to Client system using SCP command.

Command: scp velociraptor_client_0.73.2_amd64.deb Client IP Address:/tmp/



I got the debian file inside the tmp directory of my ubuntu desktop client machine

Extract the debian file

```
root@gokul:/tmp# dpkg -i velociraptor_client_0.73.2_amd64.deb
Selecting previously unselected package velociraptor-client.
(Reading database ... 163803 files and directories currently installed.)
Preparing to unpack velociraptor_client_0.73.2_amd64.deb ...
Unpacking velociraptor-client (0.73.2) ...
Setting up velociraptor-client (0.73.2) ...
Created symlink /etc/systemd/system/multi-user.target.wants/velociraptor_client.service → /etc/syste
md/system/velociraptor_client.service.
root@gokul:/tmp#
```

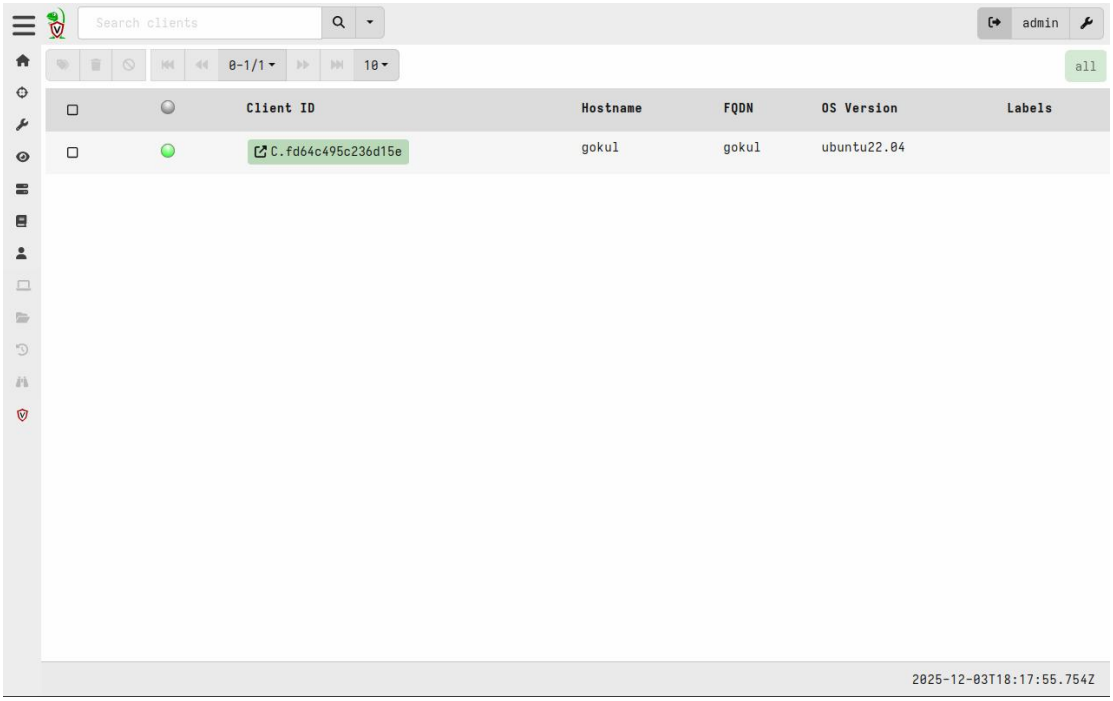
Command: dpkg -i /tmp/velociraptor_client_0.73.2_amd64.deb

Check the status

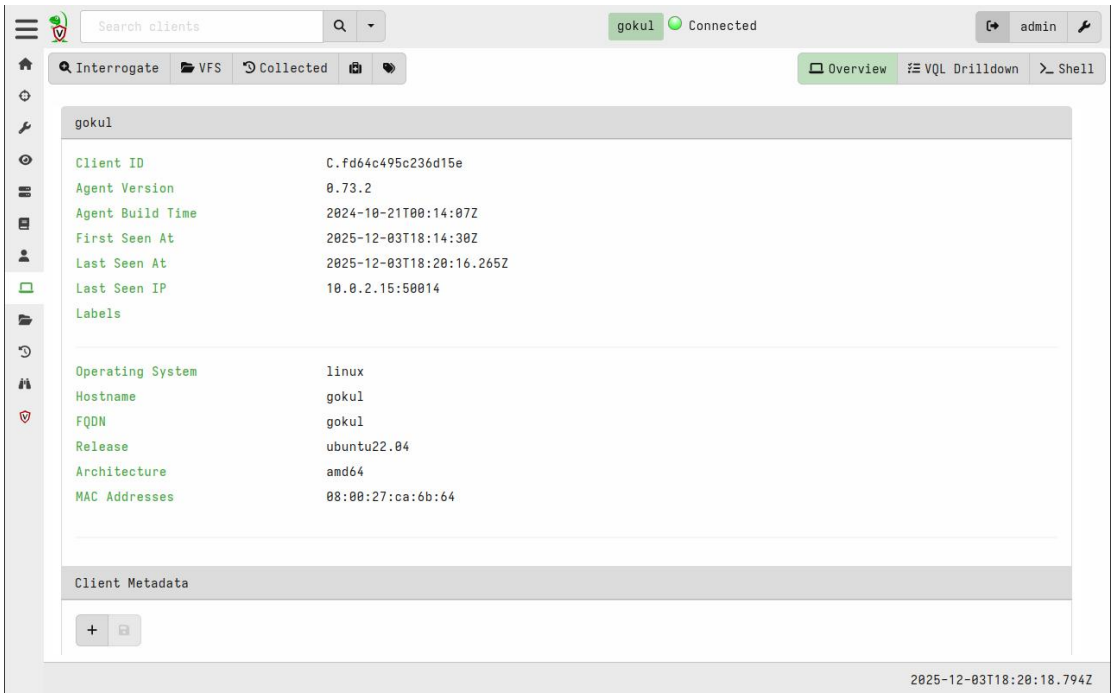
```
root@gokul:/tmp# systemctl status velociraptor_client.service
● velociraptor_client.service - Velociraptor client
     Loaded: loaded (/etc/systemd/system/velociraptor_client.service; enabled; vendor preset: enabl
     Active: active (running) since Wed 2025-12-03 16:28:53 UTC; 5min ago
   Main PID: 23286 (velociraptor_cl)
      Tasks: 8 (limit: 11291)
     Memory: 27.3M
        CPU: 1min 27.391s
     CGroup: /system.slice/velociraptor_client.service
             └─23286 /usr/local/bin/velociraptor_client --config /etc/velociraptor/client.config.ya

Dec 03 16:28:53 gokul systemd[1]: Started Velociraptor client.
lines 1-11/11 (END)
```

My client velo also successfully running

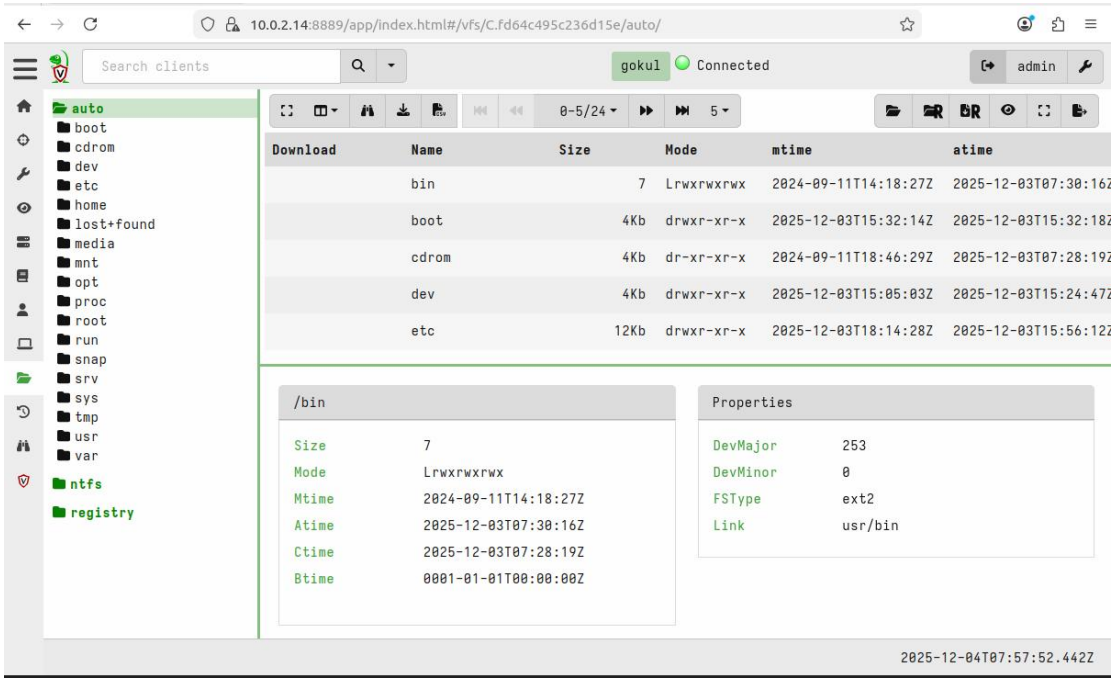
Command: systemctl status velociraptor_client.service



My velo server successfully added my client ubuntu device



This is the overview of my client ubuntu machine



I can see my full directories and the files inside those directories of my client ubuntu system

Conclusion

By setting up Velociraptor Server,i create a dedicated endpoint forensics and threat-hunting platform. This environment allows SOC analysts and incident responders to collect forensic artifacts, run VQL queries, and investigate suspicious activity across multiple endpoints in real time.