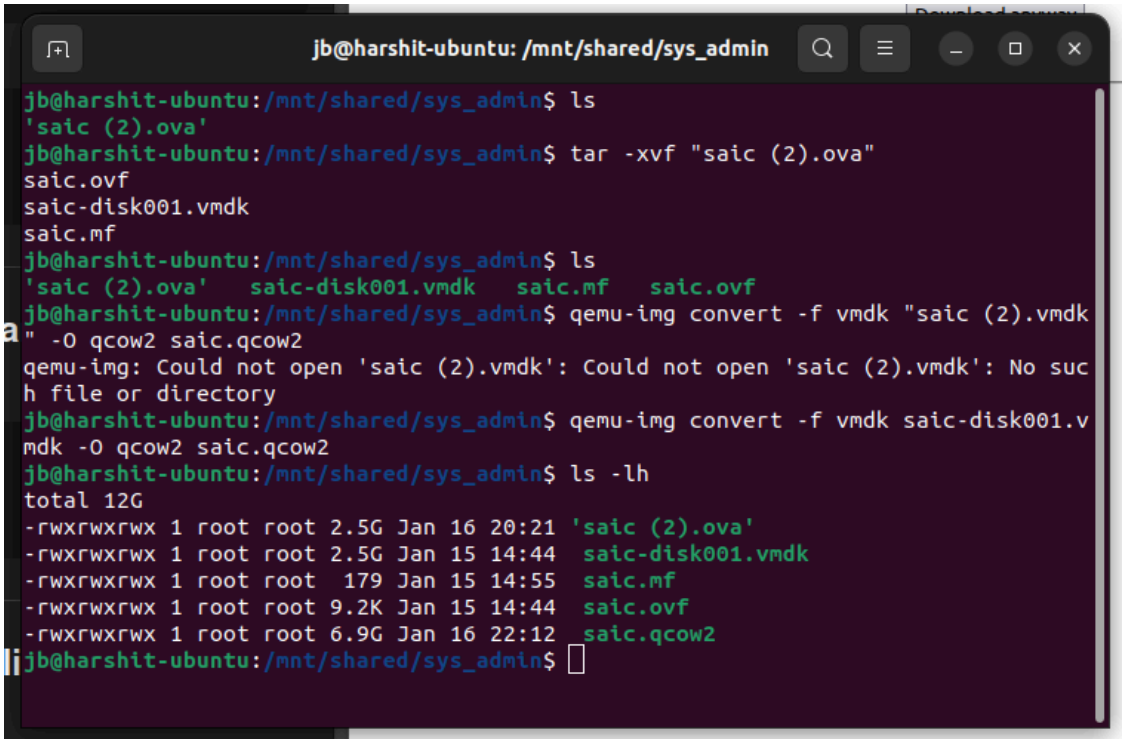
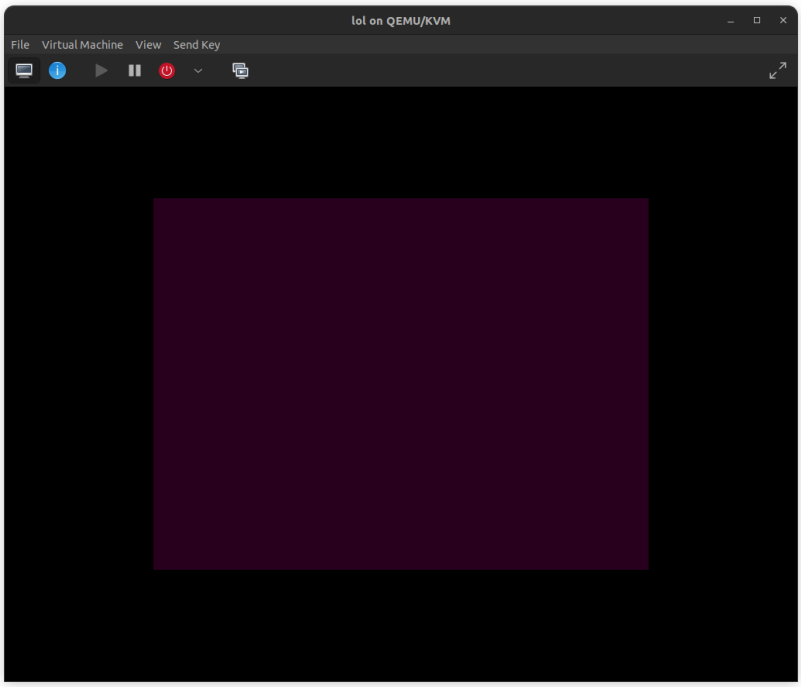
Starting on the last day I was not able to complete this one

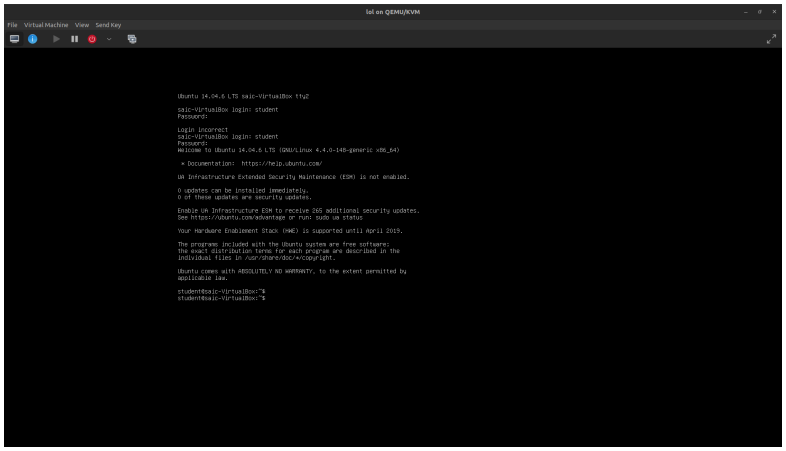
I first started with downloading the virt-manager for KVM/QEMU virtualization as I had read it is better on linux than oracle virtualbox (which i have used before)

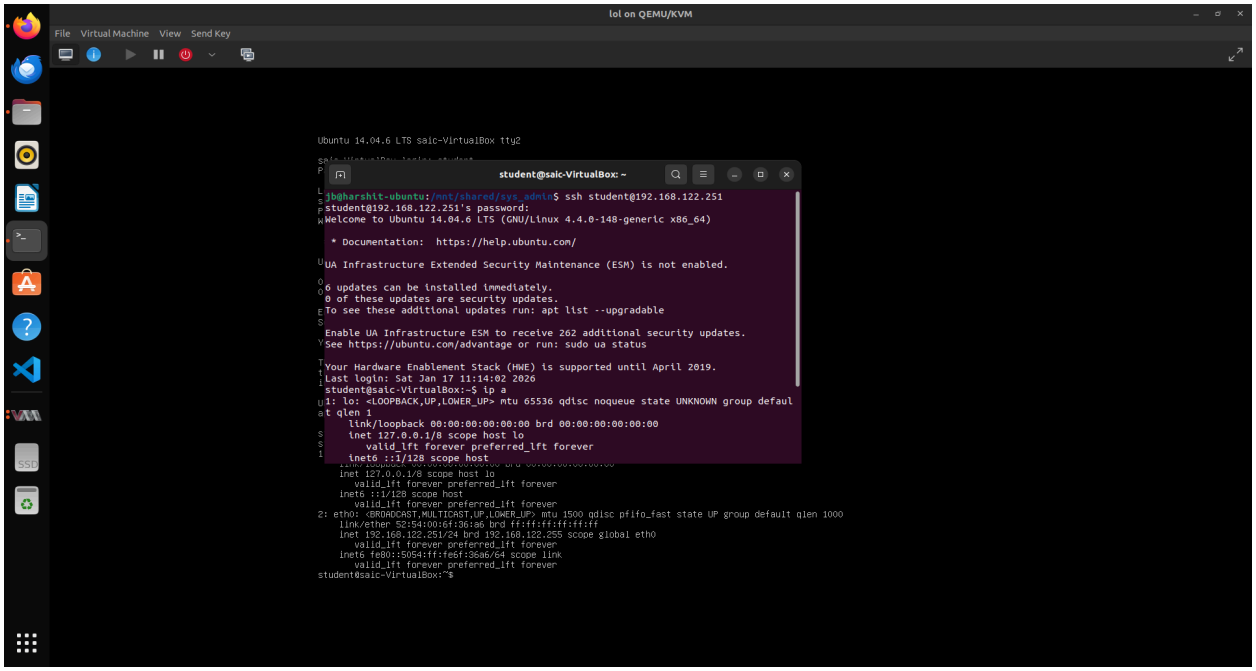


Initially i was getting this blank screen then fount out that we need to click on the ctrl+alt+f2 button inside the vm after going full screen



Since i had to use chatgpt and that requires copy paste i basically accessed it via SSH through my ubuntu terminal





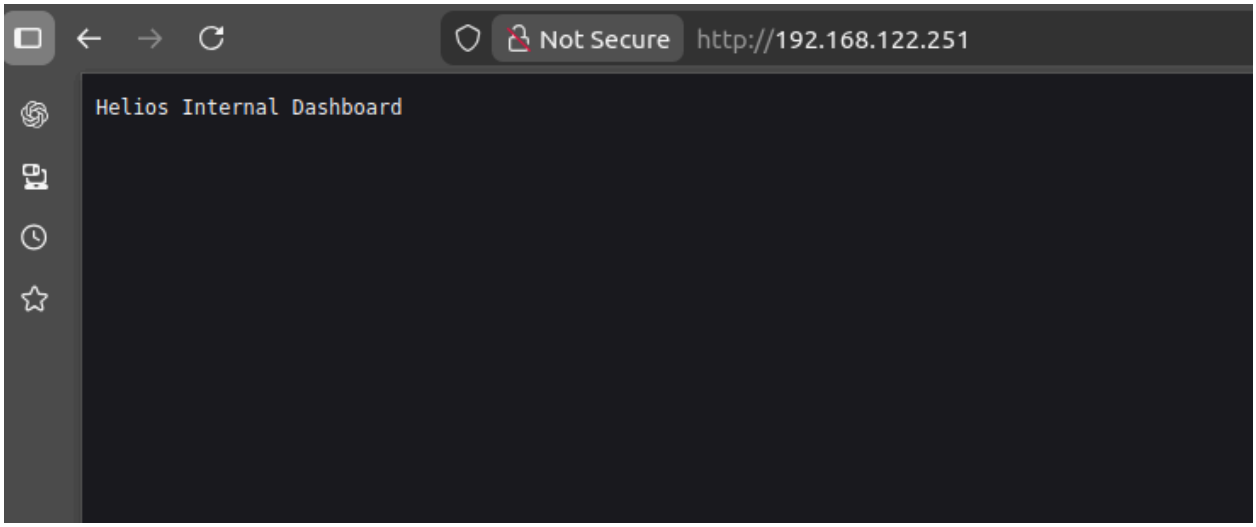
Checked the ip4

```
student@saic-VirtualBox:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP gr
oup default qlen 1000
    link/ether 52:54:00:6f:36:a6 brd ff:ff:ff:ff:ff:ff
    inet 192.168.122.251/24 brd 192.168.122.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::5054:ff:fe6f:36a6/64 scope link
       valid_lft forever preferred_lft forever
student@saic-VirtualBox:~$
```

Then i identified open services

```
student@saic-VirtualBox:~$ ss -tulpn
Netid  State      Recv-Q Send-Q     Local Address:Port       Peer Address:Port
tcp    UNCONN     0      0                *:5353                  *:*
tcp    UNCONN     0      0                *:30003                 *:*
tcp    UNCONN     0      0                *:631                   *:*
tcp    UNCONN     0      0                *:37736                 *:*
tcp    UNCONN     0      0         127.0.1.1:53                   *:*
tcp    UNCONN     0      0                *:68                    *:*
tcp    UNCONN     0      0               :::55528                :::*
tcp    UNCONN     0      0               :::5353                 :::*
tcp    UNCONN     0      0               :::45731                :::*
tcp    LISTEN     0      5         127.0.0.1:9000                 *:*
tcp    LISTEN     0      5         127.0.0.1:5000                 *:*
tcp    LISTEN     0      128              *:80                    *:*
tcp    LISTEN     0      5         127.0.1.1:53                   *:*
tcp    LISTEN     0      128              *:22                    *:*
tcp    LISTEN     0      128       127.0.0.1:631                  *:*
tcp    LISTEN     0      128             :::22                   :::*
```

Going to the ip it showed



Then i checked the backend code

```
student@saic-VirtualBox:~$ cat /opt/helios/web/server.py
from BaseHTTPServer import BaseHTTPRequestHandler, HTTPServer
import os

INTERNAL_TOKEN = None
SERVICE_TOKEN = None

# load env manually (systemd doesn't export cleanly in 14.04)
with open("/etc/helios/web.env") as f:
    for line in f:
        if line.startswith("INTERNAL_TOKEN"):
            INTERNAL_TOKEN = line.strip().split("=")[1]

with open("/etc/helios/worker.env") as f:
    for line in f:
        if line.startswith("SERVICE_TOKEN"):
            SERVICE_TOKEN = line.strip().split("=")[1]

class Handler(BaseHTTPRequestHandler):
    def do_GET(self):
        if self.path == "/":
            self.send_response(200)
            self.end_headers()
            self.wfile.write("Helios Internal Dashboard")
            return

        if self.path == "/internal/export":
            # Trust failure #1: localhost assumption
            if self.client_address[0] != "127.0.0.1":
                self.send_response(403)
                self.end_headers()
                return

            # Trust failure #2: internal token
            token = self.headers.get("X-Internal-Token")
            if token != INTERNAL_TOKEN:
                self.send_response(403)
                self.end_headers()
                return

            self.send_response(200)
            self.end_headers()
            self.wfile.write(
                "SERVICE_TOKEN={}\nUsed by helios-worker\n".format(SERVICE_TOKEN)
            )
            return

        self.send_response(404)
        self.end_headers()

HTTPServer(("127.0.0.1", 5000), Handler).serve_forever()

student@saic-VirtualBox:~$
```

It has an internal endpoint
GET /internal/export

Then i checked the internal token

```
student@saic-VirtualBox:~$ cat /etc/helios/web.env
INTERNAL_TOKEN=9877981220e470cf3cfc49e73d98ba5a
student@saic-VirtualBox:~$
```

Called internal endpoint from within the VM: and got service token

```
student@saic-VirtualBox:~$ curl -H "X-Internal-Token: 9877981220e470cf3cfc49e73d98ba5a" \
> http://127.0.0.1:5000/internal/export
SERVICE_TOKEN=313b96f83a85fb32b31ec70d86d5e40c
Used by helios-worker
student@saic-VirtualBox:~$
```

Also i found that the worker accepts POST only

```
student@saic-VirtualBox:~$ ps aux | grep control.py
ops         917  0.0  0.5  41632 10552 ?        Ss   13:23   0:02 /usr/bin/python /opt/helios/work
student   11297  0.0  0.1  15960  2272 pts/0    S+   18:06   0:00 grep --color=auto control.py
student@saic-VirtualBox:~$
```

After that i tried determining the authentication field but it returned 403 everytime 😕