11/14/2025

Building an Isolated Blue Team & Digital Forensics Laboratory

A Virtualization Project for Cyber Defence & Monitoring

Lohith.G

ABOUT THE AUTHOR

Lohith G is a cybersecurity practitioner and Bachelor of Computer Applications (BCA) graduate currently enrolled in the Certified Penetration Tester (CPT) program.

With a strong background as a Senior Laptop Service Technician, he possesses deep technical expertise in hardware repair, BIOS configuration, and system recovery. This project integrates his practical knowledge of IT infrastructure with advanced offensive and defensive cybersecurity concepts, demonstrating a capability to design, build, and secure complex virtualized environments.

My Personal DFIR & Blue Team Lab

1. Introduction:

Project Objective: To design and build a lightweight, cost-effective, and isolated virtual lab environment using VirtualBox. The lab will serve as a platform for conducting digital forensics investigations, malware analysis, and blue team security monitoring exercises in a safe and controlled manner.

Scope : The lab will initially consist of three core virtual machines: an analyst workstation, a victim/target machine, and a simple network monitoring server. It will be configured on an isolated network to prevent accidental contamination.

Guiding Principles:

- **Ethical & Legal:** All activities within this lab are for educational purposes only, using authorized software and simulated scenarios.
- **Isolation:** The lab network must be segregated from the host machine's primary LAN.
- **Repeatability:** The lab must be easily reset to a "clean" state using VM snapshots.
- **Documentation:** Every step, from network design to tool installation, will be documented here.

2. Lab Architecture:

2.1 Host Hardware : ThinkPad X13 (Intel i5-10310U, 16 GB RAM)

2.2 Host Software : Windows, VirtualBox (v6.x or v7.x).

2.3 Virtual Network Design :

We will use a VirtualBox **"Host-Only Network"** (e.g., vboxnet0).

Purpose : This creates a private network that includes your host machine and all the lab VMs. It's isolated from your home router and the internet, which is critical for safety.

IP SCHEMA :

- **Network:** 192.168.56.0/24
- **Host Adapter (vboxnet0):** 192.168.56.1 (Your "gateway" for managing VMs)
- **Analyst VM:** 192.168.56.101
- **Victim VM:** 192.168.56.102
- **Server VM:** 192.168.56.103

Lab Architecture

VM Name	Operating System	Purpose	vCPUs	RAM	Network Adapter
ANALSYT-VM	SIFT Workstation (Ubuntu-based)	Forensic analysis, Wireshark, Volatility, etc.	2	4-6 GB	Host-Only (vboxnet0)
VICTIM-VM	Windows 10 (Evaluation)	The "target" machine for analysis (e.g., malware, log collection)	2	2-4 GB	Host-Only (vboxnet0)
LOGGER-VM	Ubuntu Server 22.04	Central log collection (Syslog-NG, basic ELK, or T-Pot)	2	2-4 GB	Host-Only (vboxnet0)

3. Lab Setup: Phase 1:

Configure VirtualBox Networks :

A. In VirtualBox: File > Host Network Manager
B. Click Create
C. Select the new adapter (e.g., vboxnet0)
D. Set the **IPv4 Address** to 192.168.56.1 and **Netmask** to 255.255.255.0.
E. Go to the DHCP Server tab and **uncheck** Enable Server. We will set static IPs for full control.

Download ISOs:

SIFT Workstation(Lab) : SIFT Workstation | SANS Institute

Windows 10/11 Enterprise(Victim) : Windows 11 Enterprise | Microsoft Evaluation Center

Ubuntu Server(Logging) : Get Ubuntu Server | Download

4. Lab Setup: Phase 2 (Build and Isolate)

This section details the step-by-step build, configuration, and isolation of each lab VM.

a. ANALYST-VM(SIFT WORKSTATION)

Objective: To import the official SIFT OVA (appliance), establish a baseline snapshot, and then isolate it onto the lab network with a static IP.

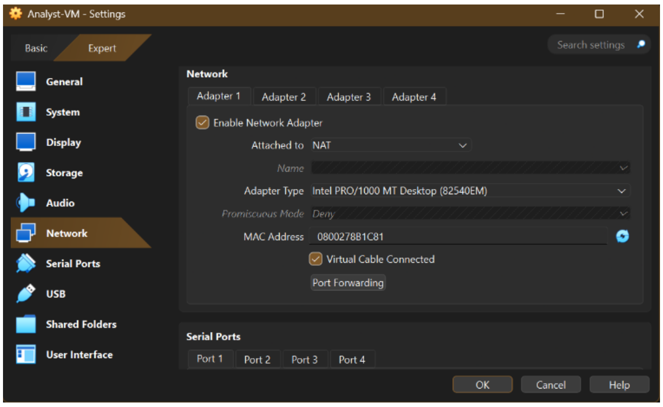
File Used: sift-22.04-20240201.ova

Base VM Settings (from import):

- RAM 8GB
- vCPU 4
- HDD 80GB

I. **Import and Baseline Snapshot:**

a) Imported the SIFT Workstation .ova file via File > Import Appliance
b) Added an Optical Drive : Before the first boot, I went to the settings > Storage, selected the IDE Controller, and clicked "Adds optical drive" to allow Guest Additions to be mounted.
c) Network : Left the adapter on default NAT to allow internet access for updates.



Analyst-VM - Settings

Network

Adapter 1

☑ Enable Network Adapter

Attached to: NAT

Name:

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Deny

MAC Address: 0800278B1C81

☑ Virtual Cable Connected

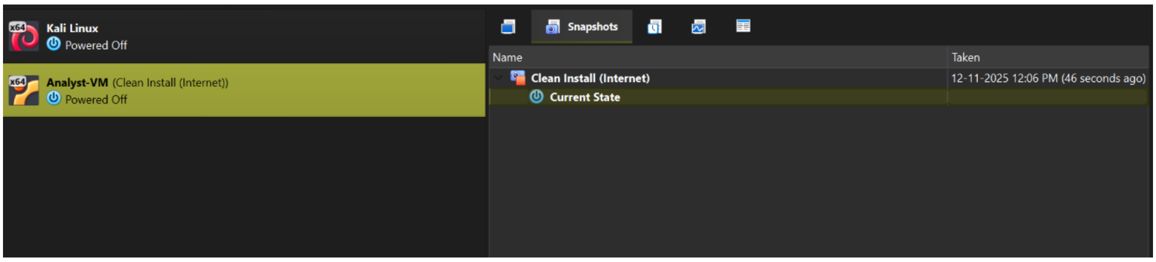
d) Start the VM(Login : Sans forensics and password : forensics)

e) Installed Guest additions : Clicked Devices > Insert Guest Additions CD Image and ran the installer from the terminal to enable clipboard and screen resizing.

f) Ran System Updates : sudo apt update , sudo apt upgrade -y

g) Shutdown the VM

h) Took Snapshot (Clean Install(Internet)), This serves as our master "template" for the SIFT workstation.

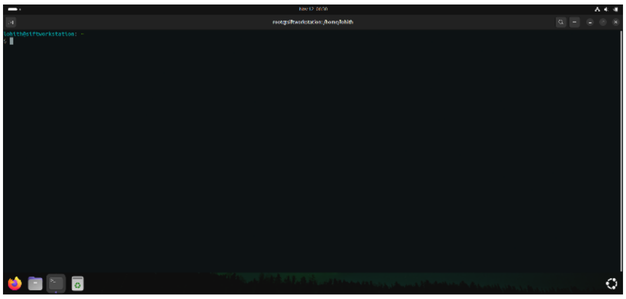


Kali Linux
Powered Off

Analyst-VM (Clean Install (Internet))
Powered Off

Name	Taken
Clean Install (Internet)	12-11-2025 12:06 PM (46 seconds ago)
Current State	

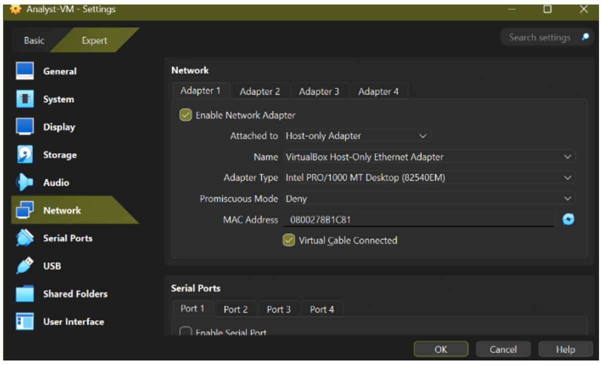
i) Successfully Installed SIFT :



```
lohith@siftworkstation: ~
$ 
```

II. Lab Isolation(Host-Only) :

a) Changed the VM's network adapter from NAT to Host-Only Adapter



Network

Adapter 1 | Adapter 2 | Adapter 3 | Adapter 4

☑ Enable Network Adapter

Attached to: Host-only Adapter

Name: VirtualBox Host-Only Ethernet Adapter

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Deny

MAC Address: 0800278B1C81

☑ Virtual Cable Connected

Serial Ports

Port 1 | Port 2 | Port 3 | Port 4

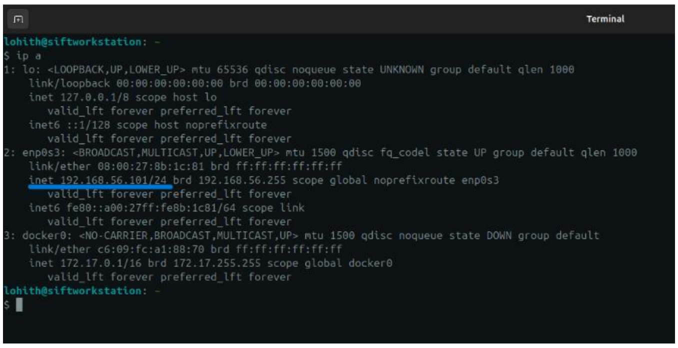
☐ Enable Serial Port

b) Started the VM

c) Identified the network adapter(enp0s3) and created the netplan configuration to set a static IP

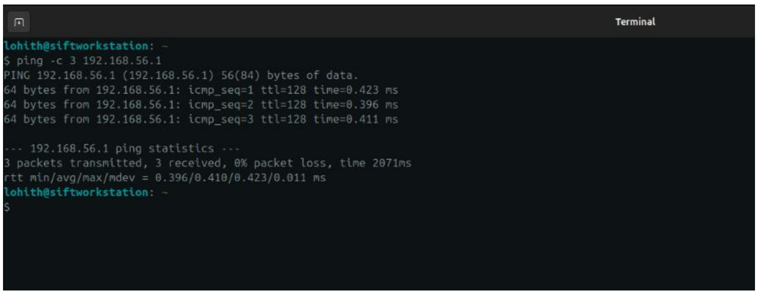
d) Applied the netplan configuration using sudo netplan apply

e) Verified Connectivity



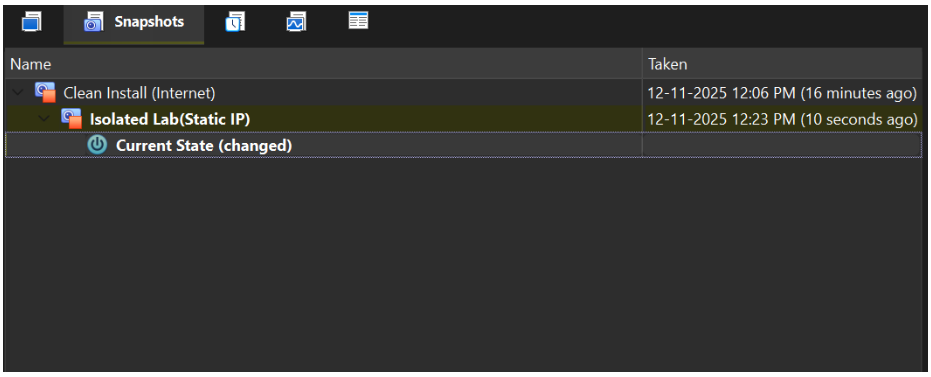
```
lohith@siftworkstation: ~
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:8b:1c:81 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.101/24 brd 192.168.56.255 scope global noprefixroute enp0s3
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe8b:1c81/64 scope link
       valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether c6:09:fc:a1:88:70 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
       valid_lft forever preferred_lft forever
lohith@siftworkstation: ~
$ 
```

f) Ping Host Check :



```
lohith@siftworkstation: ~
$ ping -c 3 192.168.56.1
PING 192.168.56.1 (192.168.56.1) 56(84) bytes of data.
64 bytes from 192.168.56.1: icmp_seq=1 ttl=128 time=0.423 ms
64 bytes from 192.168.56.1: icmp_seq=2 ttl=128 time=0.396 ms
64 bytes from 192.168.56.1: icmp_seq=3 ttl=128 time=0.411 ms

--- 192.168.56.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2071ms
rtt min/avg/max/mdev = 0.396/0.410/0.423/0.011 ms
lohith@siftworkstation: ~
$
```

g) Shut down the VM and took Snapshot – Isolated Lab(Static IP)



Name	Taken
Clean Install (Internet)	12-11-2025 12:06 PM (16 minutes ago)
Isolated Lab(Static IP)	12-11-2025 12:23 PM (10 seconds ago)
Current State (changed)	

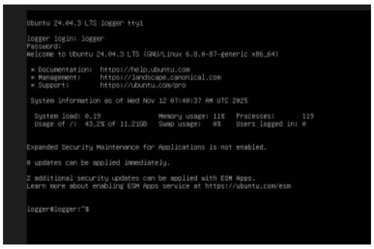
b. <u>LOGGER-VM (Ubuntu Server)</u>:

- Objective : To build a lightweight server, pre-install syslog-ng for log collection, and isolate it with a static IP.
- ISO Used : Ubuntu-24.04-live-server-amd64.iso

  Base VM Settings:
  - ➢ RAM : 2GB
  - ➢ vCPU's : 2
  - ➢ HDD : 20GB

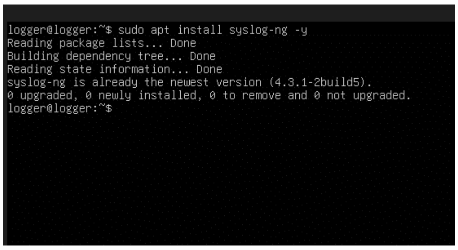
i. <u>Internet Phase Build</u> :

- Created a VM named Logger-VM with NAT
- Installed Ubuntu Server 24.04 server . During setup, selected "Install OpenSSH Server" to allow remote management.



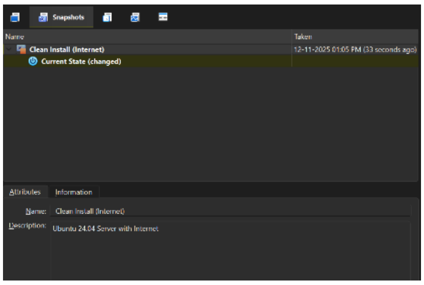
```
Ubuntu 24.04.3 LTS logger tty1

logger login: logger
Password:
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.8.0-87-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Wed Nov 12 07:40:37 AM UTC 2025

  System load: 0.19              Memory usage: 11%   Processes:       119
  Usage of /:  43.2% of 11.21GB  Swap usage:   0%    Users logged in: 0


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

2 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm


logger@logger:~$
```

- Once installed, logged in and installed syslog-ng , removing the default rsyslog to prevent conflicts:



```
logger@logger:~$ sudo apt install syslog-ng -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
syslog-ng is already the newest version (4.3.1-2build5).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
logger@logger:~$
```

- Shutdown the server using sudo shutdown now
- Took Snapshot Clean Install(Internet)



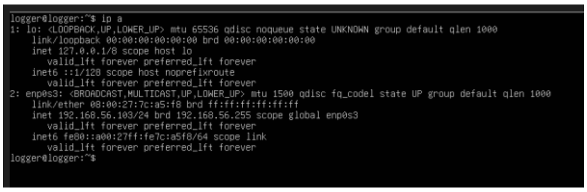
Name	Taken
Clean Install (Internet)	12-11-2025 01:05 PM (33 seconds ago)
Current State (changed)	

Name: Clean Install (Internet)

Description: Ubuntu 24.04 Server with Internet

ii. <u>Lab-Phase Isolation (Host-Only)</u> :

- Changed the VM's adapter from NAT to Host-Only Adapter

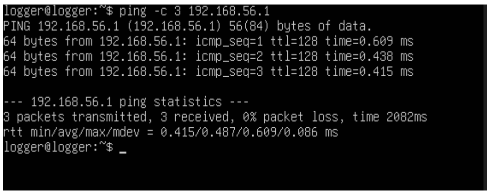


```
logger@logger:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:7c:a5:f8 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.103/24 brd 192.168.56.255 scope global enp0s3
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe7c:a5f8/64 scope link
       valid_lft forever preferred_lft forever
logger@logger:~$
```

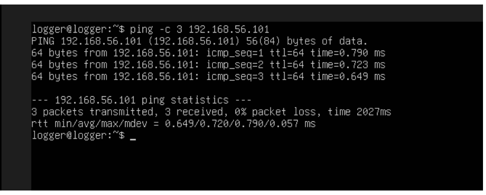
- Started the VM and identified the adapter name
- Set the static IP by creating /etc/netplan/01-netcfg.yaml :

```
network:
 version: 2
 renderer: networkd
 ethernets:
  enp0s3: #<-- Adapter name
   dhcp4: no
   dhcp6: no
   addresses:
    - 192.168.56.103/24
   gateway4: 192.168.56.1
   nameservers:
    addresses: [1.1.1.1]
```

- Applied the configuration using sudo netplan apply
- Verified connectivity to the host and the ANALYST-VM

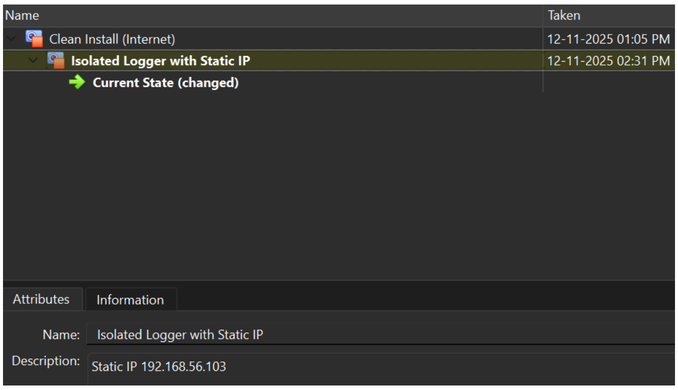


```
logger@logger:~$ ping -c 3 192.168.56.1
PING 192.168.56.1 (192.168.56.1) 56(84) bytes of data.
64 bytes from 192.168.56.1: icmp_seq=1 ttl=128 time=0.609 ms
64 bytes from 192.168.56.1: icmp_seq=2 ttl=128 time=0.438 ms
64 bytes from 192.168.56.1: icmp_seq=3 ttl=128 time=0.415 ms

--- 192.168.56.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2082ms
rtt min/avg/max/mdev = 0.415/0.487/0.609/0.086 ms
logger@logger:~$ _
```



```
logger@logger:~$ ping -c 3 192.168.56.101
PING 192.168.56.101 (192.168.56.101) 56(84) bytes of data.
64 bytes from 192.168.56.101: icmp_seq=1 ttl=64 time=0.790 ms
64 bytes from 192.168.56.101: icmp_seq=2 ttl=64 time=0.723 ms
64 bytes from 192.168.56.101: icmp_seq=3 ttl=64 time=0.649 ms

--- 192.168.56.101 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2027ms
rtt min/avg/max/mdev = 0.649/0.720/0.790/0.057 ms
logger@logger:~$ _
```

- Shutdown the server and took snapshot Isolated Lab(Static IP)



Name	Taken
Clean Install (Internet)	12-11-2025 01:05 PM
Isolated Logger with Static IP	12-11-2025 02:31 PM
Current State (changed)	

Name: Isolated Logger with Static IP

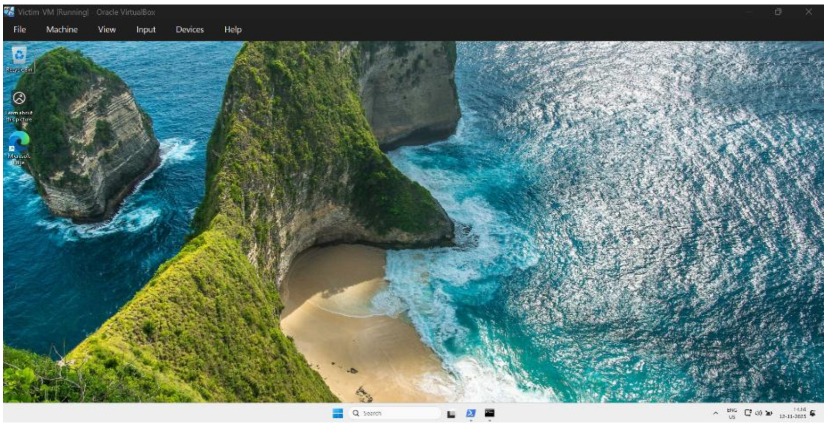
Description: Static IP 192.168.56.103

c. **<u>VICTIM-VM (Windows 11 LTSC)</u>**:

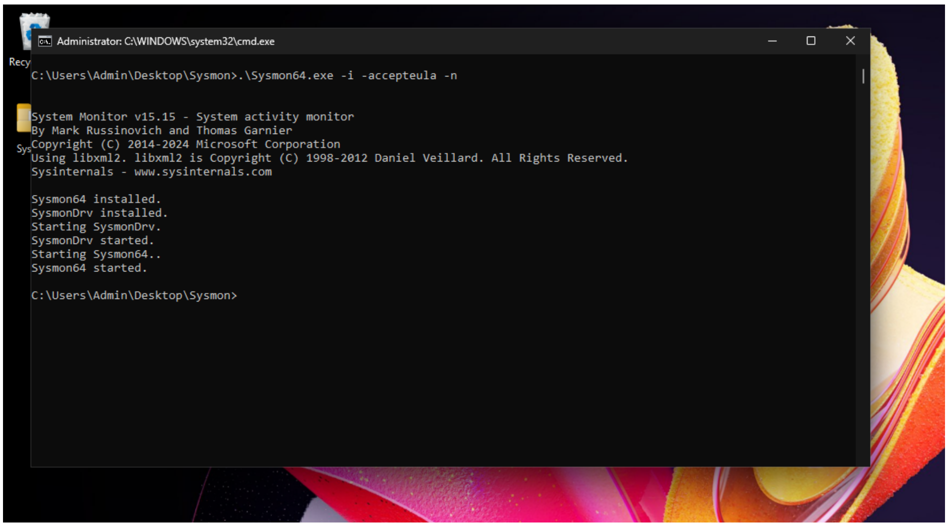
- Objective : To build a "victim" workstation, pre-install monitoring tools, and isolate it with a static IP.
- ISO Used : Windows-11-LTSC-2024-Eval.iso
- Base VM :
  - ➢ RAM : 4GB
  - ➢ vCPU's : 2
  - ➢ HDD : 60GB

i. <u>Internet-Phase Build</u> :

- Created a VM named VICTIM-VM with its network adapter set to NAT to provide internet for setup.
- Installed Windows 10 LTSC from the ISO. During setup, selected **"Offline account"** to create a local user.



- Installed VirtualBox Guest Additions (VBoxWindowsAdditions-amd64.exe) to improve performance and enable drag-and-drop.
- Downloaded and extracted the **Sysinternals Suite**.
- Installed **Sysmon** using an Admin PowerShell to provide deep system logging



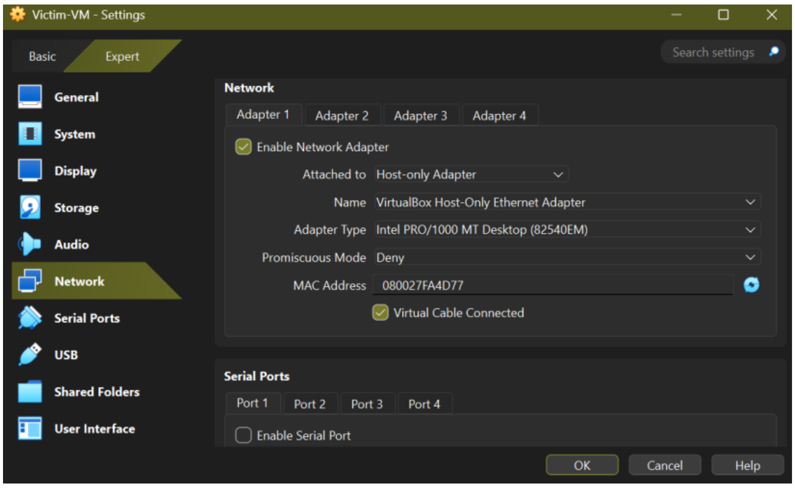
Administrator: C:\WINDOWS\system32\cmd.exe

```
C:\Users\Admin\Desktop\Sysmon>.\Sysmon64.exe -i -accepteula -n


System Monitor v15.15 - System activity monitor
By Mark Russinovich and Thomas Garnier
Copyright (C) 2014-2024 Microsoft Corporation
Using libxml2. libxml2 is Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.
Sysinternals - www.sysinternals.com

Sysmon64 installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting Sysmon64..
Sysmon64 started.

C:\Users\Admin\Desktop\Sysmon>
```

- Shut down the VM from the Start Menu.
- Took Snapshot - Clean Install (Internet). This serves as the master "template" for the victim machine.

ii. Lab-Phase Isolation (Host-Only) :

- Changed the VM’s adapter from NAT to Host-Only Adapter



Victim-VM - Settings

Network

Adapter 1

☑ Enable Network Adapter

Attached to: Host-only Adapter

Name: VirtualBox Host-Only Ethernet Adapter

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Deny

MAC Address: 080027FA4D77

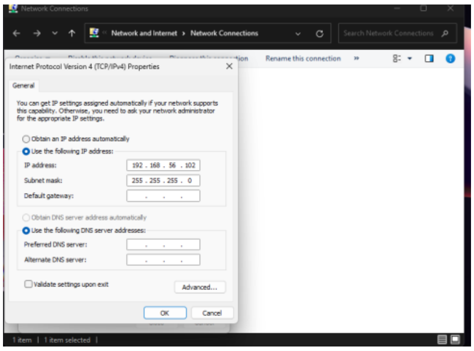
☑ Virtual Cable Connected

Serial Ports

Port 1

☐ Enable Serial Port

- Started the VM and configured the static IP using the ncpa.cpl GUI :
  - ➢ IP address : 192.168.56.102
  - ➢ Subnet Mask : 255.255.255.0
  - ➢ Gateway/DNS : Left Blank


Network Connections
Network and Internet › Network Connections
Search Network Connections
Rename this connection
1 item | 1 item selected |

Internet Protocol Version 4 (TCP/IPv4) Properties

General

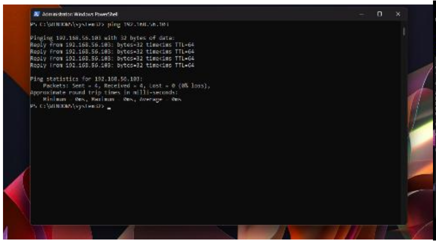
You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically
● Use the following IP address:
IP address: 192 . 168 . 56 . 102
Subnet mask: 255 . 255 . 255 . 0
Default gateway: . . .

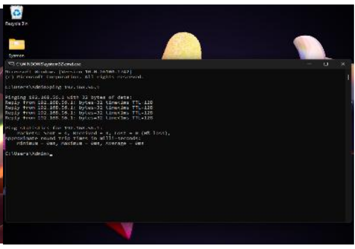
○ Obtain DNS server address automatically
● Use the following DNS server addresses:
Preferred DNS server: . . .
Alternate DNS server: . . .

□ Validate settings upon exit

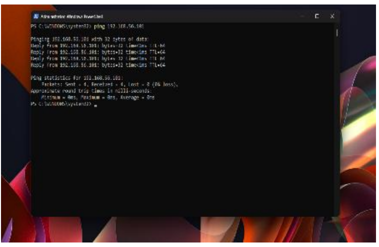
- Enabled ICMP (ping) in the Windows Defender Firewall to allow discovery from other lab VMs:
- Using Enable-NetFirewallRule -DisplayGroup "File and Printer Sharing"
- Verified connectivity to the host (.1), ANALYST-VM (.101), and LOGGER-VM (.103).



```
PS C:\WINDOWS\system32> ping 192.168.56.103

Pinging 192.168.56.103 with 32 bytes of data:
Reply from 192.168.56.103: bytes=32 time<1ms TTL=64
Reply from 192.168.56.103: bytes=32 time<1ms TTL=64
Reply from 192.168.56.103: bytes=32 time<1ms TTL=64
Reply from 192.168.56.103: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.56.103:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\WINDOWS\system32>
```



```
Microsoft Windows [Version 10.0.26100.1742]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin>ping 192.168.56.1

Pinging 192.168.56.1 with 32 bytes of data:
Reply from 192.168.56.1: bytes=32 time<1ms TTL=128
Reply from 192.168.56.1: bytes=32 time<1ms TTL=128
Reply from 192.168.56.1: bytes=32 time<1ms TTL=128
Reply from 192.168.56.1: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.56.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Admin>
```



```
PS C:\WINDOWS\system32> ping 192.168.56.101

Pinging 192.168.56.101 with 32 bytes of data:
Reply from 192.168.56.101: bytes=32 time<1ms TTL=64
Reply from 192.168.56.101: bytes=32 time<1ms TTL=64
Reply from 192.168.56.101: bytes=32 time<1ms TTL=64
Reply from 192.168.56.101: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.56.101:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\WINDOWS\system32>
```

- Shut down the VM and took Snapshot-Isolated Lab(Static IP)

5. Lab Setup: Phase 3 (Live Configuration)

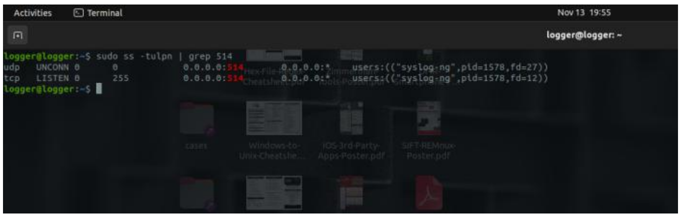
This phase details the configuration of services to connect the lab VMs, establishing the centralized logging pipeline. All VMs are started from their Isolated Lab (Static IP) snapshots.

a. Configuring the LOGGER-VM (syslog-ng) :
   - Started the LOGGER-VM from its snapshot.
   - Created a new configuration file to tell syslog-ng to accept network logs:

FILE -> /etc/syslog-ng/conf.d/net-listen.conf

```
source s_network {
  network(ip("0.0.0.0") port(514) transport("tcp"));
  network(ip("0.0.0.0") port(514) transport("udp"));
};

destination d_remote_logs {
  file("/var/log/remote/$HOST/$PROGRAM.log"
    owner("root") group("root") perm(0600) create_dirs(yes)
  );
};

log {
  source(s_network);
  destination(d_remote_logs);
  flags(final);
};
```

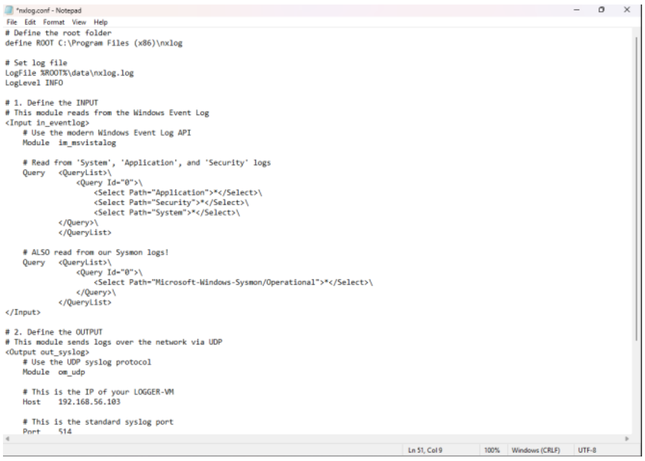
- Manually created the remote directory to ensure permissions were correct: sudo mkdir -p /var/log/remote.
- Restarted the service : sudo systemctl restart syslog-ng
- Verified the listener : sudo ss -tulpn | grep 514



```
logger@logger:~$ sudo ss -tulpn | grep 514
udp   UNCONN 0      0             0.0.0.0:514        0.0.0.0:*    users:(("syslog-ng",pid=1578,fd=27))
tcp   LISTEN 0      255           0.0.0.0:514        0.0.0.0:*    users:(("syslog-ng",pid=1578,fd=12))
logger@logger:~$ 
```

b. Configuring the VICTIM-VM(NXLOG)

- Started the VM.
- Installed NXLOG Community edition.
- Configured C:\Program Files\nxlog\conf\nxlog.conf to collect standard Event Logs and Sysmon logs, forwarding them to 192.168.56.103

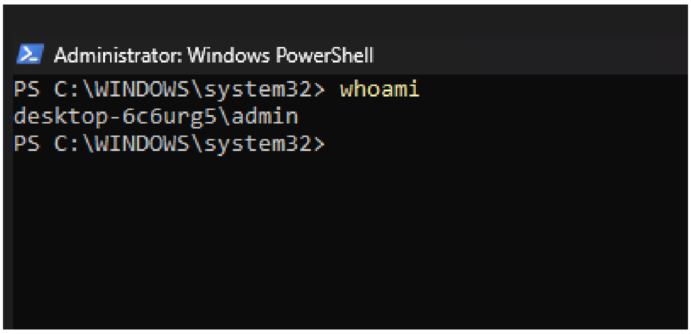


```
# Define the root folder
define ROOT C:\Program Files (x86)\nxlog

# Set log file
LogFile %ROOT%\data\nxlog.log
LogLevel INFO

# 1. Define the INPUT
# This module reads from the Windows Event Log
<Input in_eventlog>
    # Use the modern Windows Event Log API
    Module  im_msvistalog

    # Read from 'System', 'Application', and 'Security' logs
    Query   <QueryList>\
                <Query Id="0">\
                    <Select Path="Application">*</Select>\
                    <Select Path="Security">*</Select>\
                    <Select Path="System">*</Select>\
            </Query>\
            </QueryList>

    # ALSO read from our Sysmon logs!
    Query   <QueryList>\
                <Query Id="0">\
                    <Select Path="Microsoft-Windows-Sysmon/Operational">*</Select>\
                </Query>\
            </QueryList>
</Input>

# 2. Define the OUTPUT
# This module sends logs over the network via UDP
<Output out_syslog>
    # Use the UDP syslog protocol
    Module  om_udp

    # This is the IP of your LOGGER-VM
    Host    192.168.56.103

    # This is the standard syslog port
    Port    514
```

- Started the nxlog service

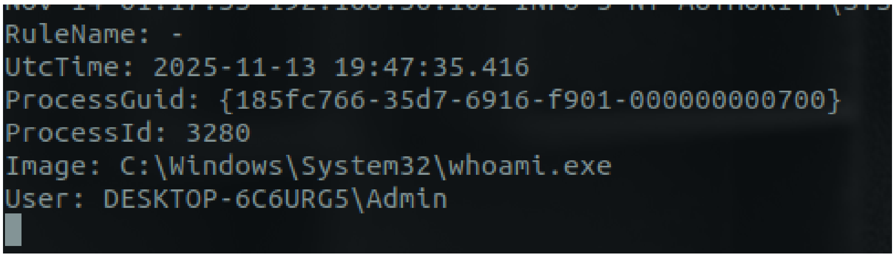
c. Verifying the Log pipeline :

- Generated "noise" on the VICTIM-VM by running whoami in PowerShell.



```
PS C:\WINDOWS\system32> whoami
desktop-6c6urg5\admin
PS C:\WINDOWS\system32>
```

- On the LOGGER-VM, monitored the incoming logs in real-time:
- Using sudo tail -f /var/log/remote/192.168.56.102/nxlog.log
- Success: Confirmed that Sysmon "Process Create" events (Event ID 1) for whoami.exe appeared instantly on the logger.



```
Nov 14 01:17:35 192.168.56.102 INFO 5 NT AUTHORITY\SYS
RuleName: -
UtcTime: 2025-11-13 19:47:35.416
ProcessGuid: {185fc766-35d7-6916-f901-000000000700}
ProcessId: 3280
Image: C:\Windows\System32\whoami.exe
User: DESKTOP-6C6URG5\Admin
```

6. Standard Operating Procedures (SOP) :

To maintain lab integrity and ensure a consistent testing environment, the following procedures must be followed.

a. Lab Start-Up Routine

- Open VirtualBox.
- Start the LOGGER-VM first. This ensures the listener is ready before logs start arriving.
- Start the VICTIM-VM
- Start the ANALYST-VM
- Verify the pipeline by checking the live log stream on the Logger:
- sudo tail -f /var/log/remote/192.168.56.102/nxlog.log

b. Lab Reset Routine (Crucial)

- Shutdown all VMs.
- For each VM, go to the Snapshots tab.
- Right-click the v2.1 - Isolated Lab (or latest working) snapshot.
- Select RESTORE.
- Uncheck "Create a snapshot of the current machine state"
- Click Restore. This guarantees that every new lab session begins with a mathematically identical, clean baseline.

7. Future Roadmap :

This project lays the foundation for advanced cybersecurity scenarios. Future enhancements planned for this lab include:

Offensive Integration: Adding a Kali Linux VM (192.168.56.104) to the isolated network to simulate active attacks (brute force, exploits) against the Victim VM.

Visualization: Upgrading the LOGGER-VM to run Elasticsearch and Kibana (ELK Stack). This will replace the raw text logs with a graphical dashboard to visualize attacks.

Malware Analysis: Creating a dedicated "Detonation" snapshot for the VICTIM-VM with Windows Defender disabled, allowing for static and dynamic analysis of real malware samples.

Active Directory: Promoting the VICTIM-VM to a Domain Controller or adding a Windows Server VM to practice AD attacks (Kerberoasting, Golden Ticket).

8. CONCLUSION :

This project successfully established a functional, isolated, and professional-grade Digital Forensics and Incident Response (DFIR) laboratory. By leveraging virtualization, we created a secure environment that mimics a real-world corporate network.

Key achievements include:

- **Full Isolation:** Implementation of a Host-Only network (192.168.56.0/24) ensures that simulated threats cannot escape to the host or internet.
- **Deep Visibility:** Integration of **Sysmon** allows for granular kernel-level monitoring of process creation, network connections, and file modifications.
- **Centralized Logging:** The **NXLog** and **syslog-ng** pipeline successfully forwards logs in real-time, enabling "over-the-shoulder" monitoring of the victim machine.
- **Forensic Readiness:** The **SIFT Workstation** is configured and ready for disk image and memory analysis.

This lab now serves as the primary workspace for Certified Penetration Tester (CPT) studies and practical skill development.

9. References & Tools Used

- **VirtualBox:** Oracle VM VirtualBox (Hypervisor)
- **SIFT Workstation:** SANS Institute (Forensics)
- **Ubuntu Server 24.04 LTS:** Canonical (Log Server)
- **Windows 10 Enterprise LTSC:** Microsoft (Target OS)
- **Sysinternals Suite (Sysmon):** Microsoft
- **NXLog Community Edition:** Log Forwarding Agent
- **Syslog-ng:** Log Collection Daemon