#### **1. System Updates**

Run the following command to update your system:

sudo apt update

Sudo apt upgrade

Sudo snap refresh

#### **2. Installing and Configuring Ghidra**

1. Download Ghidra from its [GitHub repository](https://github.com/NationalSecurityAgency/ghidra).
2. Extract the downloaded file:

unzip {file\_name}.zip

1. Install the required dependencies:

sudo apt install openjdk-21-jdk

1. Navigate to the Ghidra directory and run the application:

cd ghidra\_11.2\_PUBLIC/  
./ghidraRun

#### **3. Installing and Configuring Radare2**

1. Clone the repository:

git clone <https://github.com/radareorg/radare2.git>

1. If git is not installed, run:

sudo apt install git

1. Navigate to the installation directory and install Radare2:

cd radare2/sys  
sudo apt update  
sudo apt install build-essential  
./install.sh

#### **4. Installing KVM for Virtualization**

1. Install KVM and dependencies:

sudo apt update  
sudo apt install qemu-kvm libvirt-daemon-system libvirt-clients bridge-utils

1. Verify installation:

lsmod | grep -i kvm

You should see kvm and either kvm\_amd or kvm\_intel depending on your CPU.

#### **5. Installing Grafana**

Install Grafana using Snap:

sudo snap install grafana

#### **6. Setting Up Directories**

Create directories for organizing files:

mkdir ISO-Images # For storing ISO files  
mkdir malware # For storing malware samples

#### **7. Configuring Virtual Machines**

1. Install virt-manager:

bash

Copy code

sudo apt install virt-manager

1. Open Virt-Manager and create a new virtual machine:
   1. Select **"Local install media"** for the installation type.
   2. Browse to the ISO-Images directory to select the appropriate ISO file.
   3. Configure processors and memory as needed.

**Optional: Change Resolution of Hyper-V Machines**

1. Open PowerShell as Administrator.
2. Run the following command to adjust screen resolution:

Set-VMVideo -VMName {machine\_name} -HorizontalResolution 2560 -VerticalResolution 1440 -ResolutionType single

#### **8. Windows Sandbox Tools**

Download and install the following tools on your Windows sandbox:

* [Procmon64 (Process Monitor)](https://docs.microsoft.com/en-us/sysinternals/downloads/procmon)
* [FakeNet-NG](https://github.com/mandiant/flare-fakenet-ng)
* [Wireshark](https://www.wireshark.org/)
* [DumpIt.exe](https://www.moonsols.com/) or FTK Imager
* Sysmon (Windows version)
* <https://github.com/olafhartong/sysmon-modular/blob/master/sysmonconfig-mde-augment.xml>

Open PowerShell to provide the configuration file to sysmon:

Cd /path/to/sysmon

Sysmon.exe -i /path/to/config\_file –accepteula

Verify with sysmon.exe -c and you should see the config file name.

#### **9. Ubuntu Sandbox Tools**

Install the following tools on your Ubuntu machine:

sudo apt install inetsim wireshark tcpdump binwalk strace ltrace lsof

* Internal logs can be accessed from /var/log.
* Install Sysmon for Linux from its [official repository](https://github.com/Sysinternals/SysmonForLinux).

#### **10. Setting Up Scripts**

1. **Automation Scripts**:
   1. **automation.sh** (Ubuntu host script):
      1. Searches for unanalyzed malware in the malware directory.
      2. Starts a Windows VM using a pre-configured snapshot.
      3. Waits for required files from the Windows VM.
      4. Processes logs and memory dumps into JSON format.
   2. **Conversion.py**:
      1. Converts Process Monitor logs from XML to JSON
   3. **Influx.py**:
      1. Ingests JSON data into InfluxDB for Grafana visualization.
2. **PowerShell Script**:
   1. Used for operations on the nested Windows machine.
   2. Includes options for testing with sshpass (not recommended for production environments).

#### **11. Running the Automation Process**

1. Place malware samples in the malware directory. Use a repository like [TheZoo](https://github.com/ytisf/theZoo) for sample malware and make sure to extract them one by one to the separate “malware” directory.
2. Start the automation process:
   1. The script identifies malware and launches the Windows VM from a snapshot.
   2. Once analysis completes, files are converted and sent to the JSON directory.
3. Analyze logs:
   1. Process Monitor and Wireshark logs are converted to JSON.
   2. Memory dumps are analyzed using Volatility and results are stored.

#### **12. Visualization with Grafana**

1. Connect Grafana to InfluxDB.
2. Import data from InfluxDB into Grafana for analysis.

This documentation provides an overview of configuring a malware analysis sandbox. Ensure that all tools and scripts are used in a controlled and isolated environment to prevent accidental harm.

13. Future Advances:

* As of currently, future advances we are looking at fully implementing linux malware, as of currently we still need user interaction with the linux machine, but considerations for linux are being taken such as different tools.