



# CUCKOO INSTALLATION AND SETUP MANUAL

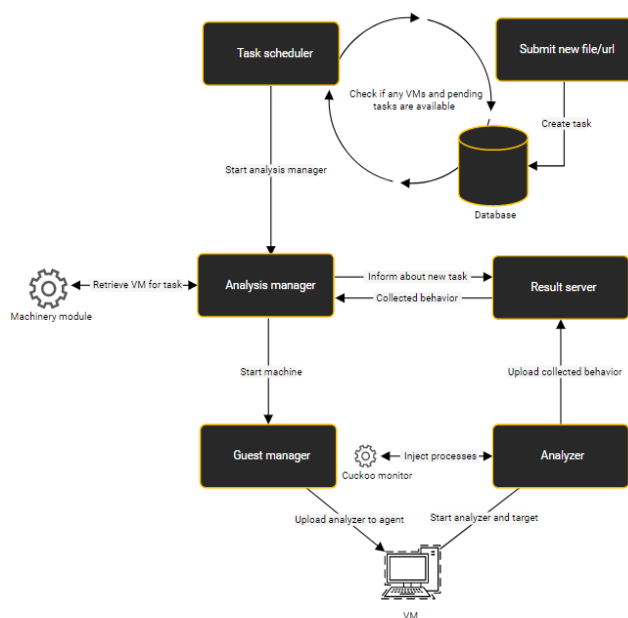
## 1. Introduction

Cuckoo is an open-source automated malware analysis system that consists of multiple modular components that work together to collect and represent the behavioral data of malware. Users can simply throw any suspicious file at cuckoo, and it will generate a detailed report outlining the behavior of the file. Cuckoo sandbox is freely available software that can analyze malicious files under Windows, macOS, Linux and android.

Some of the features that cuckoo sandbox is capable of are:

- a. Different malicious files such as office documents, pdf files, emails, executables can be analyzed along with the malicious websites as well.
- b. Network traffic trace in PCAP (Packet capture) format as well as screenshot is taken during the execution of malware.
- c. Using Yara it can perform advanced memory analysis of the infected virtualized system.
- d. Memory dumps of the malware process as well as full memory dumps of the machine.

Once a file is submitted in the Cuckoo installed laptop or PC a new entry will be made into the database which generates the new task ID and performs all the analysis which can be seen in the flow diagram below.



<https://hatching.io/static/images/blog/cuckoo-sandbox-architecture/cuckoo-diagram4.svg>

## **2. Installation Procedure**

The latest version of cuckoo is version 2.0.7. Every other version has a different manual to install it. Here, we will be writing a tutorial and small details on how to install the latest version of cuckoo 2.0.7. We faced multiple challenges such as time consuming, system error, space error while installing the requirement for cuckoo installation. There are different ways of installing cuckoo found in the web browser which barely works when we follow all steps. Maybe it's because of different systems or indeterminate instructions. The process that we tried by ourselves works on 3 different laptops which is why we are sure that it will work on other machines too.

### **2.1 Installing Ubuntu using VirtualBox**

The cuckoo sandbox setup is performed in Windows-10 (x86) using virtual box (VMWare). First, we install Oracle VM VirtualBox in Windows 10 and install Ubuntu-20.04.03 inside that VirtualBox. Here are the links to download the VirtualBox and Ubuntu 20.04.03 iso.

<https://ubuntu.com/download/desktop>

<https://www.virtualbox.org/wiki/Downloads>

We set the RAM - 6 GB (at least 4 GB required) and Hard Drive Space - 60 GB (at least 40 GB required) for VMware as we will be installing Windows 7 inside the Ubuntu with 2 GB RAM and 20 GB Hard Disk Space.

According to the cuckoo sandbox requirement and installation guide, we performed most of the syntax command in the terminal of ubuntu.

## **2.2 Commands and Screenshots**

### **2.2.1 Installing Prerequisite package for cuckoo installation**

First update all the package and upgrade using this command:

```
$ sudo apt-get update
```

```
$ sudo apt-get upgrade -y
```

Create a new user cuckoo as we don't want cuckoo to run as root.

```
$ sudo adduser cuckoo (cuckoo can be changed into any other username)
```

```
$ sudo adduser cuckoo sudo
```

Install curl which is a tool of python used to transfer data requests to and from server

```
sudo apt-get install curl
```

```
curl https://bootstrap.pypa.io/pip/2.7/get-pip.py -o get-pip.py
```

```
bisitaul@bisitaul: ~  
bisitaul@bisitaul:~$ sudo apt-get update && sudo apt-get upgrade -y  
Hit:1 http://us.archive.ubuntu.com/ubuntu focal InRelease  
Hit:2 http://us.archive.ubuntu.com/ubuntu focal-updates InRelease  
Hit:3 http://security.ubuntu.com/ubuntu focal-security InRelease  
Hit:4 http://us.archive.ubuntu.com/ubuntu focal-backports InRelease  
Reading package lists... Done  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
Calculating upgrade... Done  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
bisitaul@bisitaul:~$ sudo adduser cuckoo  
Adding user 'cuckoo' ...  
Adding new group 'cuckoo' (1001) ...  
Adding new user 'cuckoo' (1001) with group 'cuckoo' ...  
Creating home directory '/home/cuckoo' ...  
Copying files from '/etc/skel' ...  
New password:  
Retype new password:  
passwd: password updated successfully  
Changing the user information for cuckoo  
Enter the new value, or press ENTER for the default  
Full Name []:  
Room Number []:  
Work Phone []:  
Home Phone []:  
Other []:  
Is the information correct? [Y/n] y  
bisitaul@bisitaul:~$ sudo adduser cuckoo sudo  
Adding user 'cuckoo' to group 'sudo' ...  
Adding user cuckoo to group sudo  
Done.  
bisitaul@bisitaul:~$ sudo apt-get install curl  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following NEW packages will be installed:  
curl  
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.  
Need to get 161 kB of archives.  
After this operation, 412 kB of additional disk space will be used.  
Get:1 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 curl amd64 7.68.0-1ubuntu2.7 [161 kB]  
Fetched 161 kB in 0s (372 kB/s)  
Selecting previously unselected package curl.  
(Reading database ... 184180 files and directories currently installed.)  
Preparing to unpack .../curl_7.68.0-1ubuntu2.7_amd64.deb ...  
Unpacking curl (7.68.0-1ubuntu2.7) ...  
Setting up curl (7.68.0-1ubuntu2.7) ...  
Processing triggers for man-db (2.9.1-1) ...  
bisitaul@bisitaul:~$
```

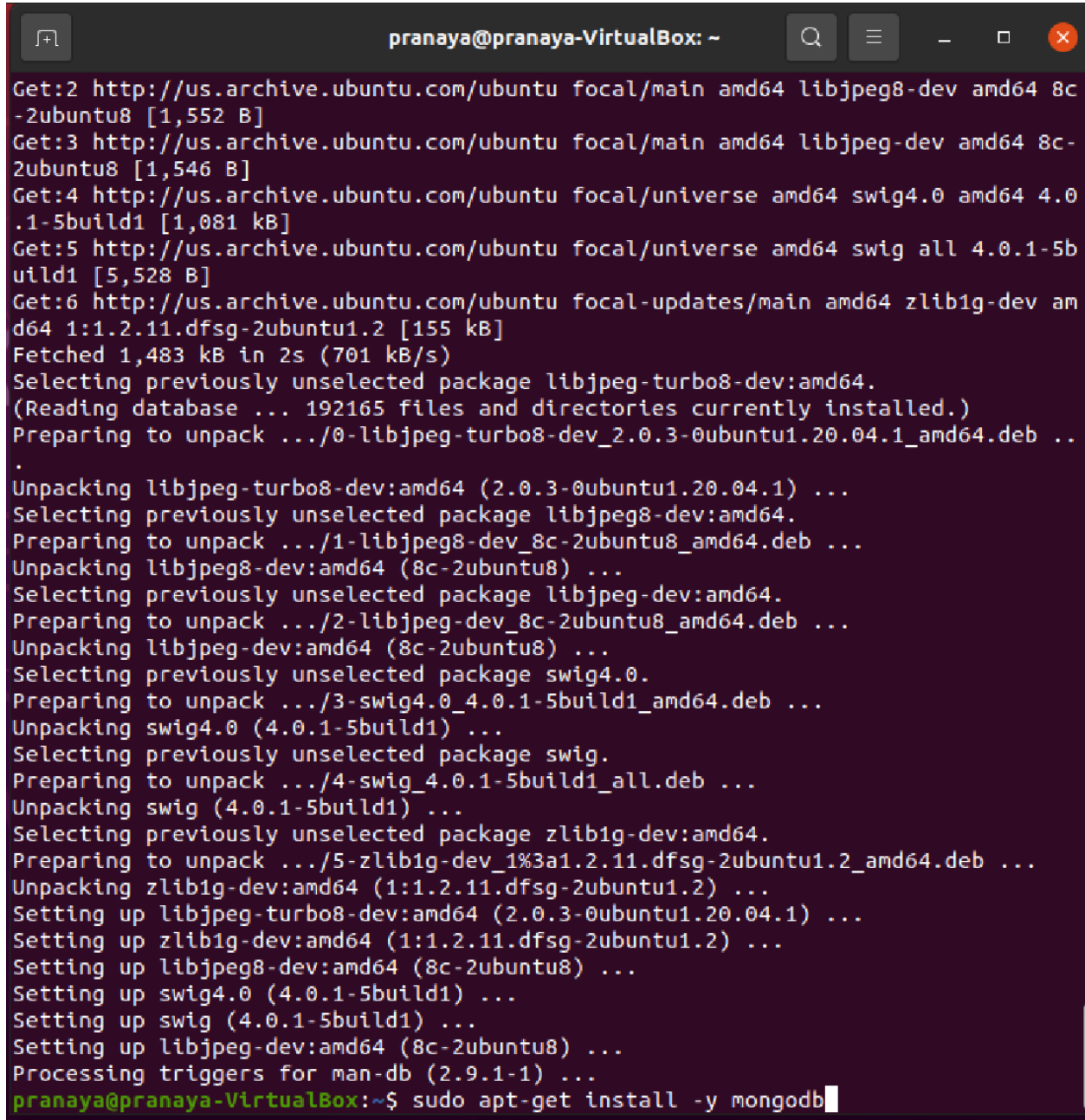
**Fig: Updating sudo and installing curl tool**

Install python libraries and packages from the apt repositories which are required to get cuckoo work in our environment.

```
$ sudo apt-get install python  
$ sudo python get-pip.py  
$ sudo apt-get install -y python-dev libffi-dev libssl-dev libfuzzy-dev libtool flex  
$ autoconf libjansson-dev git  
$ sudo apt-get install -y python-setuptools  
$ sudo apt-get install -y libjpeg-dev zlib1g-dev swig
```

Cuckoo web interface required mongoDB to be installed and enabled since it will be used to analyze the results of submitted tasks on cuckoo. It will work as a backend to the Django web interface.

```
$ sudo apt-get install -y mongodb
```

A terminal window titled 'pranaya@pranaya-VirtualBox: ~' showing the output of the command 'sudo apt-get install -y mongodb'. The terminal displays the installation of various dependencies: libjpeg8-dev, libjpeg-dev, swig4.0, swig, and zlib1g-dev. It shows the process of fetching packages, selecting previously unselected packages, and unpacking them. The installation of MongoDB itself is not yet complete, as the prompt is still at the end of the command line.

```
pranaya@pranaya-VirtualBox: ~  
Get:2 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libjpeg8-dev amd64 8c-2ubuntu8 [1,552 B]  
Get:3 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libjpeg-dev amd64 8c-2ubuntu8 [1,546 B]  
Get:4 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 swig4.0 amd64 4.0.1-5build1 [1,081 kB]  
Get:5 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 swig all 4.0.1-5build1 [5,528 B]  
Get:6 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 zlib1g-dev amd64 1:1.2.11.dfsg-2ubuntu1.2 [155 kB]  
Fetched 1,483 kB in 2s (701 kB/s)  
Selecting previously unselected package libjpeg-turbo8-dev:amd64.  
(Reading database ... 192165 files and directories currently installed.)  
Preparing to unpack .../0-libjpeg-turbo8-dev_2.0.3-0ubuntu1.20.04.1_amd64.deb ...  
Unpacking libjpeg-turbo8-dev:amd64 (2.0.3-0ubuntu1.20.04.1) ...  
Selecting previously unselected package libjpeg8-dev:amd64.  
Preparing to unpack .../1-libjpeg8-dev_8c-2ubuntu8_amd64.deb ...  
Unpacking libjpeg8-dev:amd64 (8c-2ubuntu8) ...  
Selecting previously unselected package libjpeg-dev:amd64.  
Preparing to unpack .../2-libjpeg-dev_8c-2ubuntu8_amd64.deb ...  
Unpacking libjpeg-dev:amd64 (8c-2ubuntu8) ...  
Selecting previously unselected package swig4.0.  
Preparing to unpack .../3-swig4.0_4.0.1-5build1_amd64.deb ...  
Unpacking swig4.0 (4.0.1-5build1) ...  
Selecting previously unselected package swig.  
Preparing to unpack .../4-swig_4.0.1-5build1_all.deb ...  
Unpacking swig (4.0.1-5build1) ...  
Selecting previously unselected package zlib1g-dev:amd64.  
Preparing to unpack .../5-zlib1g-dev_1%3a1.2.11.dfsg-2ubuntu1.2_amd64.deb ...  
Unpacking zlib1g-dev:amd64 (1:1.2.11.dfsg-2ubuntu1.2) ...  
Setting up libjpeg-turbo8-dev:amd64 (2.0.3-0ubuntu1.20.04.1) ...  
Setting up zlib1g-dev:amd64 (1:1.2.11.dfsg-2ubuntu1.2) ...  
Setting up libjpeg8-dev:amd64 (8c-2ubuntu8) ...  
Setting up swig4.0 (4.0.1-5build1) ...  
Setting up swig (4.0.1-5build1) ...  
Setting up libjpeg-dev:amd64 (8c-2ubuntu8) ...  
Processing triggers for man-db (2.9.1-1) ...  
pranaya@pranaya-VirtualBox:~$ sudo apt-get install -y mongodb
```

**Fig: Installing MongoDB**

PostgreSQL is required for cuckoo installation. Here is the command to install postgresql.

```
$ sudo apt-get install -y postgresql libpq-dev
```

Install and setup Virtual box using these commands:

```
$ sudo apt-get install -y virtualbox
```

```
pranaya@pranaya-VirtualBox: ~  
The database cluster will be initialized with locale "en_US.UTF-8".  
The default database encoding has accordingly been set to "UTF8".  
The default text search configuration will be set to "english".  
  
Data page checksums are disabled.  
  
fixing permissions on existing directory /var/lib/postgresql/12/main ... ok  
creating subdirectories ... ok  
selecting dynamic shared memory implementation ... posix  
selecting default max_connections ... 100  
selecting default shared_buffers ... 128MB  
selecting default time zone ... America/Denver  
creating configuration files ... ok  
running bootstrap script ... ok  
performing post-bootstrap initialization ... ok  
syncing data to disk ... ok  
  
Success. You can now start the database server using:  
  
    pg_ctlcluster 12 main start  
  
Ver Cluster Port Status Owner    Data directory          Log file  
12  main     5432 down   postgres /var/lib/postgresql/12/main /var/log/postgresql/  
/postgresql-12-main.log  
update-alternatives: using /usr/share/postgresql/12/man/man1/postmaster.1.gz to  
provide /usr/share/man/man1/postmaster.1.gz (postmaster.1.gz) in auto mode  
Setting up sysstat (12.2.0-2ubuntu0.1) ...  
  
Creating config file /etc/default/sysstat with new version  
update-alternatives: using /usr/bin/sar.sysstat to provide /usr/bin/sar (sar) in  
auto mode  
Created symlink /etc/systemd/system/multi-user.target.wants/sysstat.service → /l  
ib/systemd/system/sysstat.service.  
Setting up postgresql (12+214ubuntu0.1) ...  
Processing triggers for systemd (245.4-4ubuntu3.13) ...  
Processing triggers for man-db (2.9.1-1) ...  
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...  
pranaya@pranaya-VirtualBox:~$ sudo apt-get install -y virtualbox
```

**Fig 2: Installing virtualbox**

Further, Install and setup volatility to do forensic analysis on memory dumps using following commands:

```
$ cd Downloads/  
$ ~/Downloads  
$ git clone https://github.com/volatilityfoundation/volatility.git  
$ cd volatility  
$ sudo python setup.py build  
$ sudo python setup.py install  
$ cd ..
```

Next step is to install and setup distorm, ssdeep, tcpdump, ujson, openpyxl, jupyter using following commands:

```
$ sudo -H pip install distorm3==3.4.4
```

we install yara using this command:

```
$ sudo -H pip install yara-python==3.6.3
$ sudo apt-get install -y ssdeep
$ ssdeep -V
```

pydeep is an optional plugin which is installed using this command:

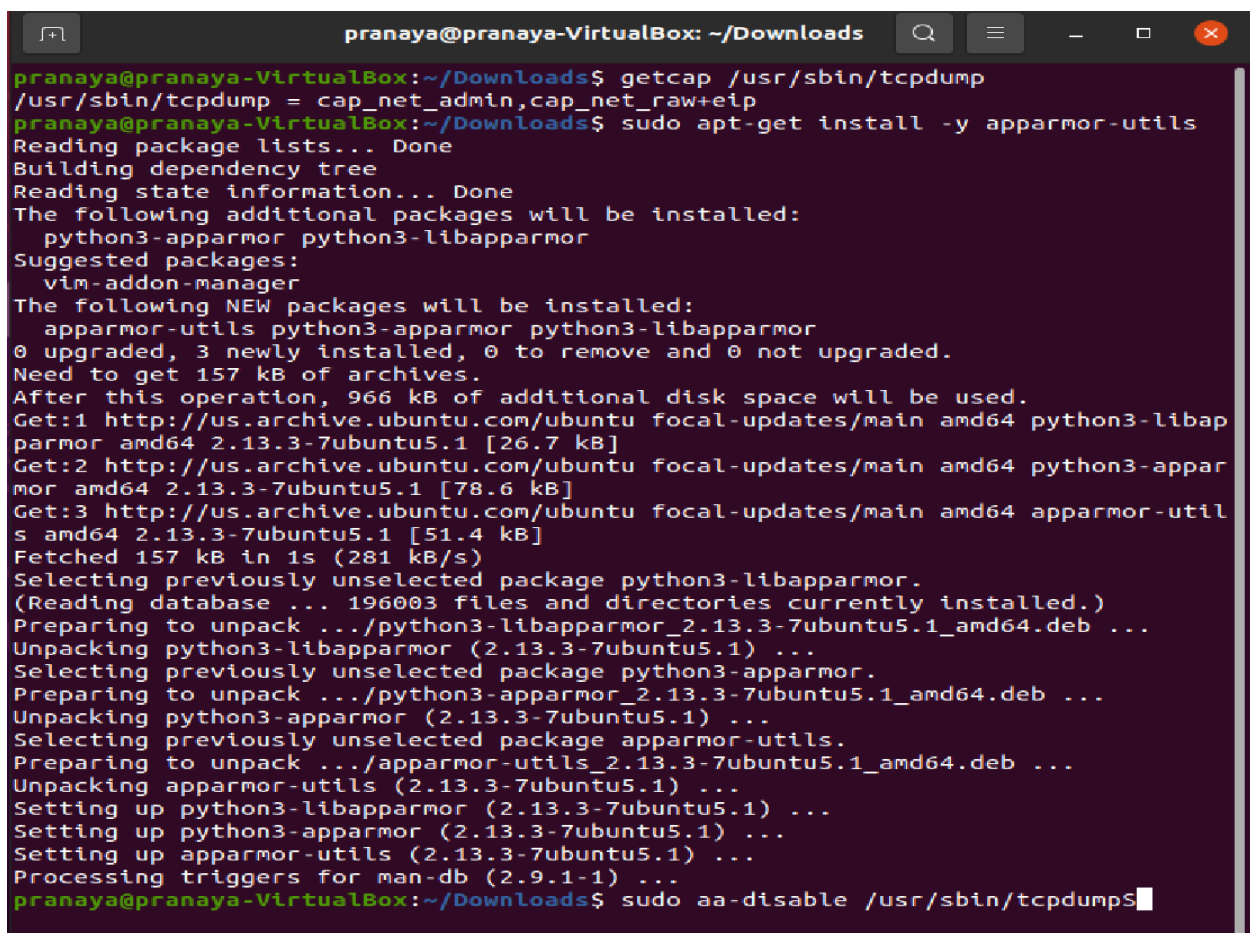
```
$ sudo -H pip install pydeep
$ pip show pydeep
```

Type Following command for the rest of the packages needed for cuckoo installation.

```
$ sudo -H pip install openpyxl
$ sudo -H pip install ujson
$ sudo -H pip install jupyter
```

Install tcpdump which is required to dump network activity performed by the malware during execution.

```
$ sudo apt-get install tcpdump
```



```
pranaya@pranaya-VirtualBox: ~/Downloads
pranaya@pranaya-VirtualBox:~/Downloads$ getcap /usr/sbin/tcpdump
/usr/sbin/tcpdump = cap_net_admin,cap_net_raw+eip
pranaya@pranaya-VirtualBox:~/Downloads$ sudo apt-get install -y apparmor-utils
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  python3-apparmor python3-libapparmor
Suggested packages:
  vim-addon-manager
The following NEW packages will be installed:
  apparmor-utils python3-apparmor python3-libapparmor
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 157 kB of archives.
After this operation, 966 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 python3-libapparmor amd64 2.13.3-7ubuntu5.1 [26.7 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 python3-apparmor amd64 2.13.3-7ubuntu5.1 [78.6 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu focal-updates/main amd64 apparmor-utils amd64 2.13.3-7ubuntu5.1 [51.4 kB]
Fetched 157 kB in 1s (281 kB/s)
Selecting previously unselected package python3-libapparmor.
(Reading database ... 196003 files and directories currently installed.)
Preparing to unpack .../python3-libapparmor_2.13.3-7ubuntu5.1_amd64.deb ...
Unpacking python3-libapparmor (2.13.3-7ubuntu5.1) ...
Selecting previously unselected package python3-apparmor.
Preparing to unpack .../python3-apparmor_2.13.3-7ubuntu5.1_amd64.deb ...
Unpacking python3-apparmor (2.13.3-7ubuntu5.1) ...
Selecting previously unselected package apparmor-utils.
Preparing to unpack .../apparmor-utils_2.13.3-7ubuntu5.1_amd64.deb ...
Unpacking apparmor-utils (2.13.3-7ubuntu5.1) ...
Setting up python3-libapparmor (2.13.3-7ubuntu5.1) ...
Setting up python3-apparmor (2.13.3-7ubuntu5.1) ...
Setting up apparmor-utils (2.13.3-7ubuntu5.1) ...
Processing triggers for man-db (2.9.1-1) ...
pranaya@pranaya-VirtualBox:~/Downloads$ sudo aa-disable /usr/sbin/tcpdump
```

### **Fig: Disabling tcpdump**

Next, Install setcap with following command:

- ```
$ sudo apt-get install libcap2-bin
```
- ```
$ sudo setcap cap_net_raw,cap_net_admin=eip /usr/sbin/tcpdump
```
- Check whether setcap is installed properly by using this command:  

```
$ getcap /usr/sbin/tcpdump
```
  - If you see the following output, then setcap is properly installed.

```
/usr/sbin/tcpdump = cap_net_admin, cap_net_raw+eip
```

Install apparmor utils which is a linux security module using following command:

```
$ sudo apt-get install -y apparmor-utils
```

```
sudo aa-disable /usr/sbin/tcpdump    //(Disable tcpdump)
```

#### **2.2.2 Cuckoo Installation**

Until now we installed and set up all the requirements to install and run cuckoo for the first time.

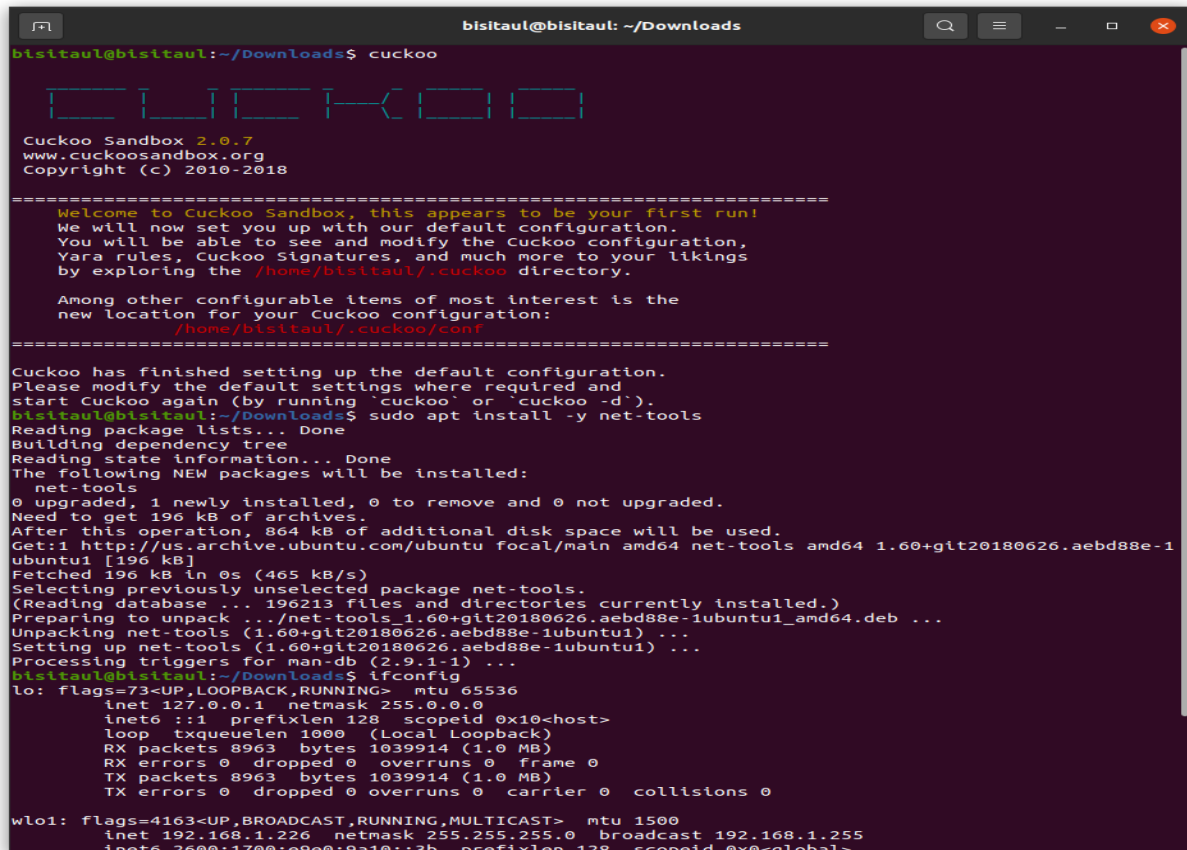
Next, install cuckoo using following command:

```
$ pip install -U pip setuptools
```

```
$ sudo -H pip install -U cuckoo
```



Command to run cuckoo:     \$ cuckoo

A terminal window titled 'bisitaul@bisitaul: ~/Downloads' showing the execution of the 'cuckoo' command. The output displays the Cuckoo Sandbox 2.0.7 logo and version information. It then shows a welcome message and instructions for configuration. The user runs 'sudo apt install -y net-tools', which outputs package list details and installation progress for 'net-tools'. Finally, the user runs 'ifconfig', showing the configuration for the 'lo' (loopback) and 'wlo1' (wireless) network interfaces.

```
bisitaul@bisitaul: ~/Downloads$ cuckoo

Cuckoo Sandbox 2.0.7
www.cuckoosandbox.org
Copyright (c) 2010-2018

Welcome to Cuckoo Sandbox, this appears to be your first run!
We will now set you up with our default configuration.
You will be able to see and modify the Cuckoo configuration,
Yara rules, Cuckoo Signatures, and much more to your likings
by exploring the /home/bisitaul/.cuckoo directory.

Among other configurable items of most interest is the
new location for your Cuckoo configuration:
/home/bisitaul/.cuckoo/conf

Cuckoo has finished setting up the default configuration.
Please modify the default settings where required and
start Cuckoo again (by running 'cuckoo' or 'cuckoo -d').
bisitaul@bisitaul:~/Downloads$ sudo apt install -y net-tools
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
 net-tools
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 196 kB of archives.
After this operation, 864 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu focal/main amd64 net-tools amd64 1.60+git20180626.aebd88e-1ubuntu1 [196 kB]
Fetched 196 kB in 0s (465 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 196213 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20180626.aebd88e-1ubuntu1_amd64.deb ...
Unpacking net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Setting up net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
Processing triggers for man-db (2.9.1-1) ...
bisitaul@bisitaul:~/Downloads$ ifconfig
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 8963 bytes 1039914 (1.0 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8963 bytes 1039914 (1.0 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlo1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.226 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 2600:1700:e9e0:9a10::3b prefixlen 128 scopeid 0x0<global>
```

**Fig: Installing Cuckoo**

Next, we need to setup virtualbox and manage host configuration. Before checking the interface configuration by using command *ifconfig*, we need to execute the following command to install *ifconfig*.

\$ sudo apt install -y net-tools

Check the interface configuration with *ifconfig*:

\$ ifconfig

Create a new interface configuration using following command:

\$ vboxmanage hostonlyif create

\$ vboxmanage hostonlyif ipconfig vboxnet0 --ip 192.168.56.1

\$ ifconfig

```
pranaya@pranaya-VirtualBox: ~/Downloads
RX packets 716  bytes 98919 (98.9 KB)
RX errors 0  dropped 0  overruns 0  frame 0
TX packets 716  bytes 98919 (98.9 KB)
TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

pranaya@pranaya-VirtualBox:~/Downloads$ vboxmanage hostonlyif create
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
Interface 'vboxnet0' was successfully created
pranaya@pranaya-VirtualBox:~/Downloads$ vboxmanage hostonlyif ipconfig vboxnet0
--ip 192.168.56.1
pranaya@pranaya-VirtualBox:~/Downloads$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
    inet6 fe80::63cc:1e88:1f85:a87c  prefixlen 64  scopeid 0x20<link>
    ether 08:00:27:37:2c:b8  txqueuelen 1000  (Ethernet)
    RX packets 205313  bytes 297637531 (297.6 MB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 50908  bytes 4781175 (4.7 MB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0x10<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 825  bytes 111246 (111.2 KB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 825  bytes 111246 (111.2 KB)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

vboxnet0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
    inet 192.168.56.1  netmask 255.255.255.0  broadcast 192.168.56.255
    ether 0a:00:27:00:00:00  txqueuelen 1000  (Ethernet)
    RX packets 0  bytes 0 (0.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 0  bytes 0 (0.0 B)
    TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

pranaya@pranaya-VirtualBox:~/Downloads$
```

**Fig: Network Interface**

```
$ sudo mkdir /opt/systemd/
$ sudo nano /opt/systemd/vboxhostonly //This will open text-editor
```

Inside the vboxhostonly copy and save the following commands. save the file and exit.

```
#!/bin/bash
hostonlyif create
vboxmanage hostonlyif ipconfig vboxnet0 --ip 192.168.56.1
```

Use these commands for network configuration:

```
$ cd /opt/systemd/
$ sudo chmod a+x vboxhostonly
$ sudo touch /etc/systemd/system/vboxhostonlynic.service
$ sudo nano /etc/systemd/system/vboxhostonlynic.service
```

Inside vboxhost onlynic.service copy and paste following lines:

```
Description=Setup VirtualBox Hostonly Adapter
After=vboxdrv.service
[Service]
Type=oneshot
ExecStart=/opt/systemd/vboxhostonly
[Install]
WantedBy=multi-user.target
```

save the file and exit

```
$ systemctl daemon-reload
$ systemctl enable vboxhostonlynic.service
```

### 2.2.3 Installing and setting up virtual machine

Next step is to set up virtualbox and create a virtual machine for windows-7 using windows 7 ISO file.

Simply, type \$ virtualbox to open the virtual box and set up windows 7 as required.

Requirement for windows-7 virtual machine:

Name: cuckoo1 (it can be any)

RAM: 2 GB (or 4 GB)

Hard Disk Space: 20GB (or more)

ISO: windows 7 (64 bit or 32 bit)

Link to Download: <https://tech-latest.com/download-windows-7-iso/#Downloads>

After setting up windows 7 in virtualbox.

Run the command:

```
$ VBoxManage modifyvm cuckoo1 --nested-hw-virt on
```

Now, start the VM and set up Windows 7. Now, we need to edit policies such as security setting, firewall setting, elevation prompt and user account control.

Please follow the following steps to edit the policy:

First, go to the start button and search for edit group policy.

Once you prompt to that follow the steps given below:

1) Windows Settings > Security Settings > Local Policies > Security Options. Scroll down to the User Account Control options >

**Step 1:** Change the security setting of “User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode” to “Elevate without prompting”.

**Step 2:** Change the security settings of “Detect application installations and prompt for elevation” to “Disabled”.

**Step 3:** Change the security settings of “Run all administrators in Admin Approval Mode” to “Disabled”

2) Computer Configuration -> Administrative Templates -> Windows Components -> Windows Update -> Configure Automatic Updates

**Step 1:** Select enabled

**Step 2:** Select Notify for download and notify for install

3) Computer Configuration > Administrative Templates > Network > Network connections > Windows Firewall > Domain Profile > Windows Firewall and change

**Step 1:** Change the security setting of “Protect all network connections” to “Disabled”

4) Computer Configuration > Administrative Templates > Windows Components > Windows Defender Antivirus then

Step 1: Change the setting of “Turn off Windows Defender Antivirus” to “Enabled”.

All the changes that were supposed to be made on Windows 7 are done after that we can move forward to complete our setup process.

### **2.2.4 Install virtualbox guest and setup network configuration**

Now, we need to install virtualbox guest inside the virtualbox that we installed previously. Go to device on the upper tab of virtualbox and select the “Install virtualbox guest edition”. Select run and wait while the process completes.

Download and install these msi on Windows 7

<https://www.python.org/ftp/python/2.7.8/python-2.7.8.amd64.msi>

<https://pypi.python.org/packages/2.7/P/Pillow/Pillow-2.5.3.win-amd64-py2.7.exe#md5=33c3a581ff1538b4f79b4651084090c8>

Now, we will upload the agent.py from ~/.cuckoo/agent to win7, which we will copy to C:\Users\\*USERNAME\*\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup

Reboot the Windows 7 Virtual Machine and sign into it.

Enable the Firewall rule on Windows 7 and minimize command window

On Virtualbox > cuckool select Network and then change Attached to: Host-only Adapter and select vboxnet0

Now we need to go back to our host ubuntu machine and type following command:

```
$ sudo apt-get install -y iptables-persistent
```

Now we need to configure IP forwarding so an internet connection gets routed from the host machine to the guest Virtual Machine. Iptables will be used to set these network forwarding rules:

check the interface configuration. Check whether you have vboxnet0.

```
$ ifconfig
$ sudo iptables -A FORWARD -o ens160 -i vboxnet0 -s 192.168.56.0/24 -m conntrack --
    ctstate NEW -j ACCEPT
$ sudo iptables -A FORWARD -m conntrack --ctstate ESTABLISHED,RELATED -j
ACCEPT
$ sudo iptables -t nat -A POSTROUTING -o ens160 -j MASQUERADE
$ sudo iptables -L
```

Now we need to enable IP forwarding in the kernel so that these settings are set to active

```
$ echo 1 | sudo tee -a /proc/sys/net/ipv4/ip_forward
$ sudo sysctl -w net.ipv4.ip_forward=1
```

```
$ sudo nano /etc/sysctl.conf
```

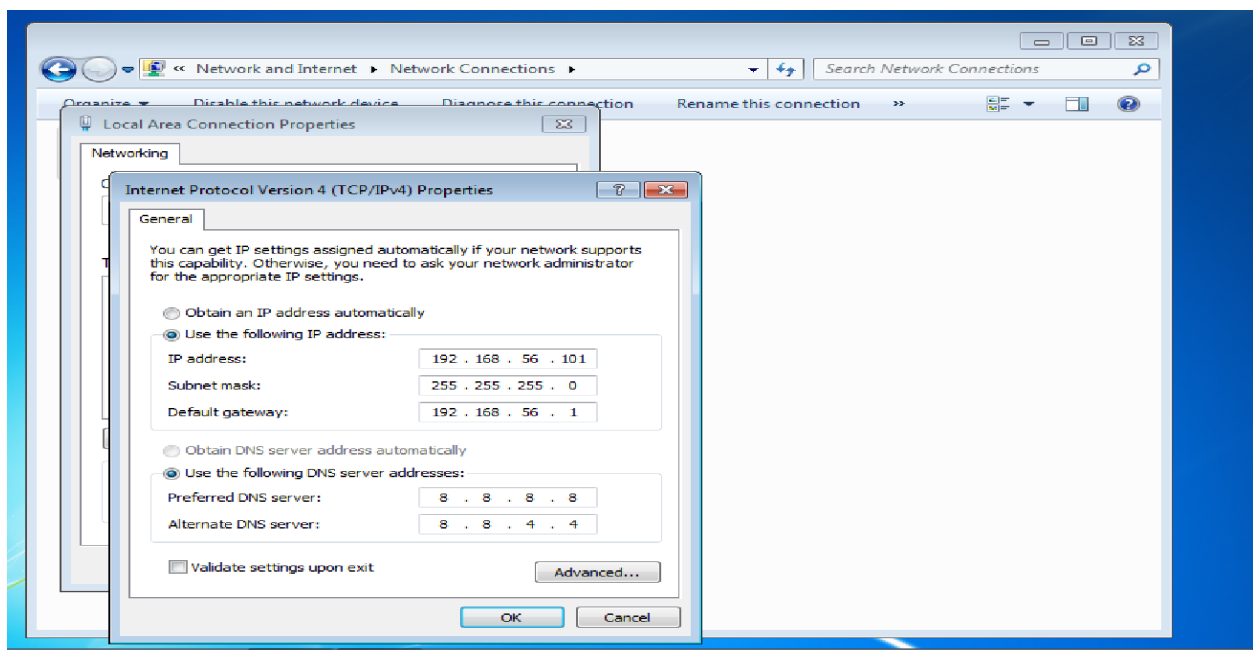
We need to uncomment the following line from sysctl.conf  
from net.ipv4.ip\_forward=1

save the file and exit.

```
$ sudo su
$ iptables-save > /etc/iptables/rules.v4
exit from sudo modus by using exit command.
```

Now we need to configure Win7 network adapter. Go to change network adapter in windows 7 and type the following IP configuration:

```
IP = 192.168.56.101
Subnet = 255.255.255.0
Gateway = 192.168.56.1
DNS = 8.8.8.8 / 8.8.4.4
```

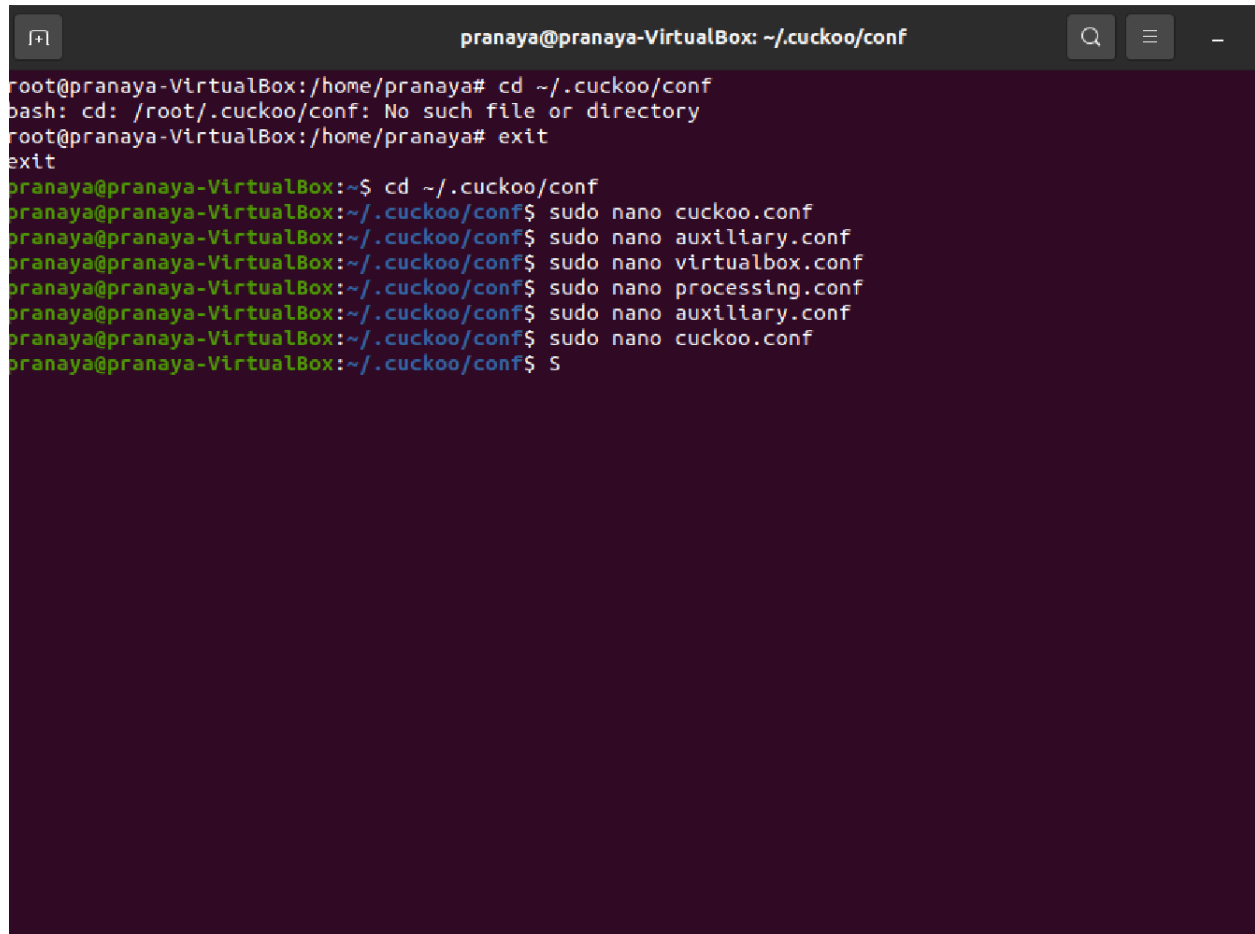


**Fig: IP Configuration on Windows 7**

Go to the machine from the screen and click on create snapshot named snapshot1.

Now, we need to change the following configuration inside cuckoo files (.conf extension).

```
$ cd ~/.cuckoo/conf
```



```
pranaya@pranaya-VirtualBox: ~/.cuckoo/conf
root@pranaya-VirtualBox:/home/pranaya# cd ~/.cuckoo/conf
bash: cd: /root/.cuckoo/conf: No such file or directory
root@pranaya-VirtualBox:/home/pranaya# exit
exit
pranaya@pranaya-VirtualBox:~$ cd ~/.cuckoo/conf
pranaya@pranaya-VirtualBox:~/.cuckoo/conf$ sudo nano cuckoo.conf
pranaya@pranaya-VirtualBox:~/.cuckoo/conf$ sudo nano auxiliary.conf
pranaya@pranaya-VirtualBox:~/.cuckoo/conf$ sudo nano virtualbox.conf
pranaya@pranaya-VirtualBox:~/.cuckoo/conf$ sudo nano processing.conf
pranaya@pranaya-VirtualBox:~/.cuckoo/conf$ sudo nano auxiliary.conf
pranaya@pranaya-VirtualBox:~/.cuckoo/conf$ sudo nano cuckoo.conf
pranaya@pranaya-VirtualBox:~/.cuckoo/conf$ S
```

**Fig: Files name need to modify**

```
$ sudo nano cuckoo.conf
```

Find and Set the following changes inside cuckoo.conf

```
machinery = virtualbox
```

```
memory_dump = yes
```

```
ip = 192.168.56.1
```

save the file and exit

```
$ sudo nano auxiliary.conf
```

```
sniffer enabled = yes
```

save the file and exit

```
$ sudo nano virtualbox.conf
```

```
virtualbox mode = gui
```

```
machines = cuckoo1
```

```
ip = 192.168.56.101
```

```
snapshot = Snapshot1
```

save the file and exit

```
$ sudo nano processing.conf
memory enabled = yes then save the file and exit
```

```
$ sudo nano memory.conf
basic guest_profile = Win7SP1x64
save the file and exit
```

```
$ sudo nano reporting.conf
enabled = no
Report.html enabled = yes
mongodb enabled = yes
save the file and exit
```

### 3. Run Cuckoo Web Server

Now, we are almost done with setting up cuckoo.

Shutdown Windows 7 and then reboot Ubuntu Virtual Machine.

Type following command to run cuckoo.

```
$ cuckoo community
```

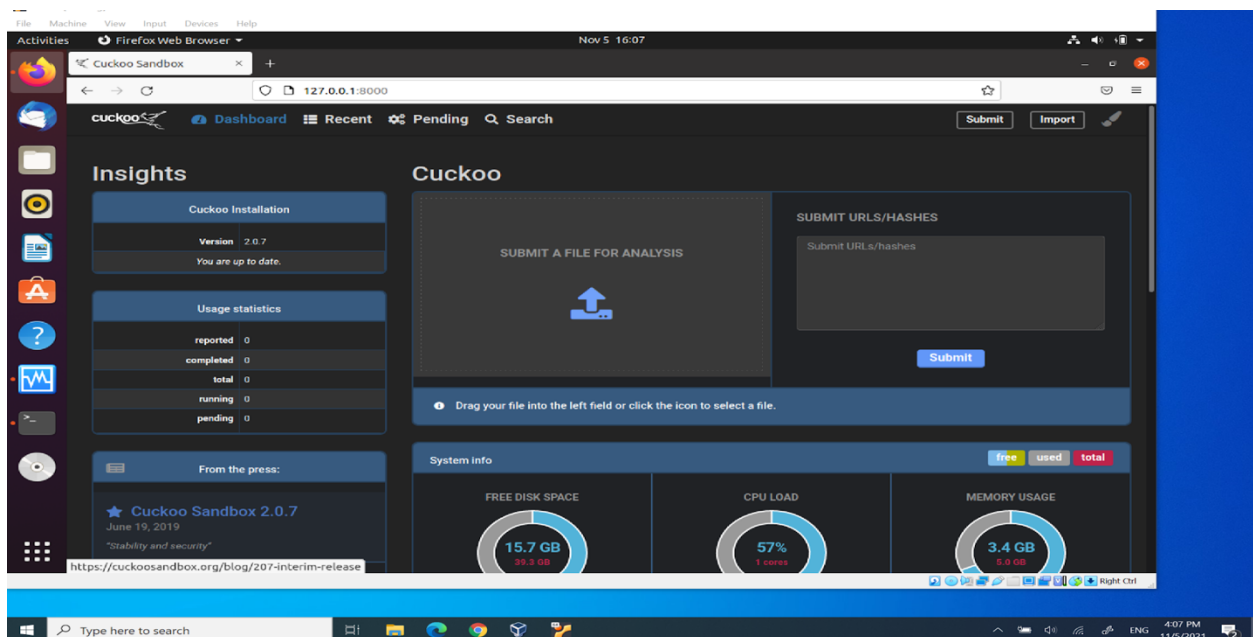
```
$ cuckoo
```

Finally, open start your cuckoo1 Virtual Machine and minimize the command window

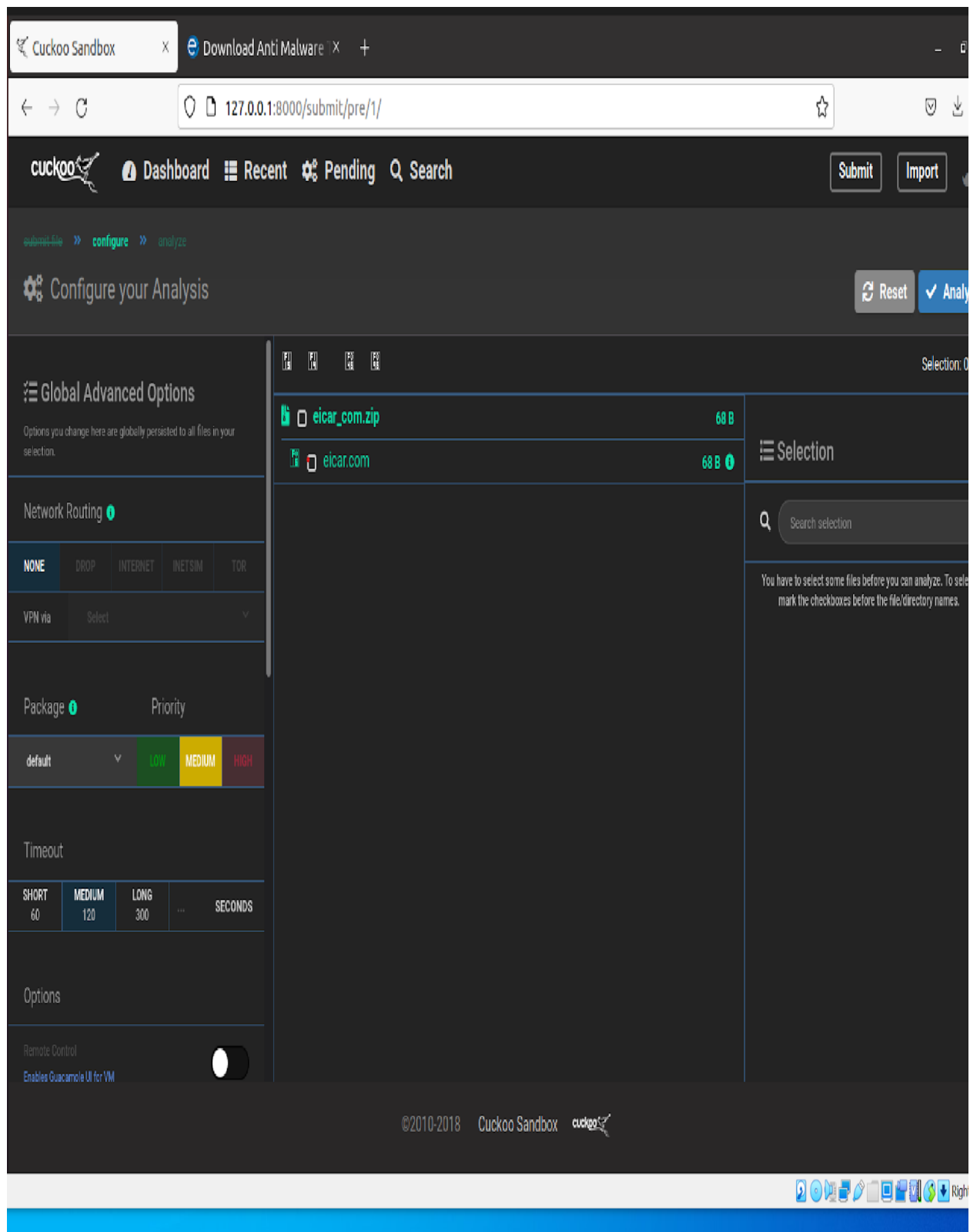
Open another terminal and type this command:

```
$ cuckoo web runserver 0.0.0.0:8000
```

Now, we can go to host browser and visit Cuckoo Web Server through: 127.0.0.1:8000 or click the link which we get after running the web server command.



**Fig: Cuckoo Web Interface**



**Fig: Uploading eicar malware on Cuckoo Interface**



## 4. Outputs

After analysis of malware, we have the reports which include a snapshot during the process, memory\_dump, analysis\_log, cuckoo\_log, etc.

Command to view the analysis log:

```
$ cd ~/.cuckoo/storage/analyses/
```

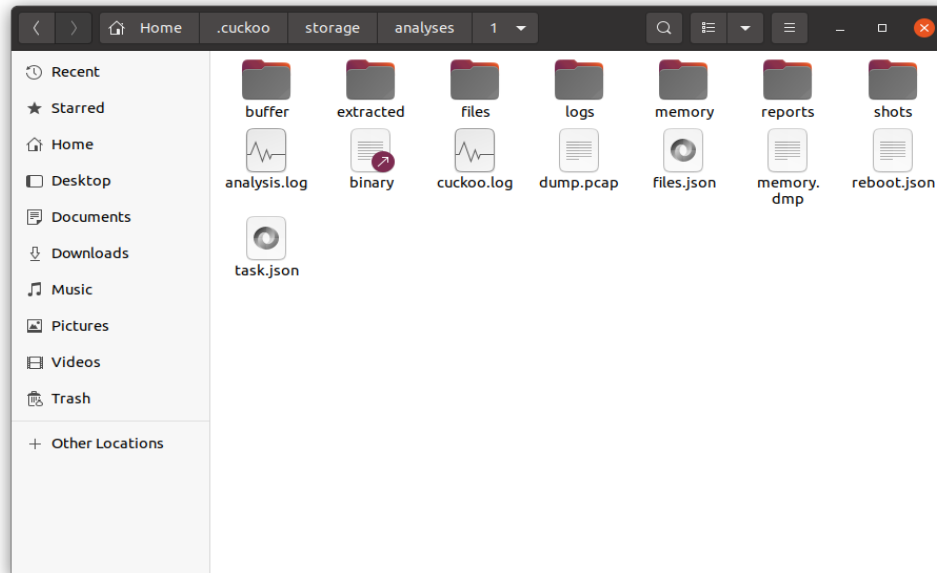


Fig: Output file of analysis report

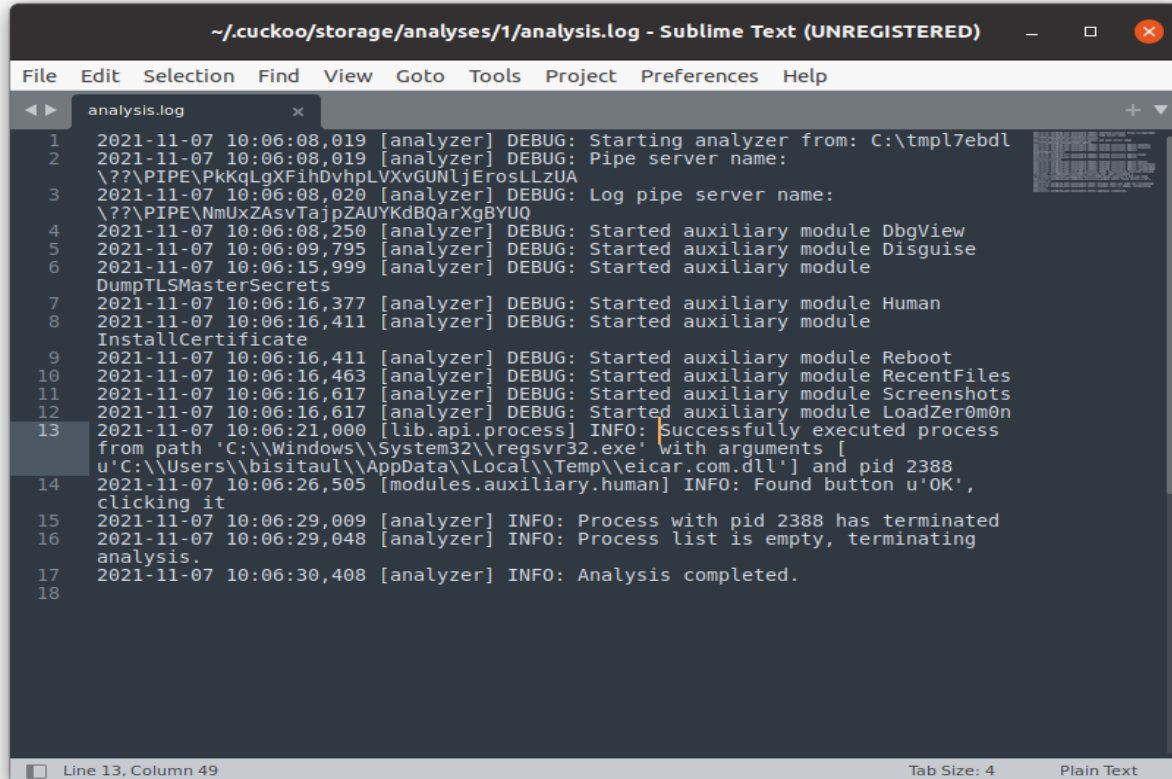
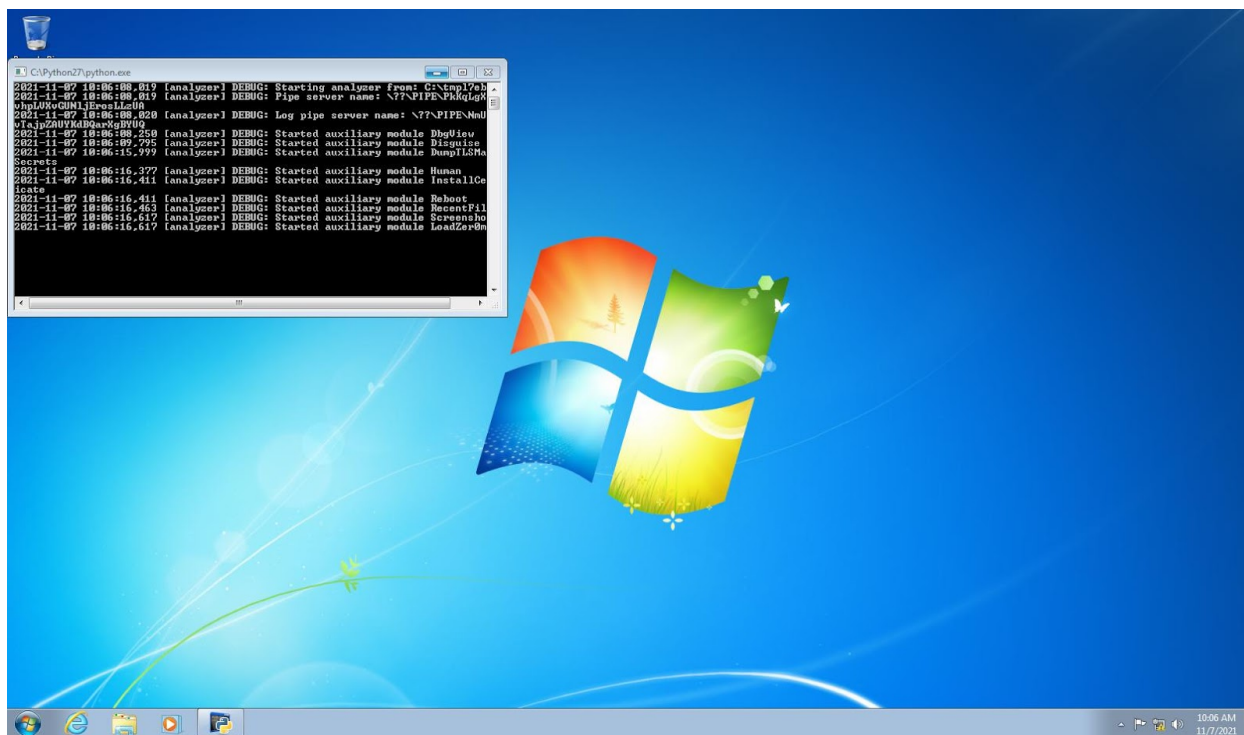
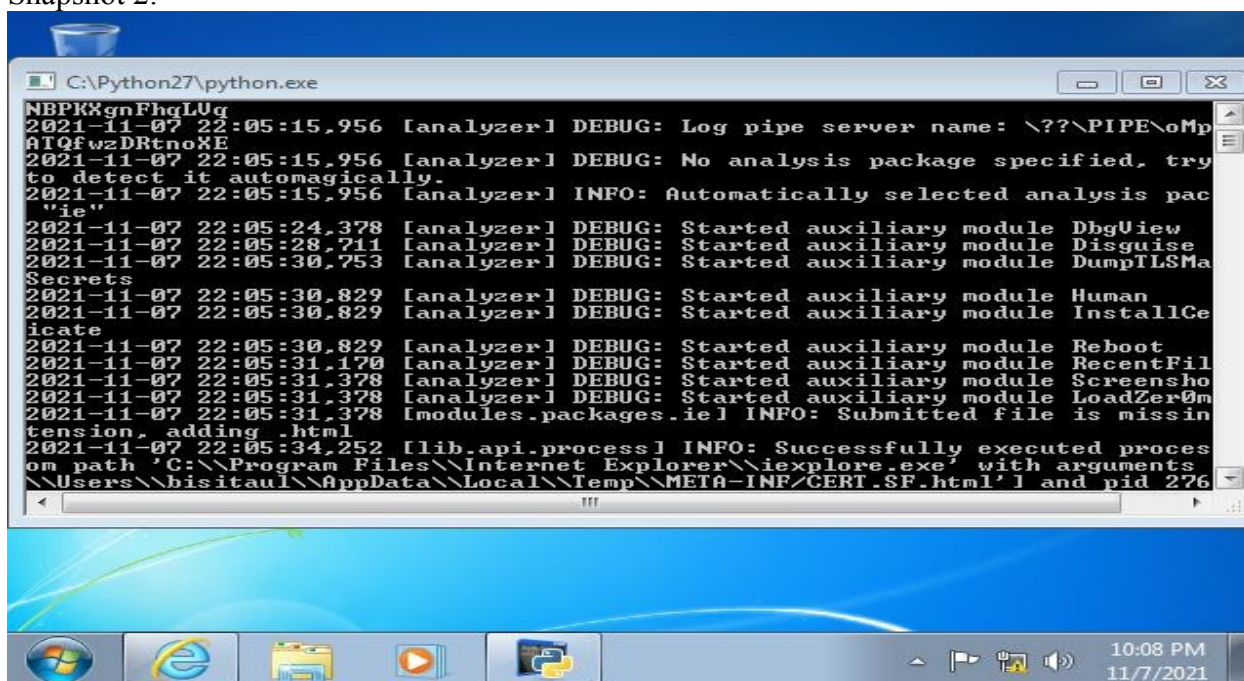


Fig: Screenshot of Analysis completed

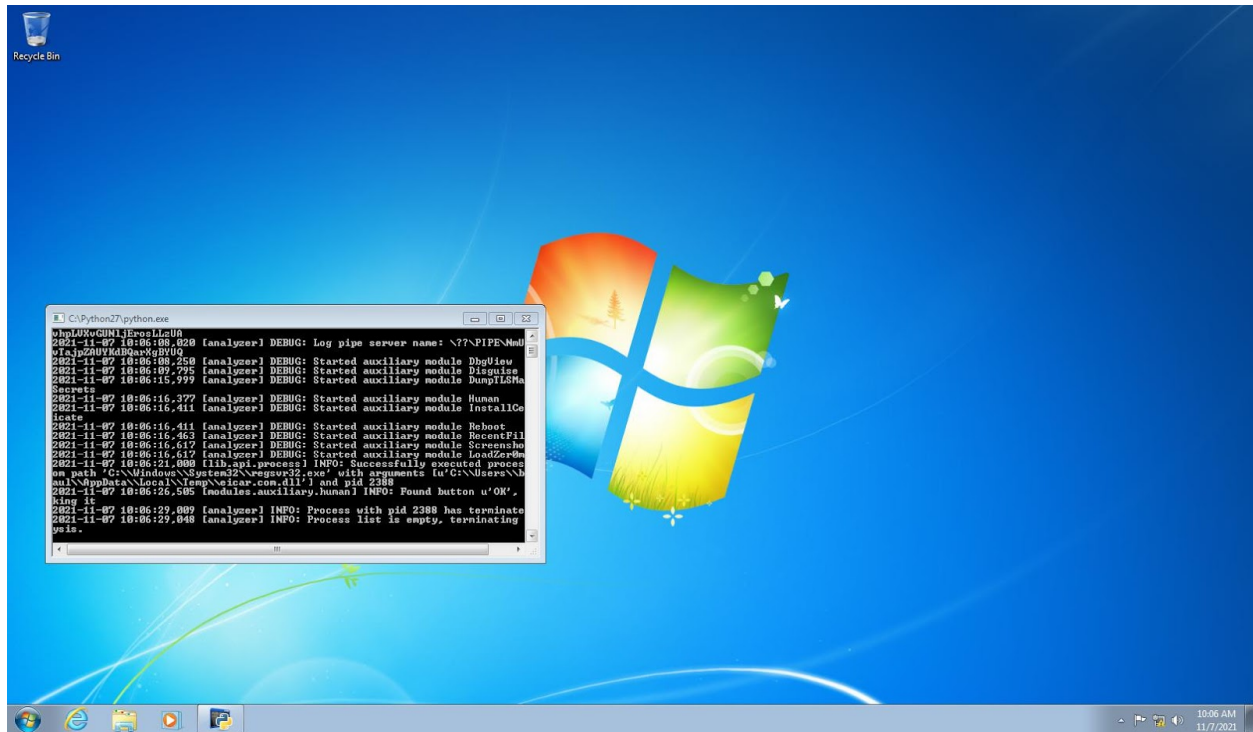
Here are the snapshots that windows take it automatically while analysis the malware file:  
Snapshot 1:



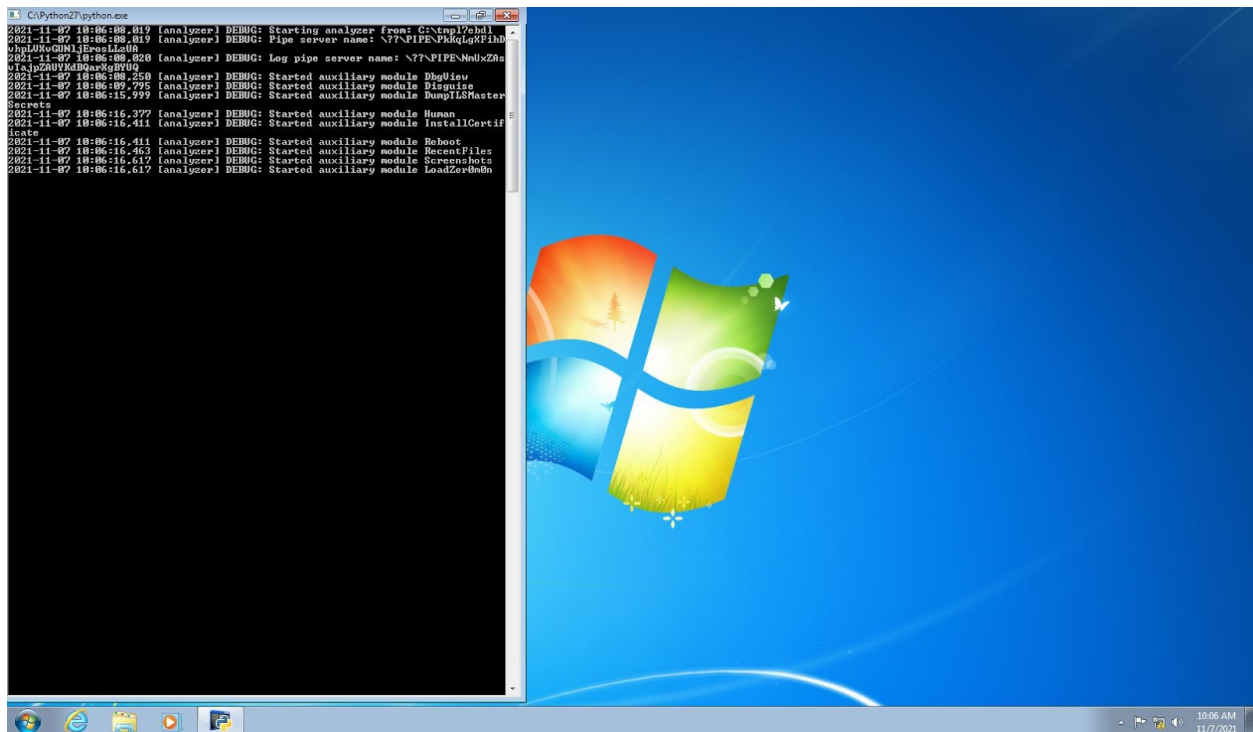
Snapshot 2:



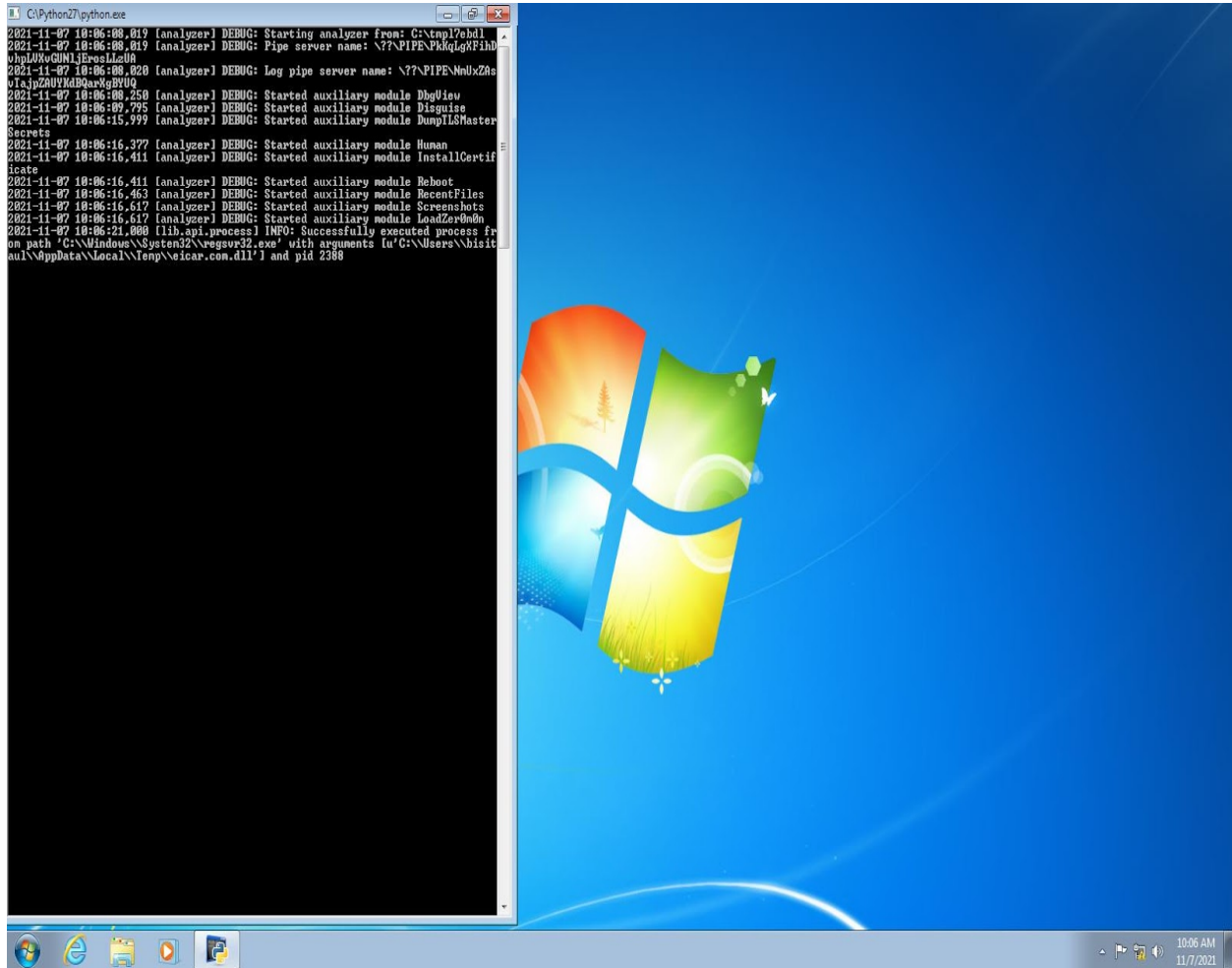
Snapshot 4:



Snapshot 5:



## Snapshot 6:



## 6. Conclusion

Cuckoo sandbox is running and working fine after successfully installing and setting the above commands. It is very important to follow each and every step and command without any errors. Now, we can analyze malware in a fully functional cuckoo sandbox either by using a terminal or web browser.

## 6. Problem faced

Installing cuckoo is not a simple task as there are many commands to be followed without missing any steps to perform it well. We need a powerful laptop or desktop to perform this task. Otherwise, it will crash frequently which makes us re-do it again and again. So, we would like to suggest to whoever plans to install this please make sure that your PC is powerful in order to make cuckoo fully functional without any issues.

### Reference

1. Written by Alwin Peppels and Ricardo van Zutphen. "Hatching - Automated Malware Analysis Solutions." *Cuckoo Sandbox Setup for People in a Hurry*, <https://hatching.io/blog/cuckoo-sandbox-setup/>.
2. "Requirements¶." *Requirements - Cuckoo Sandbox v2.0.7 Book*, <https://cuckoo.readthedocs.io/en/latest/installation/host/requirements/>.
3. "Automated Malware Analysis." *Cuckoo Sandbox - Automated Malware Analysis*, <https://cuckoosandbox.org/>.
4. Bridewell Consulting, director. *Bridewell Consulting*, 29 July 2021, <https://www.bridewellconsulting.com/automating-malware-analysis-with-cuckoo-sandbox>. Accessed 9 Nov. 2021.
5. "Cuckoo Installation on Ubuntu 20." *Utopian Cyber Knight*, 4 Sept. 2020, <https://utopianknight.com/malware/cuckoo-installation-on-ubuntu-20/>.