Written by: Aaron Jang and Andrei Guevorkian

Github: **https://github.com/aaronhhjang/MalwareClassification**

Our setup steps closely followed the official Cuckoo Sandbox steps (https://cuckoo.sh/docs/installation/host/requirements.html), as well as the following Youtube series (don't watch other videos about installing and configuring Cuckoo):

- https://www.youtube.com/watch?v=x7X9DHHxO3I Cuckoo Installation Part 1 | 2018 (Download and Install Cuckoo)
- https://www.youtube.com/watch?v=lYDnp26dXG8 Cuckoo Installation Part 2 | 2018 (Guest and network configuration)
- https://www.youtube.com/watch?v=Ur8ui2HIi8k Cuckoo Installation Final Part | 2018 (Cuckoo Configuration & Malware Analysis)

The following video gives an overview of Cuckoo Sandbox, as well as a little demonstration of running a malware: https://www.youtube.com/watch?v=V4z2tLRCuIY

==========

Minimum Required Specs for VMs

Ubuntu: 18.04.5 https://releases.ubuntu.com/18.04/
- RAM: 8GB, if possible 16GB
- DISK: 150GB, if possible 300GB (especially if there is a lot of malware to analyze)
- Name of the machine: cuckoo

Windows7 (ask from the lab):
- RAM: 4 GB, if possible 8GB
- DISK: 50GB
- Name of the machine: cuckoo1

ENVIRONMENT: ***cuckoo*** (ubuntu)
Commands to run: (volatility is an optional tool to do forensic analysis on memory dumps)

sudo apt-get update
sudo apt-get install git -y
sudo apt-get install python python-pip python-dev libffi-dev libssl-dev -y
sudo apt-get install python-virtualenv python-setuptools -y
sudo apt-get install libjpeg-dev zlib1g-dev swig -y
sudo apt-get install mongodb -y
sudo apt-get install postgresql libpq-dev -y
sudo apt install virtualbox -y

sudo apt-get install tcpdump apparmor-utils -y
sudo aa-disable /usr/sbin/tcpdump
sudo groupadd pcap
sudo usermod -a -G pcap cuckoo
sudo chgrp pcap /usr/sbin/tcpdump
sudo setcap cap_net_raw,cap_net_admin=eip /usr/sbin/tcpdump
sudo git clone https://github.com/volatilityfoundation/volatility
sudo apt-get install swig
sudo pip install m2crypto
sudo usermod -a -G vboxusers cuckoo
sudo pip install -U pip setuptools
sudo pip install -U cuckoo
sudo mkdir /opt/cuckoo
sudo chown cuckoo:cuckoo /opt/cuckoo
cuckoo --cwd /opt/cuckoo
sudo apt install net-tools
sudo apt-get install virtualbox-guest-utils
sudo mount -t vboxsf sharedfolder ~/Desktop/

Next, open VirtualBox and create inner VM (named ***cuckoo1*** (Windows))
vboxmanage hostonlyif create
vboxmanage hostonlyif ipconfig vboxnet0 --ip 192.168.56.1
sudo iptables -A FORWARD -o ens32 -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 -A POSTROUTING -t nat -j MASQUERADE
sudo su
echo 1 > /proc/sys/net/ipv4/ip_forward
sudo sysctl -w net.ipv4.ip_forward=1
exit

Set up the internet configuration (we think we used NAT for both *cuckoo1 and cuckoo*.
Install google chrome, install python 2.7, and Python Pillow (https://pypi.org/project/Pillow/#files） (all shown in the video part 2).

Restart the computer, and mount the Cuckoo agent (found in /opt/cuckoo).
Disable Windows Firewall.
While having agent.py running, take a snapshot of *cuckoo1.*

Next, video 3 of the playlist shows how to submit malware through the GUI interface. Here are the two commands needed to submit (1 or many malware through the CLI):

Cmd in terminal	*comments/ description*
`$ cuckoo -d`	*start cuckoo in TERMINAL1* and keep it running.
`$ cuckoo submit '<path for the directory>'`	*Open TERMINAL2. Execute the submitted file*
`$ cd /opt/cuckoo/storage/analyses/<##>`	*Go to the directory that contains the "reports"*
`$ xdg-open .`	*Open current directory with file explorer*

NOTES BEFORE RUNNING MALWARE:

Things to double check before running everything at once (script):

- Make the environment look as real as possible (win7 honyepot). Add adobe, microsoft word, java, etc. (https://cuckoo.readthedocs.io/en/latest/installation/guest/requirements/)
- Make sure virtualbox guest addition is disabled (which was used to share files between host/guest)
- Remove any hint that malware is being run in a virtual environment (https://cuckoo.readthedocs.io/en/latest/introduction/sandboxing/)
  Cuckoo has already set up some anti-virtualization detection techniques (such as moving and clicking the mouse), but we can do better(?)

SOME COMMON PROBLEMS ENCOUNTERED:

Everytime we shut down the Linux VM and come back, we have to reconfigure the network adapter through hostonlyif config command:
vboxmanage hostonlyif create
vboxmanage hostonlyif ipconfig vboxnet0 --ip 192.168.56.1

We would come sometimes be confronted by the problem where we would have cuckoo running, but when we submit a file, cuckoo tells us “cuckoo1: not ready yet”, and when you open up VBox manually, instead of having the final state in “ready”, as it was last left at, it now displays that the state is “aborted”. We tried restoring previous states of the VM and trying to run those, but those would abort as well. This told us that there might be some internal issue, possibly regarding RAM, or the lack thereof. Simply restarting the outer VM (linux) did the trick.

$ fuser -k 8000/tcp ← In case port 8000 is open (for submissions through the web interface), kill it to allow file submissions through the terminal

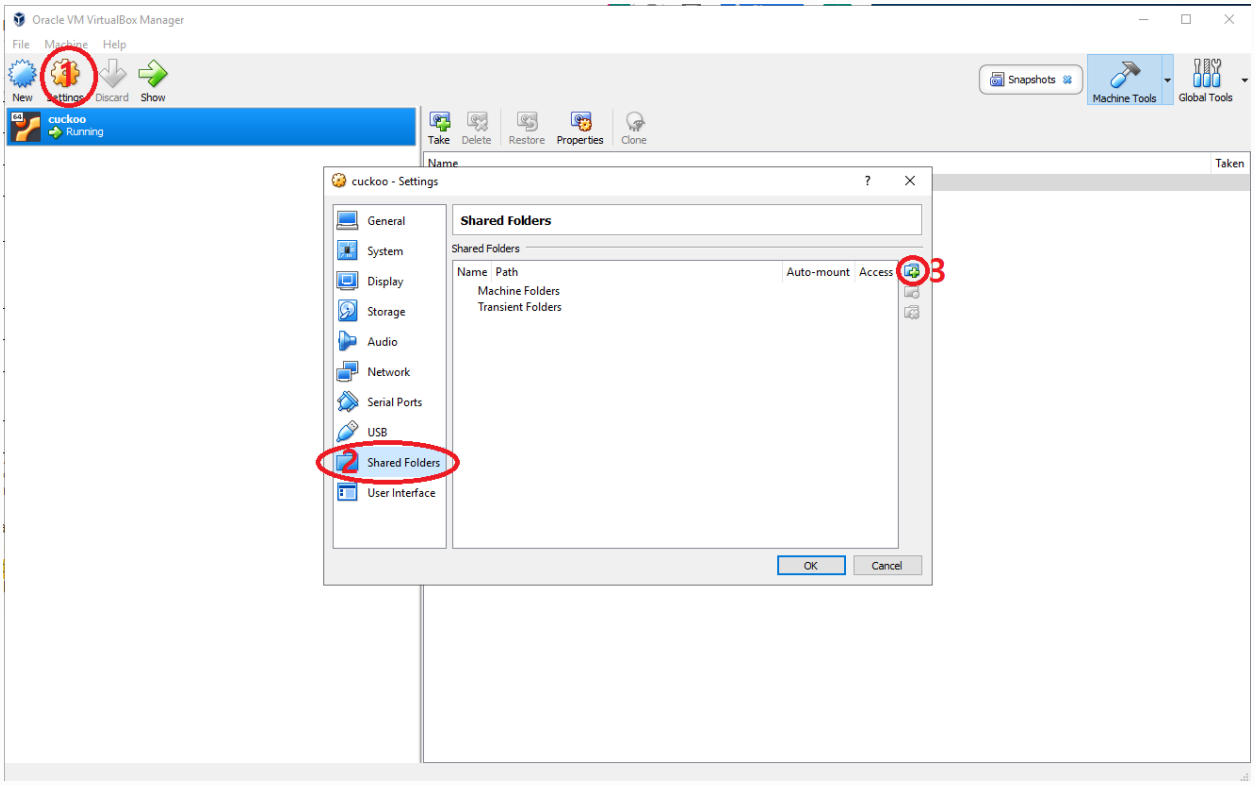
==========

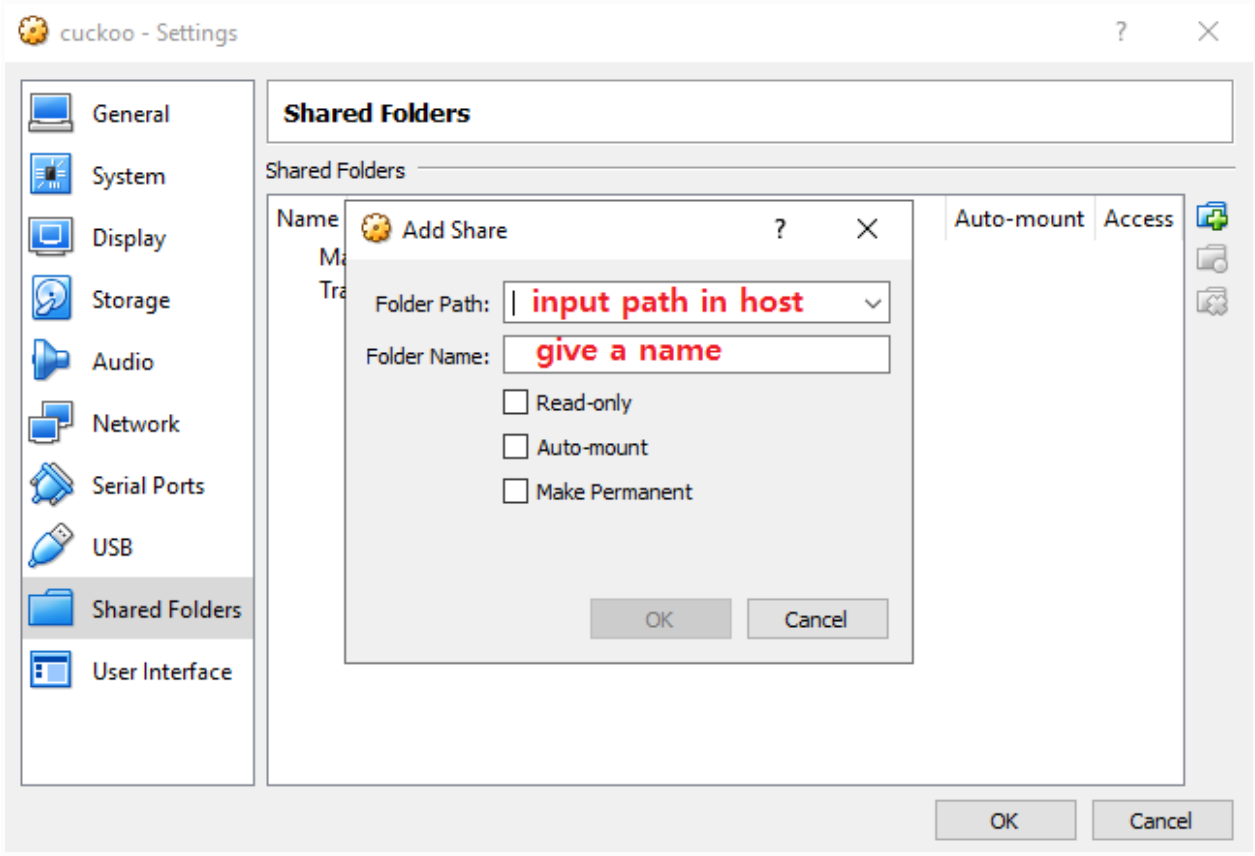
Here is explanation of all directories under results:
https://cuckoo.readthedocs.io/en/latest/usage/results/

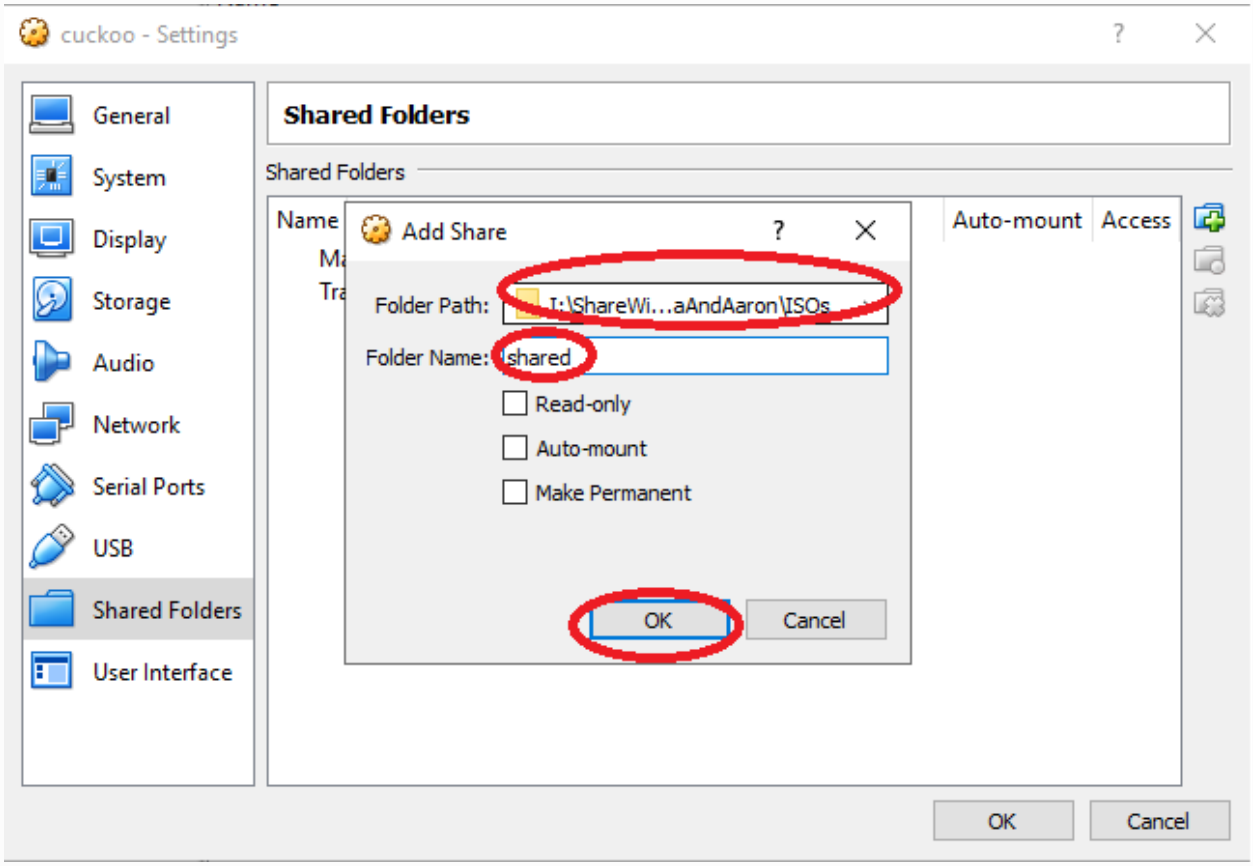
From here can disable the creation of certain reports (to save space):
https://cuckoo.readthedocs.io/en/latest/installation/host/configuration/#reporting-conf

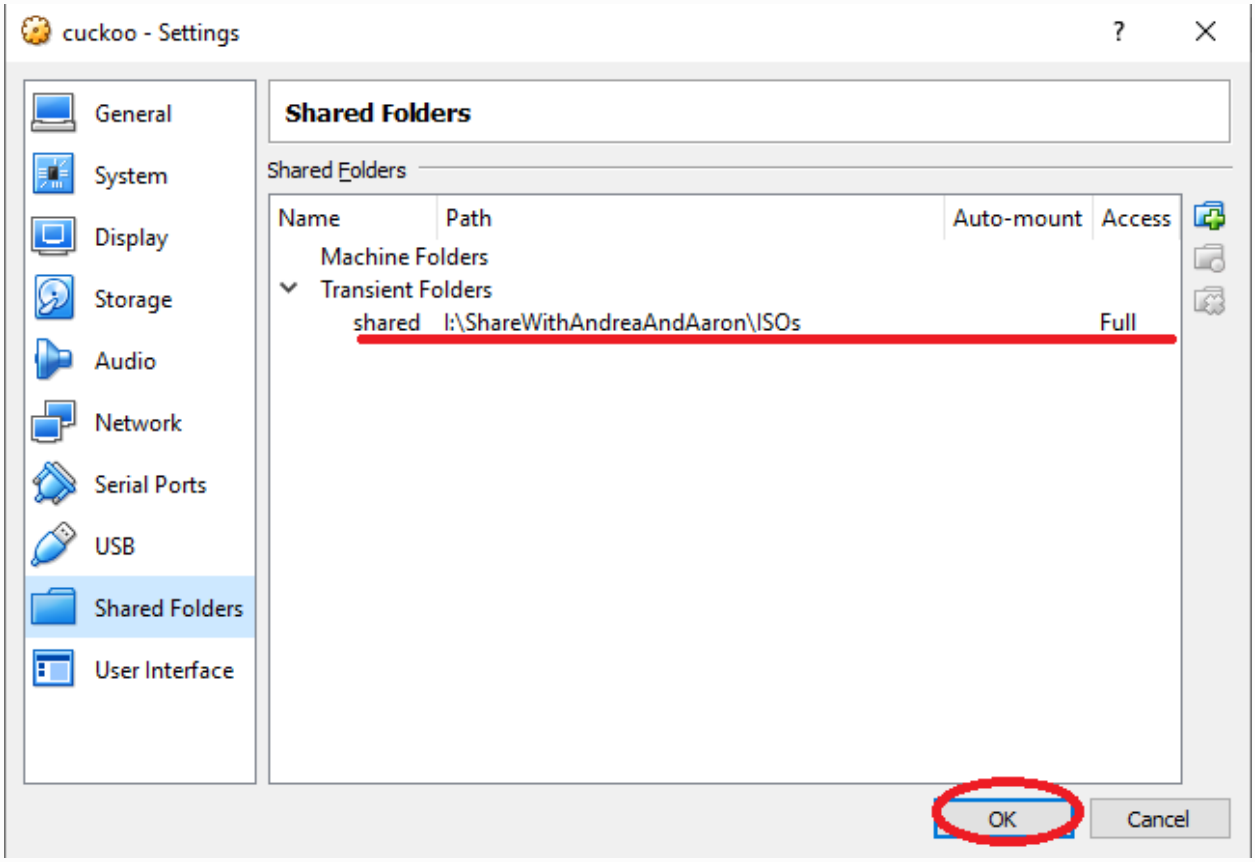
TO GET VBOX GUEST ADDITION:
sudo apt-get install virtualbox-guest-utils
sudo mount -t vboxsf HOST_SHARE GUEST_SHARE
AFTER SETTING INNER VM TO NAT (DNS server not responding):
https://www.youtube.com/watch?v=JMOOayRpII4&ab_channel=TechFixIT

Setting up folder sharing between host and guest:




cuckoo - Settings
General
System
Display
Storage
Audio
Network
Serial Ports
USB
Shared Folders
User Interface
Shared Folders
Shared Folders
Name
Auto-mount
Access
Add Share
Folder Path:
input path in host
Folder Name:
give a name
Read-only
Auto-mount
Make Permanent
OK
Cancel
OK
Cancel


cuckoo - Settings
General
System
Display
Storage
Audio
Network
Serial Ports
USB
Shared Folders
User Interface
Shared Folders
Shared Folders
Name
Auto-mount
Access
Add Share
Folder Path:
I:\ShareWi...aAndAaron\ISOs
Folder Name:
shared
Read-only
Auto-mount
Make Permanent
OK
Cancel
OK
Cancel


cuckoo - Settings
General
System
Display
Storage
Audio
Network
Serial Ports
USB
Shared Folders
User Interface
Shared Folders
Shared Folders
Name
Path
Auto-mount
Access
Machine Folders
Transient Folders
shared
I:\ShareWithAndreaAndAaron\ISOs
Full
OK
Cancel