

CS 455 – Computer Security Fundamentals

Dr. Chen-Yeou (Charles) Yu

- More about the Vulnerability
- Introduction to Kali Linux
- Install Kali Linux
- Hands-on for Kali Linux
- Let's do some Footprinting job. Shall we? ^_^

More about the Vulnerability

- Not all the systems are perfect
- This is not limited to the computer systems.
 - It could be anything, including your **peripherals**. For example, any hardware that are driven by firmware.
 - Your camera may betray you by sending your picture to someone
 - Your printer might get a Trojan so someone knows what are the documents you are printing
 - Even worse, are you sure the WiFi AP in your house is only used by you and your roommate?
 - Checkout this crazy hacker's video in the YouTube (how Hackers crack any WiFi password?! set strong WiFi password now!)
 - <https://www.youtube.com/watch?v=QGzTCL1KkeY>
 - I don't care how he cracked the password but one thing caught our eyes. Yes! The system! What is the OS this guy using!? Kali Linux!

Introduction to Kali Linux

- Kali Linux is a Debian-derived Linux distribution designed for digital forensics and penetration testing
- We can overly simplified that as a Debian Linux + tons of computer security related tool
- Getting Excited? Let's see how to install Kali Linux
- We will use this as our working environment for a while. Since this is the early weeks, let's get our hands dirty

Install Kali Linux

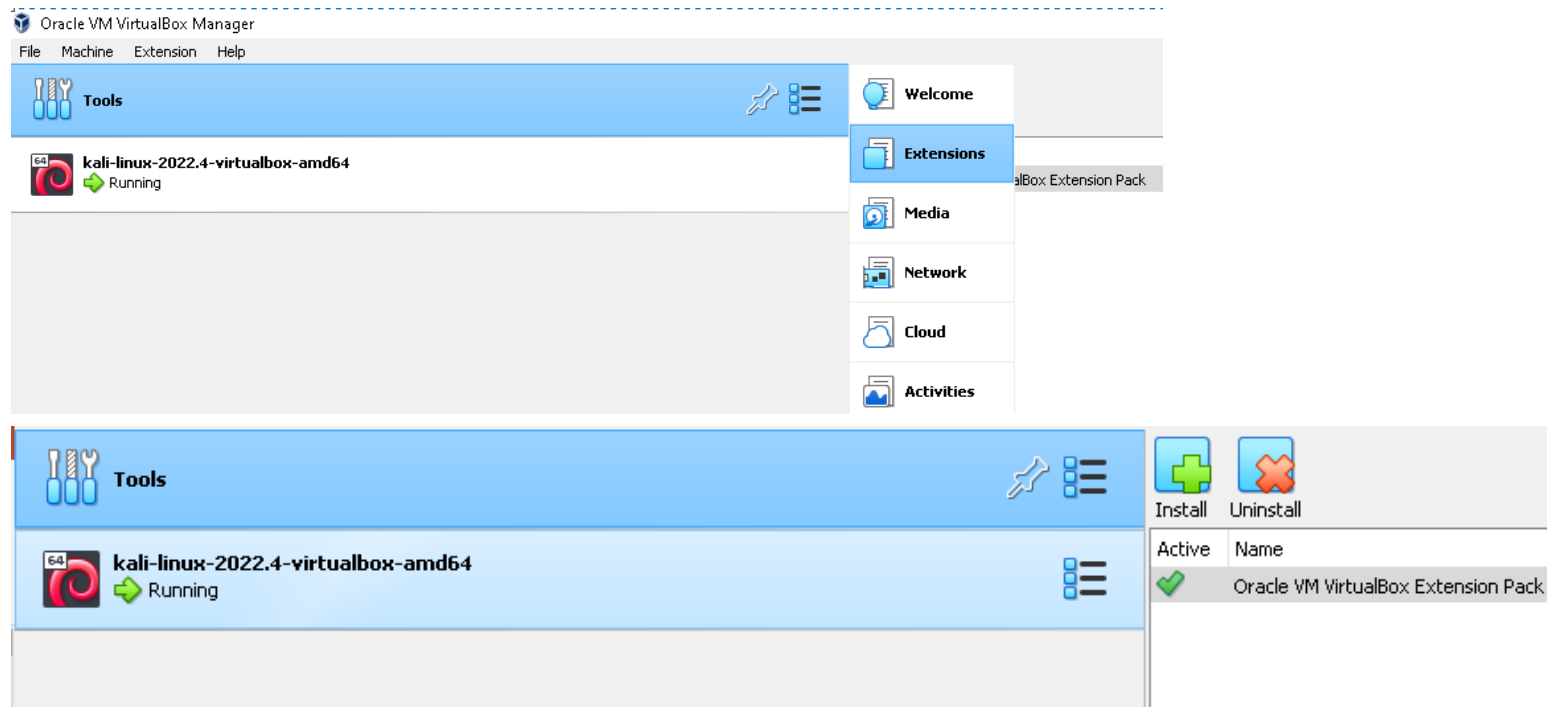
- The way to install and configure Kali Linux are many
- If you have a spare laptop / desktop, you can use download the system image and flash into a USB drive. Then, you can boot from the USB drive and follow the regular installation
- If you have a Raspberry Pi (ARM based), the Kali Linux distribution image supports Raspberry Pi zero, 1, 2, 3 and 4!
- For Intel platform laptop / desktop, it has 32 bit and 64 bit versions
- For mobile phones, you can even install Kali Linux onto your Android phones!
- If you want, you can even setup a dual boot to have your Windows co-existing with Kali

Install Kali Linux

- The most easiest and simple way is to use the VirtualBox to install Kali into the virtual machine. (I want it to co-exist with my Windows 10)
- Kali not only supports VirtualBox but also Vmware, both of the virtual machine
- I give this demo by using (downloading) the following:
 - VirtualBox 7.0.6 platform packages for Windows Hosts (VirtualBox-7.0.6-155176-Win.exe) ← main VirtualBox
 - Oracle_VM_VirtualBox_Extension_Pack-7.0.6a-155176 (enable the system tweaks in virtual machine)
 - kali-linux-2022.4-virtualbox-amd64.7z ← This is compressed a 7-Zip file, you can either decompress that by using WinRAR or 7-Zip
 - VC_redist.x64.exe ← Visual C++ 2019 redistributable (This is needed by the installation process of main VirtualBox)

Install Kali Linux

- Before installing the VirtualBox, you need to install Visual C++ 2019 redistributable first
- When finishing, you will need to install the Extension pack in the VirtualBox



Install Kali Linux (choose the correct image)

kali.org/get-kali/#kali-platforms


Agents Truman State Unive... COMPSAC THE FIRST IEEE SERV... Game Theory guillaume-chevalier... Non-Convex region US_VISA jaungiers/LSTM-Ne... LaTeX Sequence Classifica... IEEE Cloud 2020 Multi-Class Text Cla... Docker Other bookmarks

KALI

GET KALI BLOG DOCUMENTATION COMMUNITY COURSES DEVELOPERS ABOUT

Choose **your** Kali |


LIGHT ☒ DARK



ARM

- ✓ Range of hardware from the leave-behind devices end to high-end modern servers
- ✗ System architecture limits certain packages
- ✗ Not always customized kernel

Works on relatively inexpensive & low powered Single Board Computers (SBCs) as well as modern ARM based laptops, which combine high speed with long battery life.




Installer Images

- ✓ Direct access to hardware
- ✓ Customized Kali kernel
- ✓ No overhead

Single or multiple boot Kali, giving you complete control over the hardware access (perfect for in-built Wi-Fi and GPU), enabling the best performance.

Recommended




Virtual Machines

- ✓ Snapshots functionality
- ✓ Isolated environment
- ✓ Customized Kali kernel
- ✗ Limited direct access to hardware
- ✗ Higher system requirements

VMware & VirtualBox pre-built images. Allowing for a Kali install without altering the host OS with additional features such as snapshots. Vagrant images for quick spin-up also available.


Recommended



Mobile

- ✓ Kali layered on Android
- ✓ Kali in your pocket, on the go
- ✓ Mobile interface (compact view)


A mobile penetration testing platform for Android devices, based on Kali Linux. Kali NetHunter consists of an NetHunter App, App Store, Kali Container, and KeX.



Cloud

- ✓ Fast deployment
- ✓ Can leverage provider's resources
- ✗ Provider may become costly
- ✗ Not always customized kernel


Hosting providers which have Kali Linux pre-installed, ready to go, without worrying about infrastructure maintenance.



Containers

- ✓ Low overhead to access Kali toolset
- ✗ Userland actions only
- ✗ Not Kali customized kernel
- ✗ No direct access to hardware


Using Docker or LXD, allows for extremely quick and easy access to Kali's tool set without the overhead of an isolated virtual machine.



Live Boot

- ✓ Un-altered host system
- ✓ Direct access to hardware
- ✓ Customized Kali kernel
- ✗ Performance decrease when heavy I/O

Quick and easy access to a full Kali install. Your Kali, always with you, without altering the host OS, plus allows you to benefit from hardware access.



WSL

- ✓ Access to the Kali toolset through the WSL framework
- ✗ Userland actions only
- ✗ Not Kali customized kernel
- ✗ No direct access to hardware

Windows Subsystem for Linux (WSL) is included out of the box with modern Windows. Use Kali (and Win-KeX) without installing additional software.

Install Kali Linux



Prebuilt Virtual Machines

Kali Linux [VMware](#) & [VirtualBox](#) images are available for users who prefer, or whose specific needs require a virtual machine installation.

These images have the [default credentials](#) "kali/kali".

[Virtual Machines Documentation](#) >

64-bit

32-bit



VMware



VirtualBox



QEMU



2.6G

[torrent](#)

[docs](#)

[sum](#)

Recommended



2.6G

[torrent](#)

[docs](#)

[sum](#)

Recommended



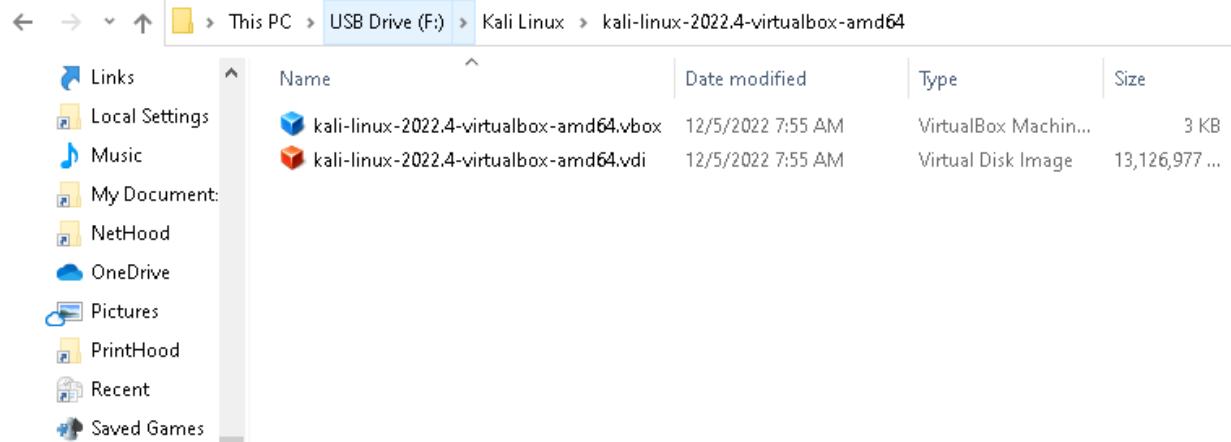
2.6G

[torrent](#)

[docs](#)

[sum](#)

Recommended

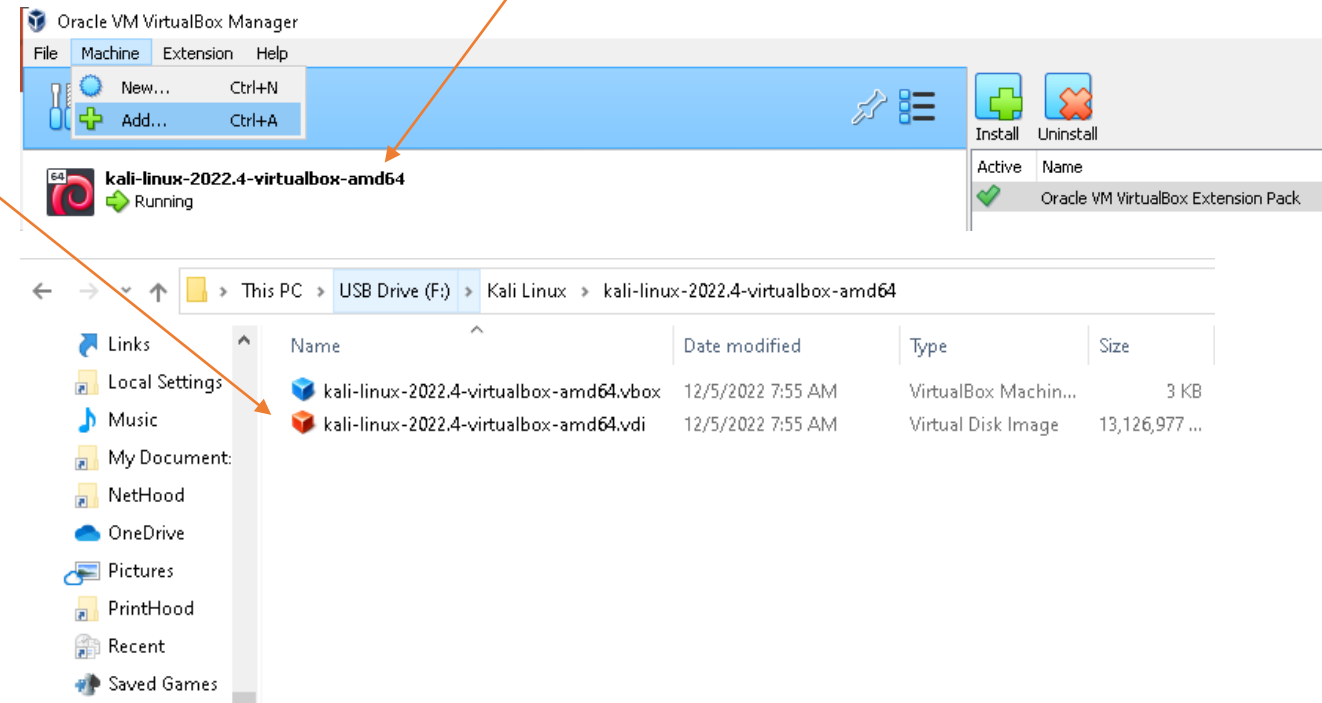


When you finished the download and decompress the .7z file, you will find out these 2 files in the folder

The .vdi file, you will need to open your VirtualBox to add the image as a new machine

Install Kali Linux

- Go back to VirtualBox by adding the .vdi file, the system image into VirtualBox. In my example, you will see the system image show up eventually.
- **Machine** → **Add** → Browse the local folder and choose this image

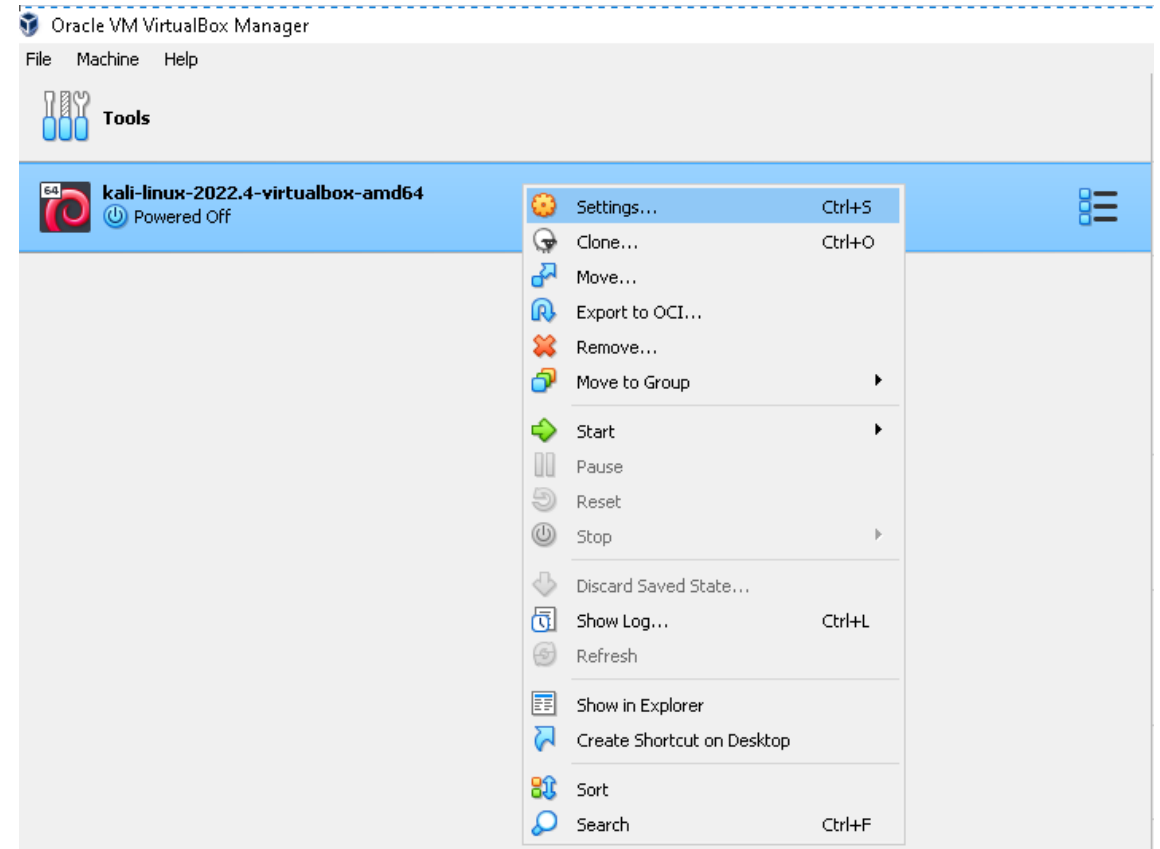


Install Kali Linux

- As you can see, I downloaded the image to the USB drive. However, I copied the image to the hard drive (SSD) then execute the previous step.
- I don't like to directly execute the image from the USB even though it is possible. Because it is too slow!

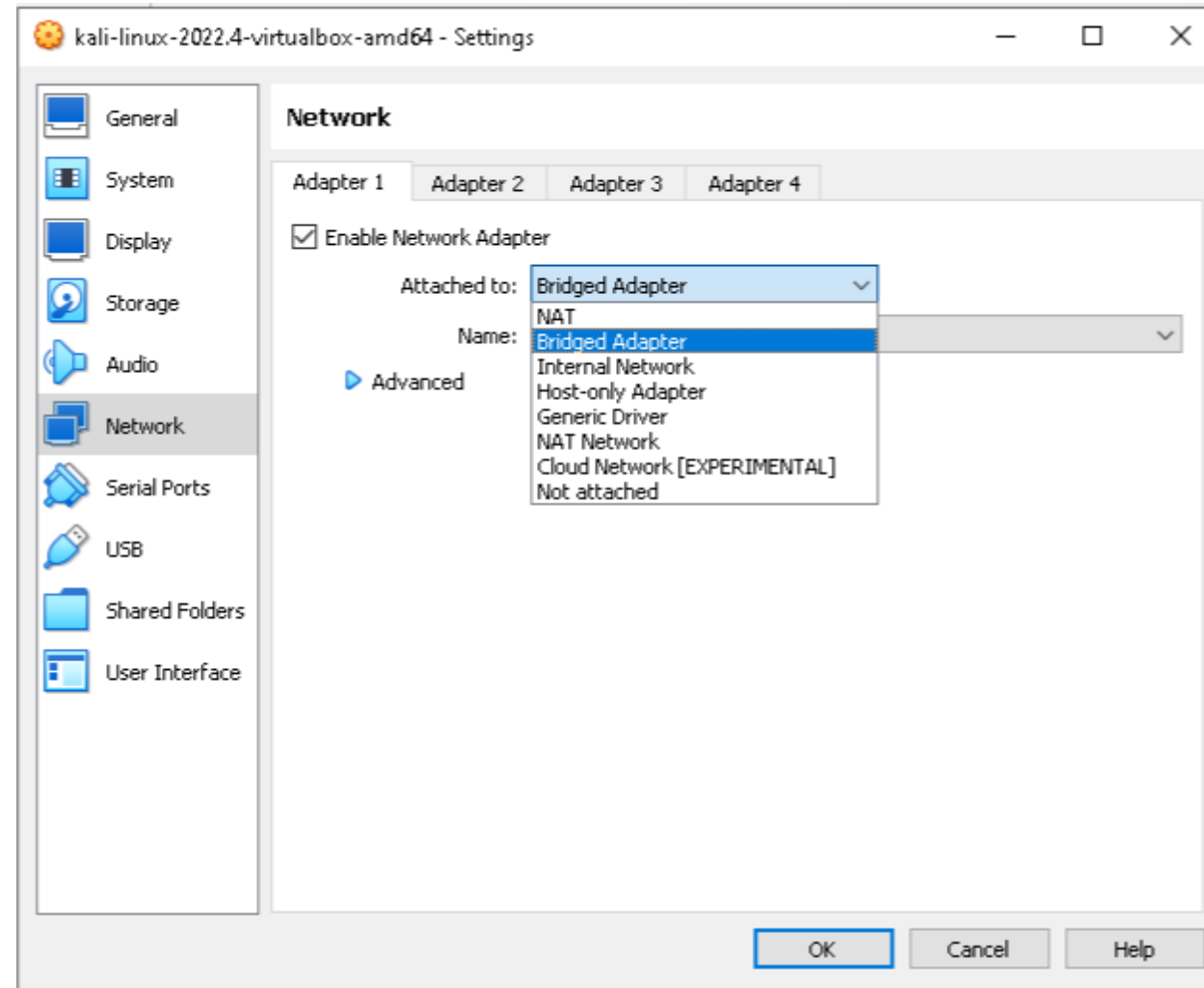
Install Kali Linux

- Go back to the VirtualBox, there are still some jobs we need to do the in the setup.
- Right click the virtual machine and click the “Settings”



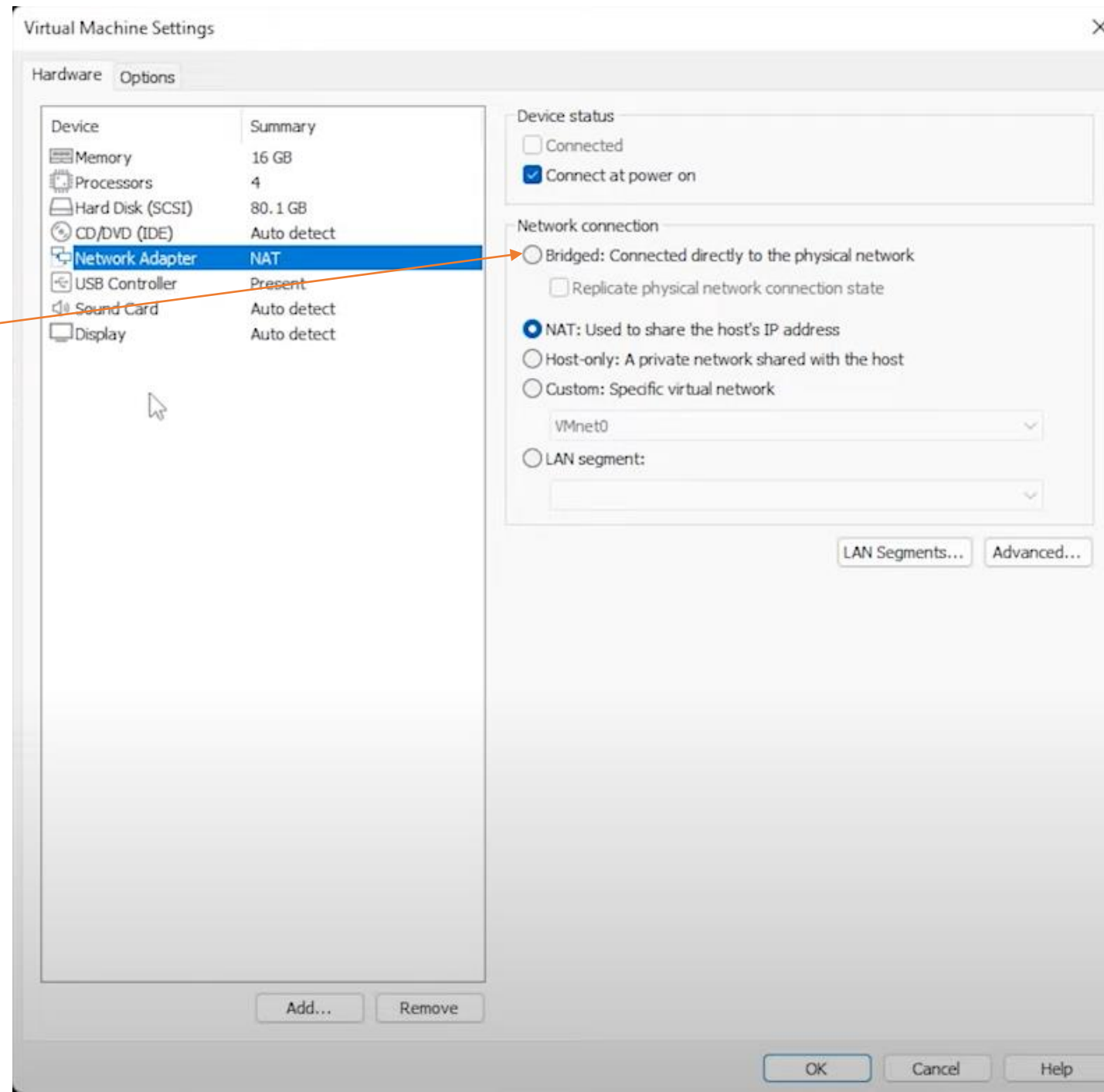
Install Kali Linux

- I choose the Bridged Adapter
- The reason why I choose this “Bridged Adapter”? I use another virtual machine software --- “VMWare” to explain this!
- Check the next page

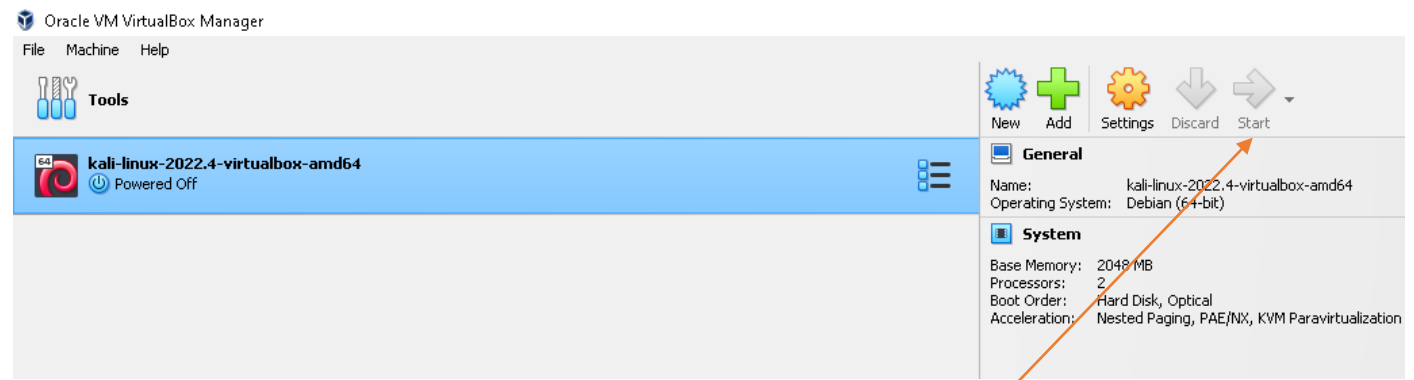


Install Kali Linux

- “Bridged” means, the virtual machine will be treated as a totally independent machine like your host computer
- For example, if my Windows Host gets 192.168.0.42, the Bridged Kali Linux might get 192.168.0.31 in the intranet



Install Kali Linux



- Click the “Start” button and you will see the Kali launched
- The default login username / password is kali / kali as it is stated in the user’s manual
- After your logged in, you might need to do lots of setup, for example, the screen resolution and the most important, updating the packages as follows:
 - `sudo apt update`
 - `sudo apt upgrade`
- Considering you might allocate 2GB for its memory (by default), it would be very slow. Have a cup of coffee! Happy updating!



File System



Home



Trash

Let's do some Footprinting job.

- Let's take our machine for examples.
- The 1st one is our sand server and the 2nd one is Dr. Alan's server
- One thing we noticed that they both answered by a server located in 24.116.0.53
- If you bring this IP address and go to the Google and ask a kind of service called "Reverse IP Lookup", we can easily found our Truman uses the internet service from a company called cableone.net

```
(kali㉿kali)-[~]  
$ nslookup sand.truman.edu  
Server:          24.116.0.53  
Address:         24.116.0.53#53  
  
Non-authoritative answer:  
sand.truman.edu canonical name = vh222004.truman.edu.  
Name:   vh222004.truman.edu  
Address: 150.243.160.10  
Name:   vh222004.truman.edu  
Address: 150.243.160.11  
  
(kali㉿kali)-[~]  
$ nslookup vh216602.truman.edu  
Server:          24.116.0.53  
Address:         24.116.0.53#53  
  
Non-authoritative answer:  
Name:   vh216602.truman.edu  
Address: 150.243.160.100
```

Let's do some Footprinting job.

- And? What amazes me is that, our “sand” server

is not hiding behind the VPN or any additional protections? What!??

- The sand server was assigned with 2 x IP addresses. It is very natural that it might have 2 x Gigabit Ethernet cards in its box

- The 2nd query is Dr. Alan's server.

Nothing special.

ptr:24.116.0.53

Find Problems

Type	IP Address	Domain Name
PTR	24.116.0.53 CABLE ONE, INC. (AS11492)	c1dns.cableone.net

```
(kali㉿kali)-[~]
$ nslookup sand.truman.edu
Server:      24.116.0.53
Address:     24.116.0.53#53

Non-authoritative answer:
sand.truman.edu canonical name = vh222004.truman.edu.
Name:   vh222004.truman.edu
Address: 150.243.160.10
Name:   vh222004.truman.edu
Address: 150.243.160.11
```

```
(kali㉿kali)-[~]
$ nslookup vh216602.truman.edu
Server:      24.116.0.53
Address:     24.116.0.53#53

Non-authoritative answer:
Name:   vh216602.truman.edu
Address: 150.243.160.100
```

Let's do some Footprinting job

- Let's take Dr. Alan's machine as an example. (150.243.160.100)
- I use the nmap to scan his machine
- 2 ports are found to be open.
- One is a http and the another is a ssh
- You can even see the version of Apache server he is using and what type of the SSH server (and also, its version)
- I will guess, he like to use the ssh to work from home and update his web pages / class schedules / course materials

```
(kali@kali)-[~]  
$ nmap -v -A -sV 150.243.160.100  
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-23 22:30 EST  
NSE: Loaded 155 scripts for scanning.  
NSE: Script Pre-scanning.  
Initiating NSE at 22:30  
Completed NSE at 22:30, 0.00s elapsed  
Initiating NSE at 22:30  
Completed NSE at 22:30, 0.00s elapsed  
Initiating NSE at 22:30  
Completed NSE at 22:30, 0.00s elapsed  
Initiating Ping Scan at 22:30  
Scanning 150.243.160.100 [2 ports]  
Completed Ping Scan at 22:30, 0.06s elapsed (1 total hosts)  
Initiating Parallel DNS resolution of 1 host. at 22:30  
Completed Parallel DNS resolution of 1 host. at 22:30, 0.06s elapsed  
Initiating Connect Scan at 22:30  
Scanning vh216602.truman.edu (150.243.160.100) [1000 ports]  
Discovered open port 80/tcp on 150.243.160.100  
Discovered open port 22/tcp on 150.243.160.100  
Completed Connect Scan at 22:30, 6.86s elapsed (1000 total ports)  
Initiating Service scan at 22:30  
Scanning 2 services on vh216602.truman.edu (150.243.160.100)  
Completed Service scan at 22:31, 7.25s elapsed (2 services on 1 host)  
NSE: Script scanning 150.243.160.100.  
Initiating NSE at 22:31  
Completed NSE at 22:31, 5.12s elapsed  
Initiating NSE at 22:31  
Completed NSE at 22:31, 0.25s elapsed  
Initiating NSE at 22:31  
Completed NSE at 22:31, 0.00s elapsed  
Nmap scan report for vh216602.truman.edu (150.243.160.100)  
Host is up (0.006s latency).  
Not shown: 986 filtered tcp ports (no-response)  
PORT      STATE SERVICE      VERSION  
20/tcp    closed ftp-data  
21/tcp    closed ftp  
22/tcp    open  ssh          OpenSSH 7.6p1 Ubuntu 4ubuntu0.5 (Ubuntu Linux; protocol 2.0)  
|_ ssh-hostkey:  
|   2048 0754d3011fb3e1947bbaa76c0d615830 (RSA)  
|   256 20b29508ef3fd39396c4332440142a76 (ECDSA)  
|   256 4dd3d7c07bae205d8092c059ae3d65cc (ED25519)  
53/tcp    closed domain  
80/tcp    open  http         Apache httpd 2.4.29 ((Ubuntu))  
|_ http-server-header: Apache/2.4.29 (Ubuntu)  
|_ http-methods:  
|_   Supported Methods: HEAD GET POST OPTIONS  
|_ http-title: Apache2 Ubuntu Default Page: It works
```

Let's do some Footprinting job

- For the rest of the ports are remaining close.
- He still gets good sense of computer security
--- all the unused ports are closed
- He is using Ububtu

```
110/tcp  closed pop3
113/tcp  closed ident
143/tcp  closed imap
443/tcp  closed https
554/tcp  closed rtsp
587/tcp  closed submission
993/tcp  closed imaps|
995/tcp  closed pop3s
5222/tcp closed xmpp-client
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

```
NSE: Script Post-scanning.
Initiating NSE at 22:31
Completed NSE at 22:31, 0.00s elapsed
Initiating NSE at 22:31
Completed NSE at 22:31, 0.00s elapsed
```