Managing Enterprise Cybersecurity MIS 4596

Unit #14

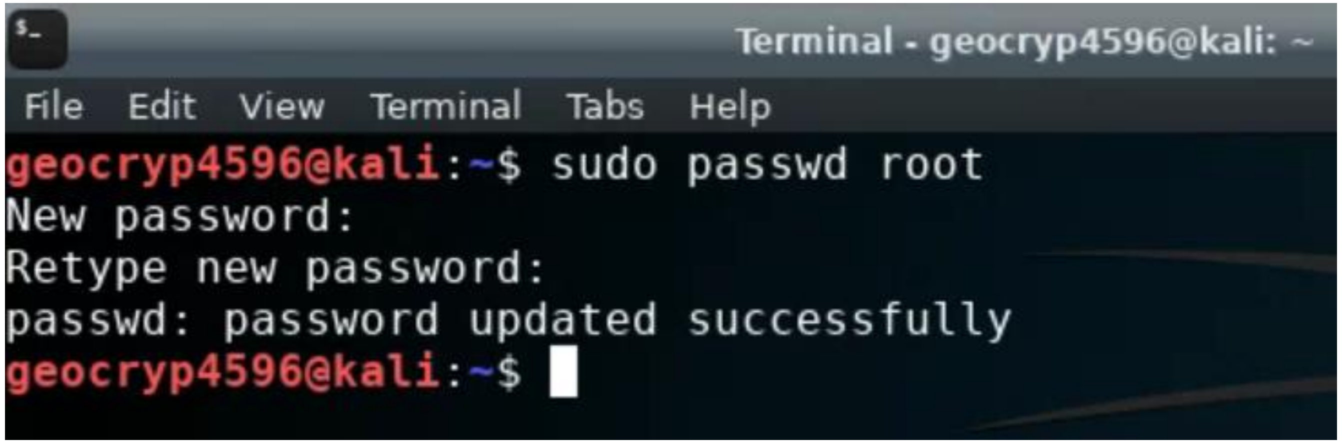
Agenda

- Reminder: Change your Kali password if you haven’t done so yet!
- Application vulnerability and security testing
- Lab 8: Vulnerability Scanning –
  - Startup and access virtual machines
  - Part 1: Nmap
  - Part 2: Nessus
- Scan results
  - Looking at a vulnerability

Reminder: Change Kali’s root Password Now (if you have not already done so)!

- Kali’s default root password is published and known to everyone
  - Login: root
  - Password: toor
- If you leave Kali running in the cloud (by mistake), someone may find it
- If they know enough to find it, they enough to login and access it
- If they use it, attack someone and create a problem – you are responsible!
- Change Kali’s root password now!
- From the $ prompt, type:

  “sudo passwd root”



```
geocryp4596@kali:~$ sudo passwd root
New password:
Retype new password:
passwd: password updated successfully
geocryp4596@kali:~$
```

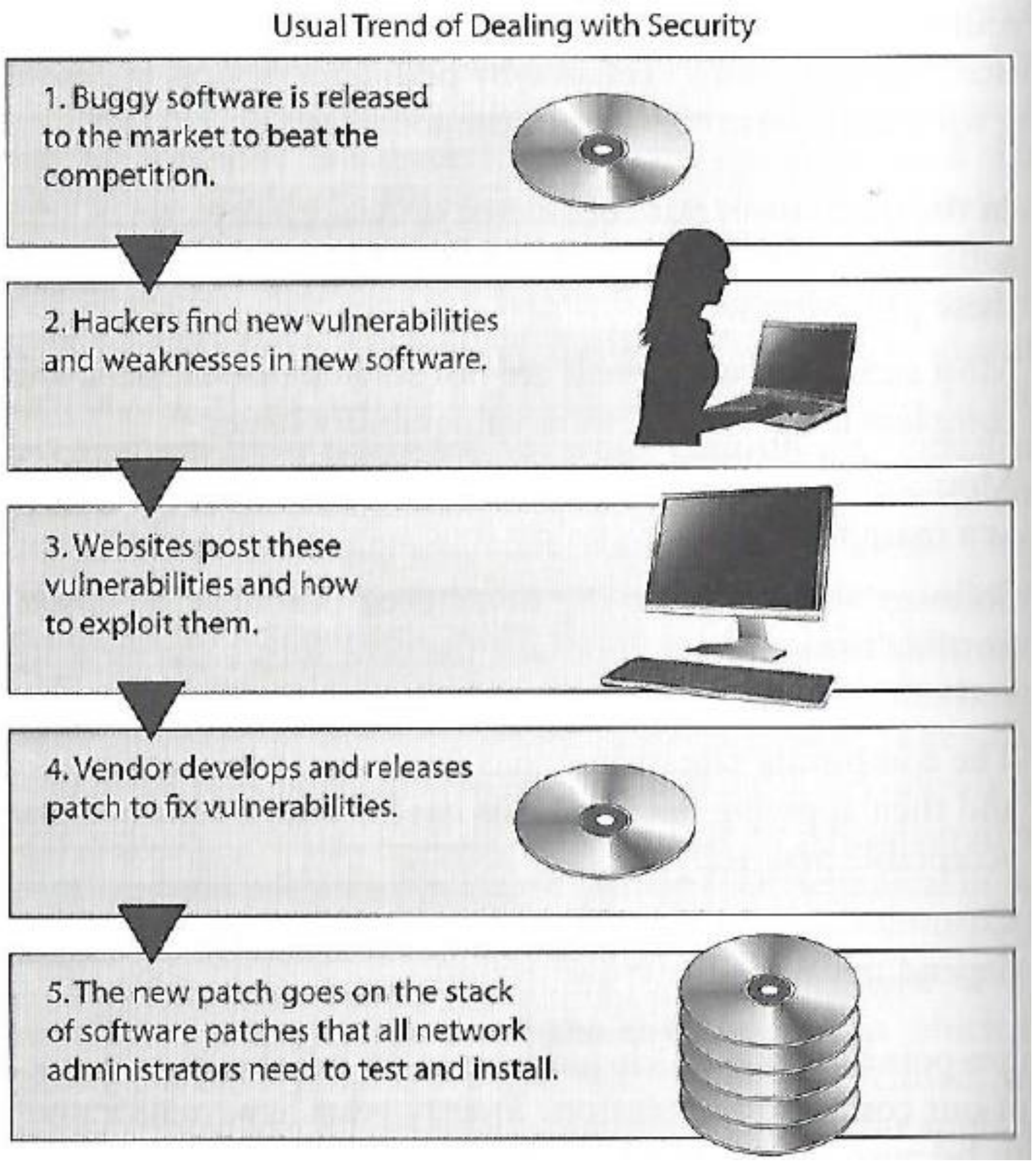
Application Security

As applications become more accessible though the web, cloud and mobile devices,

organizations are being forced to abandon their reactive approach to security and, instead,

to take a proactive approach by minimizing risk directly in the software they buy, create and use to serve themselves and their customers

Usual trend



Usual Trend of Dealing with Security

1. Buggy software is released to the market to beat the competition.

2. Hackers find new vulnerabilities and weaknesses in new software.

3. Websites post these vulnerabilities and how to exploit them.

4. Vendor develops and releases patch to fix vulnerabilities.

5. The new patch goes on the stack of software patches that all network administrators need to test and install.

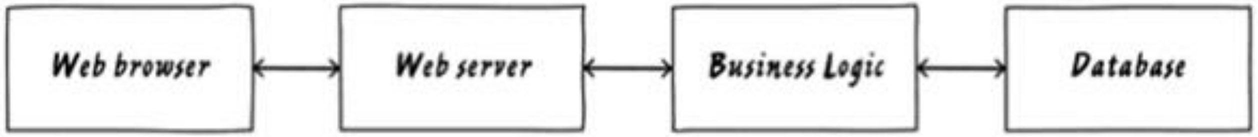
Harris, S. and Maymi F. (2016) CISSP All-in-One Exam Guide, McGraw-Hill Education, p. 1080

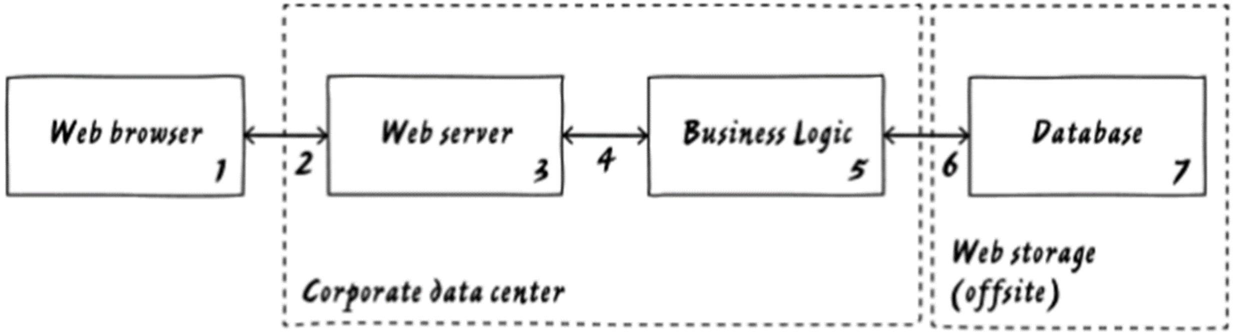
Software security, includes threat and attack surface analysis…

***Attack surface** is what is available to be used by an attacker against the application itself*

Goal of attack surface analysis is to identify and reduce the amount of code and functionality accessible to untrusted users

Development team should reduce the attack surface as much as possible to remove "resources" that can be used as avenues for the attacker to use





- *How do you know the web browser is used by the person you expect?*
- *Is it OK for data to go from one "box" to the next without being authenticated?*
- *Is it OK for data to go from one "box" to the next without being encrypted?*
- *What happens if someone made unauthorized modifications to data in the database?*

OWASP

Application Security Verification Standard 4.0.3

Final

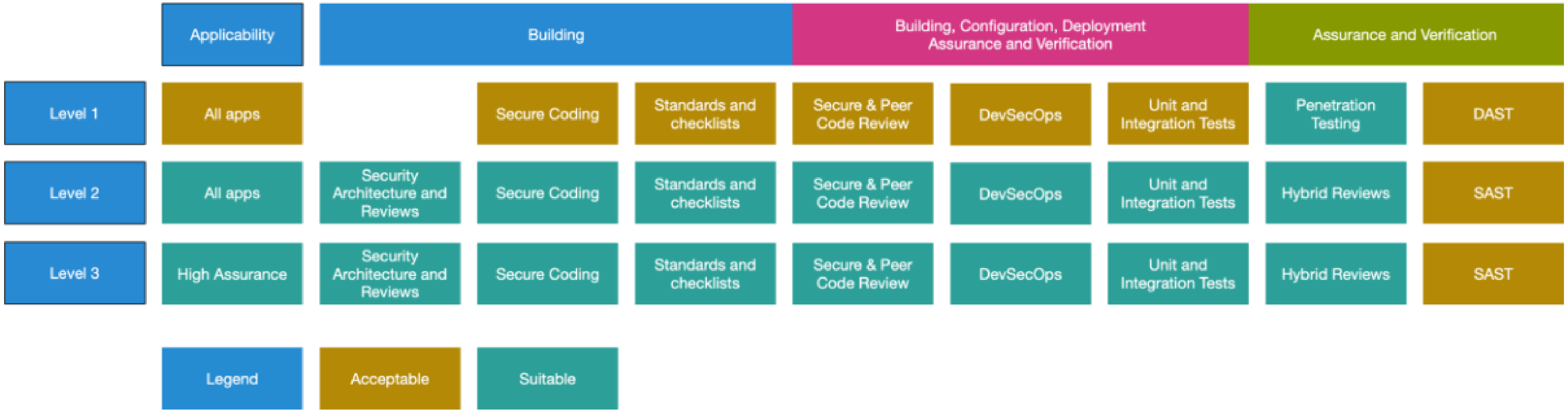
October 2021

Application Security Verification Levels

The Application Security Verification Standard defines three security verification levels, with each level increasing in depth.

- ASVS Level 1 is for low assurance levels, and is completely penetration testable
- ASVS Level 2 is for applications that contain sensitive data, which requires protection and is the recommended level for most apps
- ASVS Level 3 is for the most critical applications - applications that perform high value transactions, contain sensitive medical data, or any application that requires the highest level of trust.

Each ASVS level contains a list of security requirements. Each of these requirements can also be mapped to security-specific features and capabilities that must be built into software by developers.



	Applicability	Building			Building, Configuration, Deployment Assurance and Verification			Assurance and Verification	
Level 1	All apps		Secure Coding	Standards and checklists	Secure & Peer Code Review	DevSecOps	Unit and Integration Tests	Penetration Testing	DAST
Level 2	All apps	Security Architecture and Reviews	Secure Coding	Standards and checklists	Secure & Peer Code Review	DevSecOps	Unit and Integration Tests	Hybrid Reviews	SAST
Level 3	High Assurance	Security Architecture and Reviews	Secure Coding	Standards and checklists	Secure & Peer Code Review	DevSecOps	Unit and Integration Tests	Hybrid Reviews	SAST

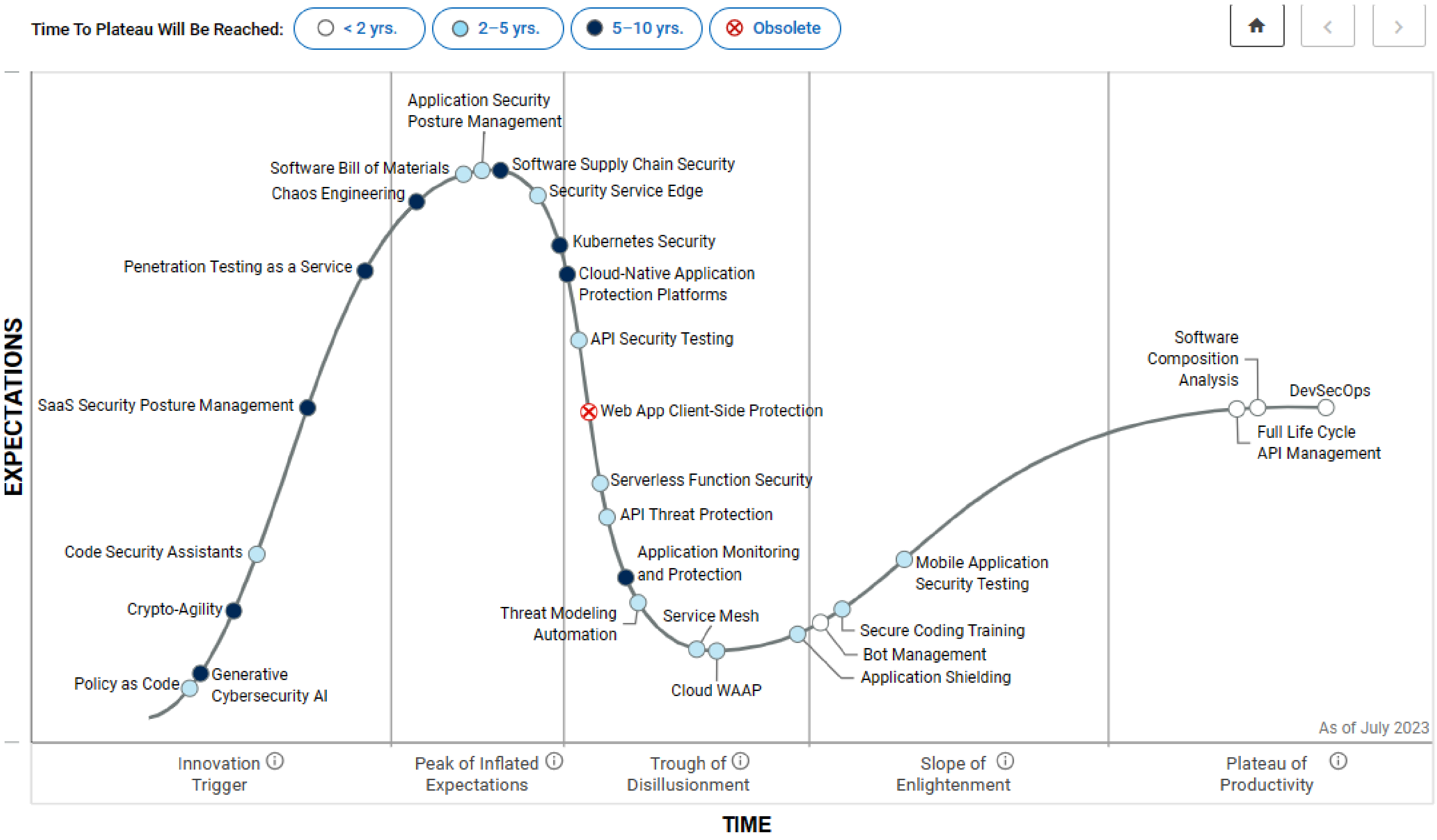
Legend | Acceptable | Suitable

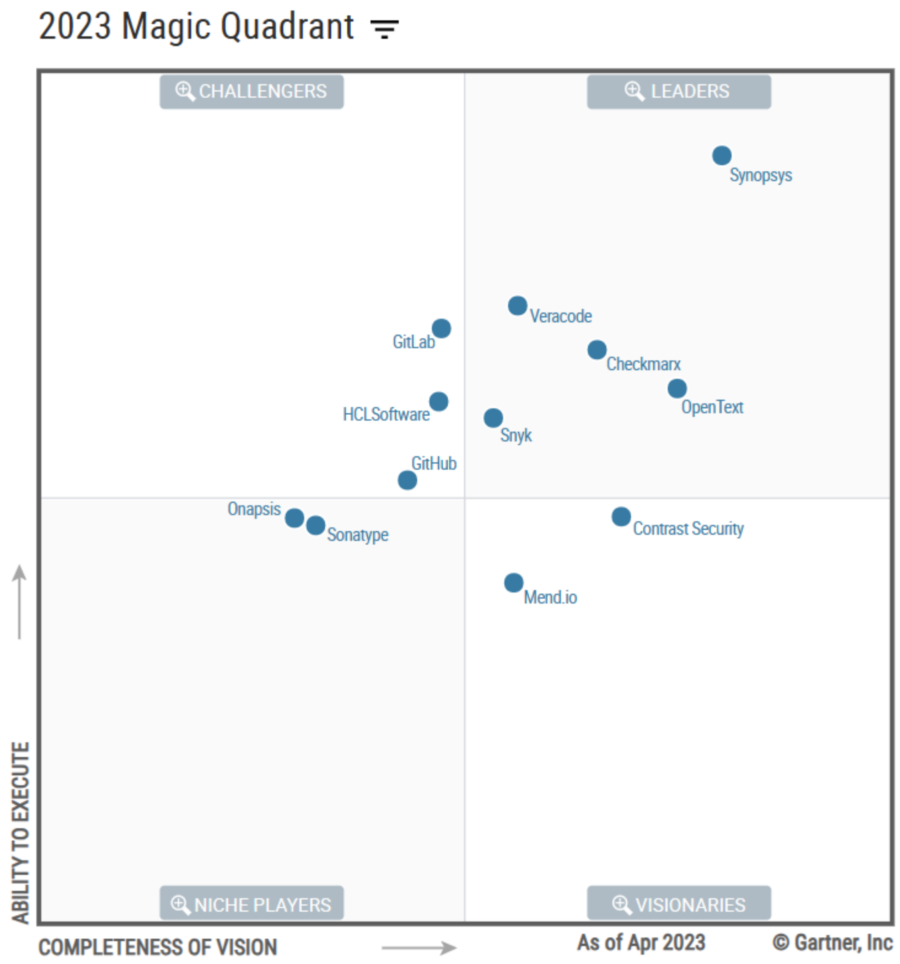
Figure 1 - OWASP Application Security Verification Standard 4.0 Levels

Application Security Testing (AST)

Fundamental Capabilities

- Static AST (SAST)
- Software Composition Analysis (SCA)
- Dynamic AST (DAST)
- API Testing





SBOM = Software Bill Of Materials

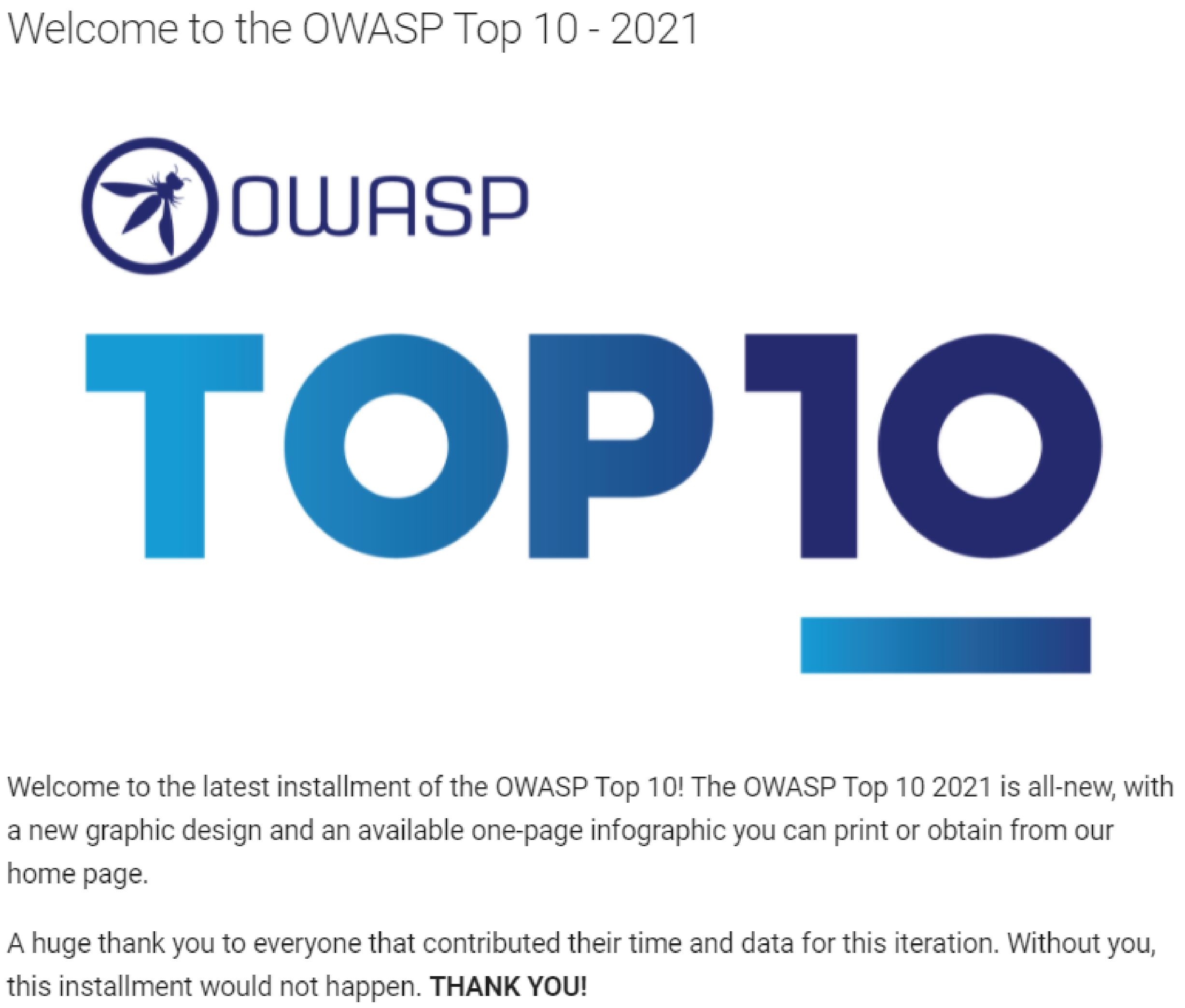
ASRTM = Application Security Requirements and Threat modeling

CSSTP = Crowdsourced Software Security Testing Platforms (i.e. bug bounties)

WAAP = cloud Web Application and API Protection

EAM = Externalized Authorization Management

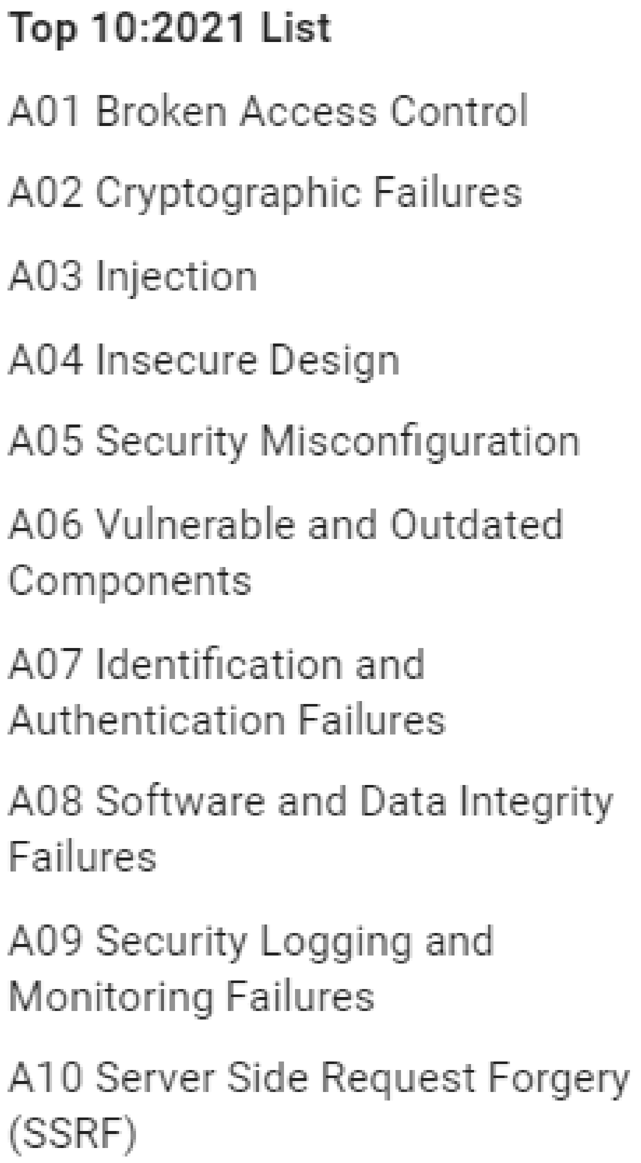
Estimated at $3.4 billion in 2022, the AST market reached a 27% year over year growth in 2021



Welcome to the OWASP Top 10 - 2021

Welcome to the latest installment of the OWASP Top 10! The OWASP Top 10 2021 is all-new, with a new graphic design and an available one-page infographic you can print or obtain from our home page.

A huge thank you to everyone that contributed their time and data for this iteration. Without you, this installment would not happen. **THANK YOU!**



Top 10:2021 List

A01 Broken Access Control

A02 Cryptographic Failures

A03 Injection

A04 Insecure Design

A05 Security Misconfiguration

A06 Vulnerable and Outdated Components

A07 Identification and Authentication Failures

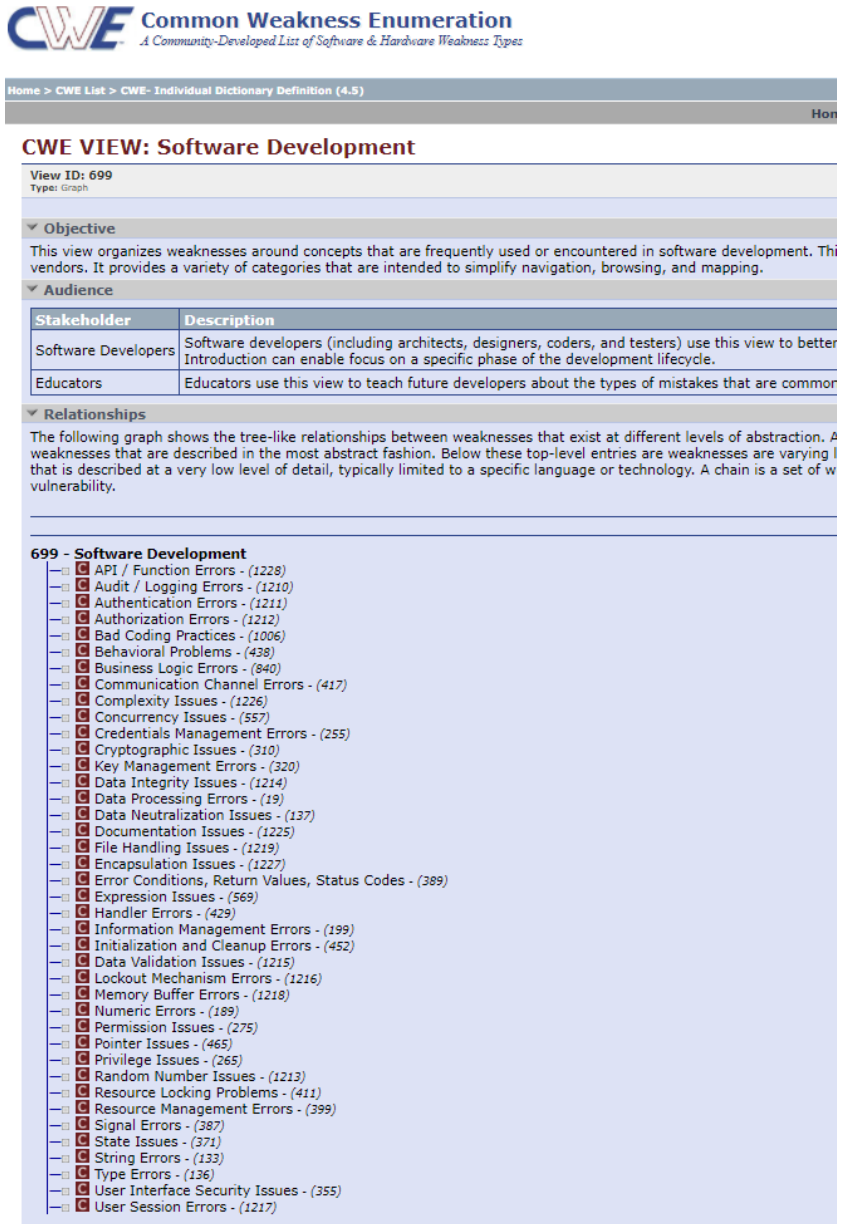
A08 Software and Data Integrity Failures

A09 Security Logging and Monitoring Failures

A10 Server Side Request Forgery (SSRF)

https://www.owasp.org/index.php/OWASP_Top_Ten_Cheat_Sheet

MITRE’s Common Application Vulnerabilities



Common Weakness Enumeration
A Community-Developed List of Software & Hardware Weakness Types

Home > CWE List > CWE- Individual Dictionary Definition (4.5)

Hon

CWE VIEW: Software Development

View ID: 699
Type: Graph

Objective

This view organizes weaknesses around concepts that are frequently used or encountered in software development. Thi
vendors. It provides a variety of categories that are intended to simplify navigation, browsing, and mapping.

Audience

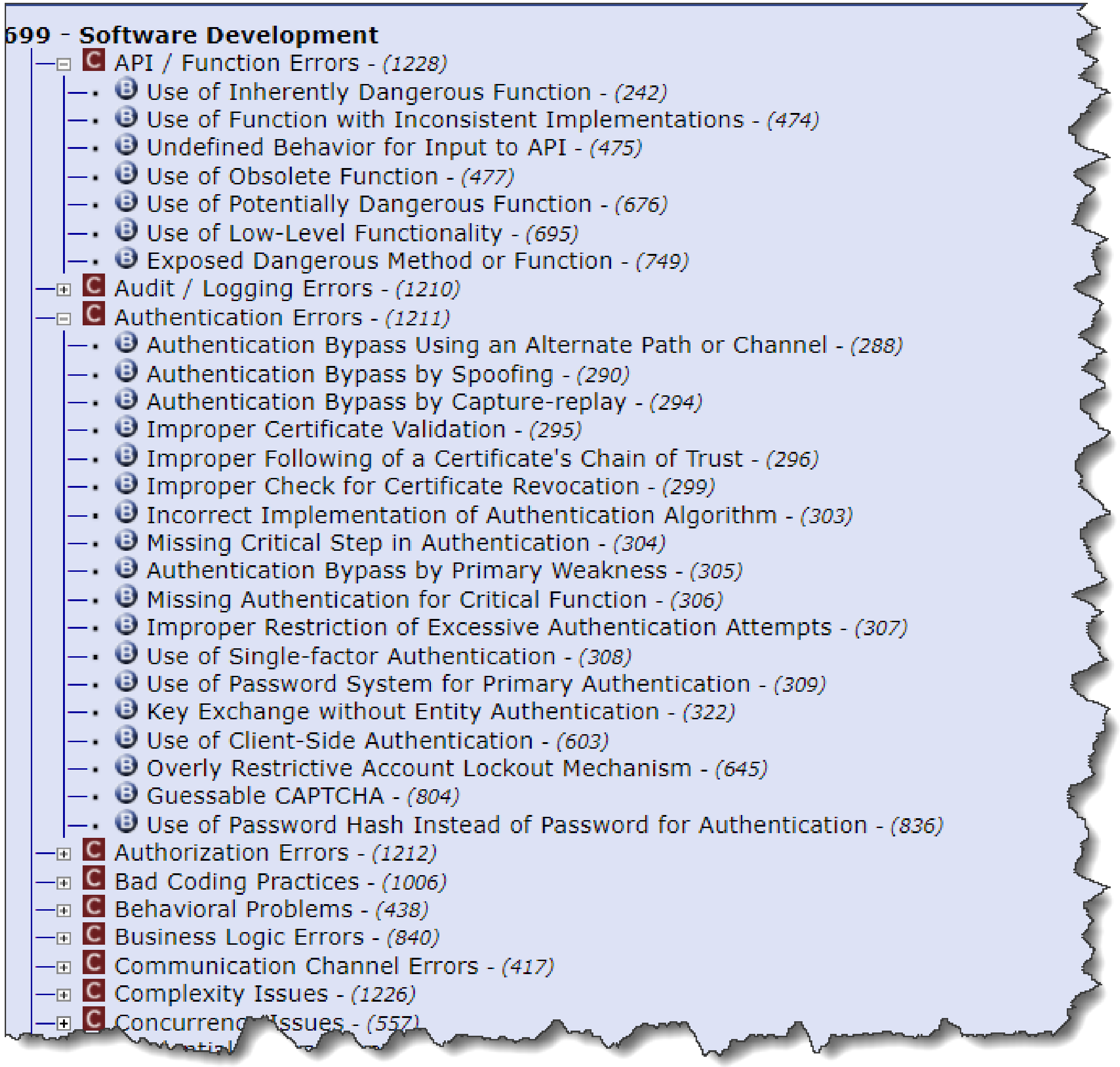
Stakeholder	Description
Software Developers	Software developers (including architects, designers, coders, and testers) use this view to better Introduction can enable focus on a specific phase of the development lifecycle.
Educators	Educators use this view to teach future developers about the types of mistakes that are commor

Relationships

The following graph shows the tree-like relationships between weaknesses that exist at different levels of abstraction. A
weaknesses that are described in the most abstract fashion. Below these top-level entries are weaknesses are varying l
that is described at a very low level of detail, typically limited to a specific language or technology. A chain is a set of w
vulnerability.

699 - Software Development

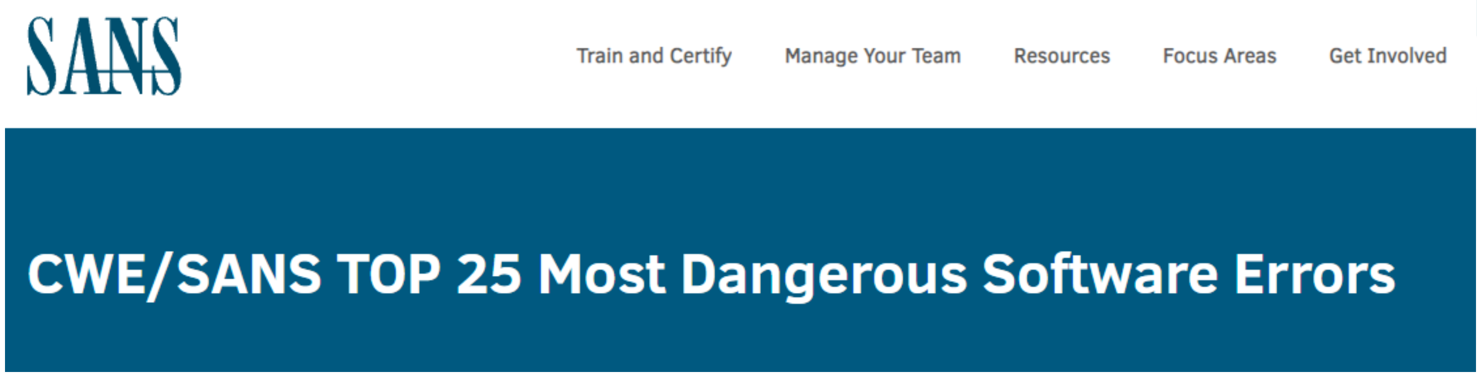
- C API / Function Errors - *(1228)*
- C Audit / Logging Errors - *(1210)*
- C Authentication Errors - *(1211)*
- C Authorization Errors - *(1212)*
- C Bad Coding Practices - *(1006)*
- C Behavioral Problems - *(438)*
- C Business Logic Errors - *(840)*
- C Communication Channel Errors - *(417)*
- C Complexity Issues - *(1226)*
- C Concurrency Issues - *(557)*
- C Credentials Management Errors - *(255)*
- C Cryptographic Issues - *(310)*
- C Key Management Errors - *(320)*
- C Data Integrity Issues - *(1214)*
- C Data Processing Errors - *(19)*
- C Data Neutralization Issues - *(137)*
- C Documentation Issues - *(1225)*
- C File Handling Issues - *(1219)*
- C Encapsulation Issues - *(1227)*
- C Error Conditions, Return Values, Status Codes - *(389)*
- C Expression Issues - *(569)*
- C Handler Errors - *(429)*
- C Information Management Errors - *(199)*
- C Initialization and Cleanup Errors - *(452)*
- C Data Validation Issues - *(1215)*
- C Lockout Mechanism Errors - *(1216)*
- C Memory Buffer Errors - *(1218)*
- C Numeric Errors - *(189)*
- C Permission Issues - *(275)*
- C Pointer Issues - *(465)*
- C Privilege Issues - *(265)*
- C Random Number Issues - *(1213)*
- C Resource Locking Problems - *(411)*
- C Resource Management Errors - *(399)*
- C Signal Errors - *(387)*
- C State Issues - *(371)*
- C String Errors - *(133)*
- C Type Errors - *(136)*
- C User Interface Security Issues - *(355)*
- C User Session Errors - *(1217)*



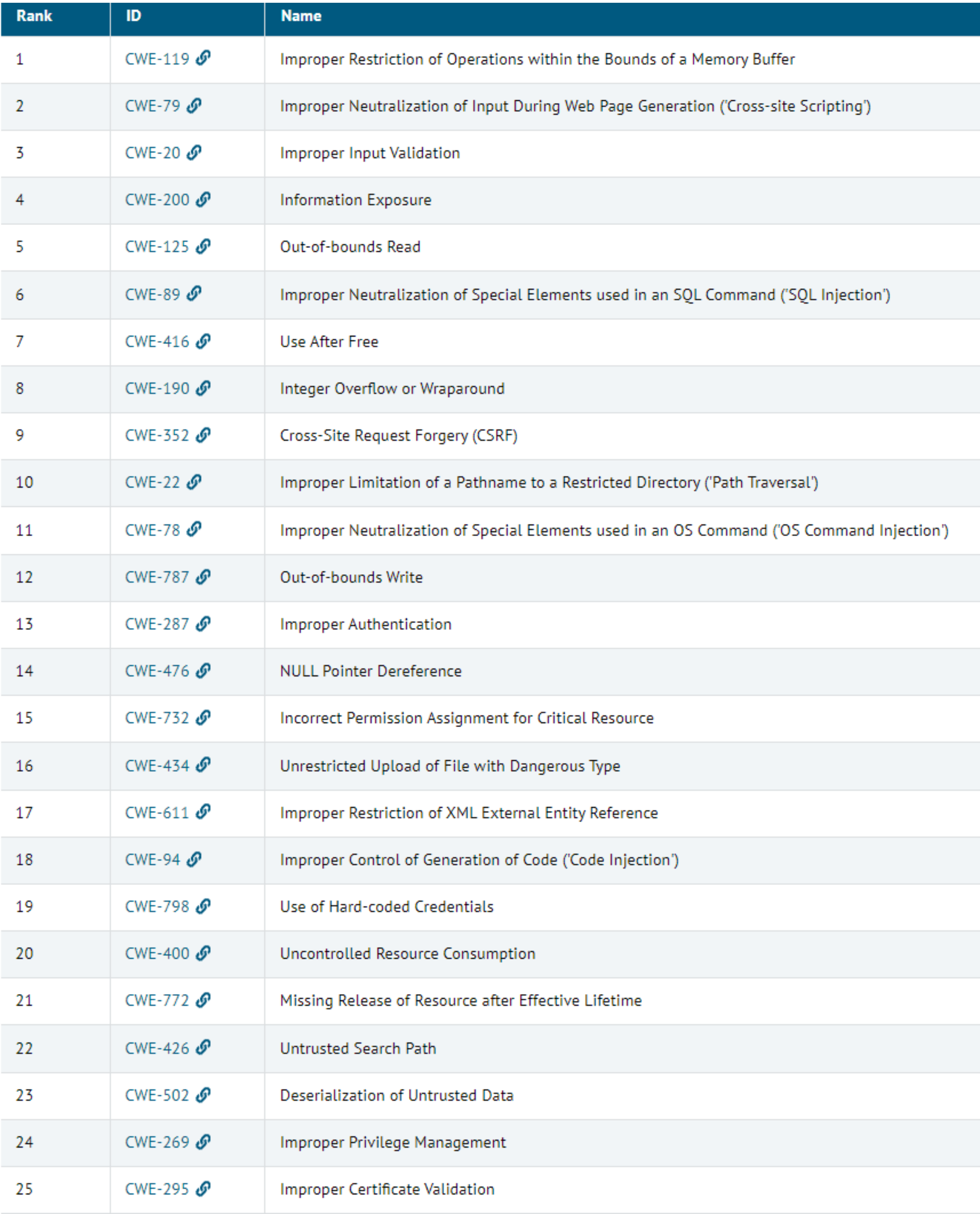
699 - Software Development

- C API / Function Errors - *(1228)*
  - B Use of Inherently Dangerous Function - *(242)*
  - B Use of Function with Inconsistent Implementations - *(474)*
  - B Undefined Behavior for Input to API - *(475)*
  - B Use of Obsolete Function - *(477)*
  - B Use of Potentially Dangerous Function - *(676)*
  - B Use of Low-Level Functionality - *(695)*
  - B Exposed Dangerous Method or Function - *(749)*
- C Audit / Logging Errors - *(1210)*
- C Authentication Errors - *(1211)*
  - B Authentication Bypass Using an Alternate Path or Channel - *(288)*
  - B Authentication Bypass by Spoofing - *(290)*
  - B Authentication Bypass by Capture-replay - *(294)*
  - B Improper Certificate Validation - *(295)*
  - B Improper Following of a Certificate's Chain of Trust - *(296)*
  - B Improper Check for Certificate Revocation - *(299)*
  - B Incorrect Implementation of Authentication Algorithm - *(303)*
  - B Missing Critical Step in Authentication - *(304)*
  - B Authentication Bypass by Primary Weakness - *(305)*
  - B Missing Authentication for Critical Function - *(306)*
  - B Improper Restriction of Excessive Authentication Attempts - *(307)*
  - B Use of Single-factor Authentication - *(308)*
  - B Use of Password System for Primary Authentication - *(309)*
  - B Key Exchange without Entity Authentication - *(322)*
  - B Use of Client-Side Authentication - *(603)*
  - B Overly Restrictive Account Lockout Mechanism - *(645)*
  - B Guessable CAPTCHA - *(804)*
  - B Use of Password Hash Instead of Password for Authentication - *(836)*
- C Authorization Errors - *(1212)*
- C Bad Coding Practices - *(1006)*
- C Behavioral Problems - *(438)*
- C Business Logic Errors - *(840)*
- C Communication Channel Errors - *(417)*
- C Complexity Issues - *(1226)*
- C Concurrenc Issues - *(557*

MITRE’s Common Weakness Enumeration



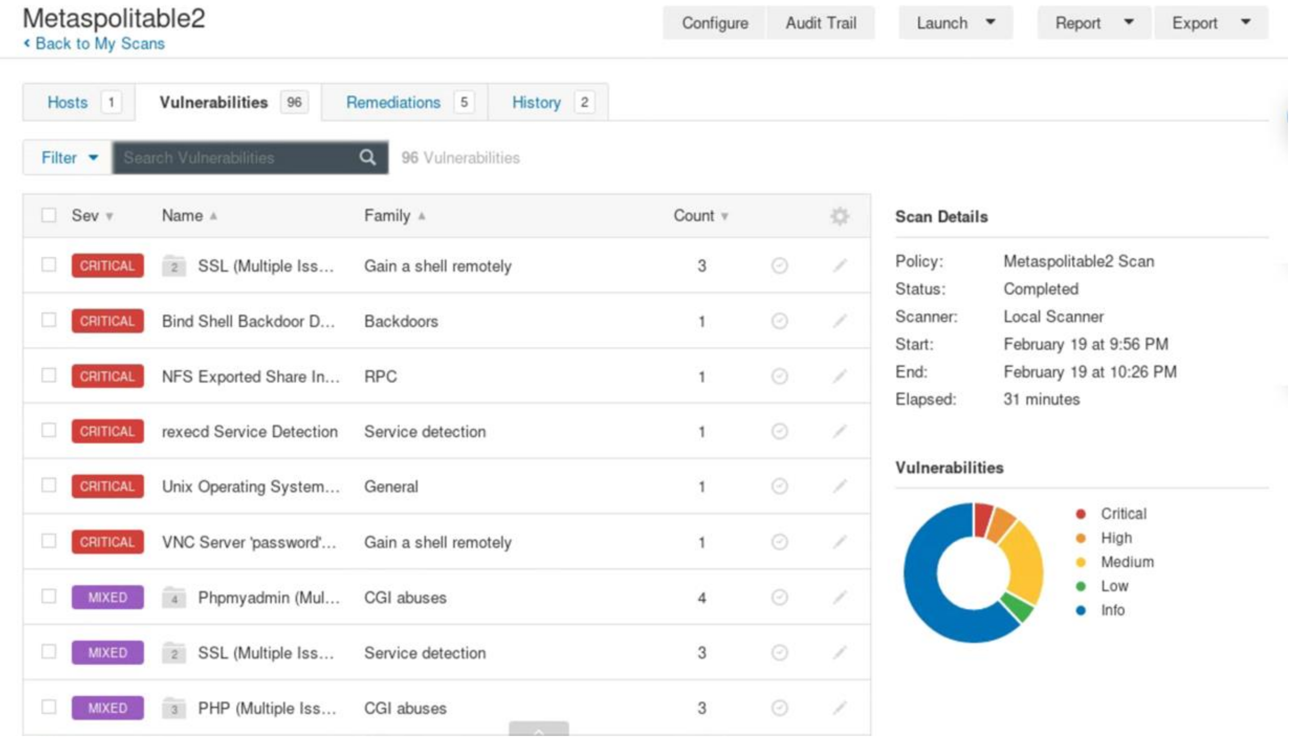
CWE/SANS TOP 25 Most Dangerous Software Errors



Rank	ID	Name
1	CWE-119	Improper Restriction of Operations within the Bounds of a Memory Buffer
2	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')
3	CWE-20	Improper Input Validation
4	CWE-200	Information Exposure
5	CWE-125	Out-of-bounds Read
6	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
7	CWE-416	Use After Free
8	CWE-190	Integer Overflow or Wraparound
9	CWE-352	Cross-Site Request Forgery (CSRF)
10	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')
11	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
12	CWE-787	Out-of-bounds Write
13	CWE-287	Improper Authentication
14	CWE-476	NULL Pointer Dereference
15	CWE-732	Incorrect Permission Assignment for Critical Resource
16	CWE-434	Unrestricted Upload of File with Dangerous Type
17	CWE-611	Improper Restriction of XML External Entity Reference
18	CWE-94	Improper Control of Generation of Code ('Code Injection')
19	CWE-798	Use of Hard-coded Credentials
20	CWE-400	Uncontrolled Resource Consumption
21	CWE-772	Missing Release of Resource after Effective Lifetime
22	CWE-426	Untrusted Search Path
23	CWE-502	Deserialization of Untrusted Data
24	CWE-269	Improper Privilege Management
25	CWE-295	Improper Certificate Validation

Vulnerability Scanning

- Scanning methods:
  - Safe
  - Destructive
- Service recognition – Determines what service is running on which ports
- Reports
  - Indicates the threat level for vulnerabilities it detects
    - Critical
    - High
    - Medium
    - Low
    - Informational
  - Description of Vulnerability
  - Risk Factor
  - CVE Number



Metaspolitable2

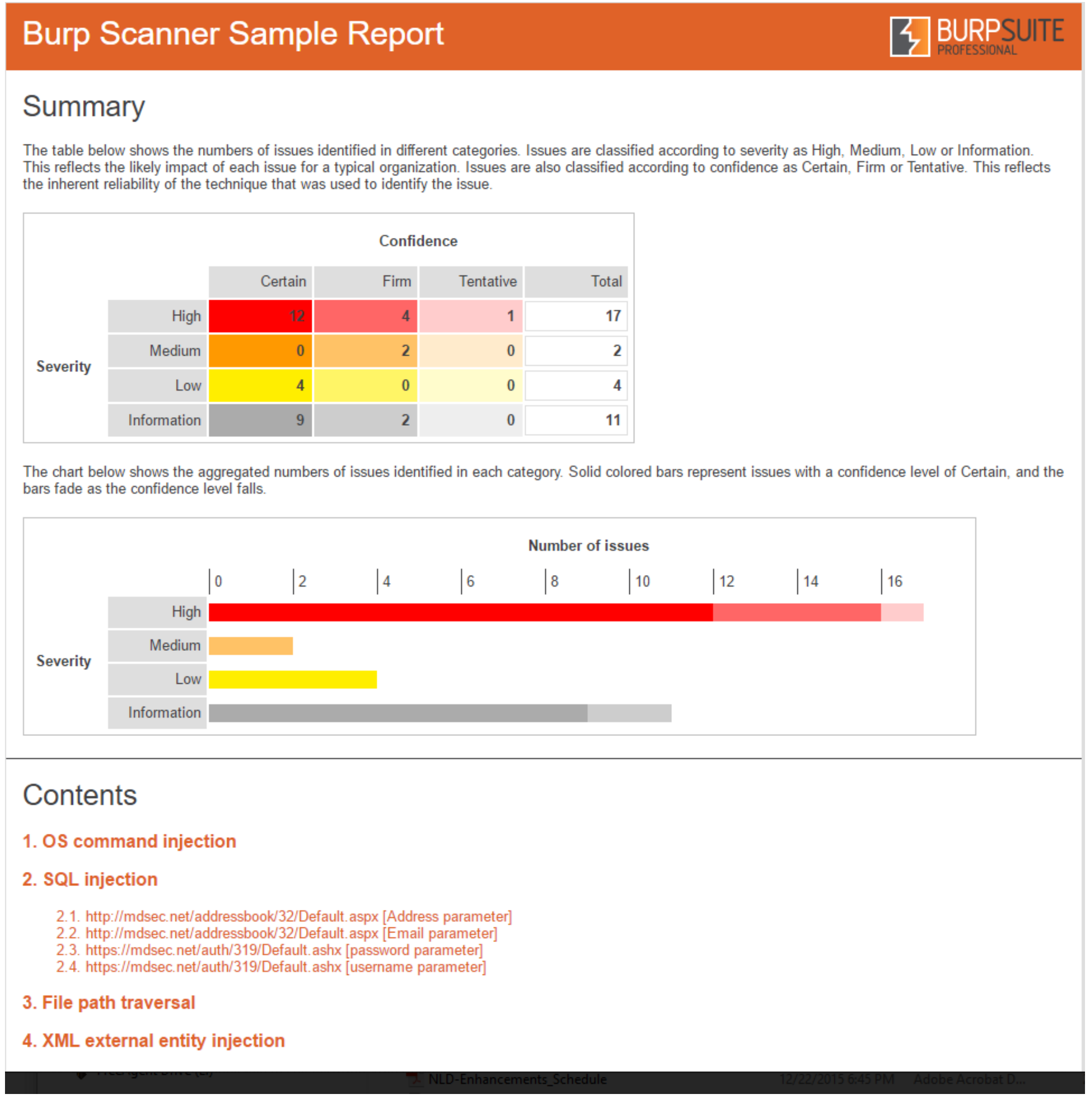
Sev	Name	Family	Count
CRITICAL	2 SSL (Multiple Iss...	Gain a shell remotely	3
CRITICAL	Bind Shell Backdoor D...	Backdoors	1
CRITICAL	NFS Exported Share In...	RPC	1
CRITICAL	rexecd Service Detection	Service detection	1
CRITICAL	Unix Operating System...	General	1
CRITICAL	VNC Server 'password'...	Gain a shell remotely	1
MIXED	4 Phpmyadmin (Mul...	CGI abuses	4
MIXED	2 SSL (Multiple Iss...	Service detection	3
MIXED	3 PHP (Multiple Iss...	CGI abuses	3

Scan Details

Policy: Metaspolitable2 Scan
Status: Completed
Scanner: Local Scanner
Start: February 19 at 9:56 PM
End: February 19 at 10:26 PM
Elapsed: 31 minutes

Vulnerabilities

Application Vulnerability Testing Reports



Burp Scanner Sample Report

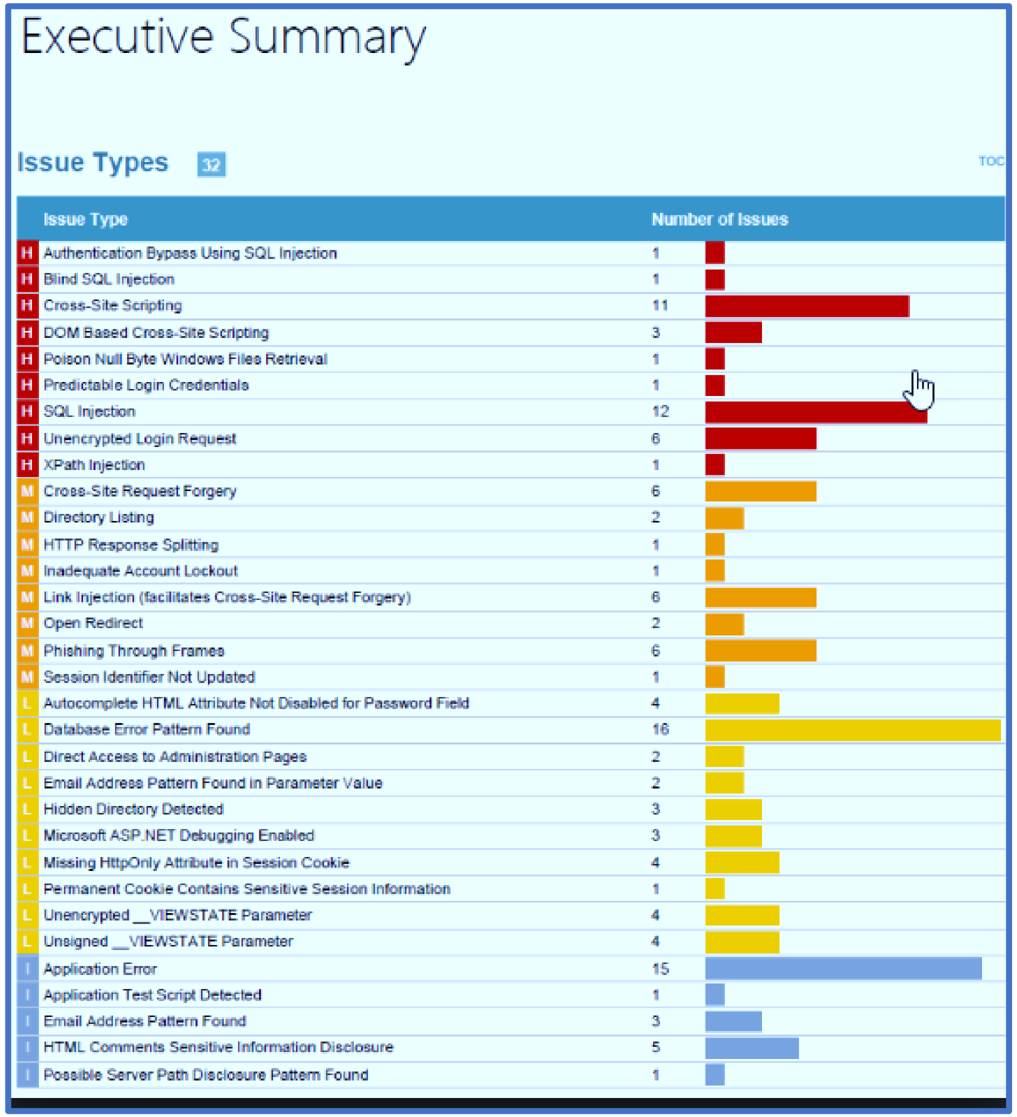
Summary

The table below shows the numbers of issues identified in different categories. Issues are classified according to severity as High, Medium, Low or Information. This reflects the likely impact of each issue for a typical organization. Issues are also classified according to confidence as Certain, Firm or Tentative. This reflects the inherent reliability of the technique that was used to identify the issue.

		Confidence			
		Certain	Firm	Tentative	Total
Severity	High	12	4	1	17
	Medium	0	2	0	2
	Low	4	0	0	4
	Information	9	2	0	11

The chart below shows the aggregated numbers of issues identified in each category. Solid colored bars represent issues with a confidence level of Certain, and the bars fade as the confidence level falls.

Contents



Executive Summary

Issue Types 32

	Issue Type	Number of Issues
H	Authentication Bypass Using SQL Injection	1
H	Blind SQL Injection	1
H	Cross-Site Scripting	11
H	DOM Based Cross-Site Scripting	3
H	Poison Null Byte Windows Files Retrieval	1
H	Predictable Login Credentials	1
H	SQL Injection	12
H	Unencrypted Login Request	6
H	XPath Injection	1
M	Cross-Site Request Forgery	6
M	Directory Listing	2
M	HTTP Response Splitting	1
M	Inadequate Account Lockout	1
M	Link Injection (facilitates Cross-Site Request Forgery)	6
M	Open Redirect	2
M	Phishing Through Frames	6
M	Session Identifier Not Updated	1
L	Autocomplete HTML Attribute Not Disabled for Password Field	4
L	Database Error Pattern Found	16
L	Direct Access to Administration Pages	2
L	Email Address Pattern Found in Parameter Value	2
L	Hidden Directory Detected	3
L	Microsoft ASP.NET Debugging Enabled	3
L	Missing HttpOnly Attribute in Session Cookie	4
L	Permanent Cookie Contains Sensitive Session Information	1
L	Unencrypted __VIEWSTATE Parameter	4
L	Unsigned __VIEWSTATE Parameter	4
I	Application Error	15
I	Application Test Script Detected	1
I	Email Address Pattern Found	3
I	HTML Comments Sensitive Information Disclosure	5
I	Possible Server Path Disclosure Pattern Found	1

Vulnerability Scanning Lab

Labs

Lab 1: Google Cloud Platform and Linux Tutorial

Lab 2: Symmetric Encryption and Hashing

Lab 3: Asymmetric Cryptography

Lab 4: Digital Certificates

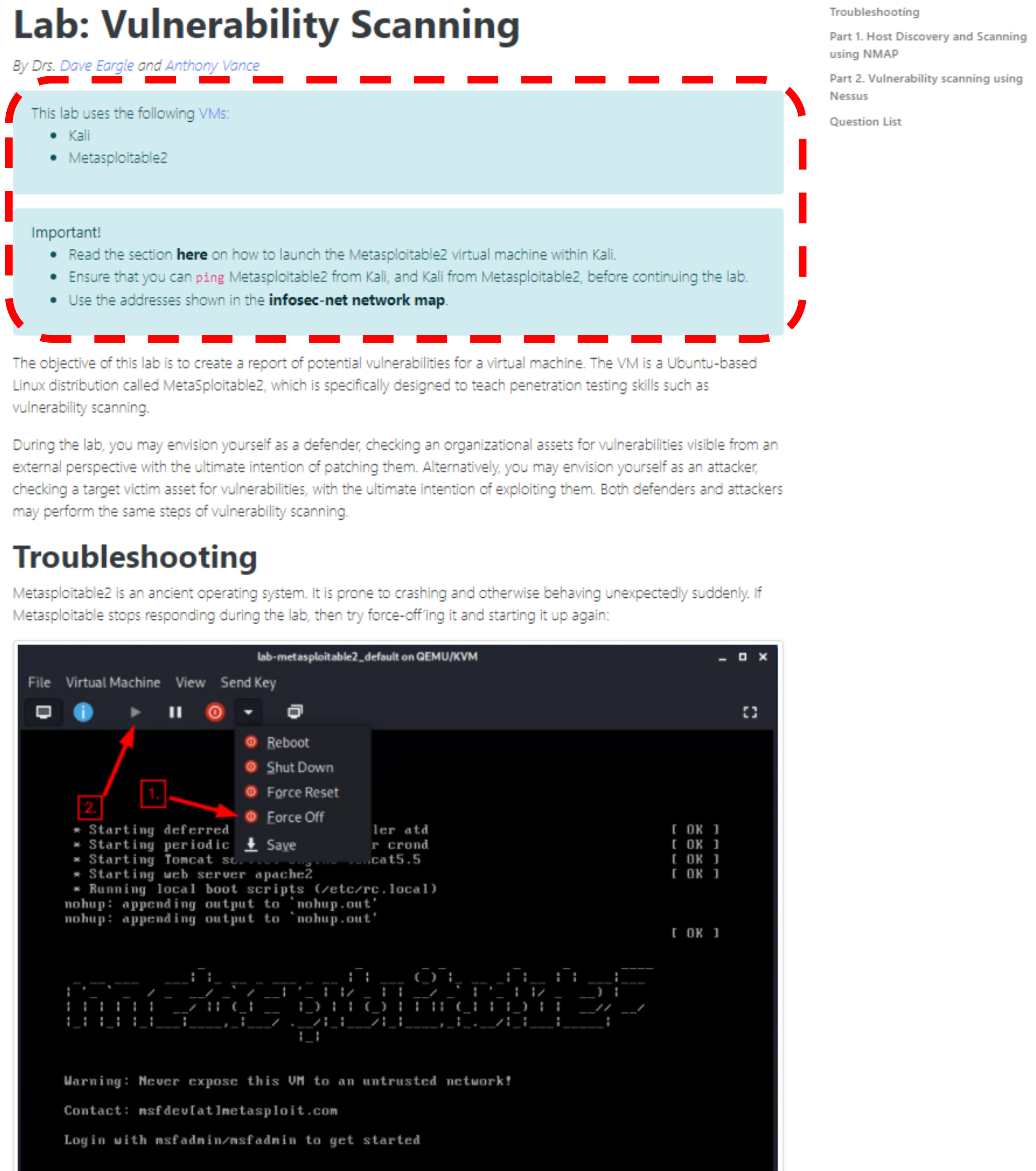
Lab 5: Password Cracking

Lab 6: Vulnerability Scanning

Lab 7: Vulnerability Exploitation

Lab 8: Web Privacy and Anonymity

Lab 9: Social Engineering



Lab: Vulnerability Scanning

By Drs. Dave Eargle and Anthony Vance

This lab uses the following VMs:

- Kali
- Metasploitable2

Important!

- Read the section **here** on how to launch the Metasploitable2 virtual machine within Kali.
- Ensure that you can `ping` Metasploitable2 from Kali, and Kali from Metasploitable2, before continuing the lab.
- Use the addresses shown in the **infosec-net network map**.

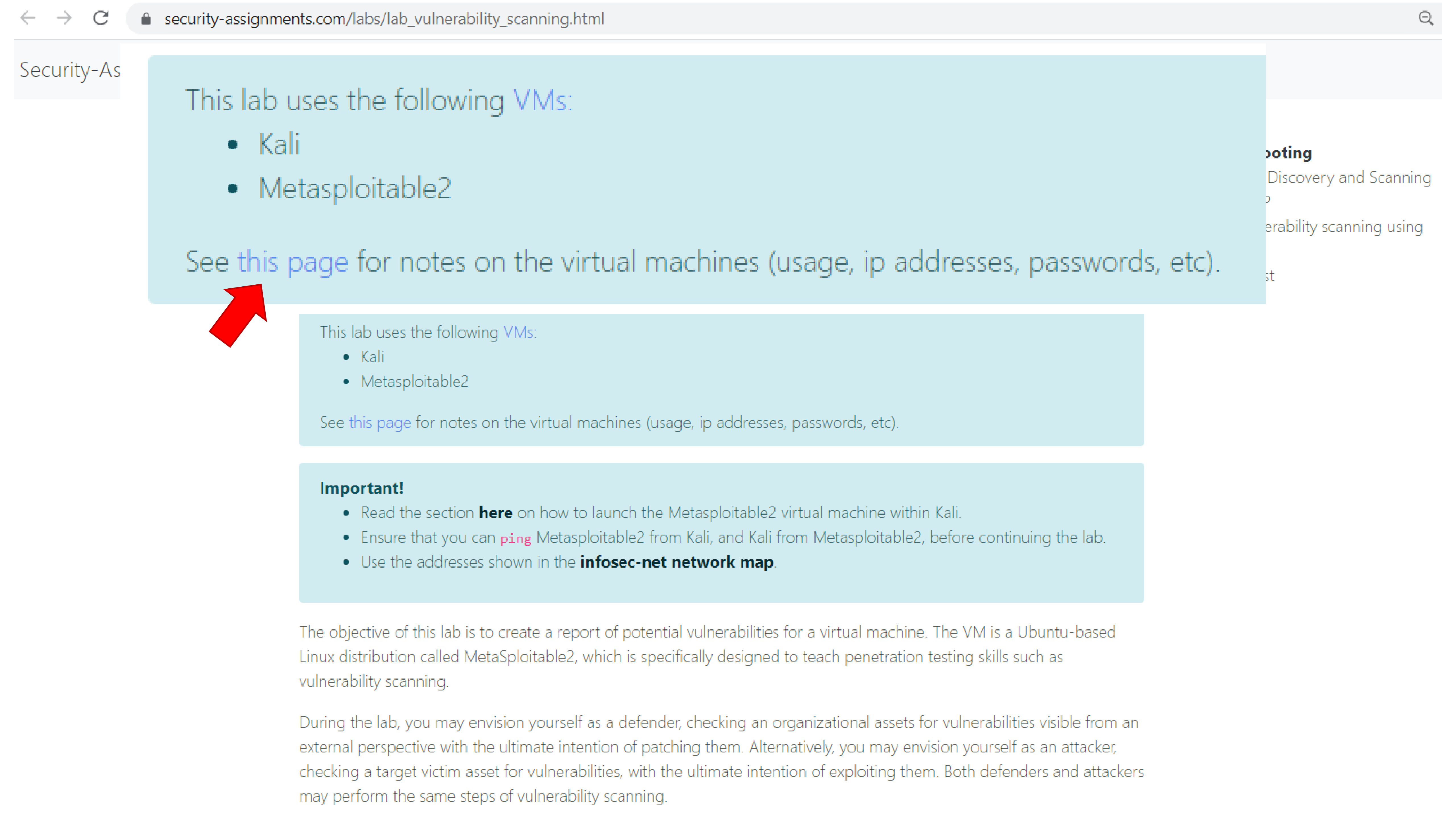
The objective of this lab is to create a report of potential vulnerabilities for a virtual machine. The VM is a Ubuntu-based Linux distribution called MetaSploitable2, which is specifically designed to teach penetration testing skills such as vulnerability scanning.

During the lab, you may envision yourself as a defender, checking an organizational assets for vulnerabilities visible from an external perspective with the ultimate intention of patching them. Alternatively, you may envision yourself as an attacker, checking a target victim asset for vulnerabilities, with the ultimate intention of exploiting them. Both defenders and attackers may perform the same steps of vulnerability scanning.

Troubleshooting

Metasploitable2 is an ancient operating system. It is prone to crashing and otherwise behaving unexpectedly suddenly. If Metasploitable stops responding during the lab, then try force-off'ing it and starting it up again:

```
 * Starting deferred                    ler atd          [ OK ]
 * Starting periodic                    r crond          [ OK ]
 * Starting Tomcat s                    cat5.5           [ OK ]
 * Starting web server apache2                           [ OK ]
 * Running local boot scripts (/etc/rc.local)
nohup: appending output to `nohup.out'
nohup: appending output to `nohup.out'
                                                         [ OK ]

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started
```



security-assignments.com/labs/lab_vulnerability_scanning.html

Security-As

This lab uses the following VMs:

- Kali
- Metasploitable2

See this page for notes on the virtual machines (usage, ip addresses, passwords, etc).

ooting

Discovery and Scanning

erability scanning using

st

This lab uses the following VMs:

- Kali
- Metasploitable2

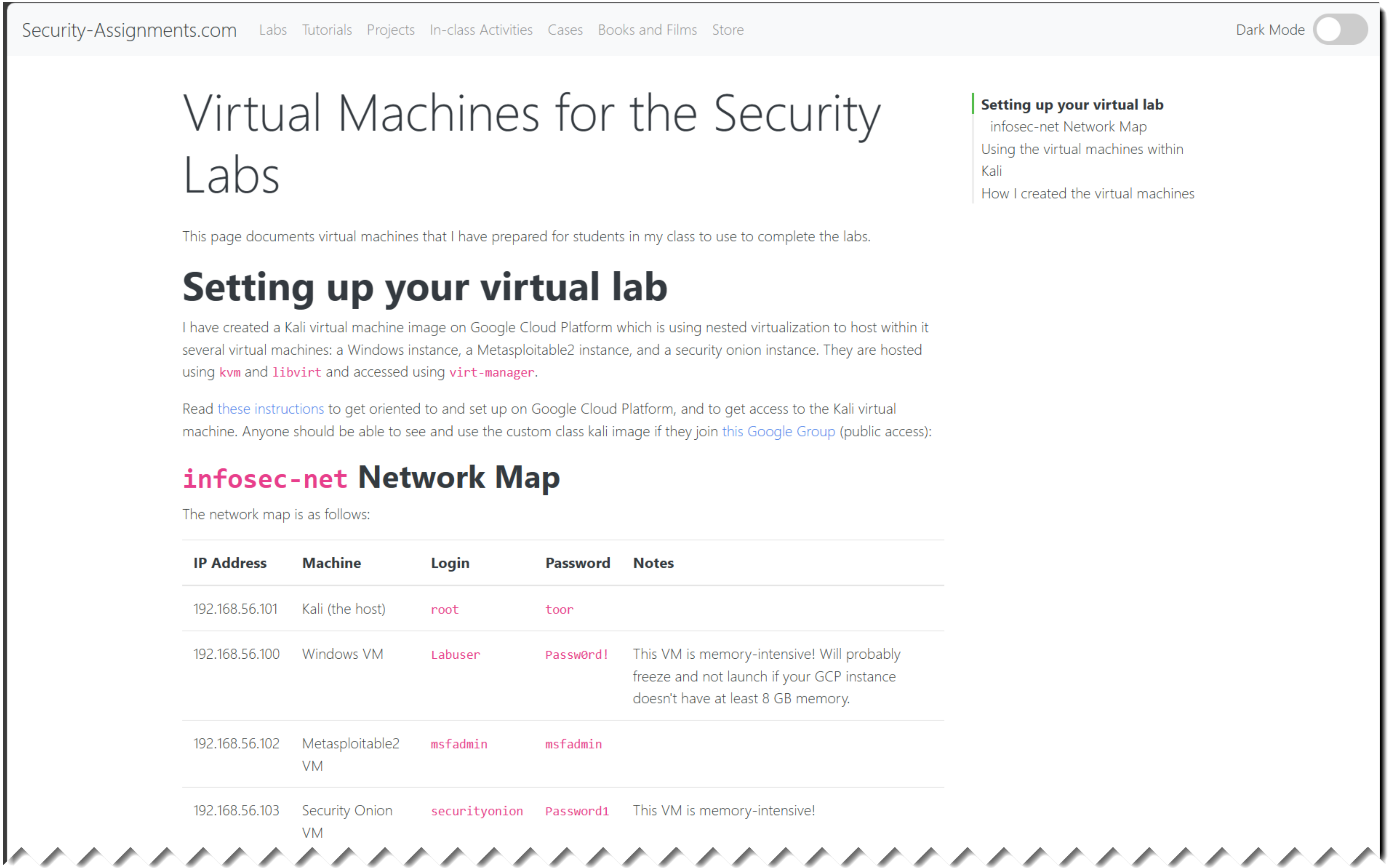
See this page for notes on the virtual machines (usage, ip addresses, passwords, etc).

Important!

- Read the section **here** on how to launch the Metasploitable2 virtual machine within Kali.
- Ensure that you can `ping` Metasploitable2 from Kali, and Kali from Metasploitable2, before continuing the lab.
- Use the addresses shown in the **infosec-net network map**.

The objective of this lab is to create a report of potential vulnerabilities for a virtual machine. The VM is a Ubuntu-based Linux distribution called MetaSploitable2, which is specifically designed to teach penetration testing skills such as vulnerability scanning.

During the lab, you may envision yourself as a defender, checking an organizational assets for vulnerabilities visible from an external perspective with the ultimate intention of patching them. Alternatively, you may envision yourself as an attacker, checking a target victim asset for vulnerabilities, with the ultimate intention of exploiting them. Both defenders and attackers may perform the same steps of vulnerability scanning.



Virtual Machines for the Security Labs

This page documents virtual machines that I have prepared for students in my class to use to complete the labs.

Setting up your virtual lab

I have created a Kali virtual machine image on Google Cloud Platform which is using nested virtualization to host within it several virtual machines: a Windows instance, a Metasploitable2 instance, and a security onion instance. They are hosted using `kvm` and `libvirt` and accessed using `virt-manager`.

Read these instructions to get oriented to and set up on Google Cloud Platform, and to get access to the Kali virtual machine. Anyone should be able to see and use the custom class kali image if they join this Google Group (public access):

`infosec-net` Network Map

The network map is as follows:

IP Address	Machine	Login	Password	Notes
192.168.56.101	Kali (the host)	`root`	`toor`	
192.168.56.100	Windows VM	`Labuser`	`Passw0rd!`	This VM is memory-intensive! Will probably freeze and not launch if your GCP instance doesn't have at least 8 GB memory.
192.168.56.102	Metasploitable2 VM	`msfadmin`	`msfadmin`	
192.168.56.103	Security Onion VM	`securityonion`	`Password1`	This VM is memory-intensive!

Using the virtual machines within Kali

1. The virtual machines are accessed using `virt-manager`. First, you should make sure that your user account is a member of the `libvirt` group.

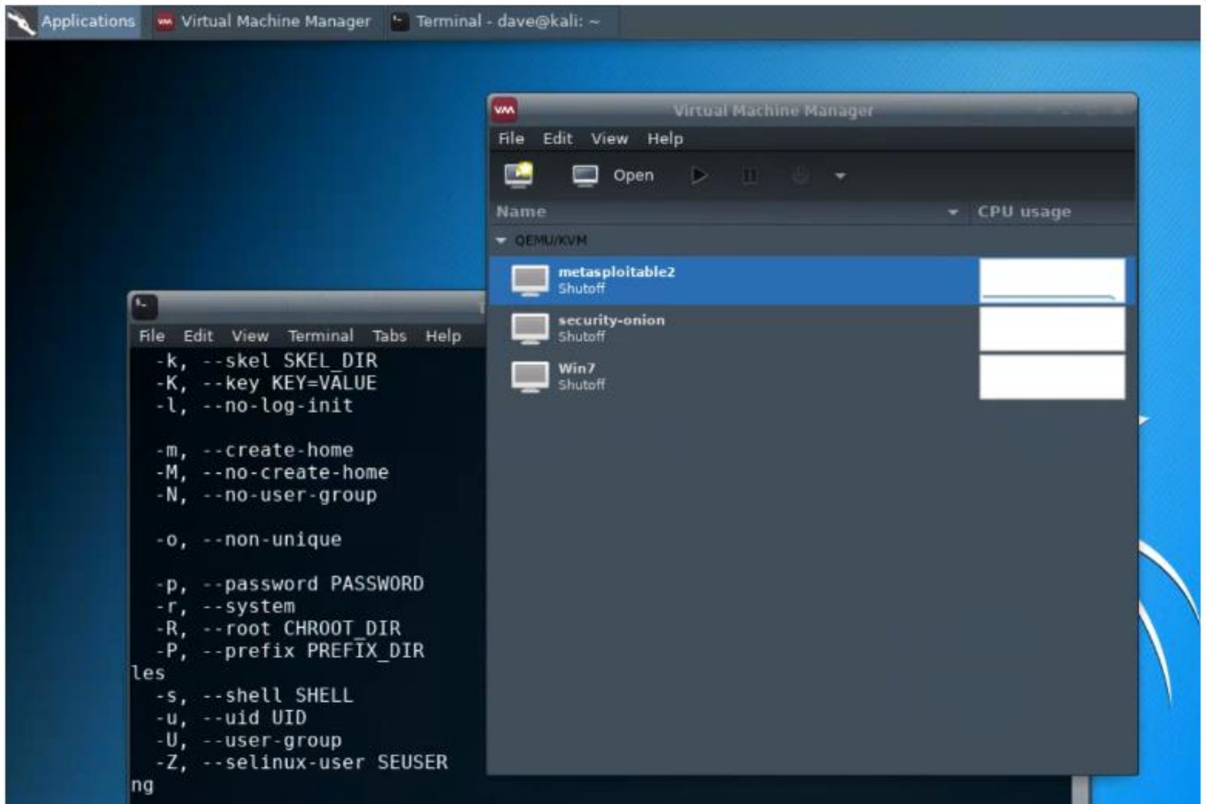
```
sudo usermod -a -G libvirt $(whoami)
```

Heads up! This will need to be run each time you create a new Kali instance.

Alternatively, log in as root (password `toor`):

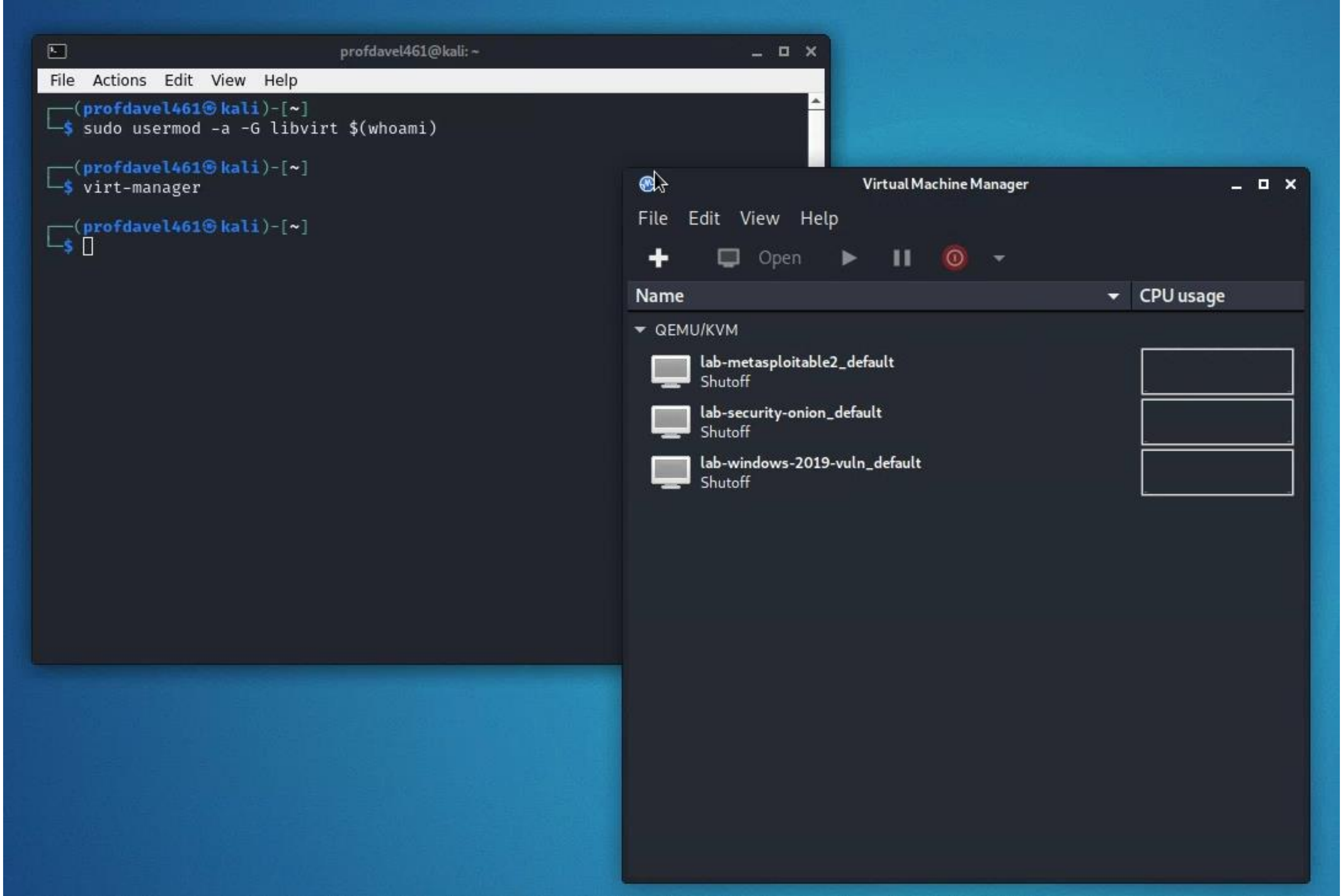
```
su root
```

2. Then, from a terminal, run `virt-manager` to get an interface such as the following:



```
  -k, --skel SKEL_DIR
  -K, --key KEY=VALUE
  -l, --no-log-init

  -m, --create-home
  -M, --no-create-home
  -N, --no-user-group

  -o, --non-unique

  -p, --password PASSWORD
  -r, --system
  -R, --root CHROOT_DIR
  -P, --prefix PREFIX_DIR
les
  -s, --shell SHELL
  -u, --uid UID
  -U, --user-group
  -Z, --selinux-user SEUSER
ng
```

[No Title]



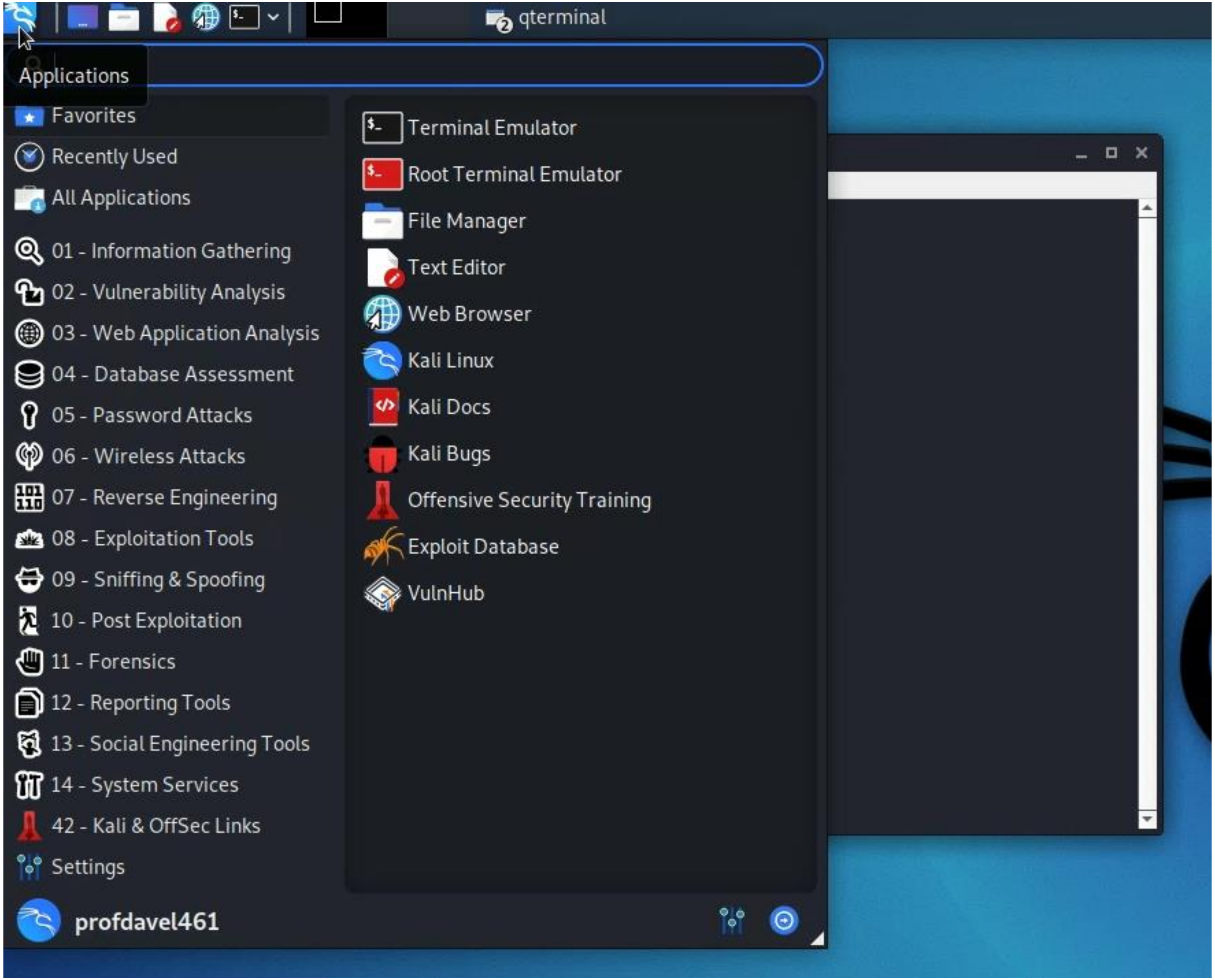
```
┌──(profdavel461㉿kali)-[~]
└─$ sudo usermod -a -G libvirt $(whoami)

┌──(profdavel461㉿kali)-[~]
└─$ virt-manager

┌──(profdavel461㉿kali)-[~]
└─$ 
```

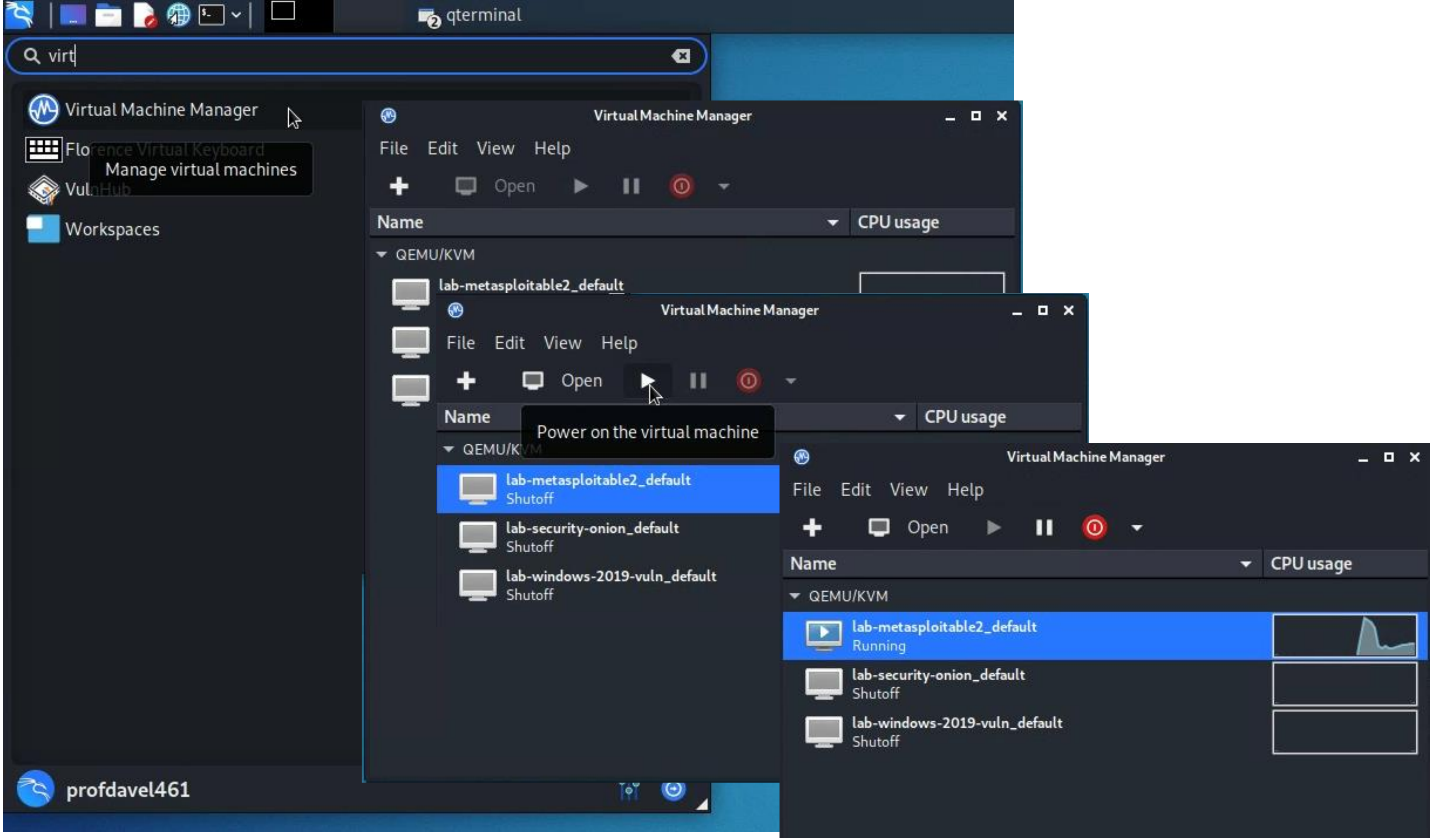
QEMU/KVM

lab-metasploitable2_default
Shutoff

lab-security-onion_default
Shutoff

lab-windows-2019-vuln_default
Shutoff


qterminal
Applications
Favorites
Recently Used
All Applications
01 - Information Gathering
02 - Vulnerability Analysis
03 - Web Application Analysis
04 - Database Assessment
05 - Password Attacks
06 - Wireless Attacks
07 - Reverse Engineering
08 - Exploitation Tools
09 - Sniffing & Spoofing
10 - Post Exploitation
11 - Forensics
12 - Reporting Tools
13 - Social Engineering Tools
14 - System Services
42 - Kali & OffSec Links
Settings
Terminal Emulator
Root Terminal Emulator
File Manager
Text Editor
Web Browser
Kali Linux
Kali Docs
Kali Bugs
Offensive Security Training
Exploit Database
VulnHub
profdavel461



virt

Manage virtual machines

Power on the virtual machine

lab-metasploitable2_default
Shutoff

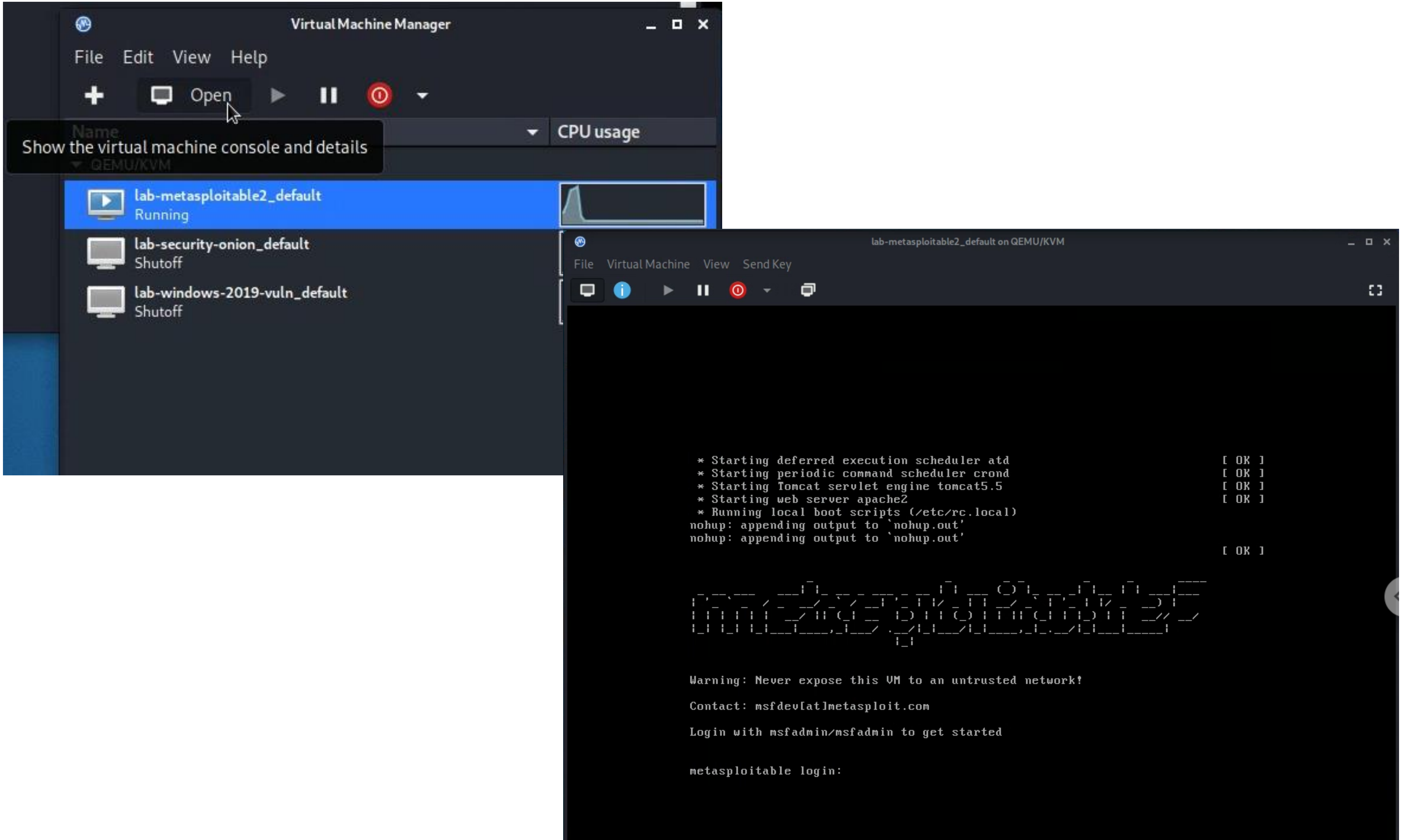
lab-security-onion_default
Shutoff

lab-windows-2019-vuln_default
Shutoff

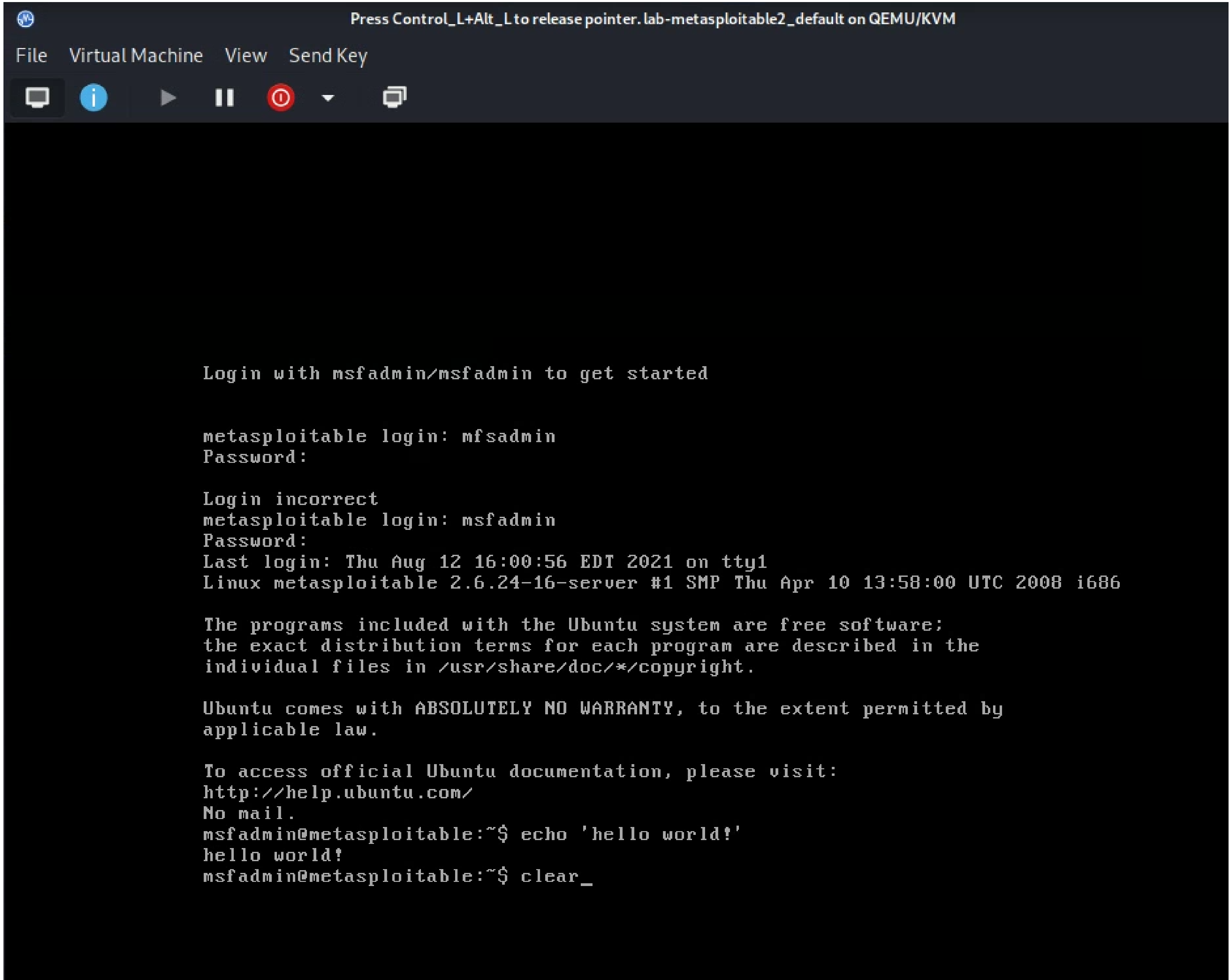
lab-metasploitable2_default
Running

lab-security-onion_default
Shutoff

lab-windows-2019-vuln_default
Shutoff

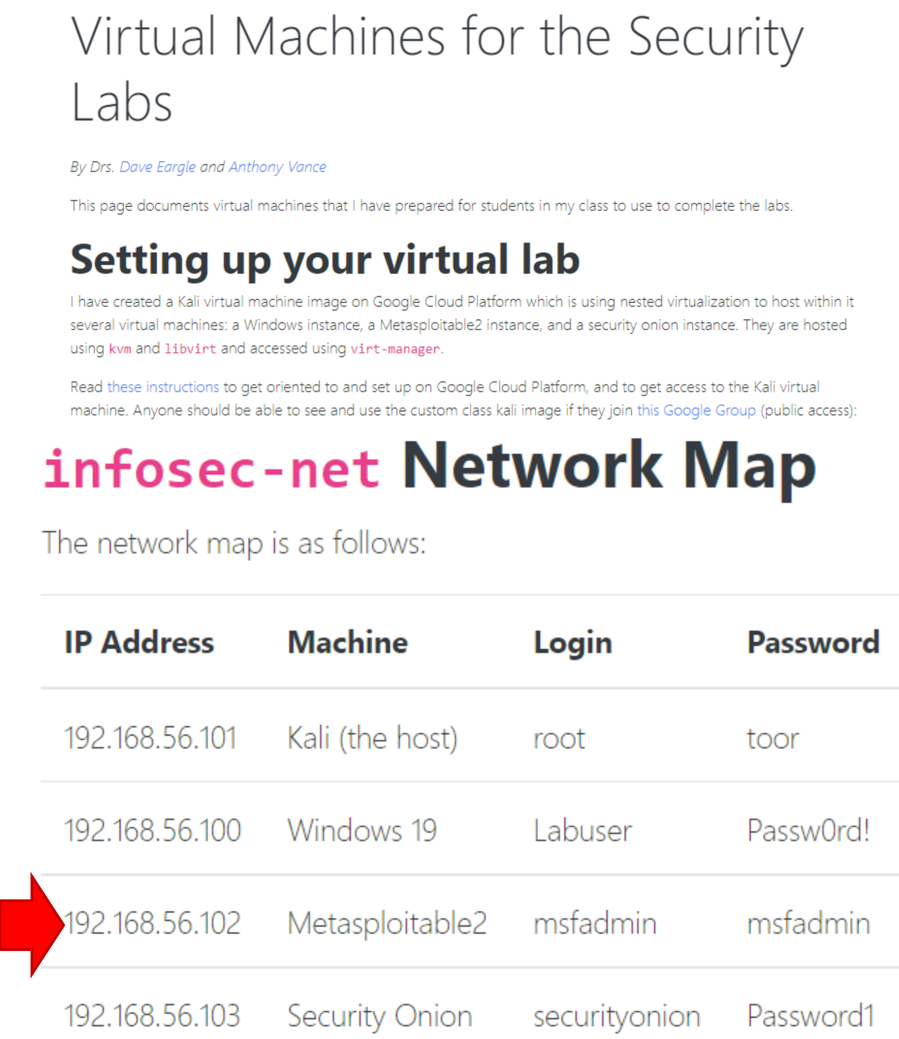

Virtual Machine Manager
File Edit View Help
Open
Name
Show the virtual machine console and details
CPU usage
QEMU/KVM
lab-metasploitable2_default
Running
lab-security-onion_default
Shutoff
lab-windows-2019-vuln_default
Shutoff
lab-metasploitable2_default on QEMU/KVM
File Virtual Machine View Send Key

```
 * Starting deferred execution scheduler atd                            [ OK ]
 * Starting periodic command scheduler crond                            [ OK ]
 * Starting Tomcat servlet engine tomcat5.5                             [ OK ]
 * Starting web server apache2                                          [ OK ]
 * Running local boot scripts (/etc/rc.local)
nohup: appending output to `nohup.out'
nohup: appending output to `nohup.out'
                                                                        [ OK ]

                _                  _       _ _        _     _      ____
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|


Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login:
```


Press Control_L+Alt_L to release pointer. lab-metasploitable2_default on QEMU/KVM
File
Virtual Machine
View
Send Key

```
Login with msfadmin/msfadmin to get started


metasploitable login: mfsadmin
Password:

Login incorrect
metasploitable login: msfadmin
Password:
Last login: Thu Aug 12 16:00:56 EDT 2021 on tty1
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ echo 'hello world!'
hello world!
msfadmin@metasploitable:~$ clear_
```

Find the IP address of your kali and metasploitable2 machines…



Virtual Machines for the Security Labs

By Drs. Dave Eargle and Anthony Vance

This page documents virtual machines that I have prepared for students in my class to use to complete the labs.

Setting up your virtual lab

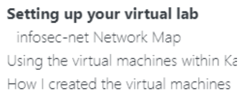
I have created a Kali virtual machine image on Google Cloud Platform which is using nested virtualization to host within it several virtual machines: a Windows instance, a Metasploitable2 instance, and a security onion instance. They are hosted using `kvm` and `libvirt` and accessed using `virt-manager`.

Read these instructions to get oriented to and set up on Google Cloud Platform, and to get access to the Kali virtual machine. Anyone should be able to see and use the custom class kali image if they join this Google Group (public access):

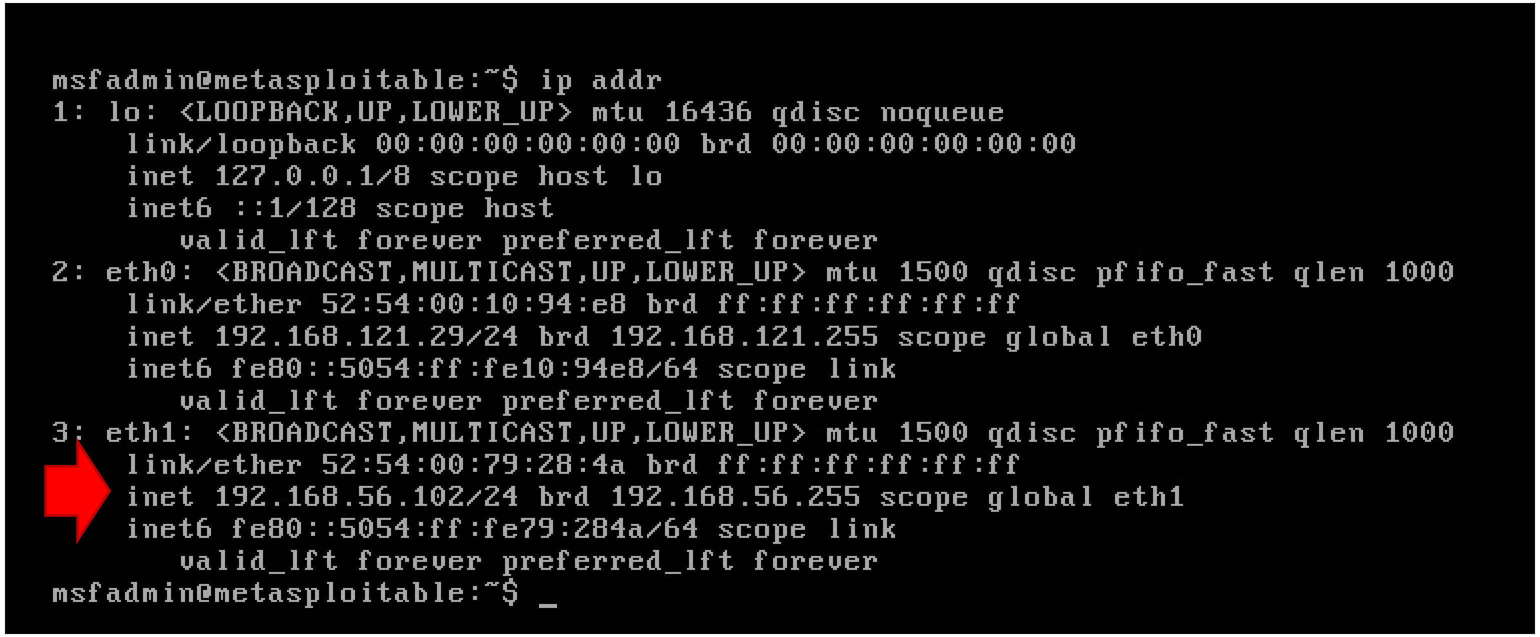
`infosec-net` Network Map

The network map is as follows:

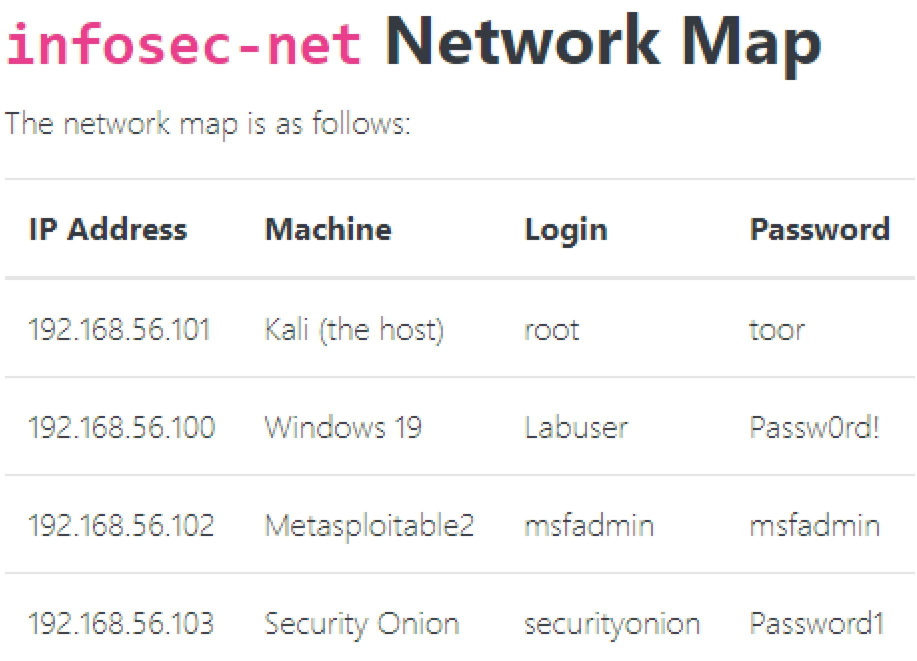
IP Address	Machine	Login	Password
192.168.56.101	Kali (the host)	root	toor
192.168.56.100	Windows 19	Labuser	Passw0rd!
192.168.56.102	Metasploitable2	msfadmin	msfadmin
192.168.56.103	Security Onion	securityonion	Password1



- You can use “ip addr” to find IP address of your metasploitable2 and kali



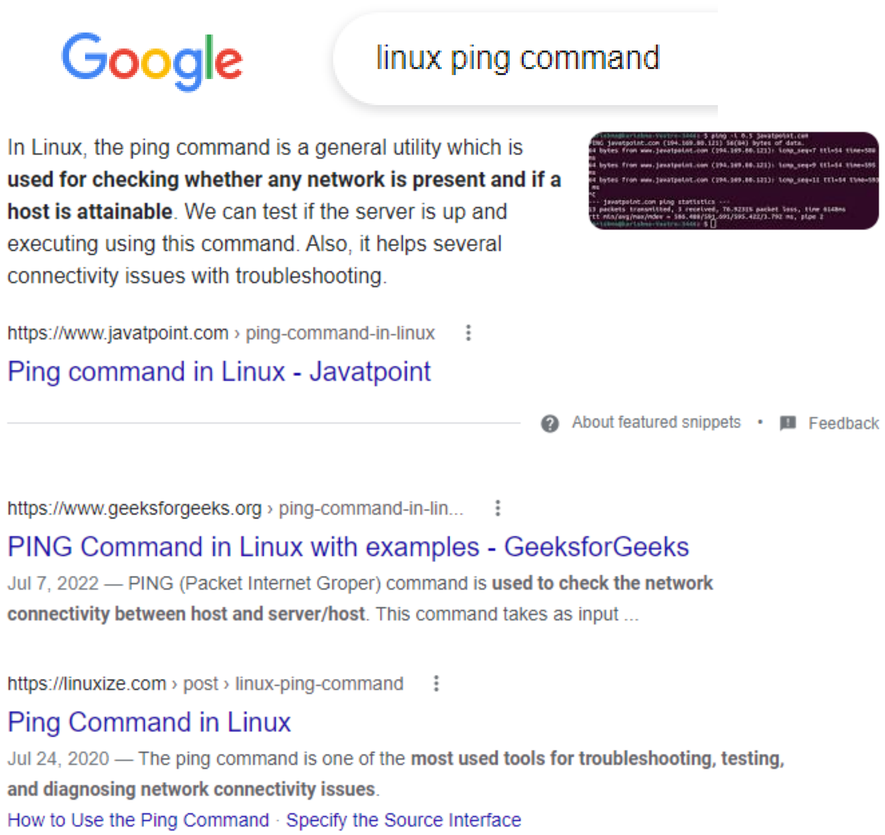
```
msfadmin@metasploitable:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 52:54:00:10:94:e8 brd ff:ff:ff:ff:ff:ff
    inet 192.168.121.29/24 brd 192.168.121.255 scope global eth0
    inet6 fe80::5054:ff:fe10:94e8/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 52:54:00:79:28:4a brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.102/24 brd 192.168.56.255 scope global eth1
    inet6 fe80::5054:ff:fe79:284a/64 scope link
       valid_lft forever preferred_lft forever
msfadmin@metasploitable:~$ _
```



infosec-net Network Map

The network map is as follows:

IP Address	Machine	Login	Password
192.168.56.101	Kali (the host)	root	toor
192.168.56.100	Windows 19	Labuser	Passw0rd!
192.168.56.102	Metasploitable2	msfadmin	msfadmin
192.168.56.103	Security Onion	securityonion	Password1



Google linux ping command

In Linux, the ping command is a general utility which is **used for checking whether any network is present and if a host is attainable**. We can test if the server is up and executing using this command. Also, it helps several connectivity issues with troubleshooting.

https://www.javatpoint.com › ping-command-in-linux

Ping command in Linux - Javatpoint

https://www.geeksforgeeks.org › ping-command-in-lin...

PING Command in Linux with examples - GeeksforGeeks

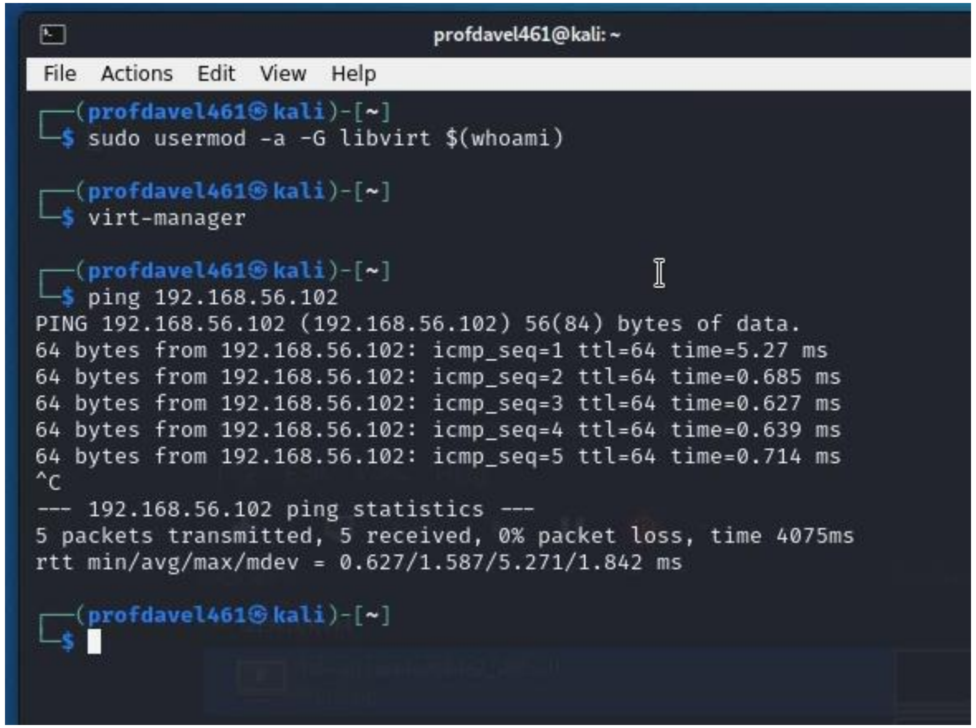
Jul 7, 2022 — PING (Packet Internet Groper) command is **used to check the network connectivity between host and server/host**. This command takes as input ...

https://linuxize.com › post › linux-ping-command

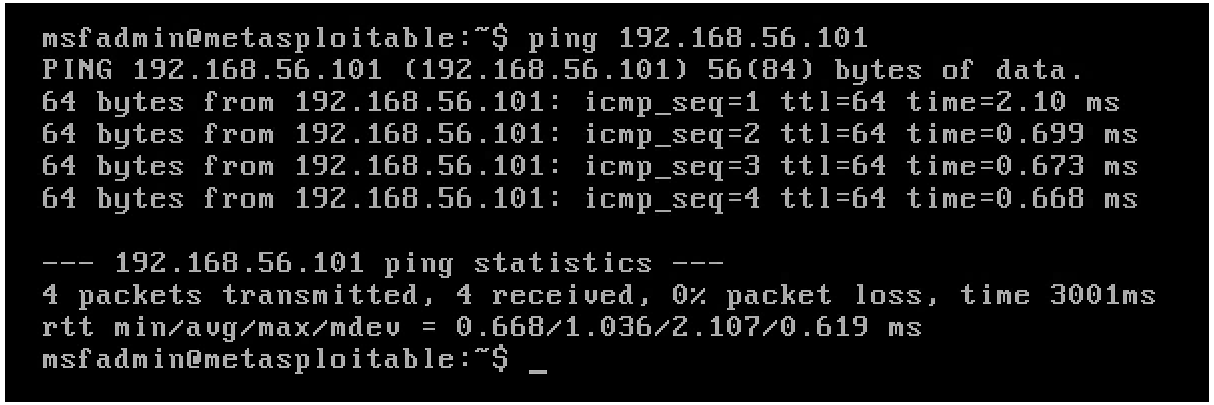
Ping Command in Linux

Jul 24, 2020 — The ping command is one of the **most used tools for troubleshooting, testing, and diagnosing network connectivity issues**.

How to Use the Ping Command · Specify the Source Interface



```
┌──(profdavel461㉿kali)-[~]
└─$ sudo usermod -a -G libvirt $(whoami)

┌──(profdavel461㉿kali)-[~]
└─$ virt-manager

┌──(profdavel461㉿kali)-[~]
└─$ ping 192.168.56.102
PING 192.168.56.102 (192.168.56.102) 56(84) bytes of data.
64 bytes from 192.168.56.102: icmp_seq=1 ttl=64 time=5.27 ms
64 bytes from 192.168.56.102: icmp_seq=2 ttl=64 time=0.685 ms
64 bytes from 192.168.56.102: icmp_seq=3 ttl=64 time=0.627 ms
64 bytes from 192.168.56.102: icmp_seq=4 ttl=64 time=0.639 ms
64 bytes from 192.168.56.102: icmp_seq=5 ttl=64 time=0.714 ms
^C
--- 192.168.56.102 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4075ms
rtt min/avg/max/mdev = 0.627/1.587/5.271/1.842 ms

┌──(profdavel461㉿kali)-[~]
└─$ 
```

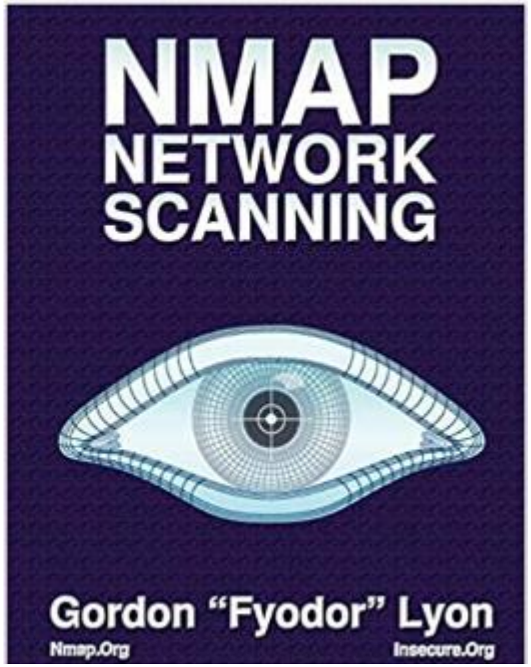


```
msfadmin@metasploitable:~$ ping 192.168.56.101
PING 192.168.56.101 (192.168.56.101) 56(84) bytes of data.
64 bytes from 192.168.56.101: icmp_seq=1 ttl=64 time=2.10 ms
64 bytes from 192.168.56.101: icmp_seq=2 ttl=64 time=0.699 ms
64 bytes from 192.168.56.101: icmp_seq=3 ttl=64 time=0.673 ms
64 bytes from 192.168.56.101: icmp_seq=4 ttl=64 time=0.668 ms

--- 192.168.56.101 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3001ms
rtt min/avg/max/mdev = 0.668/1.036/2.107/0.619 ms
msfadmin@metasploitable:~$ _
```

Note: If you lose your mouse…

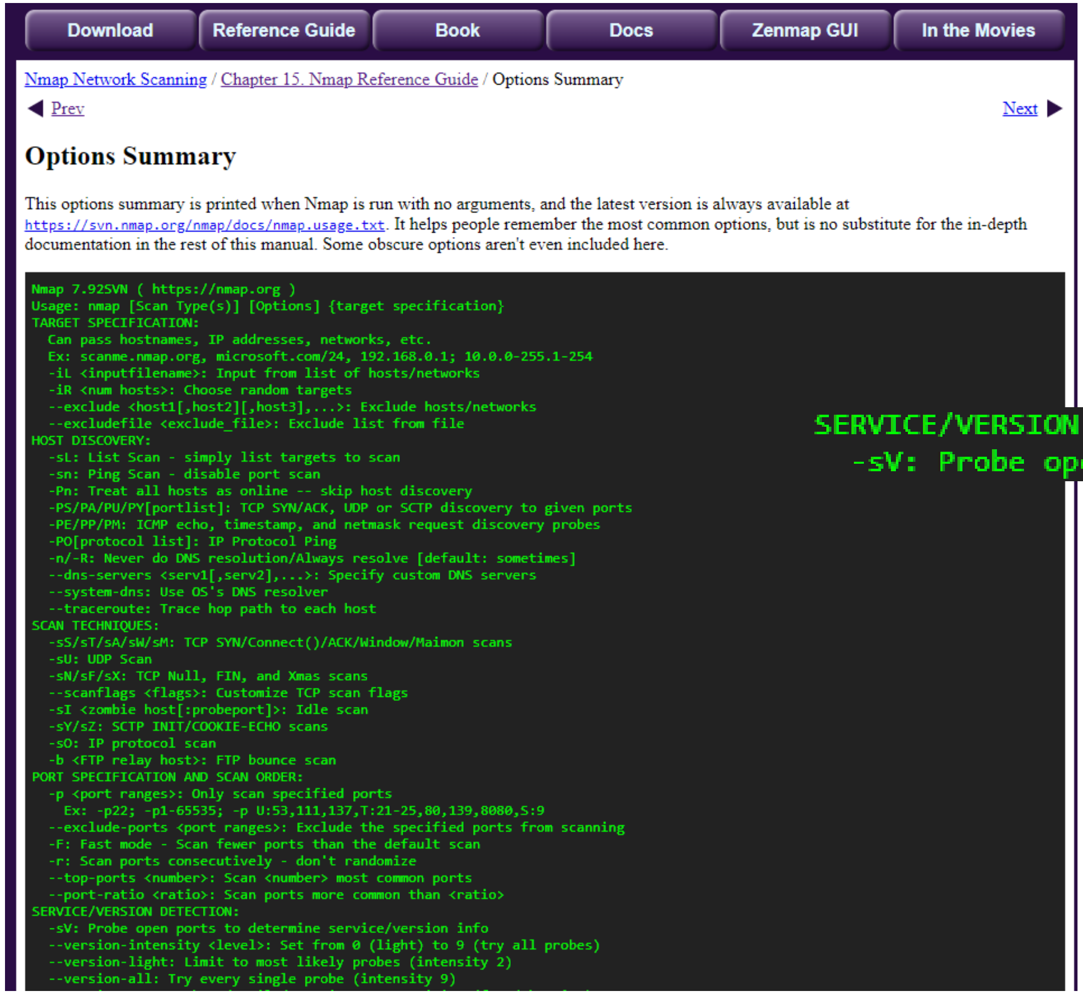
- …as you move between your computer’s environment and your kali linux window and your virtual machine windows

- Try to hit Ctrl & Alt keys simultaneously



https://nmap.org/book/man.html

https://nmap.org/book/man-briefoptions.html



Nmap Network Scanning / Chapter 15. Nmap Reference Guide / Options Summary

◀ Prev Next ▶

Options Summary

This options summary is printed when Nmap is run with no arguments, and the latest version is always available at https://svn.nmap.org/nmap/docs/nmap.usage.txt. It helps people remember the most common options, but is no substitute for the in-depth documentation in the rest of this manual. Some obscure options aren't even included here.

```
Nmap 7.92SVN ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PU/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[protocol list]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host
SCAN TECHNIQUES:
  -sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
  -sU: UDP Scan
  -sN/sF/sX: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -sI <zombie host[:probeport]>: Idle scan
  -sY/sZ: SCTP INIT/COOKIE-ECHO scans
  -sO: IP protocol scan
  -b <FTP relay host>: FTP bounce scan
PORT SPECIFICATION AND SCAN ORDER:
  -p <port ranges>: Only scan specified ports
    Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9
  --exclude-ports <port ranges>: Exclude the specified ports from scanning
  -F: Fast mode - Scan fewer ports than the default scan
  -r: Scan ports consecutively - don't randomize
  --top-ports <number>: Scan <number> most common ports
  --port-ratio <ratio>: Scan ports more common than <ratio>
SERVICE/VERSION DETECTION:
  -sV: Probe open ports to determine service/version info
  --version-intensity <level>: Set from 0 (light) to 9 (try all probes)
  --version-light: Limit to most likely probes (intensity 2)
  --version-all: Try every single probe (intensity 9)
```

```
SERVICE/VERSION DETECTION:
  -sV: Probe open ports to determine service/version info
```

Part 1. Host Discovery and Scanning using NMAP

NMAP is the de facto standard of host discovery and port scanning and has a host of features that make the tool very robust. In this section of the lab, you'll try a few of NMAP's features.

> Throughout the lab, you should replace `<IP.addr.of.metasploitable2>` with the actual IPv4 address of Metasploitable.

1. Open a "Terminal Emulator" window in Kali.
2. Run all nmap commands as root – you'll get more information as root for some commands.
    - "Get root" in your shell (i.e., `sudo -s` or `su root`).
3. Run `nmap`. Take a quick look at the available options.
4. Use `nmap` to determine whether the your Metasploitable2 VM is live using a "ping scan":

    ```
    nmap -sn <IP.addr.of.metasploitable2>
    ```

    The ping scan not only sends an ICMP request, but also an ARP ping, TCP pinging, and other techniques to determine if a host is live on the network.

    > Question 1: What kind of information is shown when you run this ping scan for Metasploitable2?

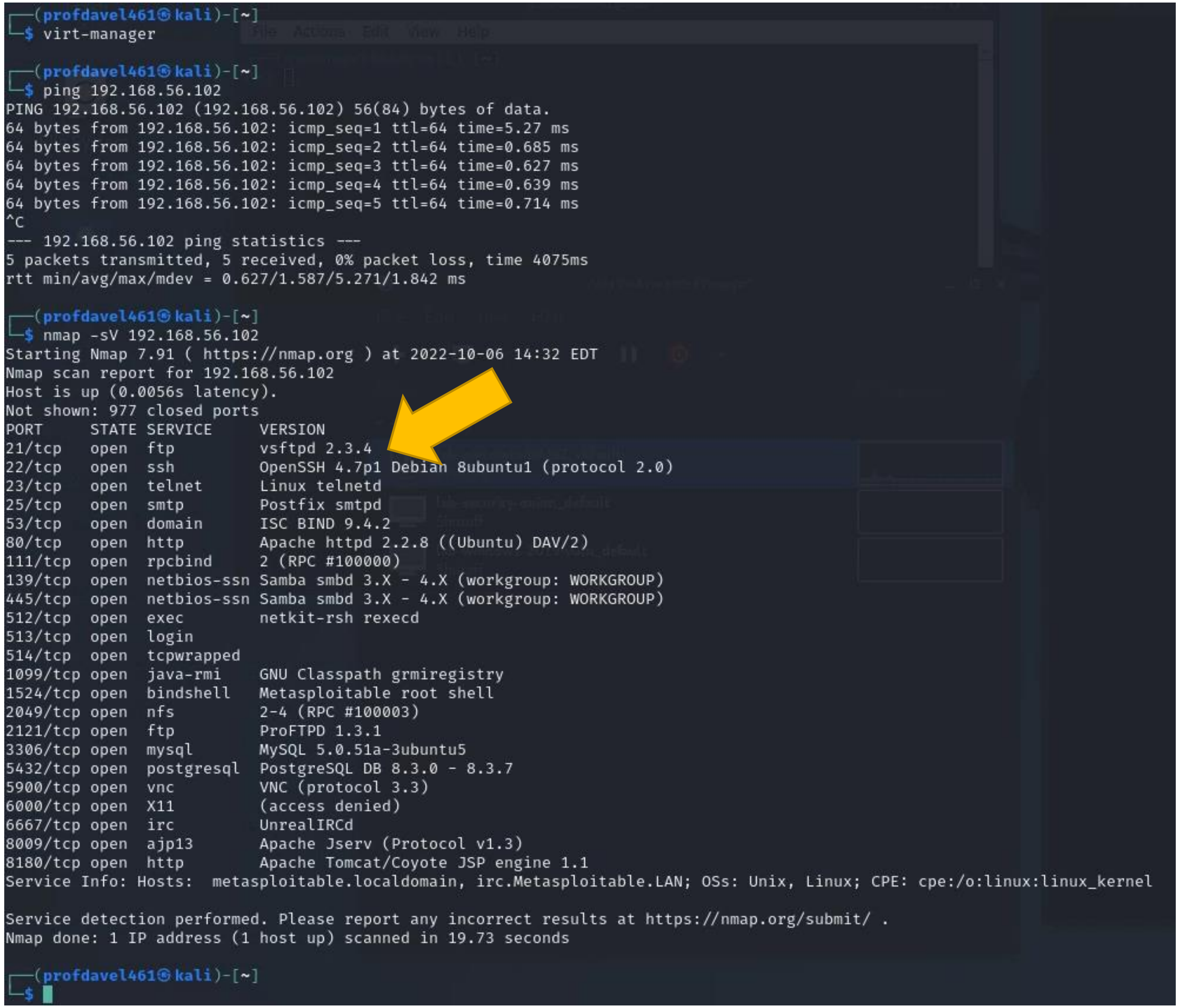
    You could also scan a *range* of IPs using CIDR block notation. See the network map for the ipv4 block of the infosec-net network. This can be fun to do if you also have your vulnerable Windows 7 vm running at the same time, although this is not required.

    ```
    nmap -sn <ipv4 CIDR block>
    ```

    You can know your network by typing `ifconfig` on either Kali or Metasploitable2, and looking for the `inet` address plus the `mask` value on the same line. For example, a "mask" of `255.255.255.0` applied to an "inet" address of 192.168.56.17 translates to a network of 192.168.56.0/ 24. (Where 24 is the number of bits to mask and it takes 8 bits to make 255, 8x3=24, so that would mask three of the '.' blocks.)

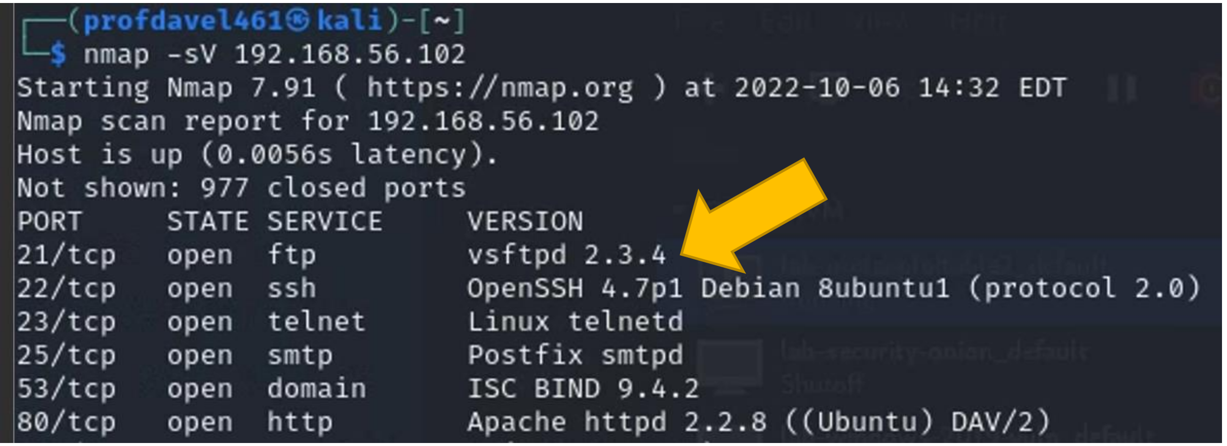
5. Once you determine that a host is live, you can use NMAP to scan for open ports. Use a TCP scan to determine which ports are open on Metasploitable2:

    ```
    nmap -sS <IP.addr.of.metasploitable2>
    ```

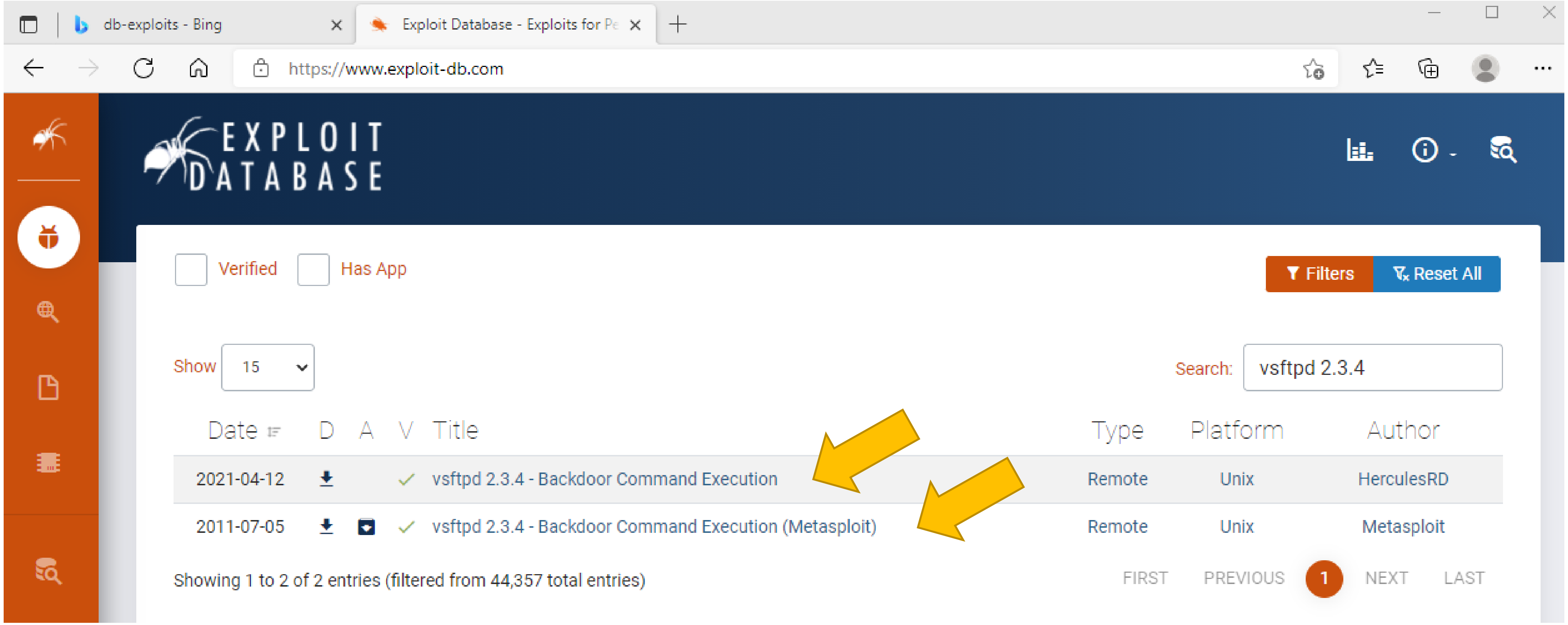


```
┌──(profdavel461㉿kali)-[~]
└─$ virt-manager

┌──(profdavel461㉿kali)-[~]
└─$ ping 192.168.56.102
PING 192.168.56.102 (192.168.56.102) 56(84) bytes of data.
64 bytes from 192.168.56.102: icmp_seq=1 ttl=64 time=5.27 ms
64 bytes from 192.168.56.102: icmp_seq=2 ttl=64 time=0.685 ms
64 bytes from 192.168.56.102: icmp_seq=3 ttl=64 time=0.627 ms
64 bytes from 192.168.56.102: icmp_seq=4 ttl=64 time=0.639 ms
64 bytes from 192.168.56.102: icmp_seq=5 ttl=64 time=0.714 ms
^C
--- 192.168.56.102 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4075ms
rtt min/avg/max/mdev = 0.627/1.587/5.271/1.842 ms

┌──(profdavel461㉿kali)-[~]
└─$ nmap -sV 192.168.56.102
Starting Nmap 7.91 ( https://nmap.org ) at 2022-10-06 14:32 EDT
Nmap scan report for 192.168.56.102
Host is up (0.0056s latency).
Not shown: 977 closed ports
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login
514/tcp  open  tcpwrapped
1099/tcp open  java-rmi    GNU Classpath grmiregistry
1524/tcp open  bindshell   Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         UnrealIRCd
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.73 seconds

┌──(profdavel461㉿kali)-[~]
└─$ 
```

Looking for vulnerable services...



```
┌──(profdavel461㉿kali)-[~]
└─$ nmap -sV 192.168.56.102
Starting Nmap 7.91 ( https://nmap.org ) at 2022-10-06 14:32 EDT
Nmap scan report for 192.168.56.102
Host is up (0.0056s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE     VERSION
21/tcp    open  ftp         vsftpd 2.3.4
22/tcp    open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet      Linux telnetd
25/tcp    open  smtp        Postfix smtpd
53/tcp    open  domain      ISC BIND 9.4.2
80/tcp    open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
```



Show 15

Search: vsftpd 2.3.4

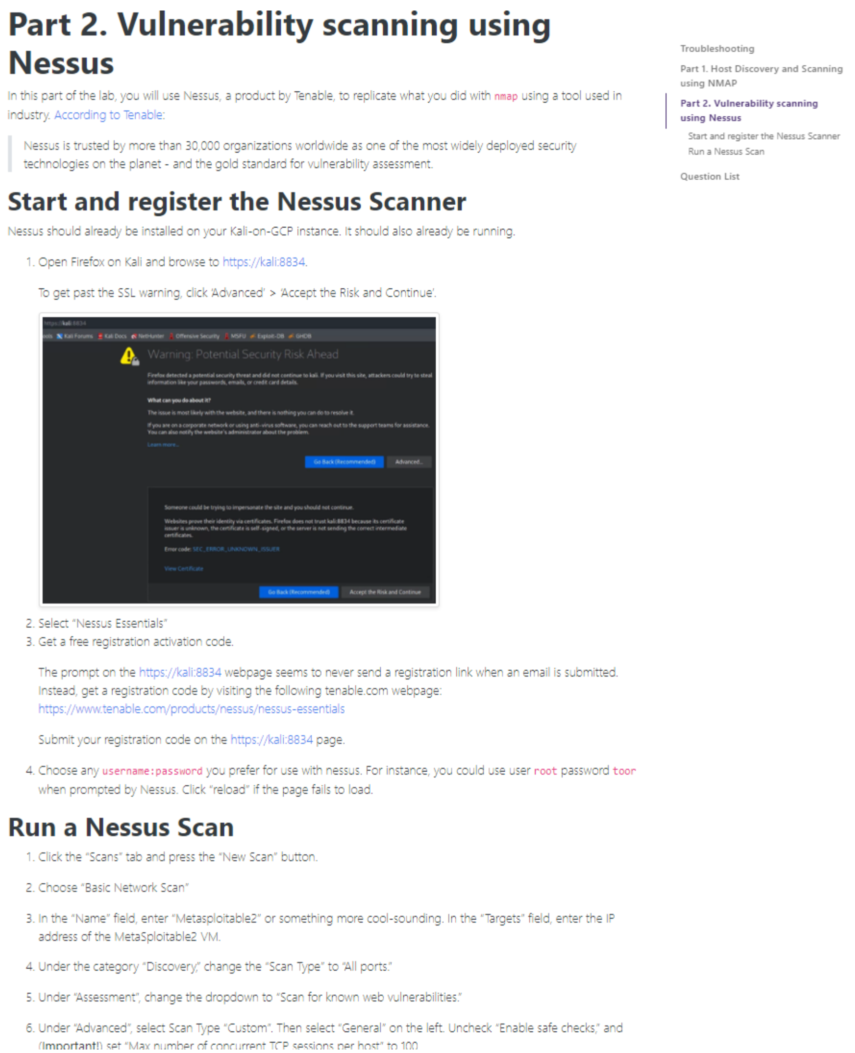
Date	D	A	V	Title	Type	Platform	Author
2021-04-12	⬇		✓	vsftpd 2.3.4 - Backdoor Command Execution	Remote	Unix	HerculesRD
2011-07-05	⬇	▣	✓	vsftpd 2.3.4 - Backdoor Command Execution (Metasploit)	Remote	Unix	Metasploit

Showing 1 to 2 of 2 entries (filtered from 44,357 total entries)

To run the Nessus portion of the vulnerability scanning lab…

You will need to complete the install and startup of Nessus

1. Startup Nessus Essentials scanner
2. Request and install your Nessus license key
3. Setup Nessus scan
4. Run Nessus scan…



Part 2. Vulnerability scanning using Nessus

In this part of the lab, you will use Nessus, a product by Tenable, to replicate what you did with `nmap` using a tool used in industry. According to Tenable:

> Nessus is trusted by more than 30,000 organizations worldwide as one of the most widely deployed security technologies on the planet - and the gold standard for vulnerability assessment.

Start and register the Nessus Scanner

Nessus should already be installed on your Kali-on-GCP instance. It should also already be running.

1. Open Firefox on Kali and browse to https://kali:8834.

   To get past the SSL warning, click 'Advanced' > 'Accept the Risk and Continue'.

   https://kali:8834

   Warning: Potential Security Risk Ahead

   Firefox detected a potential security threat and did not continue to kali. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

   What can you do about it?

   The issue is most likely with the website, and there is nothing you can do to resolve it.

   If you are on a corporate network or using anti-virus software, you can reach out to the support teams for assistance. You can also notify the website's administrator about the problem.

   Someone could be trying to impersonate the site and you should not continue.

   Websites prove their identity via certificates. Firefox does not trust kali:8834 because its certificate issuer is unknown, the certificate is self-signed, or the server is not sending the correct intermediate certificates.

   Error code: SEC_ERROR_UNKNOWN_ISSUER

2. Select "Nessus Essentials"
3. Get a free registration activation code.

   The prompt on the https://kali:8834 webpage seems to never send a registration link when an email is submitted. Instead, get a registration code by visiting the following tenable.com webpage: https://www.tenable.com/products/nessus/nessus-essentials

   Submit your registration code on the https://kali:8834 page.

4. Choose any `username:password` you prefer for use with nessus. For instance, you could use user `root` password `toor` when prompted by Nessus. Click "reload" if the page fails to load.

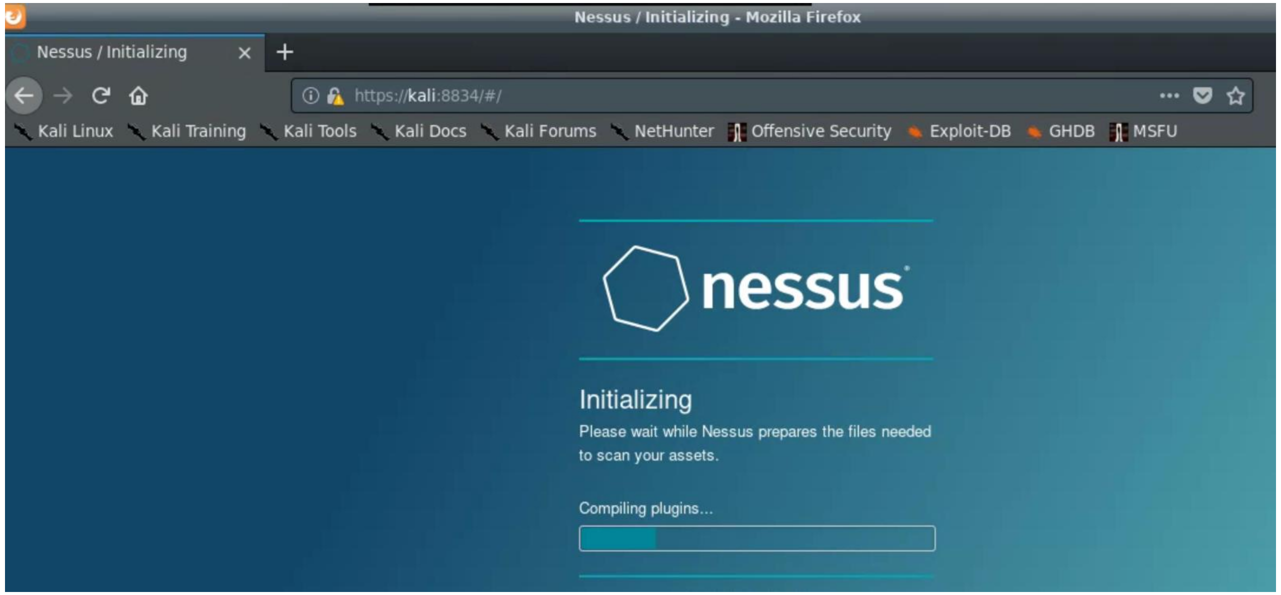
Run a Nessus Scan

1. Click the "Scans" tab and press the "New Scan" button.
2. Choose "Basic Network Scan"
3. In the "Name" field, enter "Metasploitable2" or something more cool-sounding. In the "Targets" field, enter the IP address of the MetaSploitable2 VM.
4. Under the category "Discovery," change the "Scan Type" to "All ports."
5. Under "Assessment", change the dropdown to "Scan for known web vulnerabilities."
6. Under "Advanced", select Scan Type "Custom". Then select "General" on the left. Uncheck "Enable safe checks," and (**Important!**) set "Max number of concurrent TCP sessions per host" to 100

Troubleshooting
Part 1. Host Discovery and Scanning using NMAP
Part 2. Vulnerability scanning using Nessus
Start and register the Nessus Scanner
Run a Nessus Scan
Question List

Starting up Nessus Essentials

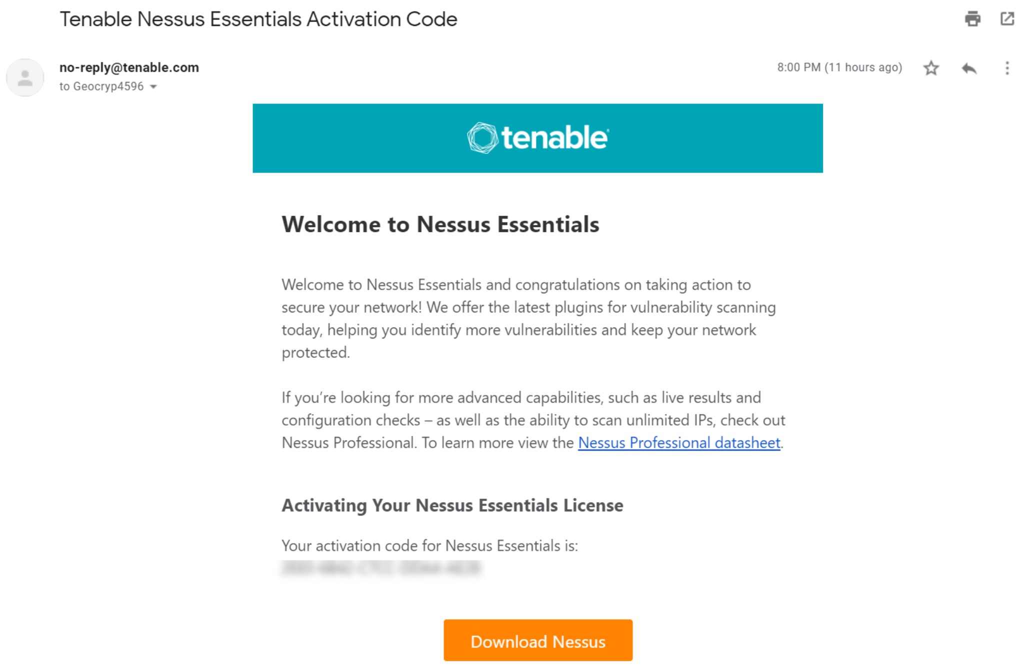
- In Kali, bring up Firefox browser
- Navigate to https://kali:8834 (Nessus is installed and listening on port 8834)
- Request and provide your Nessus activation code, it will show up by email



Initializing

Please wait while Nessus prepares the files needed to scan your assets.

Compiling plugins...



Tenable Nessus Essentials Activation Code

no-reply@tenable.com
to Geocryp4596

8:00 PM (11 hours ago)

Welcome to Nessus Essentials

Welcome to Nessus Essentials and congratulations on taking action to secure your network! We offer the latest plugins for vulnerability scanning today, helping you identify more vulnerabilities and keep your network protected.

If you're looking for more advanced capabilities, such as live results and configuration checks – as well as the ability to scan unlimited IPs, check out Nessus Professional. To learn more view the Nessus Professional datasheet.

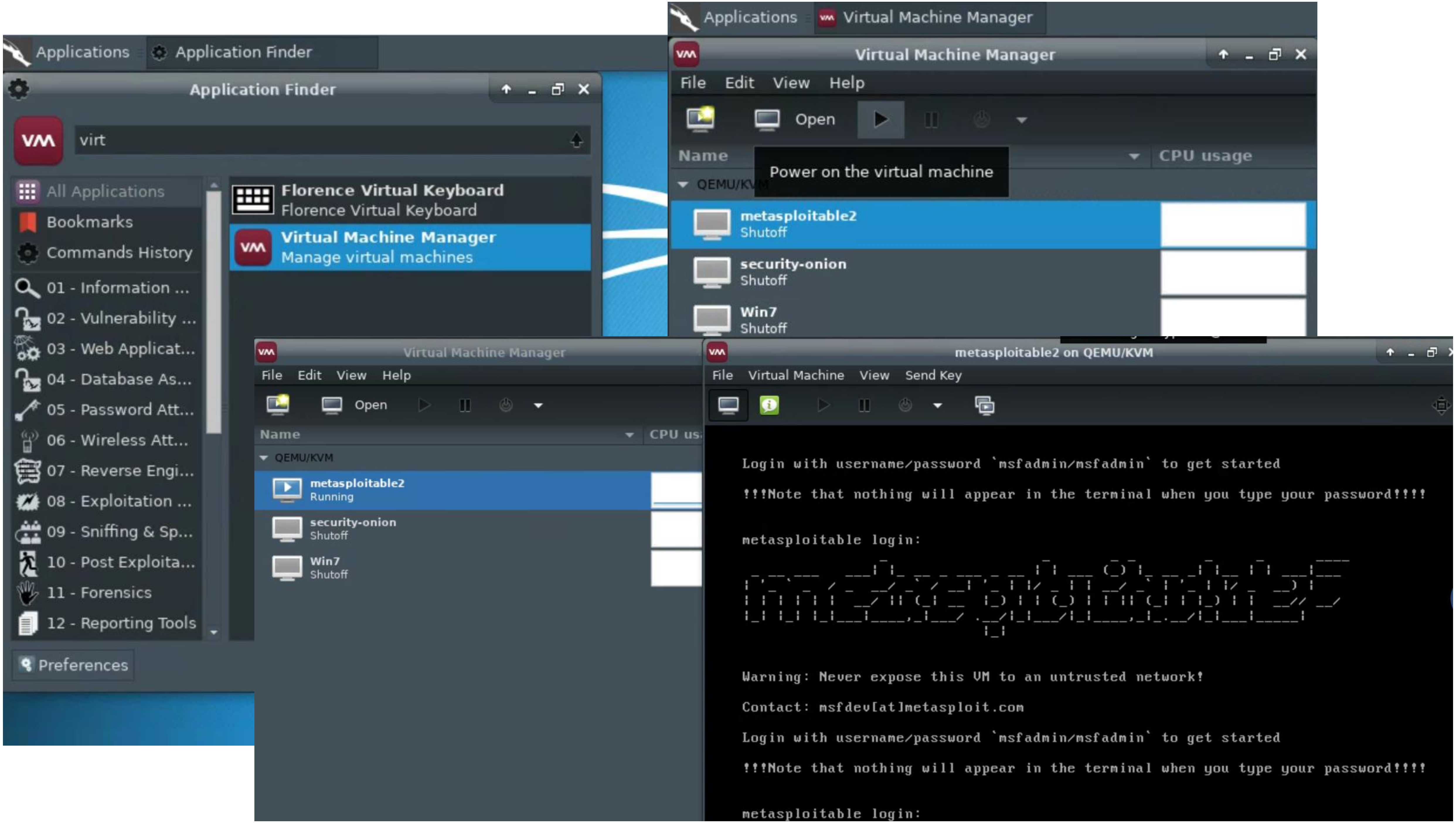
Activating Your Nessus Essentials License

Your activation code for Nessus Essentials is:

[redacted]

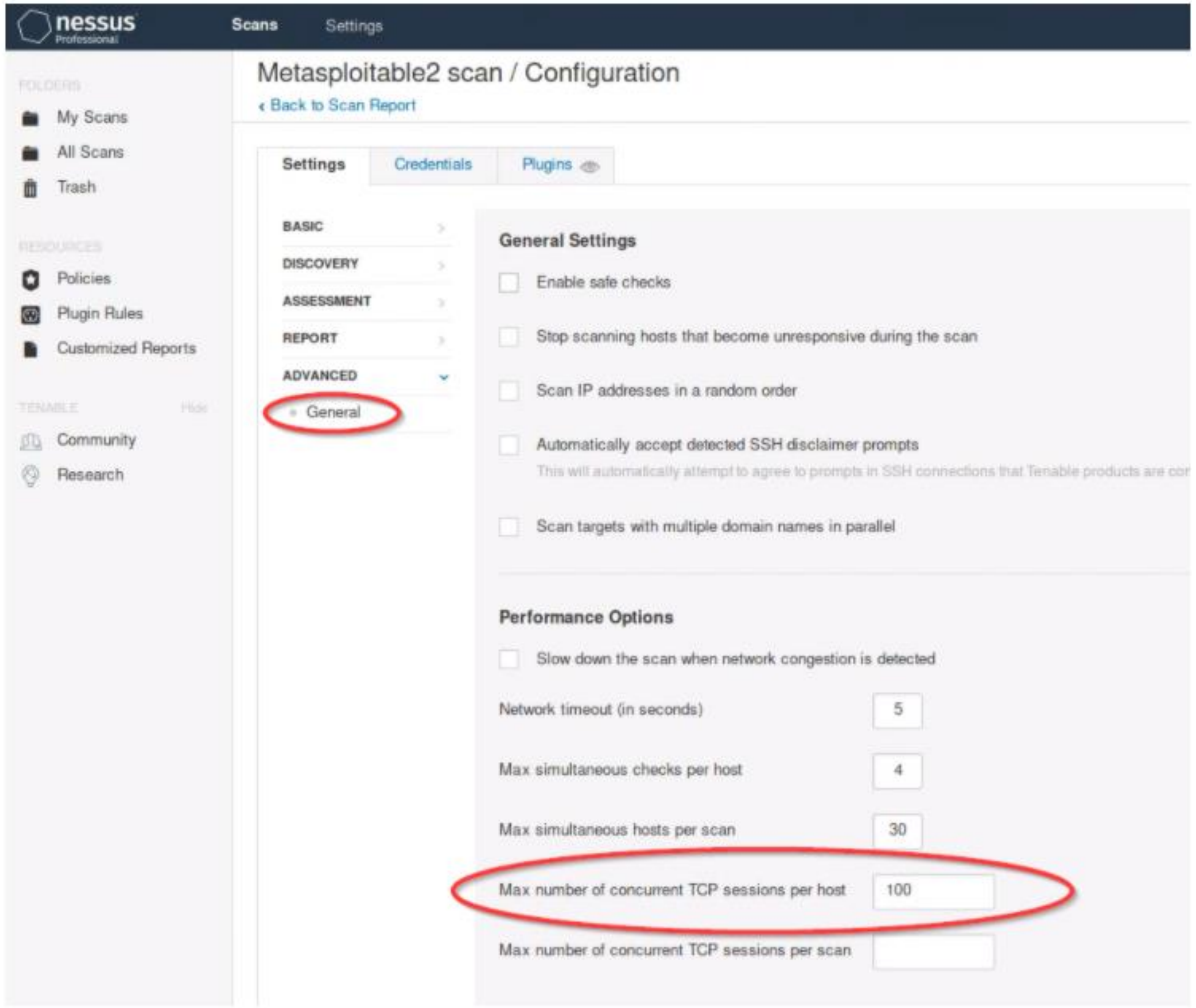
To run the Nessus portion of the vulnerability scanning lab…

- You will need to complete the install and startup of Nessus
  1. Startup Nessus Essentials scanner
  2. Request and install your Nessus license key
  3. Start up Metaspolitable2
  4. Setup Nessus scan
  5. Run Nessus scan…



```
Login with username/password `msfadmin/msfadmin` to get started
!!!Note that nothing will appear in the terminal when you type your password!!!!

metasploitable login:

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with username/password `msfadmin/msfadmin` to get started

!!!Note that nothing will appear in the terminal when you type your password!!!!

metasploitable login:
```

Follow lab instructions to create a vulnerability scan of Metasploitable2



Metasploitable2 scan / Configuration

‹ Back to Scan Report

General Settings

□ Enable safe checks

□ Stop scanning hosts that become unresponsive during the scan

□ Scan IP addresses in a random order

□ Automatically accept detected SSH disclaimer prompts

This will automatically attempt to agree to prompts in SSH connections that Tenable products are cor

□ Scan targets with multiple domain names in parallel

Performance Options

□ Slow down the scan when network congestion is detected

Network timeout (in seconds) 5

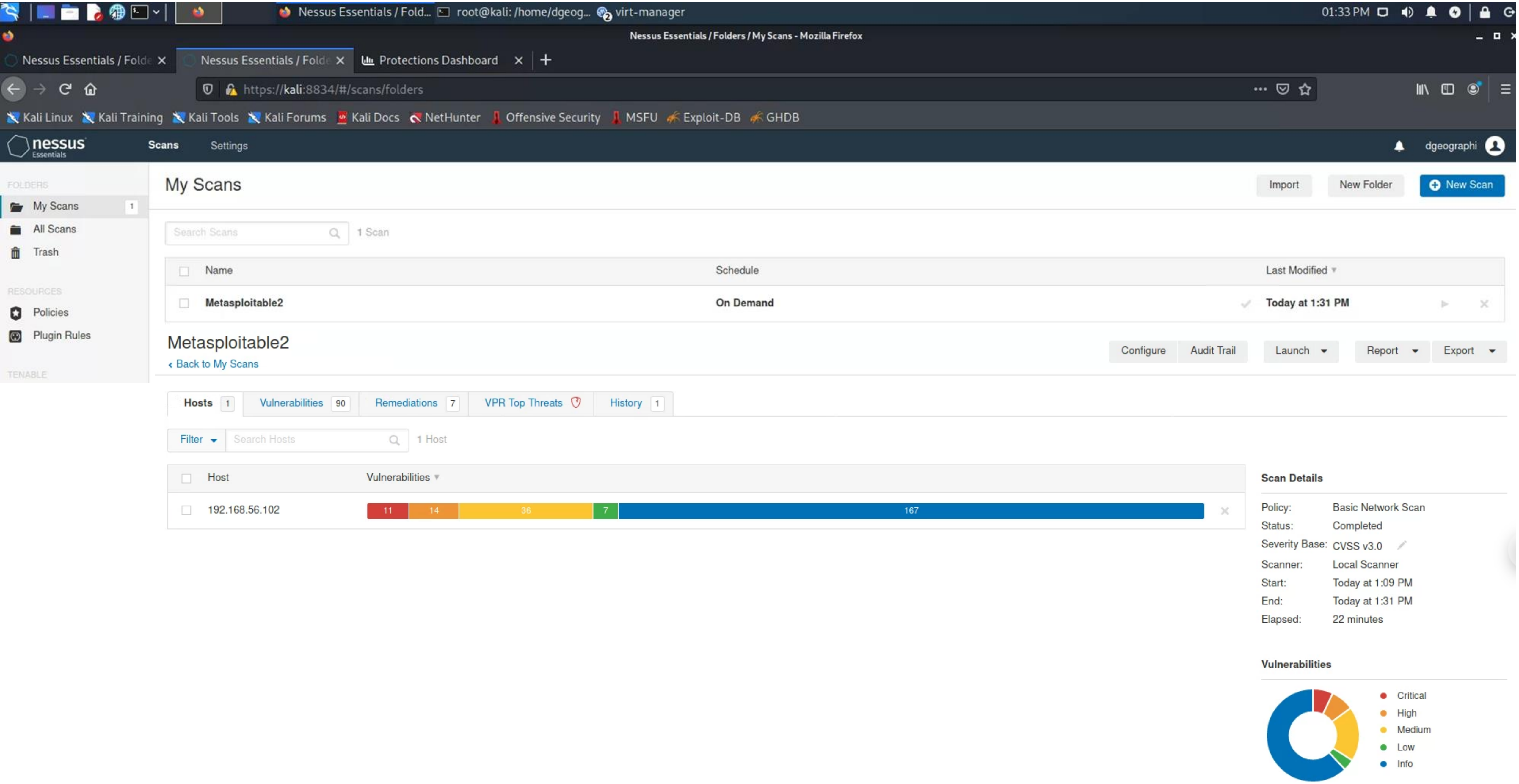
Max simultaneous checks per host 4

Max simultaneous hosts per scan 30

Max number of concurrent TCP sessions per host 100

Max number of concurrent TCP sessions per scan

Run the Nessus computer vulnerability scan (it may take ~20 - 30+ minutes)…



My Scans

1 Scan

Name	Schedule	Last Modified
Metasploitable2	On Demand	Today at 1:31 PM

Metasploitable2

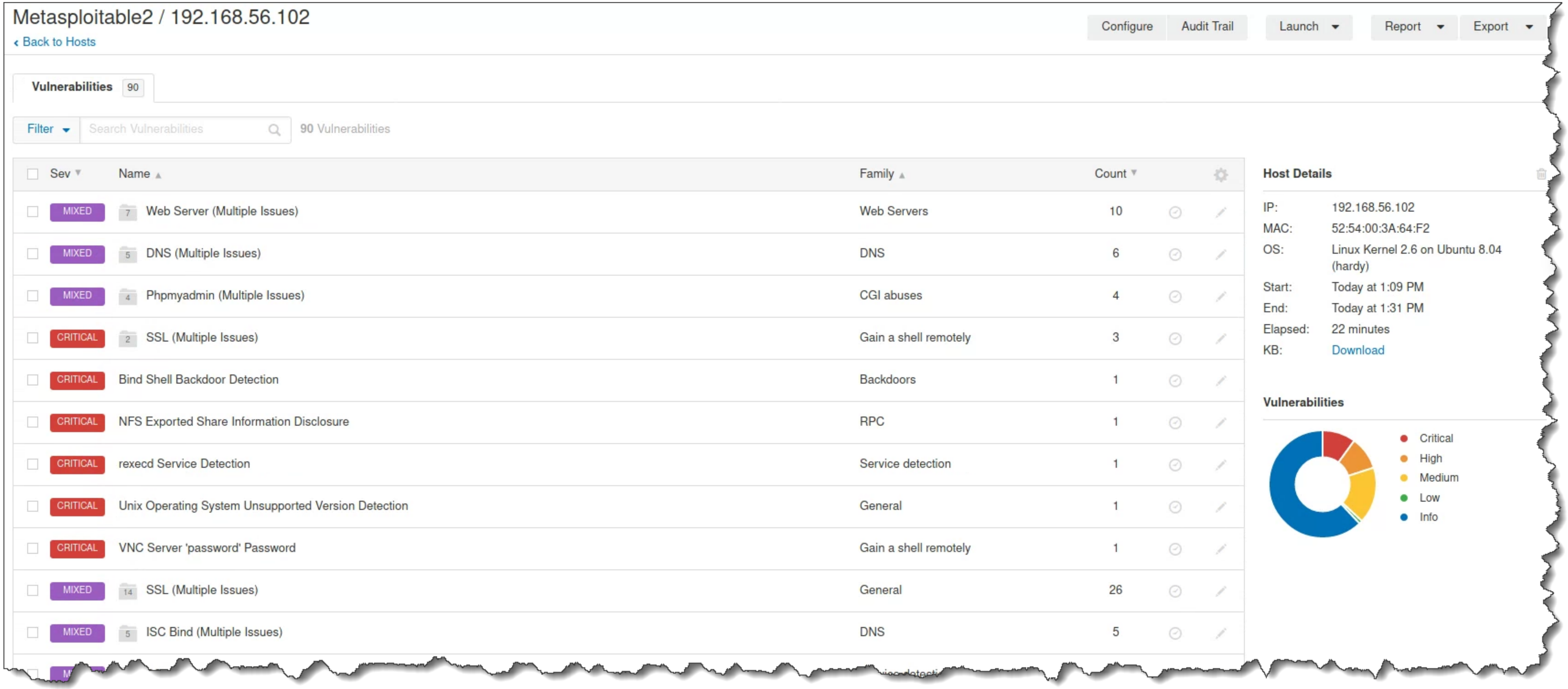
1 Host

Host	Vulnerabilities
192.168.56.102	11 14 36 7 167

Scan Details

Policy:	Basic Network Scan
Status:	Completed
Severity Base:	CVSS v3.0
Scanner:	Local Scanner
Start:	Today at 1:09 PM
End:	Today at 1:31 PM
Elapsed:	22 minutes

Vulnerabilities



Metasploitable2 / 192.168.56.102

‹ Back to Hosts

Vulnerabilities 90

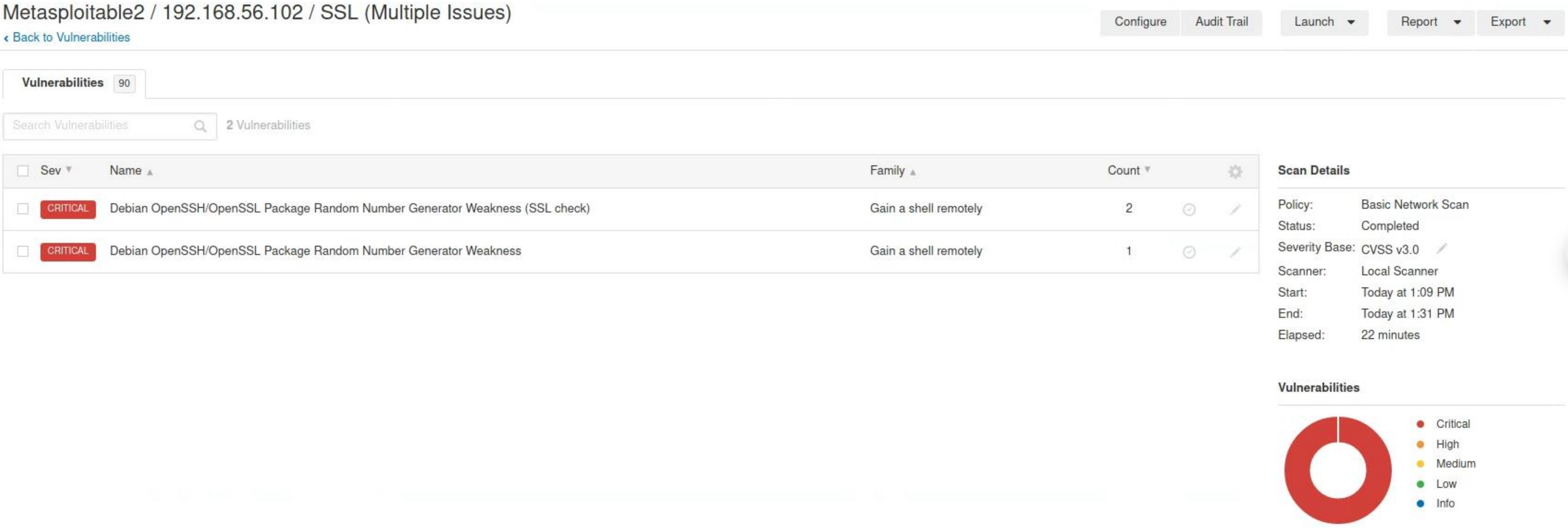
90 Vulnerabilities

□	Sev	Name	Family	Count
□	MIXED	7 Web Server (Multiple Issues)	Web Servers	10
□	MIXED	5 DNS (Multiple Issues)	DNS	6
□	MIXED	4 Phpmyadmin (Multiple Issues)	CGI abuses	4
□	CRITICAL	2 SSL (Multiple Issues)	Gain a shell remotely	3
□	CRITICAL	Bind Shell Backdoor Detection	Backdoors	1
□	CRITICAL	NFS Exported Share Information Disclosure	RPC	1
□	CRITICAL	rexecd Service Detection	Service detection	1
□	CRITICAL	Unix Operating System Unsupported Version Detection	General	1
□	CRITICAL	VNC Server 'password' Password	Gain a shell remotely	1
□	MIXED	14 SSL (Multiple Issues)	General	26
□	MIXED	5 ISC Bind (Multiple Issues)	DNS	5

Host Details

IP:	192.168.56.102
MAC:	52:54:00:3A:64:F2
OS:	Linux Kernel 2.6 on Ubuntu 8.04 (hardy)
Start:	Today at 1:09 PM
End:	Today at 1:31 PM
Elapsed:	22 minutes
KB:	Download

Vulnerabilities



Metasploitable2 / 192.168.56.102 / SSL (Multiple Issues)

‹ Back to Vulnerabilities

2 Vulnerabilities

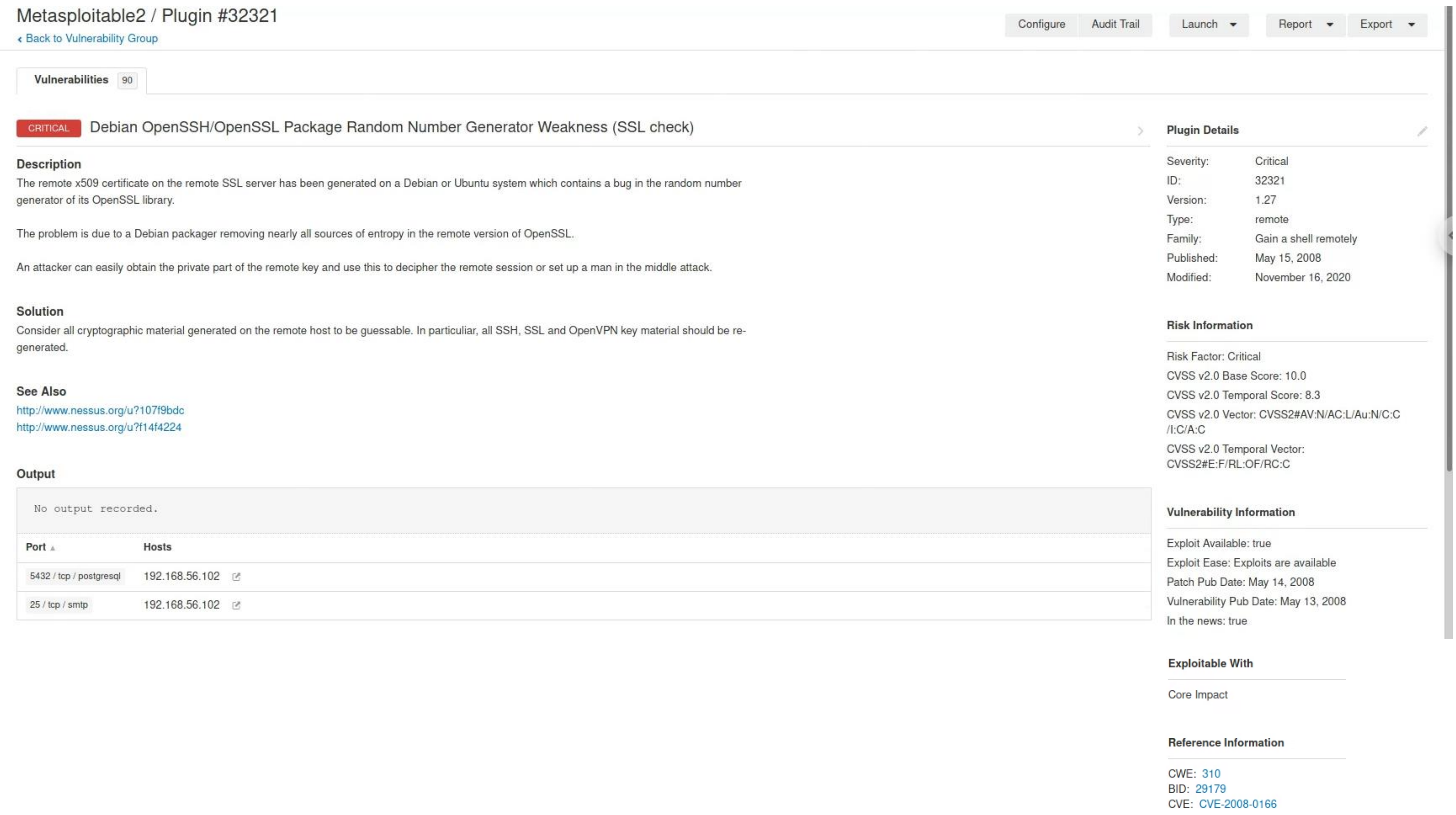
□	Sev	Name	Family	Count
□	CRITICAL	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)	Gain a shell remotely	2
□	CRITICAL	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness	Gain a shell remotely	1

Scan Details

Policy:	Basic Network Scan
Status:	Completed
Severity Base:	CVSS v3.0
Scanner:	Local Scanner
Start:	Today at 1:09 PM
End:	Today at 1:31 PM
Elapsed:	22 minutes

Vulnerabilities

- ● Critical
- ● High
- ● Medium
- ● Low
- ● Info



Metasploitable2 / Plugin #32321

CRITICAL Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)

Description

The remote x509 certificate on the remote SSL server has been generated on a Debian or Ubuntu system which contains a bug in the random number generator of its OpenSSL library.

The problem is due to a Debian packager removing nearly all sources of entropy in the remote version of OpenSSL.

An attacker can easily obtain the private part of the remote key and use this to decipher the remote session or set up a man in the middle attack.

Solution

Consider all cryptographic material generated on the remote host to be guessable. In particuliar, all SSH, SSL and OpenVPN key material should be re-generated.

See Also

http://www.nessus.org/u?107f9bdc
http://www.nessus.org/u?f14f4224

Output

```
No output recorded.
```

Port ▲	Hosts
5432 / tcp / postgresql	192.168.56.102
25 / tcp / smtp	192.168.56.102

Plugin Details

Severity:	Critical
ID:	32321
Version:	1.27
Type:	remote
Family:	Gain a shell remotely
Published:	May 15, 2008
Modified:	November 16, 2020

Risk Information

Risk Factor: Critical
CVSS v2.0 Base Score: 10.0
CVSS v2.0 Temporal Score: 8.3
CVSS v2.0 Vector: CVSS2#AV:N/AC:L/Au:N/C:C/I:C/A:C
CVSS v2.0 Temporal Vector: CVSS2#E:F/RL:OF/RC:C

Vulnerability Information

Exploit Available: true
Exploit Ease: Exploits are available
Patch Pub Date: May 14, 2008
Vulnerability Pub Date: May 13, 2008
In the news: true

Exploitable With

Core Impact

Reference Information

CWE: 310
BID: 29179
CVE: CVE-2008-0166

See Also… links

- https://lists.debian.org/debian-security-announce/2008/msg00152.html
- https://lists.ubuntu.com/archives/ubuntu-security-announce/2008-May/000705.html

Agenda

- ✓Change your Kali password!
- ✓Application vulnerability and security testing
- ✓Lab 8: Vulnerability Scanning – Part 2: Nessus
- ✓Scan results
- ✓Looking at a vulnerability