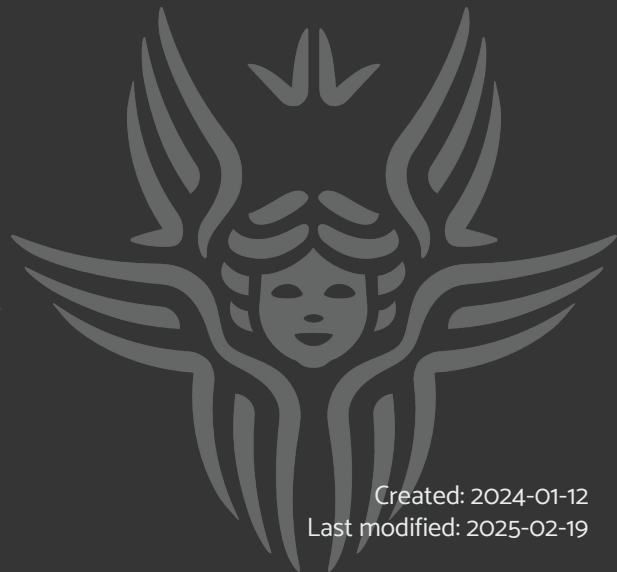
ETHL - Ethical Hacking Lab

0x00 - Introduction

Davide Guerri - davide[.]guerri AT uniroma1[.]it
Ethical Hacking Lab
Sapienza University of Rome - Department of Computer Science



Created: 2024-01-12
Last modified: 2025-02-19

ToC

1 Course Objectives

2 Students Assessment Criteria

3 Rules of Engagement

4 Understanding Security Testing

5 Kill chain & ATT&CK

6 How to Practice

Course Objectives

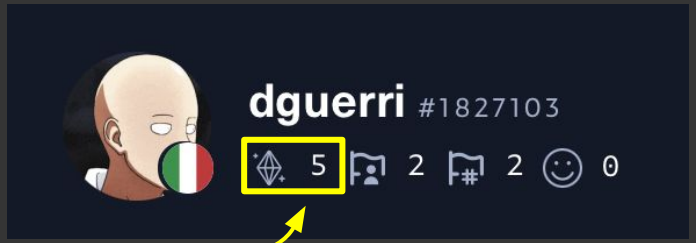
- Get a theoretical understanding of UNIX hacking techniques
  - Complementing the main course program
- Learn and practice hacking tools and techniques
  - Reinforce learning by testing real vulnerabilities
- Learn to think and act like a red-teamer and defend like a blue-teamer
  - Adversary's tactics, techniques, and procedures
  - Develop an adversarial mindset

Students Assessment Criteria

Info on ETH website - TL;DR:

- **[optional] Lab assignments: 49%** of final grade - more on this in a bit
- Final Written Exam: 51% of final grade
  - 3 questions on the main course topics
  - **2 questions on the lab topics**

Students Assessment Criteria

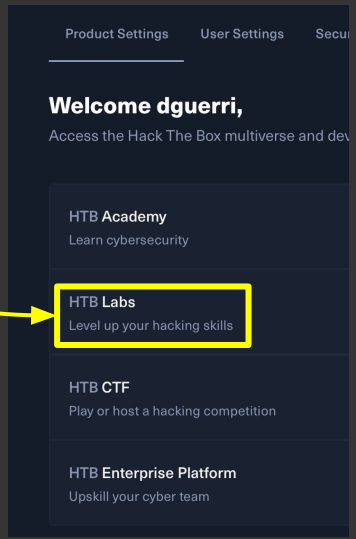


Bonus: Hack The Box

If you get a satisfactory grade, 45+ points on HTB => +1 on the final grade

Note: This is HTB Labs

- Pwn machines to get points
- **WARNING**: as machines are retired, you lose points
- More info on the ETH website



Students Assessment Criteria

[Optional] Lab assignment

- Split in **groups** of up to 4 people
  - If you don't have a group, one will be assigned to you
- Create a vulnerable VM - **50% of lab assignment grade**
  - Including a **report** describing the intended attack paths
- Hack into another group's VM - **50% of lab assignment grade**
  - Randomly assigned by us - be ethical, no cheating
  - Create a **detailed report** of tactics and techniques used and your findings

Students Assessment Criteria

[Optional] Lab assignment

- A form will be sent for assignment sign-up and to form teams
- **Please do not apply if you are unsure whether you can commit to completing it**
  - There have been complaints about the effort put in by certain team members
  - Some individuals withdrew close to the deadline

This year, behaviors like those above will result in a **grade penalty**

Rules of Engagement

"Ethical Hacking"

Hack into systems, escalate privileges, research and find vulnerabilities, do social engineering

With an authorization and always doing the ***right*** thing

Rules of Engagement

Be ethical, do the right thing...

For instance, if you find a new vulnerability, **practice responsible disclosure**

- Coordinated vulnerability disclosure
  - Allowing time to create and deploy patches before disclosing

Ok, now that we read and understood the disclaimer...

Understanding Security Testing

Understanding Security Testing

Vulnerability Assessment *vs* **Penetration Testing** *vs* **Red Teaming**

- These are not the same
- No silver bullet for assessing and mitigating security risk for a company or an institution

Understanding Security Testing

Vulnerability Assessment

Targets: systems (networks, servers, laptops, applications)

Focus: breadth of vulnerability coverage

Methodology: primarily automated with manual intervention and triage (false positives)

Limitations: may not identify specific attack paths leading to critical compromise

Understanding Security Testing

Penetration testing

Targets: systems (networks, servers, laptops, applications)

Focus: depth and achieving the objective (e.g., capturing the “flag”)

Methodology: exploiting identified vulnerabilities to establish a foothold and reach the objective

Limitations: leveraging the path of least resistance may not uncover alternative attack paths or offer comprehensive system security evaluation

Understanding Security Testing

Red teaming

Targets: Systems, processes, physical security, ...

Focus: Emulating real-world attackers, exploiting vulns across various attack vectors

Methodology: Advanced techniques like social engineering, zero-day exploits, and physical intrusion

Limitations: May cause disruption to normal operations, operational security and ethical considerations must be addressed

Kill chain & ATT&CK

Kill chain & ATT&CK

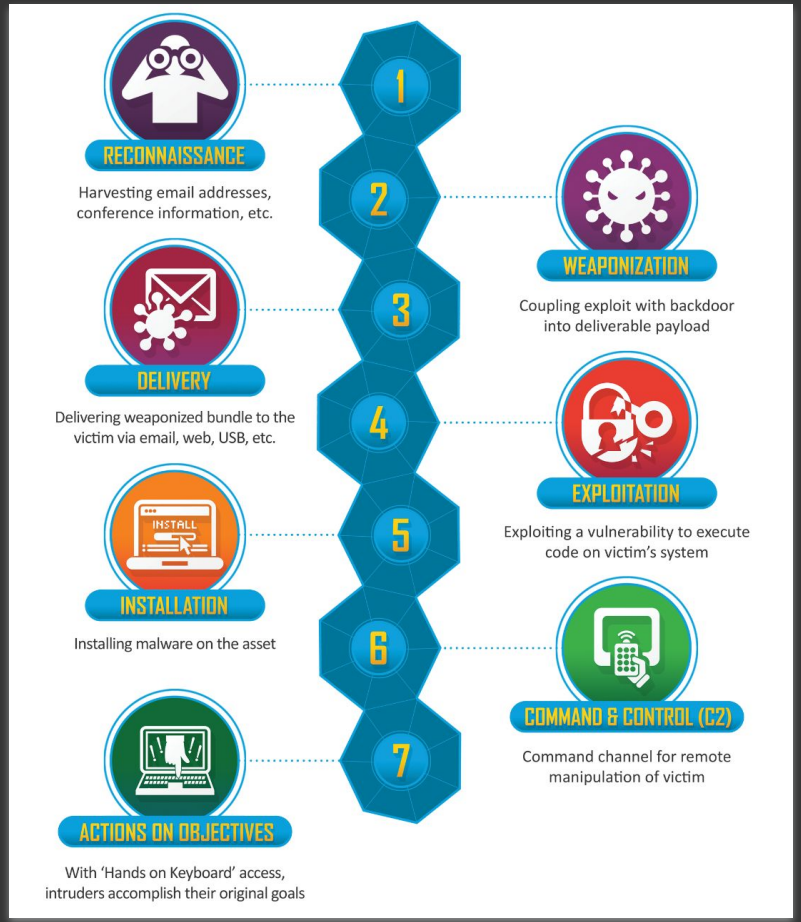
A cybersecurity kill chain is a framework

- Visualizes the attacker's journey from initial reconnaissance to exfiltrating sensitive data or doing harm
- Understanding each stage enables us to
  - Implement targeted defenses and disrupt their progress
  - Assess risk and attack paths within a realistic set

Kill chain & ATT&CK

Lockheed Martin KC - 7 stages

1. Reconnaissance
2. Weaponization
3. Delivery
4. Exploitation
5. Installation
6. Command and Control (C2)
7. Actions and Objectives



Kill chain & ATT&CK

Stage 1 - Reconnaissance

- Attackers gather information about target systems and vulnerabilities
- Techniques include social engineering, open-source intelligence (OSINT), and network scanning
- Reconnaissance can take place both online and offline and can be completely passive (e.g., Shodan, Google dorks, **X509 transparency**, ...)

Kill chain & ATT&CK

Stage 2 - Weaponization

- Attackers create or modify malicious tools to exploit identified vulnerabilities
- Techniques include developing custom malware, modifying existing tools, and using exploit kits
- **Example**: Trojanizing a legit software of interest for the victim - or researching 0-days vulnerabilities for the systems the victim is using

Kill chain & ATT&CK

Stage 3 - Delivery

- Attackers deliver the weaponized payload to the target system
- Techniques include phishing emails, malicious websites, infected USB drives, and watering hole attacks (e.g., supply chain attacks...)
- **Example**: Sending a spear-phishing email with a malicious attachment to a targeted employee

Kill chain & ATT&CK

Stage 4 - Exploitation

- Exploiting vulnerabilities (or people, or physical security, ...)
- Privilege escalation and Lateral movement
- **Example**: Attackers navigate through the network, exploiting interconnected systems and misconfigurations to reach their target data or resources

Kill chain & ATT&CK

Stage 5 - Installation

- Attackers establish persistence by installing malicious software and tools
- This ensures continued access and control even after detection or system restarts
- **Example**: Installing a backdoor to maintain remote access to the compromised system

Kill chain & ATT&CK

Stage 6 - Command & Control (C2)

- Attackers establish communication channels, including encrypted connections, covert channels (e.g., DNS, social media, ...), to avoid detection
- Send instructions to the infected system, to download additional malware, steal data, launch attacks, or perform other malicious activities
- **Data exfiltration**: Transferred data to the attacker through the C2 channels

Kill chain & ATT&CK

Stage 7 - Actions and Objectives

- Attackers achieve their goals, such as data theft, disruption, or installing ransomware
- The specific actions depend on the attacker's motivation and target
- **Example**: Stealing sensitive financial data or deploying ransomware to extort money

Kill chain & ATT&CK

MITRE ATT&CK

- ATT&CK® stands for **A**dversarial **T**actics, **T**echniques, **& C**ommon **K**nowledge
- Knowledge base and model for cyber adversary behaviour

Kill chain & ATT&CK

MITRE ATT&CK

- Provide a taxonomy of individual adversary actions understood by both offensive and defensive sides of cybersecurity
- Maps techniques to specific APT (Advanced Persistent Threat) groups

Reconnaissance	Resource Development	Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
10 techniques	7 techniques	9 techniques	12 techniques	19 techniques	13 techniques	40 techniques	15 techniques	29 techniques	9 techniques	17 techniques	16 techniques	9 techniques	13 techniques
Active Scanning (2)	Acquire Infrastructure (6)	Drive-by Compromise	Command and Scripting Interpreter (8)	Account Manipulation (4)	Abuse Elevation Control Mechanism (4)	Abuse Elevation Control Mechanism (4)	Adversary-in-the-Middle (2)	Account Discovery (4)	Exploitation of Remote Services	Adversary-in-the-Middle (2)	Application Layer Protocol (4)	Automated Exfiltration (1)	Account Access Removal
Gather Victim Host Information (4)	Compromise Accounts (2)	Exploit Public-Facing Application	Container Administration Command	BITS Jobs	Access Token Manipulation (5)	Access Token Manipulation (5)	Brute Force (4)	Application Window Discovery	Internal Spearphishing	Archive Collected Data (3)	Communication Through Removable Media	Data Transfer Size Limits	Data Destruction
Gather Victim Identity Information (3)	Compromise Infrastructure (6)	External Remote Services	Deploy Container	Boot or Logon Autostart Execution (15)	BITS Jobs	BITS Jobs	Credentials from Password Stores (5)	Browser Bookmark Discovery	Lateral Tool Transfer	Audio Capture	Data Encoding (2)	Exfiltration Over Alternative Protocol (3)	Data Encrypted for Impact
Gather Victim Network Information (6)	Develop Capabilities (4)	Hardware Additions	Exploitation for Client Execution	Boot or Logon Initialization Scripts (5)	Boot or Logon Autostart Execution (15)	Build Image on Host	Exploitation for Credential Access	Cloud Infrastructure Discovery	Remote Service Session Hijacking (2)	Automated Collection	Data Obfuscation (3)	Exfiltration Over C2 Channel	Data Manipulation (3)
Gather Victim Org Information (4)	Establish Accounts (2)	Phishing (3)	Inter-Process Communication (2)	Browser Extensions	Boot or Logon Initialization Scripts (5)	Deobfuscate/Decode Files or Information	Forced Authentication	Cloud Service Dashboard	Remote Services (6)	Browser Session Hijacking	Dynamic Resolution (3)	Exfiltration Over Other Network Medium (1)	Defacement (2)
Phishing for Information (3)	Obtain Capabilities (6)	Replication Through Removable Media	Native API	Compromise Client Software Binary	Create or Modify System Process (4)	Deploy Container	Forge Web Credentials (2)	Cloud Service Discovery	Replication Through Removable Media	Clipboard Data	Encrypted Channel (2)	Exfiltration Over Physical Medium (1)	Disk Wipe (2)
Search Closed Sources (2)	Stage Capabilities (5)	Supply Chain Compromise (3)	Scheduled Task/Job (6)	Create Account (3)	Domain Policy Modification (2)	Direct Volume Access	Input Capture (4)	Cloud Storage Object Discovery	Software Deployment Tools	Data from Cloud Storage Object	Fallback Channels	Exfiltration Over Web Service (2)	Endpoint Denial of Service (4)
Search Open Technical Databases (5)		Trusted Relationship	Shared Modules	Create or Modify System Process (4)	Escape to Host	Domain Policy Modification (2)	Modify Authentication Process (4)	Container and Resource Discovery	Taint Shared Content	Data from Configuration Repository (2)	Ingress Tool Transfer	Scheduled Transfer	Firmware Corruption
Search Open Websites/Domains (2)		Valid Accounts (4)	Software Deployment Tools	Event Triggered Execution (15)	Event Triggered Execution (15)	Execution Guardrails (1)	Network Sniffing	Domain Trust Discovery	Use Alternate Authentication Material (4)	Data from Information Repositories (3)	Multi-Stage Channels	Transfer Data to Cloud Account	Inhibit System Recovery
Search Victim-Owned Websites			System Services (2)	External Remote Services	Exploitation for Privilege Escalation	Exploitation for Defense Evasion	OS Credential Dumping (8)	File and Directory Discovery		Data from Local System	Non-Application Layer Protocol		Network Denial of Service (2)
			User Execution (3)	Hijack Execution Flow (11)	Hijack Execution Flow (11)	File and Directory Permissions Modification (2)	Steal Application Access Token	Group Policy Discovery		Data from Network Shared Drive	Non-Standard Port		Resource Hijacking
			Windows Management Instrumentation	Implant Internal Image	Process Injection (11)	Hide Artifacts (9)	Steal or Forge Kerberos Tickets (4)	Network Service Scanning		Data from Removable Media	Protocol Tunneling		Service Stop
				Modify Authentication Process (4)	Scheduled Task/Job (6)	Hijack Execution Flow (11)	Steal Web Session Cookie	Network Share Discovery		Data Staged (2)	Proxy (4)		System Shutdown/Reboot
				Office Application Startup (6)	Valid Accounts (4)	Impair Defenses (9)	Two-Factor Authentication Interception	Network Sniffing		Email Collection (3)	Remote Access Software		
				Pre-OS Boot (5)		Indicator Removal on Host (6)	Unsecured Credentials (7)	Password Policy Discovery		Input Capture (4)	Traffic Signaling (1)		
				Scheduled Task/Job (6)		Indirect Command Execution		Peripheral Device Discovery		Screen Capture	Web Service (3)		
				Server Software Component (4)		Masquerading (7)		Permission Groups Discovery (3)		Video Capture			
				Traffic Signaling (1)		Modify Authentication Process (4)		Process Discovery					
				Valid Accounts (4)		Modify Cloud Compute Infrastructure (4)		Query Registry					
						Modify Registry		Remote System Discovery					
						Modify System Image (2)		Software Discovery (1)					
						Network Boundary Bridging (1)		System Information Discovery					
						Obfuscated Files or Information (6)		System Location Discovery (1)					
						Pre-OS Boot (5)		System Network Configuration Discovery (1)					
						Process Injection (11)		System Network Connections Discovery					
						Reflective Code Loading		System Owner/User Discovery					
						Rogue Domain Controller		System Service Discovery					
						Rootkit		System Time Discovery					
						Signed Binary Proxy Execution (13)		Virtualization/Sandbox Evasion (3)					
						Signed Script Proxy Execution (1)							
						Subvert Trust Controls (6)							
						Template Injection							
						Traffic Signaling (1)							
						Trusted Developer Utilities Proxy Execution (1)							
						Unused/Unsupported Cloud Regions							
						Use Alternate Authentication Material (4)							
						Valid Accounts (4)							
						Virtualization/Sandbox Evasion (3)							
						Weaken Encryption (2)							
						XSL Script Processing							

Kill chain & ATT&CK

MITRE ATT&CK®

How to practice

How to Practice

Occasionally, there will be hands-on activities during lab lessons.

- However, practicing at home is essential to reinforce learning, complete lab scenarios, and prepare for the exam (both assignments and the written test)

A virtualization environment is recommended to create a semi-isolated testing setup

- VirtualBox is a free multi-platform virtualization platform
- KVM - free virtualization solution for Linux
- UTM - good free option for macOS - both x86 and Apple M* (based on QEMU)

How to Practice - Recommended setup

Recommended setup: virtualized environment with

- An Attack Box, with internet connectivity
    - Capable of creating a VPN to work with **Hack The Box** and **Try Hack Me**
- Vulnerable virtual machines
- Docker daemon with vulnerable docker containers
- An additional internal network connecting the attack box to vulnerable assets

How to Practice - Attack Box

Attack Box

- Starting from the next lesson, have a **Linux box ready with:**

  nmap, **Metasploit**, and **Burp Suit** (community edition)

Recommended, ready to use, Linux distributions: Kali Linux or Parrot Os

How to Practice - vulnerable machines

For “offline” testing, and demo purposes, we will use **Metasploitable 2**

- Test environment to perform penetration testing and security research

How to install

- Download `metasploitable-linux-2.0.0.zip` from Rapid7
- Unzip it
- [optional] Convert `Metasploitable.vmdk` to a format you can run
  - Not needed on VirtualBox

How to Practice - vulnerable machines

Make sure you have connectivity between your attack box (e.g., Kali) and the box with Metasploitable

- Do not expose Metasploitable to the Internet
- If needed, log in with `msfadmin/msfadmin`
    - Access is needed to configure the network, if using Virtual Box internal networks (i.e., no DHCP)

How to Practice - vulnerable machines

Later in the course, for web security, we will use OWASP Juice-Shop:

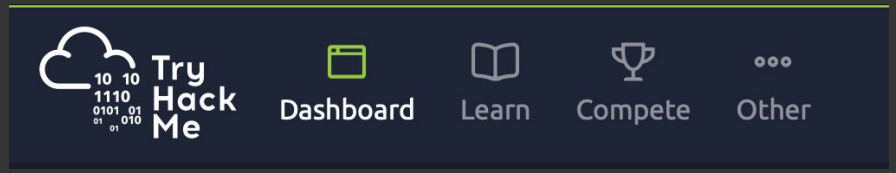
- Probably the most modern and sophisticated, insecure web application!

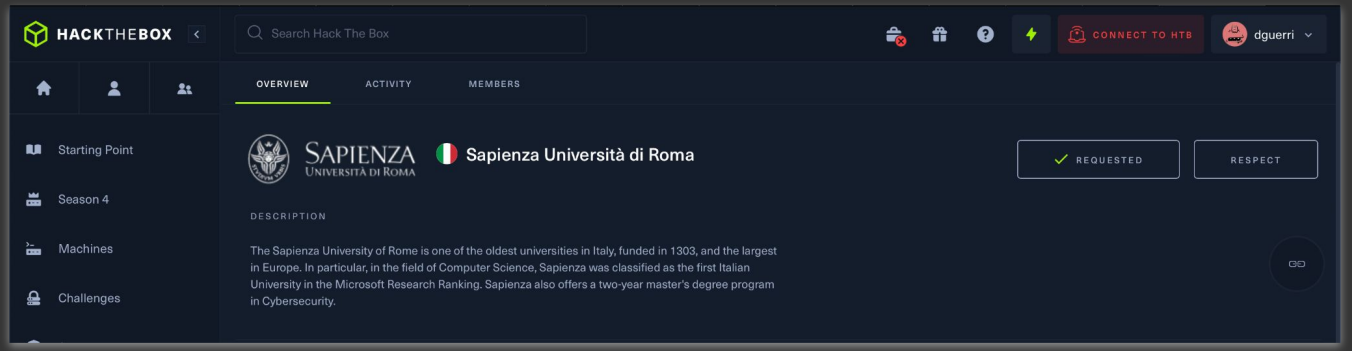
Several options to practice with Juice Shop:

- **Try Hack Me (free) - https://tryhackme.com/room/owaspjuiceshop**
- **Docker**
- **Vagrant**
- **Local**

How to Practice - vulnerable machines

Create an account on HackTheBox (HTB) and/or TryHackMe (THM)

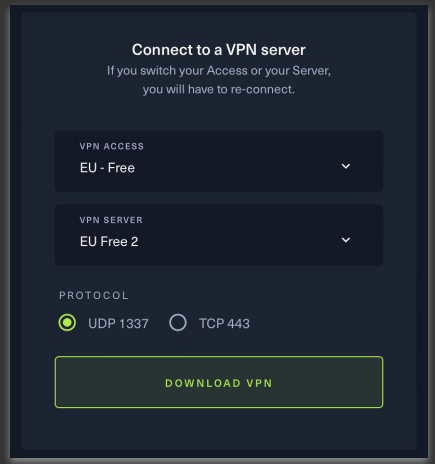




How to Practice - vulnerable machines

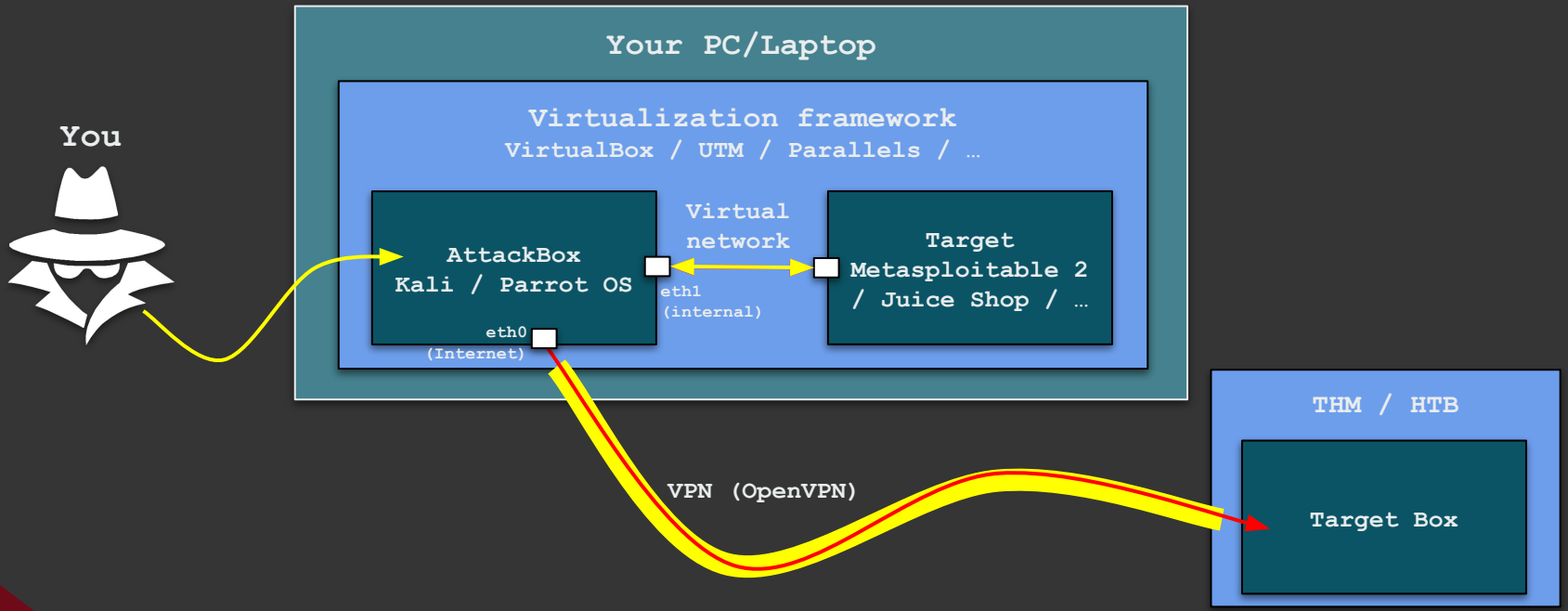
With HTB and THM, using your attack box:

- Set up a VPN from your attack box
  - Instructions/tutorials and configuration files on the respective websites
- Follow learning paths
- Spin up vulnerable machines
- Attack machines



Recap

How to Practice



How to Practice

More `[optional but useful]` community-driven resources

- Team Sapienza on Hack The Box
- CTF (Capture The Flag) hacking team, more info at Hackappatoi on Github

For more information and to join the above teams

join htbsapienza telegram group

Links

Practice

- VirtualBox Test Builds
- TryHackMe
- Hack The Box (HTB)

Extras

- Shodan
- Hacking Google
- Lockheed Martin KC
- MITRE ATT&CK