2026 ASEE ANNUAL CONFERENCE June 21–24, 2026 | Charlotte Convention Center, NC

Host Sponsor UNC Charlotte

ASEE AMERICAN SOCIETY FOR ENGINEERING EDUCATION

Paper ID #50934

Crafting Practical Scenarios in an Ethical Hacking and Penetration Testing Course: A Cyber Kill Chain Model Approach

Dr. Te-shun Chou, East Carolina University

Dr. Te-Shun Chou is a Professor in the Department of Technology Systems within College of Engineering and Technology at East Carolina University. He serves as the program coordinator of the Master program in Information and Cybersecurity Technology for the Department. His teaching and research interests focus on cybersecurity, computer networking, AI, and technology education. Dr. Chou has experience in supervising both undergraduate and graduate students in thesis, practicum, and grant-supported research projects.

Crafting Practical Scenarios in an Ethical Hacking and Penetration Testing Course: A Cyber Kill Chain Model Approach

Abstract

Building a strong defense system requires a deep understanding of hacker behavior and system vulnerabilities. This makes Ethical Hacking and Penetration Testing courses vital components of cybersecurity education. In the past, a course was developed that integrated both theoretical knowledge and practical examples to help students build a solid foundation in cybersecurity principles. In addition, the course offered hands-on lab exercises that were structured around the Cyber Kill Chain model. By mapping hands-on exercises to the seven stages of the Kill Chain, students gained a structured and realistic understanding of how cyberattacks unfold from initial reconnaissance to achieve attacker goals. To align with the goals of engineering technology education, students engaged with industry-standard software, scripting techniques, and network security applications.

This paper presented a comprehensive review of the practical lab exercises integrated into the course curriculum. These exercises not only bridged the gap between theoretical concepts and real-world applications but also provided students with hands-on experience. Additionally, the paper analyzed the results of an exit survey conducted among students who completed the course. The survey assessed the effectiveness of the labs for future improvements in both structure and content.

Keywords: Cybersecurity, ethical hacking, penetration testing, vulnerability scanning, exploitation, and post-exploitation

1. Introduction

In today's digital age, hacking occurs more frequently than ever before. By 2025, businesses could face cybercrime costs as high as $10.5 trillion, with a potential increase to $15.63 trillion by 2029 [1]. Consequently, there is an urgent need to develop defensive measures robust enough to counter ever-more complex cyber threats. To establish a robust defense system, it is essential to understand both the strategies employed by hackers and the vulnerabilities that exist within computer systems.

Ethical Hacking and Penetration Testing courses are therefore essential to cybersecurity education. These courses blend theoretical concepts with practical experience, enabling students to establish a strong understanding of cybersecurity fundamentals. Through hands-on labs, students learn to identify and exploit system vulnerabilities, moving beyond theoretical coverage to active applications. This experiential learning is crucial for cultivating the practical skills required to safeguard and defend against cyber threats.

This paper provided a comprehensive overview of the practical hands-on lab exercises implemented in an Ethical Hacking and Penetration Testing course. The labs were designed to replicate real-world scenarios and guided students through various stages of Cyber Kill Chain model, including Reconnaissance, Weaponization, Delivery, Exploitation, Installation, Command & Control (C2), and Action on Objectives. Each stage was designed to deepen their understanding of system weaknesses and the ways in which attackers leverage them. By

simulating real-world scenarios, students were better prepared to mitigate and respond to actual cyber threats. Furthermore, this paper examined the results of an exit survey conducted among students in the course. The survey assessed the overall effectiveness of the course in preparing students for professional roles in cybersecurity and furthermore provided insights into the strengths and areas for improvement in the curriculum.

This paper is organized as follows: Section 2 describes the lab environment. Section 3 illustrates tools and scripting language used for the labs. Section 4 demonstrates the designed labs. Section 5 shows Kali tools labs. Section 6 exhibits Python scripting labs. Section 7 examines the survey assessment and its results. Finally, we conclude our work in the last section.

2. Lab Environment

To enable students' engagement in activities such as information gathering, vulnerability scanning, exploitation, and post-exploitation, a controlled lab environment was created using VirtualBox. The lab environment provided a safeguarded infrastructure to replicate real-world scenarios without the risk of actual data breaches. To help students practice different operating systems, both windows-based and Linux-based operating systems were installed. Kali Linux was used as the attacker. Win Server and CentOS were used as victims with vulnerabilities. These systems provided a realistic and challenging learning environment for students to practice both their technical and observational skills.

3. Tools and Scripting Languages

Designing cybersecurity labs using specialized tools such as Kali Linux and scripting languages like Python provided students with a practical and flexible learning environment that closely mirrors real-world security operations. Kali Linux offered a comprehensive collection of preinstalled penetration testing and forensic tools, allowing students to explore a variety of attack techniques and gain hands-on experience with industry-standard tools.

Incorporating Python scripting into labs further enhanced learning by encouraging automation, customization, and deeper technical insight. Rather than relying solely on point-and-click tools, Python enabled students to understand the underlying logic of security tools, such as how packets are crafted, credentials are tested, logs are analyzed, and so on. It helped students transition from tool users to security practitioners who can adapt to modern threats. Together, Kali Linux and Python created a powerful foundation for experiential cybersecurity education.

There are numerous types of cyberattacks and each one possesses its own unique methods, targets, and consequences. Due to the vast and ever-evolving landscape of cybersecurity threats, it is impossible to cover every single type of attack within the scope of this course. Therefore, we have carefully selected the most important and prevalent types of cyberattacks to focus on in our lab sessions. Fourteen labs were designed in total, with nine labs using Kali Linux tools (SQL Injection, Cross-Site Scripting, Exploitation and Collecting Victim Data, ARP Poisoning, Phishing Attack, IP Spoofing, Man-in-the-Middle, Password Cracking, and Denial of Service) and 5 labs using Python scripts (Malware, Port Scanning, Sniffing, IP Scanning, and Command and Control Channel Establishment). By concentrating on these critical areas, the course ensured

that students develop a solid foundation in the most significant and common cyber threats. Also, this dual-use approach supported students' development of practical skills by working with widely adopted tools and scripting techniques in the real world.

4. Cyber Kill Chain Model Based Lab Design

There are many cybersecurity models, such as the MITRE ATT&CK framework [2], the Diamond Model [3], and the NIST Cybersecurity Framework [4], used to analyze threats and guide defense strategies. Among them, the Cyber Kill Chain model [3] is a very popular and established model for understanding and breaking down the sequential stages of a cyberattack, from initial reconnaissance to achieving the attacker's objective. Designing hands-on cybersecurity labs based on the Cyber Kill Chain model provided students with a structured and realistic understanding of how cyberattacks unfold from start to finish. By aligning each lab with a specific stage of the Kill Chain, students can clearly see how individual attack techniques fit into a broader attack lifecycle. This approach moved beyond isolated tool usage and further emphasized attacker intent, sequence, and dependencies between stages. Hands-on activities also reinforced students' defensive skills while analyzing and responding to real cyber incidents.

5. Kali Linux Labs

5.1. ARP Poisoning

ARP poisoning is most commonly used inside a local network after the attacker has gained access. The goal is to associate the attacker's Media Access Control (MAC) address with the Internet Protocol (IP) address of an authorized device on the network. During the lab, students learned to trace a Layer 2 attack through the Cyber Kill Chain. They began with Reconnaissance using Ettercap to scan the internal subnet and identify critical systems. During Weaponization, they configured Ettercap for ARP spoofing to corrupt ARP tables. The Delivery stage was executed when they sent forged ARP replies to both target devices, poisoning their ARP caches and redirecting traffic through the students' attacking machine. This action directly enabled Exploitation, establishing a Man-in-the-Middle position that allows attackers to have the ability to conduct post-exploitation activities, such as intercepting, modifying, or exfiltrating sensitive internal communications. This real-world example showed how an attacker can exploit weaknesses in the ARP protocol to silently intercept internal traffic. It also demonstrated how ARP poisoning serves as a gateway to Actions on Objectives, such as file transfers and authentication credentials. Furthermore, it emphasized the importance of implementing Layer 2 security measures, especially in environments with shared network access or limited segmentation.

5.2. Denial of Service (DoS) Attack

The primary objective of a DoS or a distributed DoS (DDoS) attack is to disrupt the availability of a service, system, or network. In just the first half of 2025, over 8 million DDoS attacks were documented worldwide [5]. In the Denial-of-Service attack lab against a web portal, students learned how an attacker disrupts service availability through the early stages of the Cyber Kill Chain. They began with Reconnaissance with the manipulation of nmap to scan the network,

identify the Windows Server hosting the IIS web service, and confirm open port 80. During Weaponization, they prepare the attack tool by configuring hping3 with SYN flood parameters designed to overwhelm the target's TCP stack. The Delivery stage was executed when they launched the flood of malicious SYN packets toward the server's IP address to saturate its network interface and consume system resources. While this lab focused on disruption rather than intrusion, the goal of the attack is to degrade service performance and deny access to legitimate users in the Actions on Objectives stage. Through hands-on traffic monitoring in Task Manager, students observed the real-time impact of the flood on network bandwidth and CPU usage. The lab illustrated how a basic DoS technique can compromise availability in under-protected environments. It also emphasized the importance of implementation of defensive strategies to identify and mitigate DoS threats early in the Kill Chain. The techniques include rate limiting, SYN flood protection, and DDoS mitigation,

5.3. Exploitation and Collecting Victim Data

During the exploitation phase, attackers actively use a discovered vulnerability to gain unauthorized access or control of a system. During the exploitation and data collection attack lab, students learned to trace a browser-based compromise through multiple stages of the Cyber Kill Chain. They began with Reconnaissance to identify vulnerabilities on a Firefox web browser. During Weaponization, they used Metasploit Framework to configure a malicious server on port 8080 to deliver the exploit and made a fake URL link that looks like a project management page. The Delivery stage occurred when the link in the vulnerable browser was clicked. The click triggered Exploitation that a hidden process (notepad.exe) was created and a Meterpreter session on the victim's machine was established. Students then moved to Installation, migrating the session to explorer.exe for persistence and initiated Command & Control through the Meterpreter channel. Finally, students achieved Actions on Objectives that used keylogging (keyscan_start and keyscan_dump) to capture sensitive keystrokes and gather credentials. The lab reflected how a single unpatched application can lead to full system compromise and data theft. The lab also demonstrated hackers' ability to easily gain deep access and extract sensitive data with little technical resistance if defenses are not properly implemented.

5.4. IP Spoofing

IP spoofing is the creation of IP packets with a false source IP address to disguise their identity or impersonate a legitimate computing system. In the lab, students learned to map the early stages of the Cyber Kill Chain to a concrete IP spoofing attack action. They began with the Reconnaissance stage using nmap and arp-scan to discover key systems in a fictitious financial institution. During Weaponization, students crafted packets with a falsified source IP and transmitted the packets in the Delivery stage. It led into Exploitation where the spoofed packets attempt to bypass IP-based authentication on the target system. By analyzing the spoofed traffic in Wireshark, students learned how easily IP trust can be manipulated and how such attacks appear on the network. The lab illustrated how successful spoofing could enable data leak in the Actions on Objectives stage. Moreover, the lab demonstrated that IP-based authentication alone is insufficient. Stronger security measures, such as TLS authentication, network segmentation, and anomaly detection, should be implemented to safeguard sensitive infrastructure.

5.5. Man-in-the-Middle (MitM) Attack

The goal of a MitM attack is to deliver malicious payloads and intercept (redirect or alter) communication between two parties. It is predicted that global losses from cybercrime are expected to reach $10.5 trillion per year and the losses are significantly driven by the growing sophistication and widespread use of MitM attacks [6]. In the scenario of a data breach, students managed to trace a MitM attack through each stage of the Cyber Kill Chain. They began with Reconnaissance by using the network scanning tool, Ettercap, to map the subnet and identify targets. Next, during Weaponization, they configured ARP poisoning to prepare the attack mechanism. The Delivery stage was executed as they poisoned ARP caches and put themselves into the communication path between the two compromised machines. This enables Exploitation, where students can actively intercept, sniff, and manipulate unencrypted HTTP traffic and capture credentials, session data, and sensitive research information. Finally, attackers can use the intercepted data for data theft, lateral movement, or corporate espionage in the Actions on Objectives stage. The lab highlighted how a simple tool can be devastating in a real-world environment when proper network security controls are missing. It reinforced the importance of defensive measure implementation, such as encryption, monitoring, and access control, on networks.

5.6. Password Cracking

The goal of password cracking is to gain unauthorized access to a system, account, or resource by exploiting weak authentication mechanisms. Research indicates that weak or stolen passwords are involved in roughly 81% of hacking-related security incidents. Additionally, approximately 20% of unauthorized access attempts targeting accounts with inadequate password protection are attributed to brute force attacks [7]. In the real-world password cracking scenario lab activity, students learned to trace a credential-based attack across the entire Cyber Kill Chain. They began with Reconnaissance using nmap to discover live hosts and identify open ports on the hosts. During Weaponization, they compiled custom wordlists and configured brute-force modules in Hydra and Metasploit. The Delivery stage occurred as they launched password-guessing attacks, John the Ripper, against the SSH service and sent malicious authentication attempts to the target. After successfully cracking weak passwords, it led to Exploitation where unauthorized root access was gained and sensitive data was exfiltrated. This access further enabled them to establish persistent sessions in the Command & Control stage as well as to steal credentials in the stage of Actions on Objectives. The lab demonstrated how weak authentication policies can lead to full system compromise. It also indicated that organizations should disable root SSH login, enforce strong password policies, monitor for brute-force attempts, and implement robust authentication mechanisms.

5.7. Phishing Attack

Phishing is the mechanism used to deliver deceptive content to induce the victim to click a link, open an attachment, or submit credentials. Phishing comprises 15% of breaches, which is the second most prevalent attack vector and costly for organizations accounting for an average USD 4.88 million per breach [8]. The lab demonstrated a phishing attack using a credential-theft scenario in the stages of Delivery and Exploitation. Before the classes, a pre-crafted Instagram

login page containing an urgent security alert was created and an Apache server to host the spoofed site was pre-configured. The Delivery stage is executed when students click the malicious link shown on the email. This led to Exploitation as they enter their credentials on the fake page, allowing the attacker to harvest sensitive login information. Through the hands-on exercise, students learned that they should never click unsolicited links, even if they appear legitimate. Students also learned that defensive measures should be implemented to protect valuable credentials. Defensive techniques include email filtering, user education, and multi-factor authentication,

5.8. SQL Injection (SQLi) Attack

SQLi attacks take advantage of a vulnerability in a web application's input handling. The attacker uses SQLi to execute unauthorized database commands, bypass authentication, or extract data. According to a study from OWASP, Injection is listed among the top 10 attacks today [9]. In the simulated SQLi attack lab, a web portal is designed to trace an exploitation-based intrusion through each stage of the Cyber Kill Chain. Students began with Reconnaissance to scan the network and identify the web server hosting the vulnerable portal. During Weaponization, students crafted malicious SQL payloads, ' OR '1'='1, and prepared an automated exploitation tool: sqlmap. The Delivery stage occurred when they inputted the payload into the login form, transmitting the malicious SQL query to the web application. This led directly to Exploitation, where the injection bypassed authentication, granted unauthorized access, and enabled database interrogation. Using sqlmap, students escalated the attack to extract sensitive data (usernames, passwords, and credit card numbers) during the stage of Actions on Objectives. The lab illustrated how a single unvalidated input can compromise an entire database. It also highlighted that deploying defensive countermeasures, such as parameterized queries, input validation and sanitization, and web application firewall rules, can effectively disrupt the Kill Chain at the Delivery and Exploitation stages.

5.9. Cross-Site Scripting (XSS) Attack

XSS attacks exploit a vulnerability in a web application's input validation or output encoding. The attacker injects malicious scripts in a victim's browser for session hijacking, credential theft, or malicious redirection. Statistical analysis indicates that over 60% of web applications are vulnerable to XSS attacks [10]. In the lab of XSS defacement attack on a forum website, students learned to map a client-side exploitation attack across the Cyber Kill Chain. Students started with using nmap to identify the server hosting the vulnerable forum during the Reconnaissance stage. During Weaponization, they crafted malicious JavaScript payloads and designed HTML injection code to alter page appearance and execute scripts in victims' browsers. The Delivery and Exploitation stages occurred when they logged in with compromised credentials (gained from the SQLi Attack lab) and submitted the malicious code through the comment form. This triggered a popup and visually defaced the page with altered backgrounds and headers in the Actions on Objectives stage. The lab helped students understand how easily webpages can be vandalized by injecting malicious scripts. It also reinforced the value of implementing defensive mechanisms such as input validation, output encoding, and client-side security controls.

6. Python Scripting Labs

6.1. IP Scanning

Scanning IP addresses on a network is a fundamental phase of reconnaissance conducted by attackers to discover active hosts and identify potential targets. This process often involves sending specially crafted packets across the local network segment to determine which devices are live and responsive. In the insider threat scenario at a pseudo company, students learned to apply the early stages (Reconnaissance, Weaponization, and Delivery) of the Cyber Kill Chain to conduct internal reconnaissance. They began with Reconnaissance by crafting a custom Python script using Scapy to map live hosts and perform ARP scanning across the internal subnet. This script served as a Weaponization stage to transform the Python script into a network enumeration tool, which was capable of identifying active IP-to-MAC address pairs. The Delivery stage occurred when they executed the script that broadcasted ARP requests across the network to gather real-time intelligence on connected devices. By revealing potential targets, the collected information could then directly enable Exploitation for follow-on attacks, such as port scanning or vulnerability probing. The lab illustrated how even low-privilege insiders can perform stealthy reconnaissance to map the attack surface. It also showed that IP scanning could be the first step toward privilege escalation or lateral movement if the insider has malicious intent, In the meantime, Techniques, such as network segmentation, ARP monitoring, and intrusion detection, can be used to detect and disrupt scanning activities during the early stages of the Cyber Kill Chain.

6.2. Command and Control (C2) Channel Establishment

In the Cyber Kill Chain, C2 itself is a distinct stage. C2 channels allow attackers to issue commands, receive data, update malware, and maintain persistence. During the lab, students learned how to map a post-exploitation persistence mechanism against a simulated company across multiple stages of the Cyber Kill Chain. After initial exploitation, students used Secure Copy Protocol (SCP) to upload a custom Python script to the server for establishing a persistent foothold during the Installation stage. This then led to the C2 stage, where students set up a C2 server on Kali Linux to open a secret TCP socket that enabled bidirectional communication with the infected host. Through this channel, students issued remote shell commands, whoami, pwd, and ifconfig, to gather information and maintain persistent access. The lab demonstrated how a C2 server acts as the central hub for executing actions in the Actions on Objectives stage for further network compromise. The lab showed defensive strategies, such as SSH key authentication, network segmentation, endpoint monitoring, and behavioral analytics, should be implemented to detect and disrupt C2 activity during the Installation and C2 stages of the Kill Chain.

6.3. Port Scanning

One of the most critical phases in the reconnaissance stage of a cybersecurity attack or a penetration test is identifying which network ports are open on a target system. This process, known as port scanning, enables the attacker to gain insights into the types of services, applications, and daemons running on the machine. In the simulated scenario of unauthorized port scanning at a company, students learned to apply the Cyber Kill Chain to the earliest phases.

Students began the hands-on activity by writing a custom Python script that automates the Reconnaissance stage. The script used the ipaddress and socket modules to scan the internal subnet as well as identify active hosts and open ports on the target systems. The script was able to map network exposure during the Weaponization stage. The Delivery stage occurred when students executed the script against the target IP that sent TCP connection probes to SSH port 22 and HTTP port 80 to determine service availability. The gathered intelligence directly enabled follow-on escalation actions in the Cyber Kill Chain, such as brute-force SSH login on port 22 and probing the web server on port 80 for vulnerabilities. While the lab only focused on the early stages of Cyber Kill Chain, it demonstrated how port scanning lays the groundwork for further attacks in later stages. It educated students that privileged access, such as firewalls, port knocking, and two-factor authentication, should be set up to mitigate the threat of port scanning.

6.4. Sniffing

Sniffing is a network reconnaissance technique used to intercept, capture, and analyze data packets traveling over a network. At its core, sniffing involves placing a network interface card (NIC) into promiscuous mode that allows the device to capture all packets on the network segment. In the scenario of unauthorized packet sniffing at a company, students learned to map a passive intelligence-gathering activity through the early stages of the Cyber Kill Chain. They began with Reconnaissance by using nmap to identify active hosts and network services. During Weaponization, they prepared a custom sniffing script in Python with Scapy. The Ethernet network interface, eth0, was configured as a passive sensor to capture IP-level traffic. The Delivery stage was executed when the script was run. It intercepted and logged live packets, which included ICMP pings, DNS queries, and unencrypted login attempts. This passive sniffing provided critical intelligence, such as active systems and potential plaintext credentials, that could be used in later attacks. The lab focused on Reconnaissance and Delivery, however it illustrated how sniffing enables Exploration, C2, and Actions on Objectives. In addition, the lab demonstrated that defensive mechanisms, like network segmentation, encryption protocols, and intrusion detection, can identify and block network sniffing during the initial phases of the Kill Chain.

6.5. Malware

Malware, short for malicious software, is a broad term that encompasses any software intentionally designed to disrupt, damage, or gain unauthorized access to a computer system, network, or device. Common malware types include Ransomware, Viruses, Worms, Trojans, Spyware, Adware, Rootkits, and Keyloggers. 92% of malware is delivered via email [7] and Ransomware was involved in 44% of data breaches [11]. In the ransomware attack lab against a fake company, students learned to map a multi-stage malware across the entire Cyber Kill Chain. They began with Reconnaissance, identifying the target server and its vulnerable anonymous FTP service. During Weaponization, they designed a custom Python script using tkinter and cryptography.fernet modules and packaged it into a stealthy Windows executable .exe file using PyInstaller. The Delivery stage occurred when they uploaded the malicious file via unauthenticated FTP to the server’s shared folder. Exploitation was triggered when the file was executed. It activated the ransomware, which encrypted the file system with the Fernet cipher and displayed a lock-screen GUI. Since students maintained control over the decryption key, the

malware remained active until decryption. Finally, the attack's end goals, such as disrupting operations and demanding ransom, can be successfully achieved in the stage of Actions on Objectives. The lab demonstrated the ransomware can be built using basic scripting and showed ransomware can cripple an organization. The lab also reinforced defensive lessons, including disabling anonymous services, deploying endpoint protection, maintaining backups, and user education, can break the Kill Chain at Delivery, Exploitation, and Installation stages.

7. Survey Assessment

To evaluate the effectiveness of the lab design, a survey was created via Google Forms and administered at the conclusion of the fall semesters in 2024 and 2025. Out of 62 total enrolled students, 43 participated in the survey. The questionnaire employed a five-point Likert scale and open-ended questions covering the lab environment, the Cyber Kill Chain Model, lab content, overall experience, and educational value. This paper focused specifically on presenting the assessment results related to the Cyber Kill Chain Model and the experience using tools and scripting languages.

Table 1. Cyber Kill Chain Based Designed Lab Assessment

Questions	Subjects
Q1.1	Reconnaissance
Q1.2	Weaponization
Q1.3	Delivery
Q1.4	Exploitation
Q1.5	Installation
Q1.6	C2
Q1.7	Action on Objectives

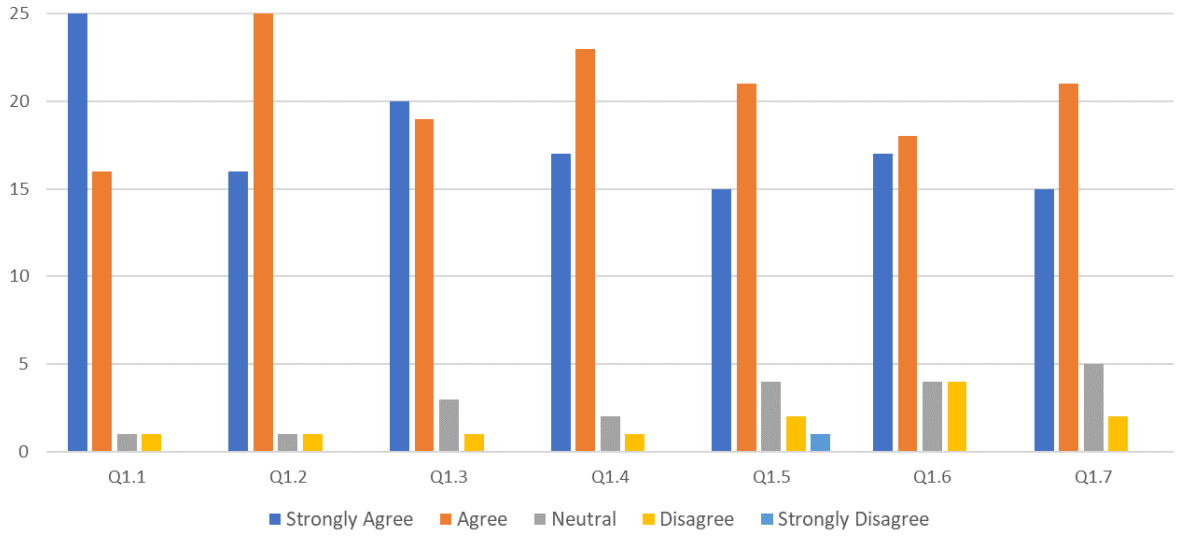

Figure 1. Survey results

Table 1 presents the questions from the Cyber Kill Chain based designed lab survey, each evaluating whether the labs effectively demonstrated techniques for a specific stage of the Cyber Kill Chain. For instance, Q1.1 asks: “The labs provided tools for gathering information on target systems, such as scanning hosts.”

As illustrated in Figure 1, the majority of students agreed that the labs properly covered the techniques across all seven stages. However, it is notable that the counts for “Neutral”, “Disagree”, and “Strongly Disagree” responses were noticeably higher for Q1.5 to Q1.7 compared to Q1.1 to Q1.4. We attributed this difference to the fact that most labs focused primarily on the first four stages (Reconnaissance, Weaponization, Delivery, and Exploitation) while only a few addressed the final three stages (Installation, C2, and Action on Objectives). To address this gap and ensure comprehensive coverage, post-exploitation activities could be added to existing labs and new labs covering the entire Cyber Kill Chain could be developed.

Table 2. Security Tools and Python Scripting

Questions	Subjects
Q2.1	After taking this course, I understand how to use tools to conduct hands-on penetration testing activities.
Q2.2	After taking this course, I understand how to use scripting language to conduct hands-on penetration testing activities.

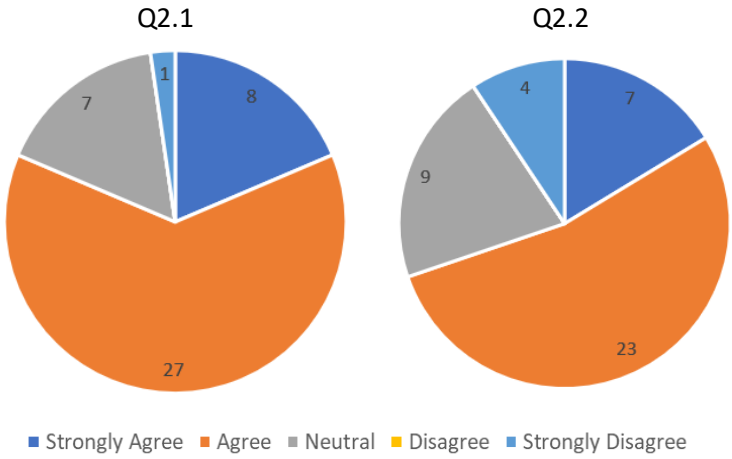

Figure 2. Assessments of Experience with Security Tools (Left) and Python Scripting (Right)

Table 2 lists the survey questions concerning students' confidence in using security tools and scripting languages for penetration testing tasks following the lab activities. Figure 2 presents the corresponding results, indicating that 80% of students responded "Agree" or higher for tools, compared to 68% for scripting languages. This indicates that students remain less comfortable employing scripting for penetration testing. To address this shortcoming, pre-lab introductory tutorials designed for beginners could equip students with the foundational scripting knowledge required for success. In addition, incorporating additional scripting exercises in other courses could be beneficial.

8. Conclusions

This paper has presented the design, implementation, and assessment of a practical lab curriculum for an Ethical Hacking and Penetration Testing course. By mapping fourteen distinct lab exercises onto the seven stages of the Kill Chain, the course provided students with a structured and realistic understanding of how cyberattacks unfold from the first stage (Reconnaissance) through to the last stage Actions on Objectives). This pedagogical framework moved students from using tools in isolation to developing a comprehensive understanding of the attacker's end-to-end process, motives, and the logical flow between stages.

The hands-on environment (built using VirtualBox with Kali Linux, Windows Server, and CentOS) provided a safeguarded yet realistic sandbox for students to practice offensive techniques and defensive principles without risk. The dual approach of utilizing both Kali Linux's pre-installed tools and custom Python scripting ensured students gained proficiency with industry-standard software and developed deeper technical insight and automation skills required of modern security practitioners.

Student survey results confirmed the overall effectiveness of the lab design, with strong agreement that the hands-on activities are well-aligned with the Kill Chain stages. However, the assessment also revealed critical insights for curriculum refinement. The lower confidence ratings for the later stages (Installation, C2, and Actions on Objectives) highlighted a coverage gap, indicating a need to develop exercises focusing on post-exploitation. To strengthen student proficiency in automation and tool customization, it is necessary to enhance the integration of scripting not only within this course but across the wider cybersecurity curriculum.

In conclusion, this Cyber Kill Chain based lab design successfully bridged the gap between theoretical cybersecurity concepts with practical, hands-on skill development. It equipped students with a structured mindset to analyze threats and reinforced the critical link between offensive techniques and defensive countermeasures. Future work will focus on enriching the curriculum with additional post-exploitation scenarios and scripting modules implementations to build student confidence and competences.

References

1. “207 Cybersecurity Stats and Facts for 2025”, VikingCloud, December 2025. [Online]. Available: https://www.vikingcloud.com/blog/cybersecurity-statistics
2. *MITRE ATT&CK*. [Online]. Available: https://attack.mitre.org/
3. H. Al-Mohannadi *et al.*, "Cyber-attack modeling analysis techniques: An overview," in *Proc. 2016 IEEE 4th Int. Conf. Future Internet Things Cloud Workshops (FiCloudW)*, 2016, pp. 69-76.
4. *NIST Cybersecurity Framework (CSF) 2.0*. [Online]. Available: https://www.nist.gov/publications/nist-cybersecurity-framework-csf-20
5. “Digital Aftershocks: Collateral Damage From DDoS Attack”, Netscout, 2025. [Online]. Available:https://www.netscout.com/threatreport
6. “Man-in-the-Middle Attacks & RADIUS Authentication”, October 2025. [Online]. Available:https://www.splashtop.com/foxpass/blog/man-in-the-middle-attacks-and-radius-authentication
7. “Password Hacking Statistics, Zipdo Education Report”, Zipdo, May 2025. [Online]. Available: https://zipdo.co/password-hacking-statistics/
8. “Cost of Data Breach Report”, IBM, 2024. [Online]. Available: https://www.ibm.com/downloads/documents/us-en/107a02e94948f4ec

9. “OWASP Top 10, The Ten Most Critical Web Application Security Risks”, OWASP, 2025. [Online]. Available: https://owasp.org/Top10/2025/0x00_2025-Introduction/
10. “Securing Websites Against XSS Attacks: Must-Know for Developers and Site Owners”, SafeAeon, April 2025. [Online]. Available: https://www.safeaeon.com/security-blog/cross-site-scripting-attack/
11. Anastasiya Novikava, “Cybersecurity statistics 2025: figures, stories, and what to do next”, Nordlayer, October 2025. [Online]. Available: https://nordlayer.com/blog/cybersecurity-statistics-of-2025/