Providing A Hands-on Advanced Persistent Threat Learning Experience Through Ethical Hacking Labs

Yen-Hung (Frank) Hu
Department of Computer Science
Norfolk State University
Norfolk, Virginia, USA
yhu@nsu.edu

Abstract—Advanced persistent threats are causing several serious cybersecurity events due to their highly stealthy characteristics, advanced technology and tools, and complicated attacking strategies, making them an imminent challenge to cybersecurity professionals. To conquer such a challenge, a thorough and dedicated defense plan must be addressed, and we believe engaging advanced persistent threat learning experiences to computer science and cybersecurity students in the early stages of their college education will be the most important part of the plan. Since there is a lack of promising approaches for engaging students in learning of advanced persistent threats, it is now an emerging issue for cybersecurity educators and researchers to investigate and develop doable and affordable advanced persistent threat learning platforms. Hands-on learning has been adopted by several fields and demonstrated promising performance improvements in the learners. Therefore, integrating hands-on learning knowledge and experiences in advanced persistent threat training for computer science and cybersecurity students will be a potential solution for mitigating such an issue. In this research, we recognize the importance of improving students' learning of advanced persistent threats. To develop a learning platform for students to learn the knowledge, skills, and abilities of advanced persistent threats, we adopt the NDG ethical hacking lab series with appropriate supplemental lectures to each stage of the lifecycle of an advanced persistent threat. We ensure our model could comply with the required knowledge units listed on NICE Cybersecurity Workforce Framework. Students are expected to connect their advanced persistent threat learning experiences to real world cybercrime cases once they have successfully completed the learning process.

Keywords—Advanced persistent threat, hands-on learning, ethical hacking, NICE cybersecurity workforce framework

This work was supported [in part] by the Commonwealth Cyber Initiative, an investment in the advancement of cyber R&D, innovation and workforce development. For more information about CCI, visit cyberinitiative.org.

I. INTRODUCTION

Advanced Persistent Threats (APTs) earn their name since they are advanced, persistent, and much more harmful than traditional cyberattacks. APTs are advanced since they adopt a full spectrum of sophisticated cyberattack techniques which usually take advantage of zero-day vulnerabilities. Such tactics make APTs capable of evading most existing cyber defense systems. APTs are persistent since they expand from their original footholds to any reachable compromised devices to survive any disruptions while keeping silent and stealthy to avoid detection. Unless there is a special need or the return is worth much more than the risk, an APT can typically reside in the compromised systems for months or even longer. APTs are much more harmful than traditional cyberattacks since they involve many more human interactions, giving APTs patterns that are less predictable and cannot be efficiently detected by traditional signature-based and behavior-based cyber defense systems [1] [2] [3]. APTs such as Stuxnet [4] [5], Duqu [4] [6], Poison Ivy [7] [8], and Solarwinds [9] [10] [10] have demonstrated the most extreme strategies that malwares can perform to cause drastic damages to their targets.

Hands-on learning has been adopted by several education and training programs in fields such as nursing, web design, software engineering, computing, and cybersecurity, and has demonstrated promising performance improvement for students [12] - [17]. We recognize the importance of improving students' performance of learning APTs. The proposed project will develop a hands-on learning model for computer science and cybersecurity students to learn the Knowledge, Skills, and Abilities (KSAs) of APTs as well as the countermeasures to defend against APT attacks. Such KSAs adopted from the NICE Cybersecurity Workforce Framework [18] will essentially comply with federal and industry standards.

We plan to adopt the NDG ethical hacking labs [19] to each phase of the APT lifecycle [20]. Students are expected to connect their learning experiences to real world cybercrime cases once they have successfully completed the training. This research ensures students gain knowledge and experiences with relevant and current APT techniques with

hands-on learning and options to experience cybersecurity research firsthand.

The remainder of this paper is organized as follows: Section II summarizes related work and recent efforts to provide perspective on the scope and importance of hands-on learning. Section III explains our research methodology. Section IV introduces the proposed hands-on learning model. Section V describes the APT lifecycle. Section VI analyzes APT Knowledge, Skills, and Abilities related to the NICE Cybersecurity Workforce Framework. Section VII maps the APT lifecycle to NICE Cybersecurity Workforce Framework. Section VIII introduces the NDG ethical hacking lab series. Section IX applies the NDG ethical hacking labs to the APT lifecycle. Section X discusses the integration of NDG ethical hacking labs with supplemental lectures for complying with NICE Cybersecurity Workforce Framework. Section XI concludes the paper with some reflections on findings and suggestions for future work to build upon them.

II. Related Works

Several researchers have studied and adopted hands-on related learning ideas and practices in their research and education projects. Although none of them includes topics related to APT education, some of them are still able to help us to develop this project.

Lisko and O'Dell [12] viewed the importance of integrating experiential learning into nursing education to improve critical thinking experiences and clinical judgement skills in students. To assess the performance of that idea, they designed a course curriculum and implemented it over a 15-week semester and studied factors related to the success of the implementation. Their research depicted a strategy for adopting experiential learning concepts and practices in an undergraduate major which involved heavy hands-on exercises and practical operations.

Bhajantri et al. [13] developed a web technology course adopting experiential learning practices such as hands-on exercises. The course emphasized a hands-on session and was taught by instructors who demonstrated the designated hands-on exercises and lead students to practice them simultaneously. The authors conducted regular assessments and concluded that hands-on exercises would attain the best outcomes for those practical oriented courses.

McLoughlin et al. [14] proposed a master level computer architecture curriculum adopting TinyCPU hands-on laboratory sessions to fix common issues caused by a traditional text-based lecture style course – students may not really understand how computers work. After two years of the implementation of that approach, their research indicated that hands-on laboratory built on TinyCPU helped students' understanding of computer architecture as well as key operations in CPU.

Holmes et al. [15] managed an undergraduate capstone open-source projects program (UCOSP) to bridge the gap between traditional classroom knowledge and the practical skills students need to succeed in the workplace. In that program, students were able to have experiential software engineering opportunities that could link work, education, and personal development. The research was conducted for 8.5 years with a total of 737 students and the results demonstrated positive feedbacks from the participants.

Shi et al. [16] created 5 educational accessibility learning labs based on a experiential learning structure to provide students the opportunity to learn foundational concepts of creating accessible software and the necessity of creating accessible software. The project was evaluated in 10 sections of a CS2 course with 276 participants. Their research results demonstrated that the proposed labs delivered positive impacts to students in creating accessible software.

Peruma et al. [17] presented a set of practical labs to enhance the creation of secure Android apps. That project, Practical Labs in Security for Mobile Applications (PLASMA), constructed several layers of security implementations for detecting and removing security vulnerabilities in mobile app development. 11 labs covering a wide range of mobile security hands-on practices were introduced and ready to be integrated into selected security and computer courses. Their research results revealed that students' interest in security was significantly improved after they were involved in the hands-on lab activities. Meanwhile, most students gained knowledge and experiences in security after they completed the lab activities.

III. Research Methodology

To thoroughly study the influence of APTs and provide education and a research plan for reducing and investigating their impacts, our research methodology includes the following 7 procedures: 1) studying and assessing hands-on learning models, 2) understanding the APT lifecycle, 3) mapping APT KSAs to the NICE Cybersecurity Workforce Framework, 4) mapping the APT lifecycle to knowledge units in the NICE Cybersecurity Workforce Framework, 5) explaining the NDG ethical hacking lab series, 6) mapping the APT lifecycle to the NDG ethical hacking labs, and 7) integrating the NDG ethical hacking labs with supplemental lectures to comply with the NICE Cybersecurity Workforce Framework. The main objectives are described below:

- Studying and assessing hands-on learning models: This procedure provides a rationale of how we selected the learning model and where it came from.
- Understanding the APT lifecycle: This procedure explains what type of APT lifecycle was selected for this research.
- Mapping APT KSAs to the NICE Cybersecurity Workforce Framework: This procedure maps required KSAs for an APT actor to NICE Cybersecurity Workforce Framework.
- Mapping APT lifecycle to knowledge units in the NICE Cybersecurity Workforce Framework: This procedure maps APT lifecycle to those required knowledge units in NICE Cybersecurity Workforce Framework for an APT actor.

- Explaining the NDG ethical hacking lab series: This procedure explains descriptions, objectives, outcomes, and contents of NDG ethical hacking labs.
- Mapping APT lifecycle to the NDG ethical hacking labs: This procedure maps the required NDG ethical hacking labs to each phase of the APT lifecycle.
- Integrating the NDG ethical hacking labs with supplemental lectures to comply with the NICE Cybersecurity Workforce Framework: This procedure explains how we implemented this project to comply with the knowledge units in NICE Cybersecurity Workforce Framework required for an APT actor. Required labs and supplemental lectures were prepared for each phase of the APT lifecycle.

IV. Hands-on Learning Model

The proposed hands-on learning model is derived from Kolb's experiential learning model [21] which emphasizes that practical experiences of a learner to a subject can enhance the individual to learn that subject.

Kolb's experiential learning model can be represented by a four-stage learning cycle and four distinct learning styles [21] [22] [23]. The four stages of the learning cycles including concrete experience, reflective observation, abstract conceptualization, and active experimentation are used to reflect four learning behaviors: feeling, watching, thinking, and doing, respectively. In this model, effective learning is observed when a learner has a concrete experience (feeling) followed by observation (watching) of and reflection on that experience which leads to the formation of abstract concepts (thinking) and conclusions which are then used for active experimentation (doing).

The four learning styles of the Kolb's model include diverging (watching and feeling), assimilating (watching and thinking), converging (thinking and doing), and accommodating (feeling and doing), and these four learning styles perfectly map to the four-stage learning cycle to demonstrate learners' special characteristics and perspectives. The matrix (see Table I) highlights the relationship between the stages of the learning cycle and the learning styles [22].

TABLE I. Relationship Between Experiential Learning Cycle and Learning Styles

	Active Experimentation (Doing)	Reflective Observation (Watching)
Concrete Experience (Felling)	**Accommodating (Feeling & Doing)**	Diverging (Watching & Feeling)
Abstract Conceptualization (Thinking)	Converging (Thinking & Doing)	Assimilating (Watching & Thinking)

Our proposed hands-on learning model adopts the accommodating learning style (highlighted in Table I) of Kolb's experiential model, focuses on hands-on (i.e., feeling and doing) and relies on intuition not logic. In this model, a learner will use other people's experiences and conduct hands-on exercises following the instruction concluded and summarized from such experiences. We adopt this learning style since we believe it is most suitable for our students now and in the near future.

V. APT Lifecycle

In general, the APT lifecycle consists of six phases: preparation, initial intrusion, expansion, persistent, search and exfiltration, and cleanup [20]. These six phases can be executed in sequence. However, adversaries may have multiple APT attacks running in parallel against the same target and each attack consists of one or more operations (see Fig. 1). Phases such as preparation and initial intrusion are the prerequisites of an APT and will be executed in series. However, phases such as expansion, persistent, and search and exfiltration are primary objectives of an APT and may be executed in parallel for efficiency. Cleanup is essential to an APT and must be executed after any operations once the previous phases have fulfilled their objectives. In this section, we will introduce these six phases in detail.

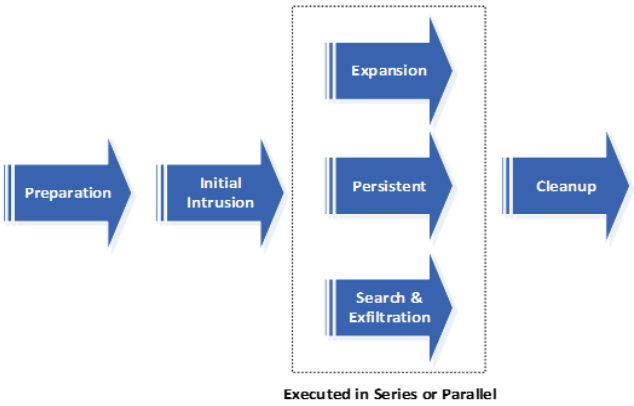

Fig. 1. APT Lifecycle

- Preparation: in this phase, APT actors will define and research targets and build or acquire hacking tools. Some common places that threat actors use to research profiles of targets are social medias, public data, and websites of the targets.
- Initial Intrusion: in this phase, APT actors will deploy initial intrusions and initiate outcome outbound connections. Phishing emails containing a malicious link or attachment are the most common tactics for initiating an intrusion.
- Expansion: in this phase, APT actors will expand access to other compromised systems in the network and seek opportunities to escalate their privilege to obtain more credentials.
- Persistent: in this phase, APT actors will employ various tactics to maintain access to the targets until there is no need of the targets.

- Search and exfiltration: in this phase, APT actors will exfiltrate the data that they searched for.
- Cleanup: in this phase, APT actors will cover their tracks and remain undetected.

VI. Mapping APT Knowledge, Skills, Abilities to the NICE Cybersecurity Workforce Framework

The components of the NICE Cybersecurity Workforce Framework [18] include category, specialty area, work role, tasks, knowledge, skills, and abilities. To represent the relationship between these components, a tree structure will be constructed. For this tree, category will be at the roots, specialty area will be at the first layer of the tree, work role will be at the second layer of the tree, and others will be on the third layer of the tree. Overall, the NICE framework has 7 categories, 33 specialty areas, 52 work roles, 1007 tasks, 630 knowledge units, 374 skills, and 176 abilities. Therefore, to represent this NICE framework, 7 trees will be constructed. Since the main purpose of the framework is to define cybersecurity roles across federal agencies, work role will be at the center stage of all components. Meanwhile, the framework clearly defines tasks needed for every work role and identify the knowledge, skills, and abilities required for conducting such tasks.

APT actor is not a work role in the existing NICE Cybersecurity Workforce Framework. Therefore, there are no pre-defined tasks, knowledge, skills, and abilities available. To study what knowledge, skills, and abilities required for conducting an APT attack and what tasks could be performed by an APT actor, we have thoroughly investigated NICE Cybersecurity Workforce Framework. We have summarized that an APT actor has performed 10 tasks and required 63 knowledge units, 37 skills and 9 abilities. The details of them are listed on Appendix A.

VII. Mapping the APT Lifecycle with Knowledge Units of the NICE Cybersecurity Framework

In this section, we have distributed 63 knowledge units of NICE Cybersecurity Workforce Framework required for an APT actor to the six phases of the APT lifecycle introduced in Section V. Overall, we observed that 11 knowledge units of the NICE Framework can be categorized into the phase of preparation, 15 knowledge units can be categorized into the phase of initial intrusion, 25 knowledge units can be categorized into the phase of expansion, 6 knowledge units can be categorized into the phase of persistent, 5 knowledge units can be categorized into the phase of search and exfiltration, and 1 knowledge unit can be categorized into the phase of cleanup, as shown in Table II. Each knowledge unit can appear only once and is arranged to be covered in the earliest phase of APT lifecycle.

TABLE II. Lifecycle and Knowledge Units of NICE Cybersecurity Workforce Framework Required for an APT Actor

APT Lifecycle	NICE Cybersecurity Workforce Framework Knowledge Units
Preparation	K0111, K0144, K0162, K0177, K0302, K0447, K0474, K0535, K0538, K0548, K0604
Initial Intrusion	K0001, K0004, K0060, K0113, K0131, K0151, K0161, K0206, K0224, K0234, K0310, K0362, K0408, K0436, K0603
Expansion	K0005, K0006, K0007, K0009, K0049, K0059, K0061, K0062, K0070, K0106, K0110, K0119, K0129, K0147, K0160, K0174, K0221, K0272, K0301, K0318, K0332, K0336, K0397, K0471, K0475
Persistent	K0002, K0013, K0259, K0298, K0367, K0523
Search and Exfiltration	K0116, K0117, K0132, K0308, K0536
Cleanup	K0003

VIII. NDG Ethical Hacking Lab Series

The ethical hacking laboratory adopted by this project is the NDG ethical hacking lab series [19] which could prepare students for the EC-Council Certified Ethical Hacking (CEH) certification [24], Offensive Security Penetration Testing with Kali Linux (PWK) certification [25], and GIAC Penetration Tester (GPEN) certification [26]. The Lab consists of 5 virtual machines: Kali Linux [27], pfSense Firewall [28], OWASP Broken Web App [29], OpenSUSE [30], and Security Onion [31]. The topology and network configuration of the Lab are depicted in Fig. 2. Details of these virtual machines are introduced below.

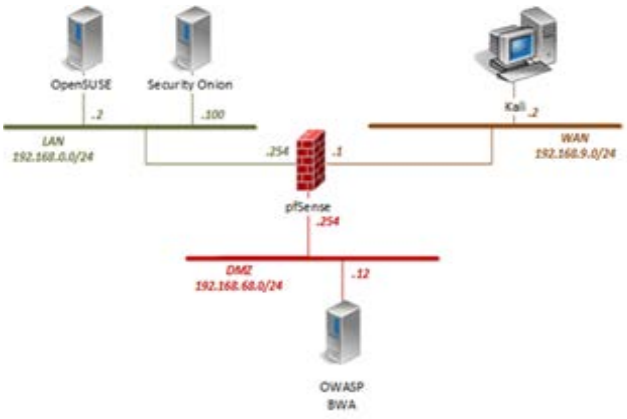

Fig. 2. NDG Ethical Hacking Lab Topology and Network Configuration [19]

- Kali Linux: Kali Linux is designed for penetrating and digital forensics. It is a Debian-derived Linux distribution maintained and funded by Offensive Security.
- pfSense Firewall: pfSense firewall is an open source firewall software based on FreeBSD. It is used to mimic a physical firewall.
- OWASP Broken Web App (BWA): OWASP BWA is a collection of vulnerable web applications that are distributed on a virtual machine in VMware format.
- OpenSUSE: OpenSUSE is an open source Linux operating system based on SUSE Linux. It is suitable for general purpose use.
- Security Onion: Security Onion is an open source platform for threat hunting, network security monitoring, and log management.

The lab series include 20 exercises [19]. Descriptions of these labs are listed on Appendix B, objectives of these labs are listed on Appendix C, and their topics are listed on Table III.

TABLE III. TOPICS OF NDG ETHICAL HACKING LABS

Lab Topic
Lab 1: Reconnaissance with Nmap & Amap
Lab 2: Social Engineering Attacks with Social Engineering Toolkit
Lab 3: Metasploit Framework Fundamentals
Lab 4: Web Pentesting with Nikto & OWSP Zap
Lab 5: Password Cracking with John the Ripper and Hashcat
Lab 6: Creating and Installing SSL Certificates
Lab 7: Vulnerability Scanning with OpenVAS
Lab 8: Enumerating SMB with enum4linux
Lab 9: Backdooring with Netcat
Lab 10: Packet Crafting with Scapy
Lab 11: Network Analysis
Lab 12: Client Side Exploitations
Lab 13: Testing Firewall Rules with Firewalking
Lab 14: Understanding SQL Commands & Injections
Lab 15: Understanding Buffer Overflows
Lab 16: Evading IDS
Lab 17: Packet Crafting with Hping
Lab 18: VNC as a Backdoor
Lab 19: Auditing Linux Systems
Lab 20: Anti-Virus Evasion

IX. APPLYING THE NDG ETHICAL HACKING LABS TO THE APT LIFECYCLE

Table IV introduces how the 20 NDG ethical hacking labs have been used to represent the APT lifecycle. We analyze the objectives and contents of each lab and identify which stages of the lifecycle are covered by them. Overall, we observe that 10 labs cover the phase of preparation, 13 labs cover the phase of initial intrusion, 13 labs cover the phase of expansion, 5 labs cover the phase of persistent, and 3 labs cover the phase of cleanup.

TABLE IV. APT LIFECYCLE AND NDG ETHICAL HACKING LABS

APT Lifecycle	**NDG Ethical Hacking Labs**
Preparation	Lab 1, Lab 2, Lab 3, Lab 4, Lab 6, Lab 7, Lab 12, Lab 13, Lab 19
Initial Intrusion	Lab 3, Lab 4, Lab 5, Lab 8, Lab 9, Lab 10, Lab 12, Lab 14, Lab 15, Lab 16, Lab 17, Lab 18, Lab 20
Expansion	Lab 3, Lab 4, Lab 5, Lab 8, Lab 9, Lab 10, Lab 12, Lab 14, Lab 15, Lab 16, Lab 17, Lab 18, Lab 20
Persistent	Lab 3, Lab 9, Lab 16, Lab 18, Lab 20
Search and Exfiltration	Lab 3, Lab 9, Lab 18
Cleanup	Lab 3, Lab 9, Lab 18

X. INTEGRATING THE NDG ETHICAL HACKING LABS WITH SUPPLEMENTAL LECTURES TO COMPLY WITH THE NICE CYBERSECURITY WORKFORCE FRAMEWORK

In the implementation of this project, we added several supplemental lectures to the selected ethical hacking labs (see Table III) to make sure the materials delivered for each phase of APT lifecycle will comply with the knowledge units required for the phase for an APT actor (see Table IV). Several supplemental lectures, but not limited to, were developed:

1. APT Labs with NDG Ethical Hacking Labs
2. Introduction to Advanced Persistent Threat

3. Introduction to Stuxnet
4. Introduction to Poison Ivy
5. Introduction to GhostNet
6. Duqu: The APT Reconnaissance Worm
7. Autopsy Forensics Browser User Guide I
8. PTK Forensics WiKi

The proposed hands-on model was implemented in one cybersecurity course in Spring 2021 with 12 graduate students and one summer camp in Summer 2020 with 4 high school students. At least 5 lectures related to APT and NDG Ethical Hacking labs were introduced to students before assigning hands-on exercises to them. Overall, 10 to 20 ethical hacking labs covering all six phases of APT lifecycle were assigned to students according to the length of each course and camp. The instructor also provided brief introductions for all selected labs. Students in the course and camp were requested to conduct all steps included in every lab as well as to record their observations and discoveries. So far, the approach of adopting hands-on ethical hacking labs in APT learning shows a promising outcome. All students in the course and camp successfully completed the course and camp with a decent grade.

XI. Conclusion

We proposed a hands-on learning model for adopting the NDG ethical hacking lab series into APT learning. In this research, we introduced negative impacts caused by APTs as well as studied several hands-on learning approaches. We clearly addressed our research methodology which consists of the following procedures: studying and examining hands-on learning models, understanding the APT lifecycle, mapping APT KSAs to the NICE Cybersecurity Workforce Framework, mapping the APT lifecycle to knowledge units of the NICE Cybersecurity Workforce Framework, explaining the NDG ethical hacking lab series, and mapping the APT lifecycle to the NDG ethical hacking labs. We also discussed our preliminary research results which indicates the proposed hands-on learning model achieves the objectives of this research.

In the future, we would like to adopt more hands-on exercises and supplemental materials as well as build appropriate assessment strategies. We also would like to expand our scope to include all four learning styles of the Kolb's model to accommodate more students with different learning styles.

References

[1] Advanced Persistent Threats (APTs), IT Governance, https://www.itgovernance.co.uk/advanced-persistent-threats-apt

[2] Ping Chen, Lieven Desmet, Christophe Huygens, A Study on Advanced Persistent Threats, Communications and Multimedia Security, CMS 2014, Lecture Notes in Computer Science, vol 8735, https://link.springer.com/chapter/10.1007/978-3-662-44885-4 5 #citeas

[3] Tyler Wrightson, Advanced Persistent Threat Hacking, The Art and Science of Hacking Any Organization, McGraw-Hill Education, 2015.

[4] Nikos Virvilis, Dimitris Gritzalis, "The Big Four - What we did wrong in Advanced Persistent Threat detection?" 2013 International Conference on Availability, Reliability and Security, 2-6 Sept. 2013, Regensburg, Germany.

[5] Paul K. Kerr, John Rollins, Catherine A. Theohary, "The Stuxnet Computer Worm: Harbinger of an Emerging Warfare Capability," Congressional Research Service, December 9, 2010, https://www.hsdl.org/?view &did=12982

[6] The Duqu 2.0 Technical Details, Kaspersky, Version 2.0, 11 June 2015, https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2018/03/07205202/The_Mystery_of_Duqu_2_0_a_sophisticated_cyberespionage_actor_returns.pdf

[7] FireEye, "POISON IVY: Assessing Damage and Extracting Intelligence," 2013, https://www.fireeye.com/content/dam/fireeye-www/global/en/current-threats/pdfs/rpt-poison-ivy.pdf

[8] Jeff Jarmoc, "RSA Compromise: Impacts on SecurID," Secureworks, March 7, 2011, https://www.secureworks.com/research/rsacompromise

[9] FireEye, "Highly Evasive Attacker Leverages SolarWinds Supply Chain to Compromise Multiple Global Victims With SUNBURST Backdoor," Dec. 13, 2020, https://www.fireeye.com/blog/threat-research/2020/12/evasive-attacker-leverages-solarwinds-supply-chain-compromises-with-sunburst-backdoor.html

[10] Christopher Bing, "Suspected Russian hackers spied on U.S. Treasury emails – sources," Reuters, Dec. 13, 2020, https://www.reuters.com/article/us-usa-cyber-treasury-exclsuive/suspected-russian-hackers-spied-on-u-s-treasury-emails-sources-idUKKBN28N0PG?edition-redirect=uk

[11] Isabella Jibilian and Katie Canales, "The US is readying sanctions against Russia over the SolarWinds cyber attack. Here's a simple explanation of how the massive hack happened and why it's such a big deal," Business Insider, April 15, 2021, https://www.businessinsider.com/solarwinds-hack-explained-government-agencies-cyber-security-2020-12

[12] Susan A. Lisko, Valeri O'dell, "Integration of Theory and Practice Experiential Learning Theory and Nursing Education", Nursing Education Perspective, Volume 31, Issue 2, Page 106-108, March 2010.

[13] Vijaykumar Bhajantri, C. Sujatha, Y. Shilpa, Manjula Pawar, "An Experiential Learning in Web Technology Course", 2016 International Conference on Learning and Teaching in Computing and Engineering (LaTICE), 31 March-3 April 2016, Mumbai, India.

[14] Ian McLoughlin, Koji Nakano, "A perspective on the experiential learning of computer architecture," 2010 IEEE/ACM International Conference on Green Computing and Communications & 2010 IEEE/ACM International Conference on Cyber, Physical and Social Computing, 18-20 Dec. 2010, Hangzhou, China.

[15] Reid Holmes, Meghan Allen, Michelle Craig, "Dimensions of Experientialism for Software Engineering Education," 2018 ACM/IEEE 40th International Conference on Software Engineering: Software Engineering Education and Training, May 27-June 3, 2018, Gothenburg, Sweden.

[16] Weishi Shi, Saad Khan, Yasmine El-Glaly, Samuel Malachowsky, Qi Yu, Daniel E. Krutz, "Experiential Learning in Computing Accessibility Education," 2020 IEEE/ACM 42nd International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), July 6–11, 2020, Seoul, Republic of Korea.

[17] Anthony Peruma, Samuel A. Malachowsky, Daniel E. Krutz, "Providing an Experiential Cybersecurity Learning Experience Through Mobile Security Labs," 2018 ACM/IEEE 1st International Workshop on Security Awareness from Design to Deployment, May 27, 2018, Gothenburg, Sweden.

[18] William Newhouse, Stephanie Keith, Benjamin Scribner, Greg Witte, "National Initiative for Cybersecurity Education (NICE) Cybersecurity Workforce Framework," NIST Special Publication

800-181 Revision 1, November 2020. https://doi.org/10.6028/NIST.SP.800-181r1

[19] NDG NetLab+, Ethical Hacking Lab Series, Network Development Group, 2016, www.netdevgroup.com

[20] Dell SecureWorks, "Lifecycle of the Advanced Persistent Threat," 2012, http://docs.media.bitpipe.com/io_10x/io_105022/item_550605/Lifecycle_of_the_Advanced_Persistent_Threat%5B1%5D.pdf

[21] David A. Kolb, Experiential learning: Experience as the source of learning and development, 1984, Prentice-Hall, Englewood Cliffs, New Jersey, USA.

[22] Saul McLeod, "Kolb's Learning Styles and Experiential Learning Cycle," SimplyPsychology, October 24, 2017, https://www.simplypsychology.org/learning-kolb.html

[23] Cao Yonghui, Liu Hui, "Study of Experiential Learning as a Model for Teaching and Learning", 2009 Second International Symposium on Knowledge Acquisition and Modeling, 30 Nov.-1 Dec. 2009, Wuhan, China.

[24] EC-Council Certified Ethical Hacker (CEH), https://www.eccouncil.org/programs/certified-ethical-hacker-ceh/

[25] Offensive Security Penetration Testing with Kali Linux (PWK), https://www.offensive-security.com/offsec/pwk-2020-update/

[26] GIAC Penetration Tester (GPEN), https://www.giac.org/certification/penetration-tester-gpen

[27] Kali Linux, https://www.kali.org/

[28] pfSense firewall, https://www.pfsense.org/

[29] OWASP Broken Web App, https://owasp.org/www-project-broken-web-applications/

[30] OpenSUSE, https://www.opensuse.org/

[31] Security Onion, https://securityonionsolutions.com/

APPENDIX A: TASKS, KNOWLEDGE UNITS, SKILLS, AND ABILITIES IN NICE CYBERSECURITY WORKFORCE FRAMEWORK RELATED TO APT ACTOR

10 Tasks Performed by an APT actor:

- T0567: Analyze target operational architecture for ways to gain access.
- T0579: Assess target vulnerabilities and/or operational capabilities to determine course of action.
- T0591: Perform analysis for target infrastructure exploitation activities.
- T0616: Conduct network scouting and vulnerability analyses of systems within a network.
- T0618: Conduct on-net activities to control and exfiltrate data from deployed technologies.
- T0624: Conduct target research and analysis.
- T0643: Deploy tools to a target and utilize them once deployed (e.g., backdoors, sniffers).
- T0653: Apply analytic techniques to gain more target information.
- T0664: Develop new techniques for gaining and keeping access to target systems.
- T0756: Operate and maintain automated systems for gaining and maintaining access to target systems.

63 Knowledge Units Need for an APT Actor:

- K0001: Knowledge of computer networking concepts and protocols, and network security methodologies.
- K0002: Knowledge of risk management processes (e.g., methods for assessing and mitigating risk).
- K0003: Knowledge of laws, regulations, policies, and ethics as they relate to cybersecurity and privacy.
- K0004: Knowledge of cybersecurity and privacy principles.
- K0005: Knowledge of cyber threats and vulnerabilities.
- K0006: Knowledge of specific operational impacts of cybersecurity lapses.
- K0007: Knowledge of authentication, authorization, and access control methods.
- K0009: Knowledge of application vulnerabilities.
- K0013: Knowledge of cyber defense and vulnerability assessment tools and their capabilities.
- K0049: Knowledge of information technology (IT) security principles and methods (e.g., firewalls, demilitarized zones, encryption).
- K0059: Knowledge of new and emerging information technology (IT) and cybersecurity technologies.
- K0060: Knowledge of operating systems.
- K0061: Knowledge of how traffic flows across the network (e.g., Transmission Control Protocol [TCP] and Internet Protocol [IP], Open System Interconnection Model [OSI], Information Technology Infrastructure Library, current version [ITIL]).
- K0062: Knowledge of packet-level analysis.
- K0070: Knowledge of system and application security threats and vulnerabilities (e.g., buffer overflow, mobile code, cross-site scripting, Procedural Language/Structured Query Language [PL/SQL] and injections, race conditions, covert channel, replay, return-oriented attacks, malicious code).
- K0106: Knowledge of what constitutes a network attack and a network attack's relationship to both threats and vulnerabilities.
- K0110: Knowledge of adversarial tactics, techniques, and procedures.
- K0111: Knowledge of network tools (e.g., ping, traceroute, nslookup).
- K0113: Knowledge of different types of network communication (e.g., LAN, WAN, MAN, WLAN, WWAN).
- K0116: Knowledge of file extensions (e.g., .dll, .bat, .zip, .pcap, .gzip).
- K0117: Knowledge of file system implementations (e.g., New Technology File System [NTFS], File Allocation Table [FAT], File Extension [EXT]).
- K0119: Knowledge of hacking methodologies.
- K0129: Knowledge of command-line tools (e.g., mkdir, mv, ls, passwd, grep).
- K0131: Knowledge of web mail collection, searching/analyzing techniques, tools, and cookies.
- K0132: Knowledge of which system files (e.g., log files, registry files, configuration files) contain relevant information and where to find those system files.
- K0144: Knowledge of social dynamics of computer attackers in a global context.
- K0147: Knowledge of emerging security issues, risks, and vulnerabilities.

- K0151: Knowledge of current and emerging threats/threat vectors.
- K0160: Knowledge of the common attack vectors on the network layer.
- K0161: Knowledge of different classes of attacks (e.g., passive, active, insider, close-in, distribution attacks).
- K0162: Knowledge of cyber attackers (e.g., script kiddies, insider threat, non-nation state sponsored, and nation sponsored).
- K0174: Knowledge of networking protocols.
- K0177: Knowledge of cyber attack stages (e.g., reconnaissance, scanning, enumeration, gaining access, escalation of privileges, maintaining access, network exploitation, covering tracks).
- K0206: Knowledge of ethical hacking principles and techniques.
- K0221: Knowledge of OSI model and underlying network protocols (e.g., TCP/IP).
- K0224: Knowledge of system administration concepts for operating systems such as but not limited to Unix/Linux, IOS, Android, and Windows operating systems.
- K0234: Knowledge of full spectrum cyber capabilities (e.g., defense, attack, exploitation).
- K0259: Knowledge of malware analysis concepts and methodologies.
- K0272: Knowledge of network analysis tools used to identify software communications vulnerabilities.
- K0298: Knowledge of countermeasures for identified security risks.
- K0301: Knowledge of packet-level analysis using appropriate tools (e.g., Wireshark, tcpdump).
- K0302: Knowledge of the basic operation of computers.
- K0308: Knowledge of cryptology.
- K0310: Knowledge of hacking methodologies.
- K0318: Knowledge of operating system command-line tools.
- K0332: Knowledge of network protocols such as TCP/IP, Dynamic Host Configuration, Domain Name System (DNS), and directory services.
- K0336: Knowledge of access authentication methods.
- K0362: Knowledge of attack methods and techniques (DDoS, brute force, spoofing, etc.).
- K0367: Knowledge of penetration testing.
- K0397: Knowledge of security concepts in operating systems (e.g., Linux, Unix.)
- K0408: Knowledge of cyber actions (i.e., cyber defense, information gathering, environment preparation, cyber-attack) principles, capabilities, limitations, and effects.
- K0436: Knowledge of fundamental cyber operations concepts, terminology/lexicon (i.e., environment preparation, cyber-attack, cyber defense), principles, capabilities, limitations, and effects.
- K0447: Knowledge of how to collect, view, and identify essential information on targets of interest from metadata (e.g., email, http).
- K0471: Knowledge of Internet network addressing (IP addresses, classless inter-domain routing, TCP/UDP port numbering).
- K0474: Knowledge of key cyber threat actors and their equities.
- K0475: Knowledge of key factors of the operational environment and threat.
- K0523: Knowledge of products and nomenclature of major vendors (e.g., security suites - Trend Micro, Symantec, McAfee, Outpost, and Panda) and how those products affect exploitation and reduce vulnerabilities.
- K0535: Knowledge of strategies and tools for target research.
- K0536: Knowledge of structure, approach, and strategy of exploitation tools (e.g., sniffers, keyloggers) and techniques (e.g., gaining backdoor access, collecting/exfiltrating data, conducting vulnerability analysis of other systems in the network).
- K0538: Knowledge of target and threat organization structures, critical capabilities, and critical vulnerabilities.
- K0548: Knowledge of target or threat cyber actors and procedures.
- K0603: Knowledge of the ways in which targets or threats use the Internet.
- K0604: Knowledge of threat and/or target systems.

37 Skills Needed for an APT Actor:

- S0001: Skill in conducting vulnerability scans and recognizing vulnerabilities in security systems.
- S0011: Skill in conducting information searches.
- S0044: Skill in mimicking threat behaviors.

- S0046: Skill in performing packet-level analysis using appropriate tools (e.g., Wireshark, tcpdump).
- S0051: Skill in the use of penetration testing tools and techniques.
- S0052: Skill in the use of social engineering techniques. (e.g., phishing, baiting, tailgating, etc.).
- S0056 Skill in using network management tools to analyze network traffic patterns (e.g., simple network management protocol).
- S0057: Skill in using protocol analyzers.
- S0063: Skill in collecting data from a variety of cyber defense resources.
- S0076: Skill in configuring and utilizing software-based computer protection tools (e.g., software firewalls, antivirus software, anti-spyware).
- S0077: Skill in securing network communications.
- S0078: Skill in recognizing and categorizing types of vulnerabilities and associated attacks.
- S0081: Skill in using network analysis tools to identify vulnerabilities. (e.g., fuzzing, nmap, etc.).
- S0089: Skill in one-way hash functions (e.g., Secure Hash Algorithm [SHA], Message Digest Algorithm [MD5]).
- S0094: Skill in reading Hexadecimal data.
- S0095: Skill in identifying common encoding techniques (e.g., Exclusive Disjunction [XOR], American Standard Code for Information Interchange [ASCII], Unicode, Base64, Uuencode, Uniform Resource Locator [URL] encode).
- S0113: Skill in performing format conversions to create a standard representation of the data.
- S0137: Skill in conducting application vulnerability assessments.
- S0156: Skill in performing packet-level analysis.
- S0158: Skill in operating system administration. (e.g., account maintenance, data backups, maintain system performance, install and configure new hardware/software).
- S0167: Skill in recognizing vulnerabilities in security systems. (e.g., vulnerability and compliance scanning).
- S0170: Skill in configuring and utilizing computer protection components (e.g., hardware firewalls, servers, routers, as appropriate).
- S0177: Skill in analyzing a target's communication networks.
- S0184: Skill in analyzing traffic to identify network devices.
- S0198: Skill in conducting social network analysis.
- S0199: Skill in creating and extracting important information from packet captures.
- S0221: Skill in extracting information from packet captures.
- S0225: Skill in identifying a target's communications networks.
- S0226: Skill in identifying a target's network characteristics.
- S0231: Skill in identifying how a target communicates.
- S0240: Skill in interpreting metadata and content as applied by collection systems.
- S0241: Skill in interpreting traceroute results, as they apply to network analysis and reconstruction.
- S0242: Skill in interpreting vulnerability scanner results to identify vulnerabilities.
- S0248: Skill in performing target system analysis.
- S0269: Skill in researching vulnerabilities and exploits utilized in traffic.
- S0293: Skill in using tools, techniques, and procedures to remotely exploit and establish persistence on a target.
- S0304: Skill to access information on current assets available, usage.

9 Abilities Needed for an APT Actor:

- A0001: Ability to identify systemic security issues based on the analysis of vulnerability and configuration data.
- A0010: Ability to analyze malware.
- A0015: Ability to conduct vulnerability scans and recognize vulnerabilities in security systems.
- A0086: Ability to expand network access by conducting target analysis and collection to identify targets of interest.
- A0092: Ability to identify/describe target vulnerability.
- A0093: Ability to identify/describe techniques/methods for conducting technical exploitation of the target.
- A0107: Ability to think like threat actors.
- A0159: Ability to interpret the information collected by network tools (e.g., Nslookup, Ping, and Traceroute).

- A0175: Ability to examine digital media on multiple operating system platforms.

Appendix B: Descriptions of NDG Ethical Hacking Labs

- Lab 1: Reconnaissance with Nmap & Amap - This lab introduces Nmap the "network mapper" and its usage to perform basic network port reconnaissance and scanning. Additionally, the use of the Amap "application mapper" tool in order to determine which applications are running on listening ports.
- Lab 2: Social Engineering Attacks with Social Engineering Toolkit - The SET toolkit or "Social Engineering Toolkit" is an effective prepackaged toolkit for performing reconnaissance against a target. This lab demonstrates the use of some of its available attacks.
- Lab 3: Metasploit Framework Fundamentals - Metasploit is a penetration testing framework that is used for conducting security assessments. The lab introduces its fundamental usage and available options to conduct a penetration test.
- Lab 4: Web Pentesting with Nikto & OWSP Zap - Enterprise applications are increasingly using web interfaces for their user interface. This lab uses two well-known web application assessment tools for conducting security assessments.
- Lab 5: Password Cracking with John the Ripper and Hashcat - This lab introduces the methodologies used for cracking both Linux and Windows passwords using two different tools. In addition, this lab examines how to create supporting wordlists in order to create dictionaries for the tools.
- Lab 6: Creating and Installing SSL Certificates - SSL (Secure Socket Layer) is used to secure communication between a client and web server throughout the internet. This lab demonstrates how to create a self-signed X.509 certificate and install it into a Linux web server.
- Lab 7: Vulnerability Scanning with OpenVAS - There are several commercial tools available for performing vulnerability scanning. In this lab, we will be using OpenVAS, an open source vulnerability scanner to perform security assessments.
- Lab 8: Enumerating SMB with enum4linux - NetBIOS is a commonly attacked program on Windows machines, however, Linux servers with a SAMBA installed also use NetBIOS. This lab addresses the vulnerabilities of NetBIOS and how to exploit them.
- Lab 9: Backdooring with Netcat - Netcat is installed in most Linux distributions. It can be used at a fundamental TCP/IP level to perform various functions. This lab explores some of the ways Netcat can be used.
- Lab 10: Packet Crafting with Scapy - Building a packet field-by-field demonstrates how someone could manipulate the packet traffic entering or leaving a network. This lab shows how to build packets layer-by-layer using Scapy, a packet manipulation tool and then implementing the finished packets to perform various network functions.
- Lab 11: Network Analysis - The ability to capture and analyze packets is an important skill when performing a security assessment or investigating a potential network breach. This lab will demonstrate how to capture and analyze network packets.
- Lab 12: Client Side Exploitations - Browsers are susceptible to exploitation and can be used to gain access to the computer system and network. In this lab, we will use the BeEF framework to specifically target the browser and exploit the browser.
- Lab 13: Testing Firewall Rules with Firewalking - Firewall rules or ACLs are fundamental in controlling ingress and egress traffic in a network. In this lab, we use a method of testing whether those rules are properly configured.
- Lab 14: Understanding SQL Commands & Injections - SQL (Structured Query Language) is used by many databases as a language to query, insert and delete elements. This lab demonstrates how to build, query, and delete elements in a database and how these skills can be used to attack a database.
- Lab 15: Understanding Buffer Overflows - Buffer overflows are programming errors whether intentional or accidental. This lab shows how to create a vulnerable program and demonstrate the vulnerability.
- Lab 16: Evading IDS - Different methods can be employed to attempt to thwart IDS detection. This lab explores the different methods that can be employed to hide from IDS systems.
- Lab 17: Packet Crafting with Hping - Hping is a TCP/IP packet assembler and analyzer. In this lab, we will use hping to create packets as well as perform different network functions with the packets.
- Lab 18: VNC as a Backdoor - The ability to get through a firewall once a system is compromised is a skill used by both hackers and pen testers. Using the open source tool TightVNC the lab will show how to create a reverse connection through the firewall.

- Lab 19: Auditing Linux Systems - Testing a system for security issues is part of a security assessment. Using the open source tool Lynis in this lab demonstrates how to assess a Linux system and evaluate what vulnerabilities exist in its configuration.
- Lab 20: Anti-Virus Evasion - The ability to package an exploit and make it undetectable to anti-virus programs is a method to gain access to a system. This lab introduces the Veil framework to create and hide exploits to bypass anti-virus detection.

APPENDIX C: OBJECTIVES OF NDG ETHICAL HACKING LABS

Lab 1: Reconnaissance with Nmap & Amap

- Reconnaissance Using network mapper (Nmap)
- Using application mapper (Amap) for Reconnaissance

Lab 2: Social Engineering Attacks with Social Engineering Toolkit

- Using the Social Engineering Toolkit (SET)
- Modifying the SET Parameters
- Test the SET Attack

Lab 3: Metasploit Framework Fundamentals

- Getting Familiar with Metasploit
- Vulnerability Scanning Using the WMAP Module
- Configuring Exploits and Payloads

Lab 4: Web Pentesting with Nikto & OWSP Zap

- Scanning With Nikto
- Scanning With OWASP Zap

Lab 5: Password Cracking with John the Ripper and Hashcat

- Generating Password Lists for Password Cracking
- Create User Accounts to be Cracked
- Password Cracking Using John the Ripper
- Password Cracking Using Hashcat

Lab 6: Creating and Installing SSL Certificates

- Creating a Self-Signed Certificate
- Configuring the Apache SSL File
- Testing the SSL Certificate

Lab 7: Vulnerability Scanning with OpenVAS

- Using OpenVAS
- Quick Scanning with OpenVAS
- Customized Scanning with OpenVAS

Lab 8: Enumerating SMB with enum4linux

- Enumerating the Samba Server with enum4linux
- Cracking Samba Users with xHydra

Lab 9: Backdooring with Netcat

- Port Scanning with Netcat
- Establishing Connections with Netcat
- Transferring Files with Netcat

Lab 10: Packet Crafting with Scapy

- Creating Packets with Scapy
- Sending Crafted Packets

Lab 11: Network Analysis

- Capturing Traffic with tcpdump
- Analyzing Traffic with Wireshark
- Analyzing Traffic with Xplico

Lab 12: Client Side Exploitations

- Hooking Browsers with BeEF Framework
- Client Exploitation with BeEF Framework

Lab 13: Testing Firewall Rules with Firewalking

- Navigating to the pfSense Dashboard
- Scan for Firewall Rules with Firewalk
- Configuring ACL Rules
- Test Configured Firewall Rules with Firewalk

Lab 14: Understanding SQL Commands & Injections

- Basic SQL Commands
- Querying with SQL
- Deleting with SQL
- SQL Injection

Lab 15: Understanding Buffer Overflows

- Writing a Buffer Overflow Program
- Run Code to Demonstrate Buffer Overflow
- Analyzing and Modifying Overflow Code

Lab 16: Evading IDS

- Initialize Network Monitoring Applications
- Test IDS Results with Regular Nmap Scan
- Test IDS Results with Low MTU Scan
- Test IDS Results with Decoy Scan

- •Test IDS Results with Spoofed MAC Scan

Lab 17: Packet Crafting with Hping

- Using Hping as an ICMP Utility
- Using Hping for Port Scanning

Lab 18: VNC as a Backdoor

- Using TightVNC
- Reversing VNC Connection

Lab 19: Auditing Linux Systems

- Getting Familiarized with Lynis
- Auditing with Lynis

Lab 20: Anti-Virus Evasion

- Creating Malicious Payloads Using the Veil Framework