Interactive Framework for Cybersecurity Education and Future Workforce Development

Sujan Ghimire[†], Muhtasim Alam Chowdhury[*], Ryan Tsang[‡], Richard Yarnell[§], Emma Heckert[*], Jaeden Carpenter[*], Yu-Zheng Lin[†], Muntasir Mamun[†], Ronald F. DeMara[§], Setareh Rafatirad[¶], Pratik Satam[†], and Soheil Salehi[*]

[†]Department of Systems and Industrial Engineering, University of Arizona, Tucson, AZ 85721
[*]Department of Electrical and Computer Engineering, University of Arizona, Tucson, AZ 85721
[‡]Department of Electrical and Computer Engineering, University of California Davis, Davis, CA 95616
[§]Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL 32826
[¶]Department of Computer Science, University of California Davis, Davis, CA 95616
{[†]sghimire, [*]mmc7, [*]carpenterjaeden, [*]emmaheckert, [†]yuzhenglin, [†]muntasir, [†]pratiksatam [*]ssalehi}@arizona.edu;
{[‡]rchtsang, [¶]srafatirad}@ucdavis.edu; {[§]richardyarnell, [§]ronald.demara}@ucf.edu

2024 IEEE Frontiers in Education Conference (FIE) | 979-8-3503-5150-7/24/$31.00 ©2024 IEEE | DOI: 10.1109/FIE61694.2024.10892909

***Abstract*—This research-to-practice paper presents a novel pedagogical tool for hardware cybersecurity education and workforce development. The growing importance of hardware security has made it essential for individuals and organizations to understand hardware security principles and best practices. However, the current educational curriculum falls short of fulfilling these emerging demands due to the rapidly changing hardware security landscape and limited opportunities for hands-on training. To address these challenges, we propose and have developed the Interactive Hardware and Cybersecurity (I-HaC) Educational Framework, a pedagogical educational framework that supplements existing courses by leveraging generative AI for individualized instruction related to hardware and cybersecurity, data mining, and applied Machine Learning (ML), as well as data visualization to enhance cybersecurity education and workforce development. The framework is designed to be utilized by graduate and undergraduate Electrical and Computer Engineering (ECE) and Computer Science (CS) students for a comprehensive introduction to cybersecurity exploits and countermeasures in an interactive manner with hands-on components. Using I-HaC, we have developed tailored lab components for a diverse range of students and intend to release I-HaC as open-source for the benefit of the ECE and CS education community.**

***Index Terms*—National Vulnerability Database (NVD), Common Vulnerability and Exposure (CVE), Common Weakness Enumeration (CWE), Hardware Security, Cybersecurity Education, Future Workforce Development**

I. Introduction

In our increasingly interconnected world, hardware security is crucial for safeguarding physical devices and their sensitive data, particularly with the rapid expansion of IoT and other connected technologies [1]. Addressing hardware security challenges brought by these advancements has created a high demand for experts and professionals with specialized knowledge of cyber and hardware security. However, the current educational curriculum falls short of fulfilling these emerging demands due to the rapidly changing hardware security landscape, limited opportunities for hands-on training, and predominantly emphasis on software security [2]–[5]. As modern computing platforms become more complex, security vulnerabilities at the hardware level, such as Spectre and Meltdown, have emerged, affecting the design considerations for future processors. Thus, the growing importance of hardware security has made it essential for individuals and organizations to understand its principles and best practices.

Our global digital infrastructure is heavily reliant on secure, robust hardware systems. To safeguard this, it is vital to equip future professionals with a comprehensive understanding of hardware protection and associated best practices. Enhanced education and skill development in hardware security can empower individuals and organizations to identify and address potential security threats proactively [6]. Moreover, a skilled hardware security workforce can foster innovation, driving the development of increasingly secure hardware products and systems. However, to create such effective and secure design solutions, understanding the evolution and impact of different vulnerabilities is critical. Unfortunately, valuable insights from vast datasets offered by organizations like the National Institute of Standards and Technology (NIST) and MITRE can be challenging to extract without a systematic framework for analyzing vulnerabilities and weaknesses in detail [7]–[9]. Previous works have often neglected to provide a comprehensive analytical tool that reveals patterns and relationships within the National Vulnerability Database (NVD) [10], which lists the Common Vulnerability and Exposures (CVEs) and MITRE's Common Weakness Enumeration (CWE) [11] databases.

In this work, we propose the Interactive Hardware and Cybersecurity (I-HaC) Educational Framework, a novel approach to cybersecurity education that emphasizes the importance of hardware security in the cyber-physical system (CPS) security domain. Our proposed framework will help the learners analyze a set of vulnerabilities and give them details about the impact a particular vulnerability has on the CPS system. Using the proposed I-HaC as a pedagogical educational framework, we can introduce students to a wide variety of cybersecurity exploits, attacks and possible countermeasures in an interactive fashion through hands-on components in a cybersecurity curriculum. Through the I-HaC educational framework, we aim to develop a set of comprehensive lab

components for senior undergraduate and graduate students with diverse backgrounds. The proposed I-HaC Educational Framework will be released as an open-source tool and the developed laboratory curriculum will be provided for the benefit of the broader community. This interactive learning experience will not only equip students with the knowledge and skills required to identify and mitigate hardware security risks but also inspire them to contribute to the advancement of hardware security research and the development of more secure products and systems.

II. Background

Hardware security education has long been hindered by significant shortcomings and challenges, posing a serious risk to the cybersecurity landscape. Traditional curricula have lacked emphasis on hardware security [12]–[14], while learners and educators have faced limited resources. Additionally, there is a shortage of industry professionals, including those who specialize in hardware security expertise, leaving a generation of professionals ill-prepared to address the growing threats targeting hardware systems [15]. These issues become even more critical in the context of IoT and CPS, where the interconnectivity of systems amplifies the consequences of hardware security breaches. As IoT and CPS devices become embedded in various sectors, such as healthcare, finance, and transportation, vulnerabilities in these systems can lead to severe consequences, including privacy breaches, disruption of critical infrastructure, and compromised safety [16], [17].

Advances in AI, particularly with LLMs, have revealed new attack models and vulnerabilities. As these tools grow more powerful, novel attack patterns emerge, prompting the development of diverse defense strategies [18]. Additionally, hardware security presents unique challenges due to its complexity, requiring a deep understanding of both hardware and software systems [19]. This complexity makes it difficult for learners to effectively grasp hardware security concepts, thereby complicating the identification and mitigation of vulnerabilities. Without a solid foundation in hardware security, future cybersecurity professionals may struggle to protect against sophisticated attacks that exploit hardware weaknesses [20].

To address these pressing issues, there is an urgent need for comprehensive and standardized hardware security education and workforce development. Integrating hardware security into traditional curricula and providing accessible resources and materials can equip the next generation of cybersecurity professionals with the knowledge and skills necessary to protect against evolving hardware-based threats [21], [22]. This empowers them to secure critical systems, detect vulnerabilities, and develop robust mitigation strategies.

While various frameworks have emerged in cybersecurity education, including the renowned Certified Information Systems Security Professional (CISSP) certification program [23], they often focus on software security, overlooking hardware security in the context of IoT and CPS [24]–[26], and lack clear and comprehensive information derived from key cybersecurity datasets such as the NVD and CWE, as mentioned in Section I [27]. Additionally, the rigorous and time-consuming nature of the CISSP certification exam poses challenges for learners and educators seeking a more accessible and practical learning approach. Furthermore, the CISSP program primarily targets professionals already working in the field, leaving a gap in cybersecurity education for individuals pursuing careers in related disciplines. In contrast, game-based learning offers an easily accessible method to enhance the cybersecurity education experience [26], [28]. Games like "Pomega," "What Can Go Wrong?," and "Bird's Life" [29] integrate gaming elements with educational information to effectively convey cybersecurity concepts. However, there are still areas within game-based learning that require attention, such as the initial step of categorizing vulnerabilities and attacks, as well as the provision of comprehensive and relevant information.

Our framework, I-HaC, addresses these existing shortcomings in hardware security education, with a particular focus on IoT and CPS. Even with the unstructured data from NVD and CWE, I-HaC manages to provide clear and comprehensive information derived from these datasets. Its user-friendly graphical interface enables learners and professionals to easily input vulnerability descriptions, extract valuable insights, and establish connections between vulnerabilities and their associated weaknesses. Leveraging established ontologies and industry standards, I-HaC offers a comprehensive understanding of hardware vulnerabilities and empowers users to select and implement effective mitigation measures. By bridging the gap in hardware security education, I-HaC contributes to the development of a robust cybersecurity workforce and strengthens the resilience of critical systems. In a rapidly evolving digital landscape, where hardware vulnerabilities pose escalating risks, it is essential to prioritize and invest in hardware security education.

III. Interactive Hardware and Cybersecurity (I-HaC) Educational Framework

To build our framework, getting to know the hardware security data and concepts to design the algorithms is important. In this section, we cover the concepts of data, algorithms, and models. We utilize the security vulnerability information available in the National Vulnerability Database (NVD) and Common Weakness Enumeration (CWE) database to build our ontology-based educational framework. Using the power of a robust hardware ontology, learners can gain a deeper understanding of the hardware layer's role in security and develop the necessary skills to design, analyze, and secure hardware systems effectively. Such formal representation of hardware weaknesses and vulnerabilities can help students perform threat modeling specific to hardware-based systems, assess the security risks associated with hardware components, identify potential attack vectors, and evaluate the impact of attacks on system integrity.

NVD provides a comprehensive database of vulnerabilities. It utilizes a standardized approach for identifying, assessing,

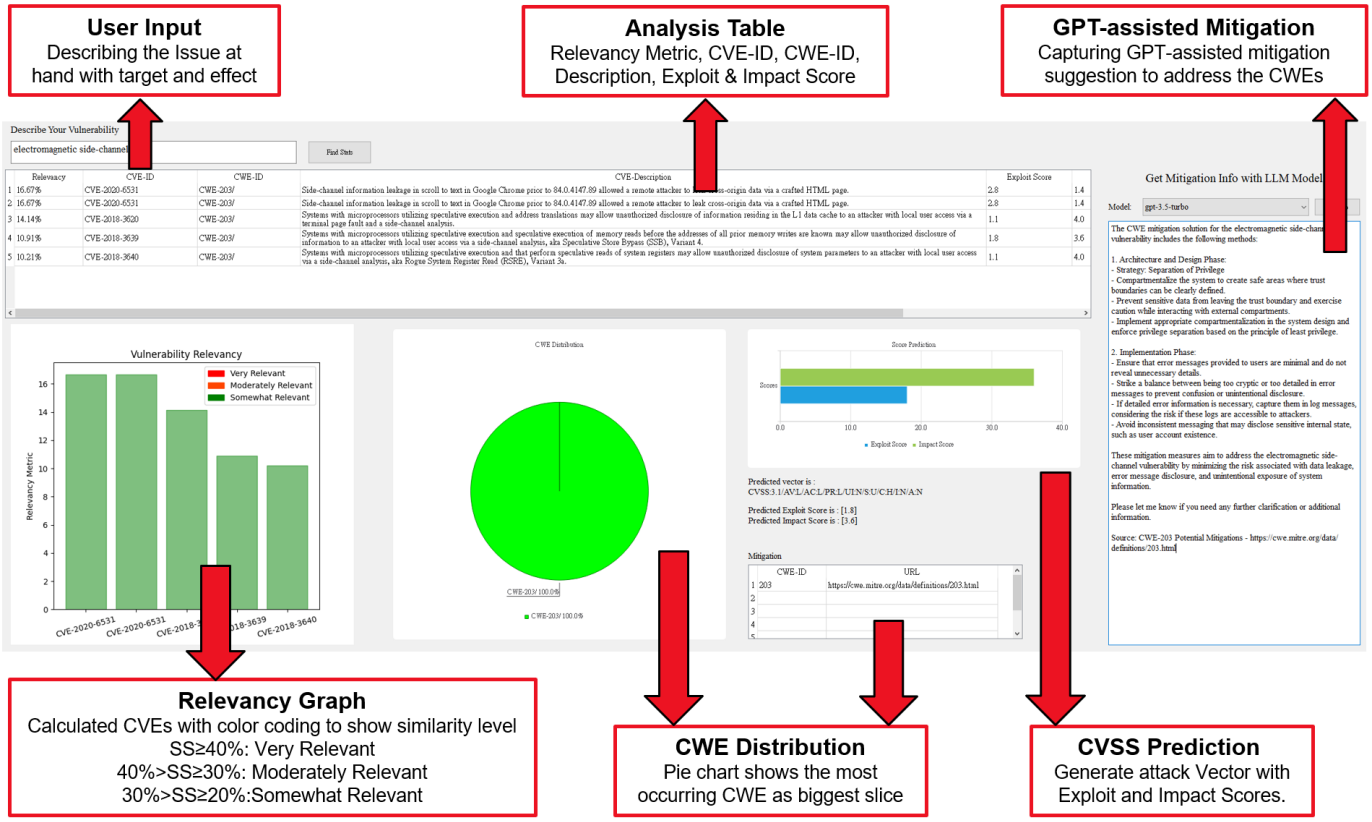

Fig. 1: Screenshot of the Graphical User Interface designed for the framework [9].

and prioritizing vulnerabilities in software and hardware products. The NVD uses JavaScript Object Notation (JSON) and receives regular updates to incorporate new information and modify existing data regarding specific threats. Additionally, CWE is a community-contributed dataset that describes common weaknesses. We use this dataset to identify the root causes of each instance of the vulnerability input and to correlate them with the relationships defined in our ontology. By doing so, we can better understand the connections and patterns among similar vulnerabilities. In the development of our ontology, we have incorporated four key concepts, namely Vulnerability, CWE, ExploitTarget, and AttackImpact. To provide a better understanding of our ontology classes, we offer a concise overview of each of these concepts below.

- ***Vulnerability:*** A system flaw that can be exploited by attackers, compromising confidentiality, integrity, or availability. It can stem from design flaws, manufacturing defects, or the insertion of malicious hardware components (hardware trojans).
- ***CWE:*** The "CWE" class describes the weakness type associated with a vulnerability.
- ***ExploitTarget:*** Refers to victim systems targeted by attackers due to existing vulnerabilities that can be exploited for unauthorized access or harm.
- ***AttackImpact:*** The "AttackImpact" class represents the potential consequences of an attacker exploiting a vulnerability in an ExploitTarget. It includes outcomes such as data theft, system compromise, or denial of service.

Below are the key object properties that capture the relationships between these classes:

- ***Exploits:*** Connects a vulnerability to its associated "ExploitTarget" class, indicating the vulnerability's target for exploitation.
- ***hasAttackImpact:*** Links the "ExploitTarget" class and the "AttackImpact" class, representing the various types of attack impacts resulting from vulnerability exploitation.
- ***TargetsCWE:*** Associates a vulnerability with its corresponding CWE, indicating the specific CWE-ID related to each vulnerability.

We construct our hardware ontology utilizing the NVD dataset from 2010 to the present. Our framework successfully detects updates in the NVD dataset and extracts the aforementioned four key concepts (Vulnerability, CWE, AttackImpact, and ExploitTarget). This involves capturing the relationship between vulnerabilities and the impact they have on victim systems when exploited by an attacker. We employ the Owlready 2.0 Python library to map these concepts and their relationships, leveraging Machine Learning (ML) techniques such as linguistic annotations and a pre-existing Natural Language Processing (NLP) corpus to dynamically relate and attach meaningful contexts to unstructured text data. Each vulnerability is then linked to a corresponding CVE-ID, enabling our ontology to model the vulnerability information along with the associated CVE-IDs. Furthermore, we establish connections between each vulnerability and a CWE-ID, categorizing the vulnerability based on its description and the

affected system.

To ensure the ontology stays up-to-date, we developed a vocabulary for updating it. This involved extracting concepts from a dictionary and repeating the process until all the aforementioned four concepts were captured. As new data became available, we assigned it to the appropriate concepts and updated our framework accordingly. Throughout the ontology development process, we utilized Protege 5.5.0 software, which supports the Ontology Web Language (OWL) and provides a user-friendly environment for ontology creation and management. Currently, our ontology comprises 1,460 axioms, including 652 logical axioms and 801 declaration axioms. It also includes 252 classes, 32 object properties, and 518 individuals, representing a comprehensive representation of the domain. To enhance the visibility of cybersecurity threats and provide an intuitive user experience, we have developed a graphical user interface (GUI) with an interactive dashboard using PyQt5, as demonstrated in Figure 1. The GUI allows users to input vulnerability descriptions and leverages our framework to extract valuable insights. Here is an overview of the process flow:

1) User input is cleaned by removing stop-words, punctuation, and irrelevant data.
2) The NLTK framework [30] is used for natural language processing to standardize the meaning of the description.
3) Tokenization and stemming are performed to further standardize the text.
4) Cosine similarity is calculated to extract relevant occurrences and map the description to existing CVEs to identify similarities in the concepts.
5) The similarity results are visualized to demonstrate the classification of the retrieved vulnerabilities and the type of CWE that was mainly portrayed in the description.
6) The established ontology is used to connect vulnerability occurrences and display relationships between them, providing a development story. OntoSpy is used to visualize the RDF models and interact with the documentation.

In the GUI, users input vulnerability descriptions, and the system processes the input to provide organized information. The top section displays a table with relevant occurrences, including CWE and CVE IDs, descriptions, and severity scores. A color-coded graph shows the relevance of estimated CVEs, providing insights into related CWEs. Another tab presents a pie chart highlighting prevalent CWEs associated with the cause. The GUI also predicts Exploit and Impact Scores using a highly accurate machine learning-based approach (up to 98.29% accuracy and 90.90% recall). Users can also explore the ontology by clicking on this tab, gaining a comprehensive understanding of the data and relationships. One notable advantage of our framework, I-HaC, is its adaptability to various security topics, particularly focused on hardware security in the context of IoT and CPS. However, its design allows for seamless adaptation to different domains and areas of cybersecurity. By defining a set of keywords, we can extract

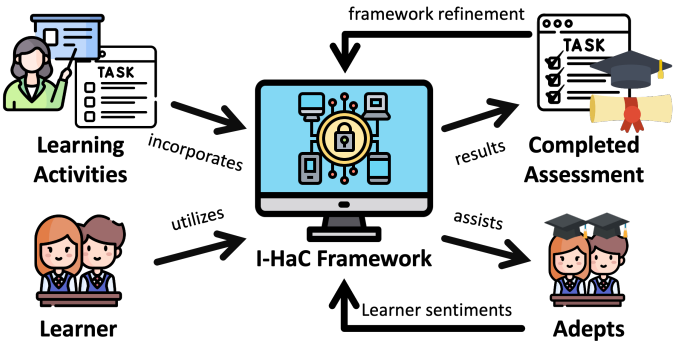

Fig. 2: I-HaC Pedagogical Flow.

information from the CWE and CVE datasets specific to software, firmware, hardware, coding security, product security, IoT security, and more. This flexibility allows our framework to effectively accommodate any cybersecurity topic.

IV. Educational Activity Design and Objectives

Figure 2 illustrates the pedagogical flow of the proposed I-HaC framework. Our I-HaC Educational Framework provides a structured and systematic way to represent, organize, and analyze information about hardware vulnerabilities. This framework helps us standardize the terminology, classification, and representation of hardware vulnerabilities, making it easier for hardware security experts, researchers, and practitioners to share and exchange information. Furthermore, the framework supports the development of new hardware security techniques and tools by providing a common basis for data analysis, decision-making, and knowledge sharing. This, in turn, will significantly enhance the efficiency and effectiveness of hardware security efforts and contribute to the continuous improvement of hardware security practices [31]. Our framework, in collaboration with industry partners, can bring real-world hardware security challenges into the classroom. Simulation and emulation tools can be utilized to demonstrate attacks and mitigation techniques. In addition, we plan to implement a web-based I-HaC Educational Framework to facilitate distance learning. The web application is designed with a user-friendly graphical user interface that incorporates all of the visual elements shown in Figure 1 to remove the necessity of local installation. By deploying the framework on cloud infrastructure, similar to the CPS-TR tool described by Satam et al. [32], such as Amazon AWS, which provides higher efficiency, scalability, ease of access, and better traceability, we can improve students' educational experiences in large class sizes. Cloud-based deployment enables students and learners to have extended access to these resources outside lab hours, which provides students with the flexibility to learn at their own pace and have more time to practice the concepts covered in each exercise. Such pedagogical approaches make the proposed I-HaC Educational Framework a highly suitable and effective educational tool for adoption in classes with fully online, hybrid, and hyflex modes of instruction (that have recently received significant attention during the pandemic) to facilitate active learning based on students' majors and background [33],

TABLE I: Proposed educational activities in the courses delivered at the University of Arizona, University of California Davis, and University of Central Florida using the I-HaC framework.

University	Course ID and Title	Brief Activity Description
UofA	ECE 413/513 Web Development and IoT	In the course project, the students are tasked to search for any vulnerabilities in their HTML, CSS, JavaScript, and firmware code and identify mitigation solutions using the tool.
	ECE 407/507 Digital VLSI Design	The course project can focus on designing secure digital VLSI systems using Verilog, Synopsys, and Cadence tools.
	SFWE 407/507 Foundations of Software Engineering	Course project to develop secure software systems and perform risk assessment and threat vector analysis.
UCDavis	EEC 172 Embedded Systems	Assignment to identify hardware and firmware vulnerabilities in sample projects to raise awareness and familiarity with potential threat vectors on deeply embedded and/or safety-critical systems.
	ECS 171 Introduction to Machine Learning	Course project to perform a comparative analysis of multiple classifiers to perform link prediction between hardware vulnerabilities and weaknesses and compare with a baseline approach.
	ECS 111 Applied Machine Learning	Course activity to interactively explore the correlations between software vulnerabilities and weaknesses to understand the application of machine learning in the computer Cyber-security domain.
UCF	EEE 4346C Hardware Security and Trusted Circuit Design	Case studies on hardware security spanning design, analysis, and synthesis of sequential logic circuits and systems.
	EEE 5790 Introduction to Secure Architectures	Learning modules covering state-of-the-art security primitives in modern processors, including Intel's Safe-Guard Extension(SGX), ARM's TrustZone, and AMD's SME and SEV.
	EEL5268 Communications and Networking for Smart Grid	Learners analyze and strengthen large-scale networks and public infrastructures to prevent and mitigate cyberattacks.

[34]. I-HaC uses learner sentiments to provide personalized activities for each learner. Furthermore, learner evaluation results are used to improve the I-HaC framework and content delivery. The objective of integrating the I-HaC framework is to enhance learners' understanding of cybersecurity concepts, including hardware security and evolving attack methods. The curriculum will familiarize students with cutting-edge tools for cybersecurity, machine learning, data analysis, and visualization. Engaging discussions, group activities, and hands-on exercises will be incorporated to support students in achieving the course learning outcomes [35], [36]. Some example activities are listed in Table I.

V. Pilot Implementation and Evaluation Plan

We plan to integrate the I-HaC framework in the courses listed in Table I. In particular, as a pilot study, we plan to create customized labs and activities for these courses tailored specifically to use the proposed I-HaC framework, which will include both video demonstrations and written documentation for its use. To evaluate the effectiveness of the framework and to improve the user/learner experience, we will conduct a pre- and post-student survey in accordance with the approaches used in [33]–[39]. This survey will gather information on students' knowledge of large language models (LLMs), hardware and software vulnerabilities, and the use of LLMs to mitigate vulnerabilities. This survey would demonstrate student knowledge comprehension and understanding of these topics before and after using the proposed I-HaC framework. In particular, students will respond to five questions, each using a 5-point Likert-type scale (Strongly Agree, Agree, Neutral, Disagree, Strongly Disagree), that will be designed to gauge their familiarity with LLMs, their experience in using LLMs for programming, and their understanding of security issues related to hardware and software vulnerabilities. Survey questions will be carefully designed to align with the course's learning objectives. The initial survey will provide a baseline understanding of student knowledge, which will be critical for tailoring course content and delivery. At the end of the semester, the post-survey results, coupled with student feedback, will be analyzed to measure the effectiveness of the course in enhancing student knowledge and confidence in using LLMs and addressing security concerns. This analysis will inform future iterations of the course and identify areas for improvement in teaching methods and materials to enhance student engagement and deepen their understanding of LLMs and security in the context of hardware design and the Internet of Things. We believe this will ultimately produce a more informed and capable cohort of engineering students. Below, we provide a few examples demonstrating the functionality of I-HaC framework for different user-defined vulnerability descriptions, where ***Example 1:*** "Use of a Cryptographic Primitive with a Risky Implementation" (Figure 3), ***Example 2:*** "Sensitive Information Uncleared Before Debug/Power State Transition" (Figure 4), and ***Example 3:*** "Improper Protection of Physical Side Channels" (Figure 5) [9].

VI. Conclusion

With the increasing use of connected devices and IoT systems, the need to be educated about hardware security in cybersecurity has risen. In an effort to better analyze the vulnerabilities and impacts provided by NVD, we developed a hardware vulnerability-focused ontology I-HaC. Our proposed framework would assist in examining a set of vulnerabilities and providing details about the impact a specific vulnerability has on the CPS system. Because each vulnerability is associated with a specific CWE ID, this ontology framework will associate the vulnerability with its respective CWE ID. We believe that this framework enables us to move closer to our goal of educating learners about the various hardware security risks and how to mitigate them.

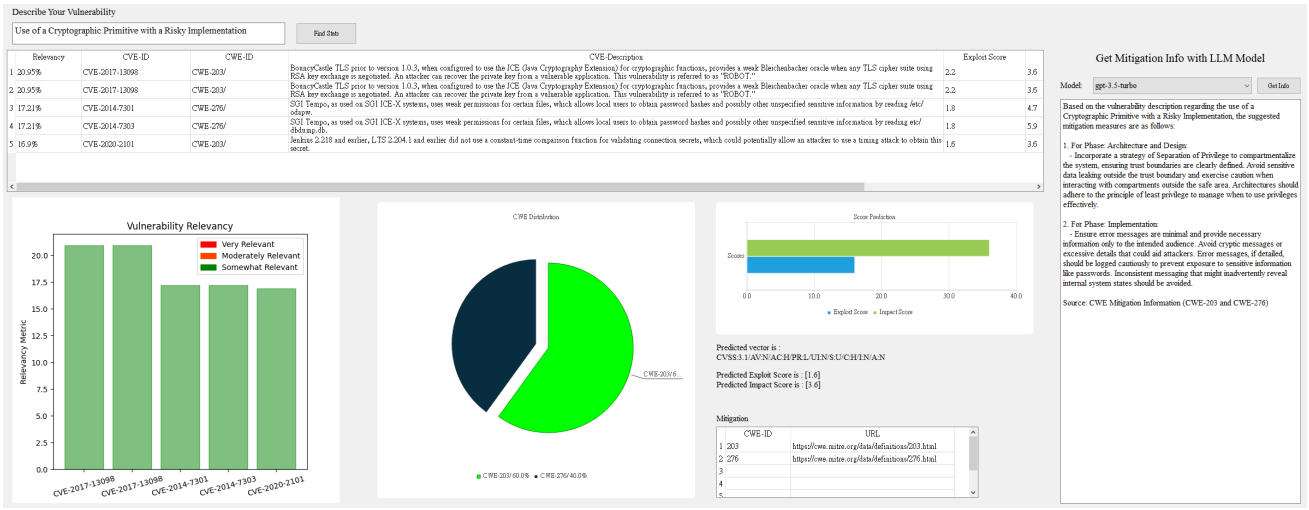

Fig. 3: Example 1 View of “Use of a Cryptographic Primitive with a Risky Implementation”.

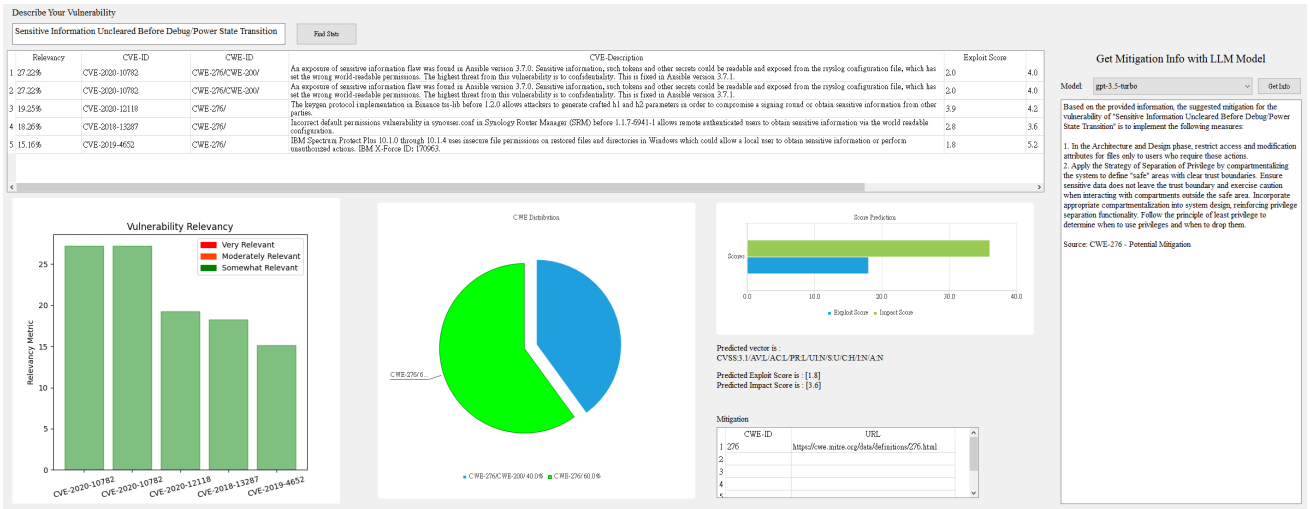

Fig. 4: Example 2 View of “Sensitive Information Uncleared Before Debug/Power State Transition” result.

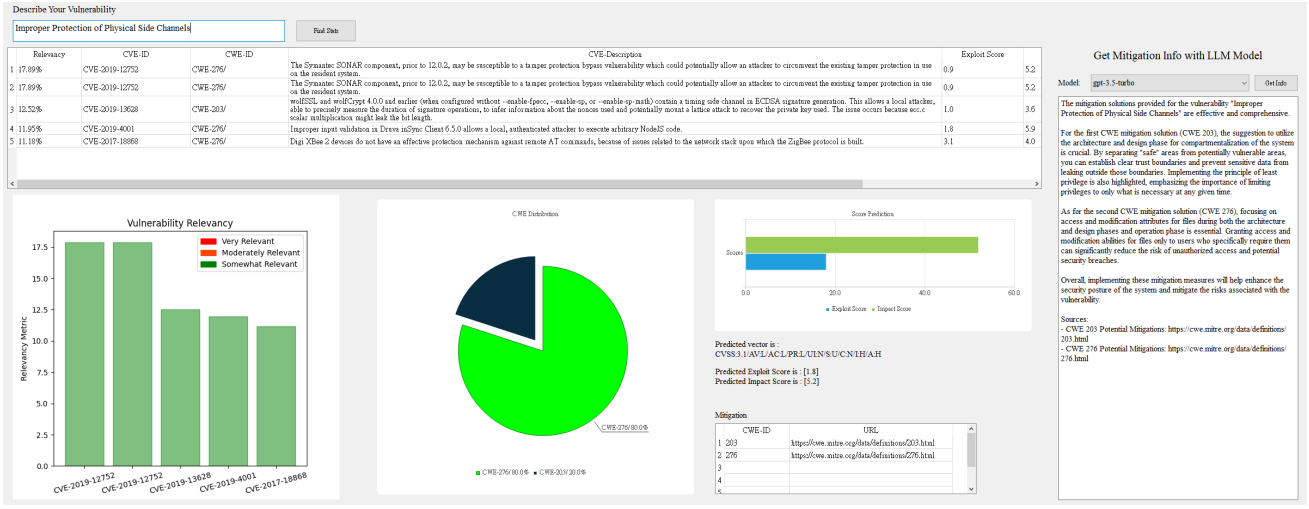

Fig. 5: Example 3 View of “Improper Protection of Physical Side Channels” result.

ACKNOWLEDGMENT

This work is supported in part by National Science Foundation (NSF) project 2335046 and the University of Arizona's Research, Innovation & Impact (RII) award for "Future Factory" and "CyberRiskMAPS".

REFERENCES

[1] W. Zhou, Y. Jia, A. Peng, Y. Zhang, and P. Liu, "The Effect of IoT New Features on Security and Privacy: New Threats, Existing Solutions, and Challenges yet to be Solved," *IEEE Internet of things Journal*, vol. 6, no. 2, pp. 1606–1616, 2018.

[2] S. Bhunia and M. Tehranipoor, *Hardware Security: a Hands-on Learning Approach*. Morgan Kaufmann, 2018.

[3] D. Lee, B. Steed, Y. Liu, and O. Ezenwoye, "Tutorial: A Lightweight Web Application for Software Vulnerability Demonstration," in *2021 IEEE Secure Development Conference (SecDev)*. IEEE, 2021, pp. 5–6.

[4] T. Srivatanakul and F. Annansingh, "Incorporating Active Learning Activities to the Design and Development of an Undergraduate Software and Web Security Course," *Journal of Computers in Education*, vol. 9, no. 1, pp. 25–50, 2022.

[5] P. Seda, J. Vykopal, V. Švábenský, and P. Čeleda, "Reinforcing Cybersecurity Hands-on Training With Adaptive Learning," in *2021 IEEE Frontiers in Education Conference (FIE)*, 2021, pp. 1–9.

[6] N. Bliss, L. A. Gordon, D. Lopresti, F. Schneider, and S. Venkatasubramanian, "A Research Ecosystem for Secure Computing," *arXiv preprint arXiv:2101.01264*, 2021.

[7] C. Bandi, S. Salehi, R. Hassan, S. Manoj P D, H. Homayoun, and S. Rafatirad, "Ontology-Driven Framework for Trend Analysis of Vulnerabilities and Impacts in IoT Hardware," in *2021 IEEE 15th International Conference on Semantic Computing (ICSC)*, 2021, pp. 211–214.

[8] R. Hassan, C. Bandi, M.-T. Tsai, S. Golchin, S. Manoj P D, S. Rafatirad, and S. Salehi, "Automated Supervised Topic Modeling Framework for Hardware Weaknesses," in *2023 IEEE 24th International Symposium on Quality Electronic Design (ISQED'23)*, 2023, p. 125–132.

[9] Y.-Z. Lin, M. Mamun, M. A. Chowdhury, S. Cai, M. Zhu, B. S. Latibari, K. I. Gubbi, N. N. Bavarsad, A. Caputo, A. Sasan, H. Homayoun, S. Rafatirad, P. Satam, and S. Salehi, "HW-V2W-Map: Hardware Vulnerability to Weakness Mapping Framework for Root Cause Analysis with GPT-assisted Mitigation Suggestion," 2023. [Online]. Available: https://arxiv.org/abs/2312.13530

[10] *NIST: National Vulnerability Database (NVD)*, retrieved: January 2023, Available at: https://nvd.nist.gov/.

[11] *MITRE: Common Weakness Exposure (CWE) Database*, retrieved: January 2023, Available at: https://cwe.mitre.org/index.html.

[12] R. S. Bell, E. Y. Vasserman, and E. C. Sayre, "A Longitudinal Study of Students in an Introductory Cybersecurity Course," in *2014 ASEE Annual Conference & Exposition*, 2014, pp. 24–61.

[13] N. Swain, "A Multi-Tier Approach to Cyber Security Education, Training, and Awareness in the Undergraduate Curriculum (CSETA)," in *2014 ASEE Annual Conference & Exposition*, 2014, pp. 24–72.

[14] M. Lukowiak, A. Meneely, S. P. Radziszowski, J. R. Vallino, and C. A. Wood, "Developing an Applied, Security-Oriented Computing Curriculum," in *2012 ASEE Annual Conference & Exposition*, 2012, pp. 25–420.

[15] B. J. Blažič, "The Cybersecurity Labour Shortage in Europe: Moving to a New Concept for Education and Training," *Technology in Society*, vol. 67, p. 101769, 2021.

[16] V. Venugopalan and C. D. Patterson, "Surveying the Hardware Trojan Threat Landscape for The Internet-of-Things," *Journal of Hardware and Systems Security*, vol. 2, no. 2, pp. 131–141, 2018.

[17] M. Beaumont, B. Hopkins, and T. Newby, "Hardware Trojans-Prevention, Detection, Countermeasures (a Literature Review)," 2011.

[18] B. S. Latibari, S. Ghimire, M. A. Chowdhury, N. Nazari, K. I. Gubbi, H. Homayoun, A. Sasan, and S. Salehi, "Automated Hardware Logic Obfuscation Framework Using GPT," in *2024 IEEE 17th Dallas Circuits and Systems Conference (DCAS)*, 2024, pp. 1–5.

[19] N. Potlapally, "Hardware Security in Practice: Challenges and Opportunities," in *2011 IEEE International Symposium on Hardware-Oriented Security and Trust*. IEEE, 2011, pp. 93–98.

[20] M. Wagner, "The Hard Truth About Hardware in Cyber-Security: It's More Important," *Network Security*, vol. 2016, no. 12, pp. 16–19, 2016.

[21] T. Aura, "Why you shouldn't study security [security education]," *IEEE security & privacy*, vol. 4, no. 3, pp. 74–76, 2006.

[22] A. Carpenter, "A Hardware Security Curriculum and its Use for Evaluation of Student Understanding of ECE Concepts," in *2018 ASEE Annual Conference & Exposition*, 2018.

[23] *CISSP: Certified Information System Security Professional*, retrieved: January 2023, Available at: https://https://www.cissp.com/.

[24] M. Taeb and H. Chi, "A Personalized Learning Framework for Software Vulnerability Detection and Education," in *2021 International Symposium on Computer Science and Intelligent Controls (ISCSIC)*. IEEE, 2021, pp. 119–126.

[25] M. Zeng and F. Zhu, "Secure Coding in Five Steps," *Journal of Cybersecurity Education, Research and Practice*, vol. 2021, no. 1, p. 5, 2021.

[26] H. Suarez and H. Kincannon, "SSETGami: Secure Software Education Through Gamification," *Proceedings on Cybersecurity Education, Research, and Practice*, 2017.

[27] A.-a. O. Affia, A. Nolte, and R. Matulevičius, "IoT Security Risk Management: A Framework and Teaching Approach," *Informatics in Education*, 2023.

[28] P. Prinetto, G. Roascio, and A. Varriale, "Hardware-based Capture-The-Flag Challenges," in *2020 IEEE East-West Design & Test Symposium (EWDTS)*. IEEE, 2020, pp. 1–8.

[29] W. A. Hill Jr, M. Fanuel, X. Yuan, J. Zhang, and S. Sajad, "A Survey of Serious Games for Cybersecurity Education and Training," in *Proceedings of the KSU Conference on Cybersecurity Education, Research & Practice*, 2020. [Online]. Available: https://digitalcommons.kennesaw.edu/ccerp/2020/Research/7/

[30] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc., 2009.

[31] F. Koushanfar and M. Potkonjak, "Hardware Security: Preparing Students for the Next Design Frontier," in *2007 IEEE International Conference on Microelectronic Systems Education (MSE'07)*. IEEE, 2007, pp. 67–68.

[32] P. Satam, C. Philipp, S. Shao, and S. Salehi, "CPS-TR: An Online Training Platform to Address Fourth Industrial Revolution Workforce Needs," in *2023 IEEE Integrated STEM Education Conference (ISEC)*, 2023, pp. 271–276.

[33] R. F. DeMara, S. Salehi, B. Chen, and R. Hartshorne, "GLASS: Group Learning At Significant Scale via WiFi-Enabled Learner Design Teams in an ECE Flipped Classroom," in *American Society for Engineering Education (ASEE) Annual Conference & Exposition*, vol. 2017-June, Columbus, OH, 2017. [Online]. Available: https://peer.asee.org/28408

[34] S. Salehi and R. F. DeMara, "Virtualized Active Learning for Undergraduate Engineering Disciplines (VALUED):A Pilot in a Large Enrollment STEM Classroom," in *2019 IEEE Frontiers in Education Conference (FIE)*, 2019, pp. 1–2.

[35] B. Chen, R. F. DeMara, S. Salehi, and R. Hartshorne, "Elevating Learner Achievement Using Formative Electronic Lab Assessments in the Engineering Laboratory: A Viable Alternative to Weekly Lab Reports," *IEEE Transactions on Education*, vol. 61, no. 1, pp. 1–10, 2018.

[36] R. F. DeMara, S. Salehi, N. Khoshavi, R. Hartshorne, and B. Chen, "Strengthening STEM Laboratory Assessment Using Student-Narrative Portfolios Interwoven with Online Evaluation," in *Proceedings of American Association for Engineering Education Southeastern Conference*, Tuscaloosa, AL, USA, 2016, pp. 13 – 15,.

[37] R. C. Yarnell, M. Hossain, R. Graterol, A. Pindoria, S. Ghimire, M. A. Chowdhury, S. Salehi, Y. Bai, and R. F. Demara, "Educational Tool-spaces for Convolutional Neural Network FPGA Design Space Exploration Using High-Level Synthesis," in *Proceedings of the Great Lakes Symposium on VLSI 2024*, ser. GLSVLSI '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 343–346. [Online]. Available: https://doi.org/10.1145/3649476.3658786

[38] R. F. DeMara, S. Salehi, and S. Muttineni, "Exam Preparation through Directed Video Blogging using Electronically-Mediated Realtime Classroom Interaction," in *American Association for Engineering Education Southeastern Conference*, Tuscaloosa, AL, USA, 2016, pp. 1–11.

[39] A. Ahmed, K. Lundqvist, C. Watterson, and N. Baghaei, "Teaching Cyber-Security for Distance Learners: A Reflective Study," in *2020 IEEE Frontiers in Education Conference (FIE)*, 2020, pp. 1–7.