Practice-Informed Pedagogy for Linux Security: Perspectives on ML-Driven Solutions for Daemon Security

Sheikh Muhammad Farjad
University of Nebraska at Omaha
Omaha, Nebraska, USA
sfarjad@unomaha.edu

Robin Gandhi
University of Nebraska at Omaha
Omaha, Nebraska, USA
rgandhi@unomaha.edu

Abstract

As machine learning (ML) continues to shape modern cybersecurity tools, computing education must evolve to help students critically engage with these technologies. One underrepresented topic in current curricula is daemon security, which involves protecting the long-running background processes common in Linux systems. These background processes (i.e., daemons) pose significant security risks in enterprise and cloud environments, yet receive little attention in operating systems or security courses. This paper reports on a semi-structured interview study with 22 academic and industry stakeholders. Using a Business Model Canvas-informed framework, we explored perspectives on ML-based approaches to daemon security and their educational implications. Participants conveyed both enthusiasm and apprehension. Although many acknowledged the potential of machine learning for real-time threat detection, concerns were also raised regarding automation, trust, and the adequacy of educational preparedness. Thematic analysis highlighted persistent gaps in instruction related to daemon awareness, system-level monitoring, and intelligent defense strategies. We propose practice-informed curriculum changes that directly address these instructional gaps, positioning this work as a concrete step toward aligning cybersecurity education with emerging system-level defense challenges.

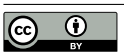
CCS Concepts

• **Social and professional topics** → **Computing education**; • **Human-centered computing** → *Empirical studies in HCI*; • **Security and privacy**;

Keywords

computing education, cybersecurity, machine learning, daemon security, linux, curriculum design

ACM Reference Format:
Sheikh Muhammad Farjad and Robin Gandhi. 2026. Practice-Informed Pedagogy for Linux Security: Perspectives on ML-Driven Solutions for Daemon Security. In *2026 ACM Southeast Conference (ACMSE 2026), April 23–25, 2026, Troy, AL, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3746467.3801511



ACMSE 2026, April 23–25, 2026, Troy, AL, USA

ACM ISBN 979-8-4007-2062-8/2026/04
https://doi.org/10.1145/3746467.3801511

1 Introduction and Motivation

Daemons are privileged background processes in Linux systems responsible for managing essential tasks such as networking, authentication, and system monitoring [29]. Their continuous execution and elevated privileges make them prime targets for cyberattacks, including privilege escalation, persistence mechanisms, and remote exploitation. Recent ransomware campaigns have exploited supply chain vulnerabilities in background services (i.e., daemons) to infiltrate critical infrastructure, emphasizing the urgent need for stronger security mechanisms [15, 26, 36].

Linux-based systems form the backbone of critical computing infrastructure across domains such as servers, cloud platforms, mobile devices, and edge environments. The Linux kernel, comprising approximately 30 million lines of code, is inherently complex and prone to security vulnerabilities [27]. The open-source nature of Linux fosters innovation and rapid development, but also introduces risks through third-party dependencies. These dependencies have increased the prevalence of software supply chain attacks, enabling malicious actors to embed vulnerabilities within widely used open-source components [2]. Although the Linux community actively monitors security issues, the decentralized and collaborative development model complicates the timely detection and prevention of such intrusions [17, 25].

Despite the central role daemons play in system operations and the growing threats targeting them, daemon-specific security concerns remain largely underrepresented in computing curricula. Educational guidelines such as those from ACM, IEEE, and NCAE-C often lack explicit coverage of daemon threats and related mitigation techniques, particularly in the context of emerging machine learning (ML)-based security solutions [19, 31, 32].

Traditional signature-based detection methods are increasingly inadequate for defending against zero-day vulnerabilities and sophisticated attacks targeting Linux systems. In contrast, recent advances in ML have shown promise for real-time anomaly detection and intrusion prevention. Yet, the application of ML to daemon security remains largely unaddressed, leaving a critical gap in both practical defenses and educational preparedness.

Addressing this gap requires a deeper understanding of how stakeholders in both academia and industry perceive daemon security, as well as their views on incorporating ML-based strategies into educational programs. Accordingly, this study investigates the following research questions:

RQ1: How do IT professionals in industry and academia perceive daemon security?
RQ2: What are their perspectives on mitigating daemon vulnerabilities through ML-based approaches?

RQ3: What are the curricular gaps pertaining to daemon security and ML-based security solutions in computing education?

To investigate these questions, we conducted semi-structured customer discovery interviews with 22 IT professionals from both industry and academia. These interviews were part of the U.S. National Science Foundation Innovation Corps (NSF I-Corps) program, an entrepreneurial training initiative that focuses on the commercialization of research projects[1]. This qualitative study, guided by the Business Model Canvas (BMC) framework, captures stakeholder perspectives, concerns, and pedagogical implications related to the integration of ML-driven solutions into daemon security practices. Our findings contribute to practice-informed educational recommendations designed to address curricular gaps and align cybersecurity education with the evolving needs of the industry.

2 Background and Related Work

Prior research on security education [14] indicates that current educational practices inadequately meet industry demands for cybersecurity professionals. This inadequacy has resulted in a skill gap, creating a shortage of qualified cybersecurity experts. The shortage is particularly pronounced in specific subfields, such as hardware reverse engineering, and researchers have linked this gap to the absence of actionable curricular guidelines [33].

A joint task force comprising ACM, the IEEE Computer Society, and AAAI recently released the latest version of the computer science curricular guidelines, CC2023 [19]. While these guidelines include reviewing major vulnerabilities in real-world operating systems as part of the CS core curriculum (OS-Protection, p. 209), they do not explicitly address Linux systems, unlike topics such as Denial of Service and Distributed Denial of Service attacks (SEC-Foundations, p. 257).

To assess how current curriculum guidelines are implemented in educational institutions, we examined the accreditation requirements of the National Centers of Academic Excellence in Cybersecurity (NCAE-C), a U.S. program for universities, managed by the National Security Agency (NSA) [23]. Compared to CC2023, the NCAE-C knowledge units (KUs) in cyber operations (CAE-CO [32]) and cyber defense (CAE-CD [31]) provide more detailed guidance on Linux and draw clearer distinctions for operating system-specific topics, such as Linux system configuration analysis (CAE-CD KUs, p. 71). Although these knowledge units are technically more rigorous, they still omit the specific topic of daemon security. They do, however, include machine learning and artificial intelligence as optional components, reflecting institutional awareness of data-driven defense mechanisms.

Simultaneously, traditional signature-based detection systems continue to fall short against zero-day exploits and polymorphic malware, prompting growing interest in ML-based anomaly detection [9, 12]. Prior research has shown that machine learning is effective for tasks such as intrusion detection and malware classification; however, its application to daemon-specific security remains in the early stages of development [3, 34]. Industry skepticism further complicates adoption, with Arp et al. [3] citing sampling bias, data snooping, and spurious correlations as common pitfalls that discourage practitioners from deploying ML-based solutions.

[1] https://www.nsf.gov/funding/initiatives/i-corps

In a computing education research context, these technical and cultural barriers underscore the importance of investigating how individuals in academia and industry perceive daemon security and the feasibility of ML-driven solutions. Moreover, they highlight the need to develop curricular materials that bridge the gap between system-level threats and practical, classroom-ready exercises.

3 Methodology

In this section, we outline and explain the interview approach used in this study, including interview design, participant recruitment, data analysis procedures, and limitations of our adapted approach.

3.1 Study Design: BMC-Incorporated Interviews

To explore perceptions of daemon security and its potential mitigation through ML-based solutions, particularly in real-world systems, we adopted the Business Model Canvas (BMC) framework [24, 28] as the foundation for our stakeholder interviews. Although the BMC was originally developed for entrepreneurial contexts, we adapted it to structure our engagement with stakeholders across both academic and industry settings. This BMC-informed approach provided two main advantages:

- **Hierarchy-Informed Thematic Emphasis.** Academic and industrial environments often involve hierarchical structures and varying levels of domain expertise. The BMC provides an organized means of capturing the intricacies of this hierarchy, which can be obscured if responses from all participants are treated as equally significant regardless of position or expertise. Not all participants are equally positioned to influence decisions or provide specialized insights. To reflect this variation, we foregrounded responses based on participants' organizational roles and levels of expertise. For example, input from senior stakeholders, such as vice presidents or innovation architects, was treated as more consequential than feedback from developers or DevOps professionals, reflecting their greater decision-making authority and broader strategic perspective. This approach allowed us to interpret and contextualize the information gathered more accurately, including themes raised by senior stakeholders even when those views were less frequent in the sample.
- **Development of a Real-World Solution.** In addition to gathering stakeholder perspectives, the BMC framework helped identify practical components such as value propositions, key activities, and user segments. These insights informed the conceptualization of a potential solution, named "DaemonSec" [10, 11], which addresses the specific challenges highlighted by participants. This solution framework also served as a bridge between real-world needs and educational applications within computing curricula.

The use of this BMC-based methodology enabled us to capture diverse stakeholder perspectives and translate them into actionable features. This process supports the development of educational content that strengthens trust, reliability, and effectiveness in daemon security, with particular emphasis on ML-based mitigation strategies.

3.2 Participants

We recruited 22 individuals from both academia and industry who work in computer science, with a preference for those in security-related roles. To ensure a diversity of perspectives, we used two recruitment strategies [18, 35]: professional outreach and direct email invitations. Through professional contacts, we engaged individuals with varying levels of experience within academic and industry environments. Snowball sampling [13] was also employed by asking participants to recommend others who met the inclusion criteria. To supplement this process, we directly contacted additional participants via email.

Participants represented a diverse demographic group. Of the 22 participants, 18 identified as male (81.8%) and 4 identified as female (18.2%). Most participants (n = 19) were based in the United States, while the rest were located in Australia, Saudi Arabia, and Pakistan. Most interviews (n = 21) were conducted synchronously via Zoom and were recorded with participant consent for annotation and analysis. One participant, due to scheduling constraints, responded asynchronously via their official email.

3.3 Interview Structure

The interviews were divided into five main sections, each designed to gather specific information about participants' knowledge, experiences, and needs related to ML-driven solutions for daemon security. The flow between sections was structured to be seamless in order to maintain participant engagement.

Before the interview began, participants were briefed on the purpose of the study and informed that their participation was voluntary. They were also given the option to skip any questions or withdraw at any time. In addition, we assured them that any identifiable or privacy-sensitive information would be removed during transcription and analysis. After obtaining consent to record the session and addressing any questions, the interviews commenced. The five sections of the interview were as follows.

(1) **IT Role.** This section explored participants' organizational roles and years of experience in their respective domains. This information was used to categorize participants within the stakeholder hierarchy.
(2) **Linux Experience.** As the study focused on daemon security in Linux environments, this section assessed participants' familiarity with Linux systems.
(3) **Security Measures.** Participants were asked about their awareness of daemons and the types of security practices they applied to both personal and work-related devices. Distinguishing between device types enabled exploration of behavioral differences between constrained and unconstrained environments.
(4) **ML-Based Approach.** This section introduced ML-based security concepts and asked participants whether they would prefer ML-based approaches over traditional methods, such as signature-based detection.
(5) **Security Automation.** Building on the previous section, this part explored participants' trust in automated security systems compared to systems involving human oversight. Questions focused on whether participants preferred full automation or favored human-in-the-loop decision-making.

At the conclusion of each interview, participants were invited to share any additional comments. All responses were recorded and incorporated into the data analysis.

3.4 Coding and Data Analysis

The interviews were conducted in three iterative rounds, with each round informed by insights from the previous one. This approach supported a dynamic process where early findings helped refine subsequent interview questions and guide the analytical focus. After each round, we reviewed the interview recordings and documented notes relevant to the evolving themes.

We used a semi-open coding approach [4] to analyze the interview data. Relying on inductive reasoning, we conducted thematic analysis [16] to allow themes to emerge from participants' responses, loosely guided by the interview structure described in Section 3.3. In addition to recurring themes, we identified new patterns that reflected gaps in current curricula, based on participants' perspectives and discussions. This process was repeated across all rounds until thematic saturation was reached, with a total of 22 participants. After completing the interviews, we reviewed and synthesized the themes identified for each participant. The results of this analysis are presented in Section 4.

3.5 Limitations

As with many interview-based studies, this work has several inherent limitations, including potential self-reporting and sampling biases, as well as concerns about generalizability. Although thematic saturation was achieved, the sample of 22 participants may not fully represent the broader population of academic and industry professionals. Most academic participants were affiliated with a single university, which may have limited the diversity of perspectives, particularly from students in programs with different curricula. Future research could address this limitation by recruiting participants from a broader range of academic institutions.

Another limitation is the reliance on self-reported perceptions, which may not accurately reflect actual behavior. This issue could be mitigated in future work by placing participants in simulated or emulated environments and observing whether their actions align with their stated preferences. In contrast to the academic sample, industry participants represented a more diverse set of roles and backgrounds. Nonetheless, the robustness of the findings could be improved by increasing both the sample size and participant diversity.

Despite these limitations, the study provides an initial understanding of how stakeholders perceive daemon security and ML-based approaches. These insights can inform the design of practical security tools and guide the development of educational interventions in computing curricula.

3.6 Ethical Considerations

We consulted the Institutional Review Board (IRB) at our university, which determined that the project does not constitute human subjects research under 45CFR46.102. This study involved voluntary semi-structured interviews with academic and industry professionals. Participants were informed of the study purpose, their right to withdraw, and their option to skip questions. Interviews were

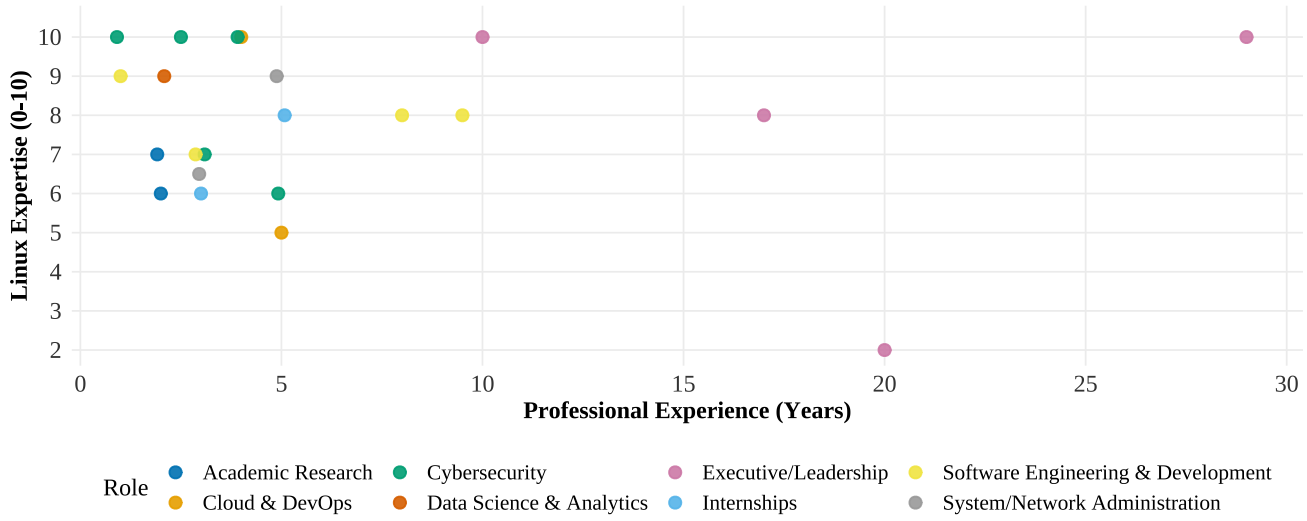


Figure 1: Participant Distribution by Professional Experience and Linux Expertise

recorded only after obtaining consent. Names and organizational affiliations were used solely for recruitment and were not included in the analysis or reporting. All identifying details were removed during transcription. We followed standard practices for confidentiality and secure data handling.

4 Results

This section reports findings from the 22 semi-structured interviews conducted in three iterative rounds, analyzed via semi-open coding and thematic analysis as described in Section 3. Quantitative summaries and qualitative themes are presented in turn.

4.1 Participant Overview & Linux Experience

To better understand patterns across professional backgrounds, interview participants (P) were categorized into eight distinct role types: Executive/Leadership, System/Network Administration, Cybersecurity, Software Engineering & Development, Cloud & DevOps, Data Science & Analytics, Academic Research, and Internships. This categorization was based on participants' job titles and responsibilities as reported during the study.

Participants' work experience ranged from 1 year (P6, P10) to 29 years (P15), with a median of 4 years. During the interviews, we asked participants to rate their familiarity (i.e., comfort level) with Linux on a scale from 1 to 10. Two participants (P3, P12) rated their familiarity as 5 or below, while the remaining participants rated it as 6 or above. The ratings ranged from 2 to 10, with a median of 8. Figure 1 provides a detailed illustration of individual participants, including their roles, self-reported Linux expertise, and professional experience.

To further assess their comfort with the Linux environment, we also asked participants to indicate their preferred user interface: Graphical User Interface (GUI) or Command Line Interface (CLI). Ten participants preferred the CLI, eight reported using both interfaces, and four preferred the GUI. Table 1 summarizes participant characteristics, pivoted by identified role. Overall, participants spanned a broad range of work experience and self-reported Linux expertise.

We observed that participants' comfort levels and interface preferences were closely aligned with their prior experience and job responsibilities. For instance, participants who primarily work with Linux systems (e.g., P1, P19, P14) preferred the command-line interface, while others (e.g., P16) favored a graphical interface depending on their specific tasks. A similar rationale applied to operating system preferences (e.g., P18).

> *"You can like quickly do most of the git operations using the GitLens GUI, but in some scenarios CLI is also much faster. So it really depends on the context." — P16*

> *"I would not prefer Linux as a personal preference because it's an open source software and I prefer having the rich GUI for my day-to-day work, but Linux is obviously used because... it's the operating system of choice for running servers." — P18*

4.2 RQ1 - Perception of Daemon Security

Limited Conceptual Familiarity. We explored participants' understanding of daemon security by asking them to identify common system entry points that malicious actors might exploit. Figure 2 presents a word cloud generated from their responses. A similar question was used to assess their familiarity with configuring background services. Surprisingly, only five participants (P5, P14, P15, P19, P20) mentioned daemons as potential entry points. Among them, only one participant (P5) explicitly used the term "daemon", while the others referred to them more generally as "services".

> *"And so you know how many daemons is probably an interesting question. It's probably… whatever the base number you'd have on a Linux… Plus, you know each*

Table 1: Participant Roles, Backgrounds, and System Interaction Preferences

Identified Role	Participant	Segment	Work Experience	Linux Expertise	Preferred UI
Executive/Leadership	P1, P5, P12, P15	Academia + Industry	10–29 years (M = 18.5)	2–10 (M = 6)	CLI + GUI
System/Network Administration	P2, P21	Academia + Industry	3–5 years (M = 4)	6.5–9 (M = 7.75)	CLI + GUI
Cybersecurity	P6, P9, P11, P19, P22	Industry	1–5 years (M = 3)	6–10 (M = 10)	CLI + GUI
Software Engineering & Development	P10, P16, P17, P18	Academia + Industry	1–9.5 years (M = 5.5)	7–9 (M = 8)	CLI + GUI
Cloud & DevOps	P3, P14	Industry	4–5 years (M = 4.5)	5–10 (M = 7.5)	CLI + GUI
Data Science & Analytics	P7	Industry	2 years (M = 2)	9 (M = 9)	CLI
Academic Research	P8, P13	Academia	2 years (M = 2)	6–7 (M = 6.5)	CLI + GUI
Internships	P4, P20	Industry	3–5 years (M = 4)	6–8 (M = 7)	CLI + GUI

Note: M denotes the median.

server is going to have maybe one to four different for the vendor… might actually be more." — P5

The lack of awareness among the remaining 17 participants, representing 77.27% of the total sample, provides empirical support for our hypothesis that daemon security is underrepresented in formal education. Despite working in IT, the participants demonstrated limited awareness of daemons and minimal experience managing background services.

Tacit Knowledge in Misconfiguration Practices. Experienced professionals (P15, P5) and those responsible for overseeing network and system security (P19) emphasized that a significant portion of their work involves addressing poorly configured services, which are a common source of security vulnerabilities. To mitigate these issues, they rely on standardized guidelines such as STIGs [8], CIS Benchmarks [5], and MITRE D3FEND [20] (P9, P15, P19).

"The number one way is supply chain attack against a product vendor. I use the VMware horizon client for work, it is still vulnerable. Other tools, Almost every Oracle tool is just a big security hole. Never run weblogic... poor configuration defaults... on downloaded packages?" — P15

"On the top of my daily vulnerability scans and the networking and the vulnerability stuff. I also deal with the hardening of the Linux OSes, so some around 20% of my time during my work is going and configuring on those backend services to the best of hardening guidelines that come from MITRE or STIGs" — P19

This reliance on vendor-provided guidelines highlights the importance of integrating such content into coursework, helping to address skill gaps while revealing the tacit knowledge often overlooked in formal education.

4.3 RQ2 - Attitude Toward ML-Based Mitigation

When briefed about the application of machine learning for daemon security, the participants expressed enthusiasm for anomaly detection capabilities, particularly in identifying subtle or zero-day threats that are difficult to detect using traditional tools. However, participants also raised several concerns. These included the interpretability of ML models, the risk of false positives and negatives, the limited availability of high-quality or representative logs for training, and the computational resources required for deployment. There was strong support for hybrid approaches that combine

Figure 2: Word Cloud for Common Entry Points

ML-driven detection with traditional signature-based methods and human oversight (P1, P5, P9, P12, P14–P16, P18–P22).

"It's unrealistic to think we could have someone in our NOC watching for and catching every security alert that pops up. If it were automated, we could catch every single one." — P1

"So an automated solution is really good for catch alls... especially when we're looking at like containers, they spin up and spin down super fast. And so how do you secure a system that is being created and deleted in a span of minutes? You can't without an automated system." — P22

These views reflect the perceived value of automation in fast-paced, large-scale environments, particularly in cloud-native contexts where infrastructure is ephemeral. Participants emphasized that manual oversight cannot scale to meet the speed and volume of modern systems, making automated solutions essential for baseline monitoring and alert generation.

"I like a combination of both, so I'm a believer of defense in depth and if you can have both, you get the things that are known with the signature based, but you get these zero days and the new security risks with solution." — P1

"...Signature-based is limited. For example, it will only match the signature which are existed till today. But on behavior-based... it will observe the behavior of unusual malicious activity." — P9

These responses support the idea that defense-in-depth remains the dominant mental model among practitioners. Signature-based detection is valued for identifying known threats, while behavior-based ML solutions are viewed as essential for detecting novel attacks. Most participants consistently endorsed combining both approaches to address the limitations of each.

Preferences for Automation vs. Human Oversight. While automation was widely appreciated for its scalability and consistency, the participants emphasized that critical decision-making should remain in human hands (P14, P21).

> *"And false negatives exist and you need a human eye on certain things, but you need the automated solution to be able to tell you what needs human eyes. But you need a human to configure the automated solution to detect what is good and what is bad." — P21*

> *"...So human intervention is required in if there is something which is beyond the automated solution." — P14*

Human oversight was considered essential for interpreting complex alerts, configuring detection parameters, and resolving ambiguous cases (P2, P15, P19). The participants also stressed the importance of minimizing alert noise and tuning detection thresholds based on operational context. And a participant (P22) even expressed privacy concerns regarding commercial ML-based security tools, underscoring the need for transparency and configurability in such systems.

4.4 RQ3 - Curricular Gaps & Learning Pathways

Ad Hoc Learning of Linux Internals. Some participants (P9, P12, P15, P19) reported acquiring Linux-related skills, such as daemon configuration and service management, through non-academic channels. This suggests a curricular blind spot in foundational operating system topics. To address this gap, computing programs should incorporate scaffolded labs that focus on the full lifecycle of daemons, from writing and hardening to implementing privilege separation.

These scaffolded labs offer valuable learning opportunities. For instance, one participant recalled their experience configuring services through such projects (P6). Similarly, another participant reported that they "enjoyed using Linux" during their graduate program (P8). These accounts highlight the importance of integrating Linux concepts into coursework through hands-on lab activities.

> *"I've set up being able to SSH into different machines. I've had projects in academia for creating websites, web servers, hosting, yeah, just hosting websites and stuff like that. " — P6*

ML as a Black Box. Although 21 out of 22 participants expressed interest in using machine learning for security purposes, the majority of them expressed concerns about data quality and model explainability. This perception of ML as a black box highlights the need for earlier integration of topics such as dataset curation and model interpretation into the curriculum. As P22 notes, transparency is a critical factor in building trust in such systems.

> *"And so for me, I don't tend to go for commercial security solutions personally, even if they are offered through work, because that removes an aspect of privacy, the best security solutions need to have the most visibility into a system." — P22*

Role-Based Awareness Disparity. Security awareness varied significantly across professional roles. Participants not directly involved in security often demonstrated limited engagement with security-related responsibilities, even within their own domains. One developer described how their organization's dedicated security team handled tasks such as patching vulnerabilities and checking for outdated software. They stated that these tasks were "solely with them" and that they (the developer themselves) "need not to worry about those" (P17).

> *"They do check if, like, you know, the software versions are updated, there is no bug in the software that can be exploited, but that was solely with them. So, as an employee, I need not to worry about those. They used to take care of that." — P17*

This response illustrates how developers often defer security responsibilities to specialized teams. While this delegation is efficient, it may limit their engagement with secure coding practices, which should be an essential part of the curriculum.

5 Discussion and Recommendations

5.1 Interpreted Gaps in Computing Education

This study highlights several important gaps in computing education, particularly related to system-level security and the responsible use of machine learning in cybersecurity contexts.

Underrepresentation of Daemon Security. Daemon processes are critical components of Linux-based systems, yet they receive little attention in current curricula (see Section 2). Most participants in this study were unfamiliar with daemon functionality or its associated security risks. This suggests that these topics are often overlooked or generalized under broader categories such as system services or side-channel attacks. As a result, students are not adequately prepared to identify or mitigate the real-world threats that target background processes.

Lack of Pedagogical Guidance for ML in Security. Although participants recognized the value of machine learning for detecting complex threats such as zero-day attacks, many expressed concerns about transparency, reliability, and misuse. In many academic programs, ML instruction tends to emphasize technical aspects of machine learning without adequately addressing how to apply or interpret these tools in practical security contexts. As a result, many students lack the skills needed to evaluate, deploy, or troubleshoot ML-based solutions in real-world environments. This highlights a broader gap at the intersection of machine learning and cybersecurity education. However, the recent initiative by NCAE-C to introduce knowledge units focused on Cyber Artificial Intelligence represents a promising step toward addressing this gap [22].

Need for Practice-Informed Pedagogy. To close these gaps, computing education must adopt approaches that integrate real-world perspectives. While the skills gap in cybersecurity has been widely acknowledged, there remains a disconnect between theoretical instruction and practical application. A curriculum that emphasizes hands-on projects, stakeholder input, and real-world systems can better equip students to meet industry needs.

Ethical and Privacy Considerations in ML-Based Security. Participants also raised ethical and privacy concerns regarding automated ML-based tools. These concerns included the lack of explainability, risks of biased decision-making, and potential overreach in system monitoring. Addressing these issues in the classroom is essential for building trust in automated systems. Integrating topics such as model interpretability, bias mitigation, and privacy-preserving techniques into security courses can help students develop a more comprehensive and responsible understanding of ML-based defense strategies.

5.2 Implications for Curriculum Design

Previous studies focusing on incorporating cybersecurity education in existing computer science curriculum emphasize the need and effectiveness of teaching cybersecurity as a crosscutting concept spanning different courses even in earlier stages of education [1, 21]. This will help in mitigating the detrimental impact of alarming practices (e.g., role-based awareness disparity) highlighted in the earlier section. Based on this, we suggest the incorporation of the topic of daemon security in different relevant courses. For example, the courses dedicated to teaching operating system concepts must provide a detailed account of the Linux ecosystem and how daemons work at the low level and how they can be exploited. Similarly, a dedicated course on cybersecurity must detail the area of daemon security without obscuring it under the umbrella term of side-channel attacks. This incorporation also needs to be accentuated through the means of standardized curricular guidelines (e.g., CC2023 [19]).

Specifically, daemon security concepts, such as privilege escalation risks, the principle of least privilege, and secure inter-process communication, can be integrated into several Knowledge Areas (KAs) described in CC2023: Operating Systems (OS), Security (SEC), Parallel and Distributed Computing (PDC), Software Engineering (SE), and Systems Fundamentals (SF). The sunflower model adopted in CC2023 provides a structural framework for this integration, grounding the base of the curriculum in the CS Core (foundational topics every graduate must know) while extending into specialized KA Core topics that allow for the specific depth required to master daemon security. This model allows institutions to tailor their curricula based on local expertise and regional needs while maintaining a unified foundational center. Mapping these concepts ensures that students understand how process isolation and protection mechanisms, which are central to daemon behavior, are enforced within the system. By treating security as a pervasive theme rather than an incidental one, the curriculum prepares graduates to design more resilient systems.

Besides daemon security, the courses focusing on machine learning should also be taught with some additional focus on its appropriate application for dealing with cybersecurity problems. This is necessary for preventing the pitfalls, which may arise from inappropriate use of ML methods for dealing with security problems as highlighted and discussed by Arp et al. [3]. These changes in curriculum design should be supplemented by theories (e.g., cognitive load theory [30]) and frameworks (e.g., ICAP framework [6]) from the domain of educational psychology to better evaluate the impact and efficacy of introduced changes [7]. These changes in curriculum design will not only equip students with cybersecurity mindsets and skills, but will also help in reducing the notorious skill gap in the cybersecurity industry. Further, it will also promote the industry-wide application and build practitioner trust for ML-driven security solutions.

6 Concluding Remarks

In this paper, we report a BMC-informed interview study with 22 academic and industry stakeholders, synthesizing themes aligned with our research questions on daemon security perceptions, ML-based mitigation, and curricular implications. This study highlights the pressing need to integrate daemon security more explicitly into computing curricula, particularly in the context of emerging ML-driven approaches to mitigating daemon-level threats. Although daemons are fundamental to the operation of Linux-based systems, most participants demonstrated limited understanding of their function and associated security risks. At the same time, while there was clear interest in applying machine learning to detect anomalous daemon behavior, participants expressed concerns regarding trust, transparency, false positives, and practical implementation. These findings reinforce that advancing ML-driven solutions for daemon security requires not only technical innovation, but also educational preparation grounded in real-world stakeholder perspectives.

Collectively, the results point to two closely connected gaps in computing education: insufficient attention to daemon-specific system security, and limited guidance on how machine learning can be responsibly and effectively applied to daemon security contexts. Addressing these gaps will require curriculum design that is both practice-informed and principled. This includes incorporating daemon security into relevant courses while introducing interpretable and applied machine learning methods tailored to system-level security use cases. By aligning educational content with practice-informed insights and the evolving demands of ML-enabled defense, computing programs can better prepare students for the evolving landscape of cybersecurity challenges.

Acknowledgments

We thank anonymous reviewers for their feedback. We also thank Gregory Bertsch (NSF I-Corps) and Gul-e-Fatima Kiani for their guidance and suggestions regarding this interview study. This work is partially supported by the U.S. Army Engineer Research and Development Center (ERDC) under the following contracts: W912HZ21C0060 and W912HZ23C0005.

References

[1] Majed Almansoori, Jessica Lam, Elias Fang, Adalbert Gerald Soosai Raj, and Rahul Chatterjee. 2023. Towards Finding the Missing Pieces to Teach Secure Programming Skills to Students. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2023)*. Association for Computing Machinery, Toronto, Canada, 973–979. https://doi.org/10.1145/3545945.3569730

[2] Florian Angermeir, Markus Voggenreiter, Fabiola Moyón, and Daniel Mendez. 2021. Enterprise-Driven Open Source Software: A Case Study on Security Automation. In *Proceedings of the 2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*. 278–287. https://doi.org/10.1109/ICSE-SEIP52600.2021.00037

[3] Daniel Arp, Erwin Quiring, Feargus Pendlebury, Alexander Warnecke, Fabio Pierazzi, Christian Wressnegger, Lorenzo Cavallaro, and Konrad Rieck. 2022. Dos and Don'ts of Machine Learning in Computer Security. In *Proceedings of the 31st USENIX Security Symposium (USENIX Security 22)*. USENIX Association, Boston, MA, 3971–3988. https://www.usenix.org/conference/usenixsecurity22/presentation/arp

[4] Melanie Birks and Jane Mills. 2015. *Grounded Theory: A Practical Guide.* Sage.

[5] Center for Internet Security. [n. d.]. *CIS Benchmarks*. https://www.cisecurity.org/cis-benchmarks

[6] Michelene T. H. Chi and Ruth Wylie. 2014. The ICAP Framework: Linking Cognitive Engagement to Active Learning Outcomes. *Educational Psychologist* 49, 4 (2014), 219–243. https://doi.org/10.1080/00461520.2014.965823

[7] James Crabb, Christopher Hundhausen, and Assefaw Gebremedhin. 2024. A Critical Review of Cybersecurity Education in the United States. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2024)*. Association for Computing Machinery, Portland, USA, 241–247. https://doi.org/10.1145/3626252.3630757

[8] Defense Information Systems Agency. [n. d.]. *Security Technical Implementation Guides (STIGs)*. https://public.cyber.mil/stigs

[9] Fatemeh Deldar and Mahdi Abadi. 2023. Deep Learning for Zero-day Malware Detection and Classification: A Survey. *ACM Comput. Surv.* 56, 2, Article 36 (Sept. 2023), 37 pages. https://doi.org/10.1145/3605775

[10] Sheikh Muhammad Farjad. 2024. DAEMONSEC: A Framework for Security Auditing of Linux Daemons. In *Student Research and Creative Activity Fair*. Omaha, USA. https://digitalcommons.unomaha.edu/srcaf/2024/PosterPresentations/65/

[11] Sheikh Muhammad Farjad. 2025. DaemonSec: Examining the Role of Machine Learning for Daemon Security in Linux Environments. *arXiv preprint arXiv:2504.08227* (2025).

[12] Sheikh Muhammad Farjad and Asad Arfeen. 2020. Cluster Analysis and Statistical Modeling: A Unified Approach for Packet Inspection. In *2020 International Conference on Cyber Warfare and Security (ICCWS)*. Islamabad, Pakistan, 1–7. https://doi.org/10.1109/ICCWS48432.2020.9292396

[13] Leo A. Goodman. 1961. Snowball Sampling. *The Annals of Mathematical Statistics* 32, 1 (March 1961), 148–170. https://doi.org/10.1214/aoms/1177705148

[14] Francois Goupil, Pavel Laskov, Irdin Pekaric, Michael Felderer, Alexander Dürr, and Frederic Thiesse. 2022. Towards Understanding the Skill Gap in Cybersecurity. In *Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education Vol. 1 (ITiCSE '22)*. Association for Computing Machinery, Dublin, Ireland, 477–483. https://doi.org/10.1145/3502718.3524807

[15] Cheng Gu, Yicheng Zhang, and Nael Abu-Ghazaleh. 2025. I Know What You Sync: Covert and Side Channel Attacks on File Systems via syncfs. In *2025 IEEE Symposium on Security and Privacy (SP)*. 3636–3652. https://doi.org/10.1109/SP61157.2025.00209

[16] Greg Guest, Kathleen MacQueen, and Emily Namey. 2025. *Applied Thematic Analysis*. SAGE Publications, Inc., Thousand Oaks, California. https://doi.org/10.4135/9781483384436

[17] Alfusainey Jallow, Michael Schilling, Michael Backes, and Sven Bugiel. 2024. Measuring the Effects of Stack Overflow Code Snippet Evolution on Open-Source Software Security. In *Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP)*. 1083–1101. https://doi.org/10.1109/SP54263.2024.00022

[18] Harjot Kaur, Sabrina Klivan, Daniel Votipka, Yasemin Acar, and Sascha Fahl. 2022. Where to Recruit for Security Development Studies: Comparing Six Software Developer Samples. In *31st USENIX Security Symposium (USENIX Security 22)*. USENIX Association, Boston, MA, 4041–4058. https://www.usenix.org/conference/usenixsecurity22/presentation/kaur

[19] Amruth N. Kumar, Rajendra K. Raj, Sherif G. Aly, Monica D. Anderson, Brett A. Becker, Richard L. Blumenthal, Eric Eaton, Susan L. Epstein, Michael Goldweber, Pankaj Jalote, Douglas Lea, Michael Oudshoorn, Marcelo Pias, Susan Reiser, Christian Servin, Rahul Simha, Titus Winters, and Qiao Xiang. 2023. *Computer Science Curricula 2023*. ACM Press and IEEE Computer Society Press and AAAI Press. https://doi.org/10.1145/3664191

[20] MITRE Corporation. [n. d.]. *MITRE D3FEND*. https://d3fend.mitre.org

[21] Azqa Nadeem. 2024. Cybersecurity as a Crosscutting Concept Across an Undergrad Computer Science Curriculum: An Experience Report. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2024)*. Association for Computing Machinery, Portland, USA, 916–922. https://doi.org/10.1145/3626252.3630821

[22] National Science Foundation and NSA National Centers of Academic Excellence in Cybersecurity. 2024. *Cyber Artificial Intelligence Knowledge Units (KUs): Strawman–Stoneman Framework*. Unclassified Technical Report Cyber_AI_KUs_Stoneman. U.S. Department of Defense, Cyber Crime Center. https://dl.dod.cyber.mil/wp-content/uploads/cae/pdf/unclass-cyber_ai_kus_stoneman.pdf

[23] National Security Agency (NSA). [n. d.]. *Centers of Academic Excellence*. https://www.nsa.gov/Academics/Centers-of-Academic-Excellence

[24] Alexander Osterwalder and Yves Pigneur. 2010. *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers*. John Wiley & Sons.

[25] Akond Rahman, Shazibul Islam Shamim, Dibyendu Brinto Bose, and Rahul Pandita. 2023. Security Misconfigurations in Open Source Kubernetes Manifests: An Empirical Study. *ACM Trans. Softw. Eng. Methodol.* 32, 4, Article 99 (May 2023), 36 pages. https://doi.org/10.1145/3579639

[26] Armin Sarabi, Ziyuan Huang, Chenlan Wang, Tai Karir, and Mingyan Liu. 2025. The Ransomware Decade: The Creation of a Fine-Grained Dataset and a Longitudinal Study. In *Proceedings of the 34th USENIX Conference on Security Symposium (SEC '25)*. USENIX Association, Seattle, USA, Article 247, 20 pages.

[27] Heyuan Shi, Shijun Chen, Runzhe Wang, Yuhan Chen, Weibo Zhang, Qiang Zhang, Yuheng Shen, Xiaohai Shi, Chao Hu, and Yu Jiang. 2024. Industry Practice of Directed Kernel Fuzzing for Open-Source Linux Distribution. In *Proceedings of the 39th IEEE/ACM International Conference on Automated Software Engineering (ASE '24)*. Association for Computing Machinery, Sacramento, USA, 2159–2169. https://doi.org/10.1145/3691620.3695278

[28] Sergio Sparviero. 2019. The Case for a Socially Oriented Business Model Canvas: The Social Enterprise Model Canvas. *Journal of Social Entrepreneurship* 10, 2 (2019), 232–251. https://doi.org/10.1080/19420676.2018.1541011

[29] William Stallings and Lawrie Brown. 2018. *Computer Security: Principles and Practice* (4th ed.). Pearson.

[30] John Sweller, Paul Ayres, and Slava Kalyuga. 2011. *Cognitive Load Theory* (1 ed.). Explorations in the Learning Sciences, Instructional Systems and Performance Technologies, Vol. 1. Springer, New York, NY. https://doi.org/10.1007/978-1-4419-8126-4

[31] U.S. Department of Defense. 2024. *Centers of Academic Excellence in Cyber Defense (CAE-CD) Knowledge Units*. Technical Report. U.S. Department of Defense. https://dl.dod.cyber.mil/wp-content/uploads/cae/pdf/unclass-cae-cd_ku.pdf

[32] U.S. Department of Defense. 2024. *Centers of Academic Excellence in Cyber Operations (CAE-CO) Knowledge Units*. Technical Report. U.S. Department of Defense. https://dl.dod.cyber.mil/wp-content/uploads/cae/pdf/unclass-cae-co_knowledge_units.pdf

[33] René Walendy, Markus Weber, Steffen Becker, Christof Paar, and Nikol Rummel. 2025. An Evidence-Based Curriculum Initiative for Hardware Reverse Engineering Education. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSETS 2025)*. Association for Computing Machinery, Pittsburgh, USA, 1176–1182. https://doi.org/10.1145/3641554.3701797

[34] Feng Wei, Hongda Li, Ziming Zhao, and Hongxin Hu. 2023. xNIDS: Explaining Deep Learning-Based Network Intrusion Detection Systems for Active Intrusion Responses. In *Proceedings of the 32nd USENIX Security Symposium (USENIX Security 23)*. USENIX Association, Anaheim, CA, 4337–4354. https://www.usenix.org/conference/usenixsecurity23/presentation/wei-feng

[35] Dominik Wermke, Noah Wöhler, Jan H. Klemmer, Marcel Fourné, Yasemin Acar, and Sascha Fahl. 2022. Committed to Trust: A Qualitative Study on Security Trust in Open Source Software Projects. In *2022 IEEE Symposium on Security and Privacy (SP)*. 1880–1896. https://doi.org/10.1109/SP46214.2022.9833686

[36] Laurie Williams, Giacomo Benedetti, Sivana Hamer, Ranindya Paramitha, Imranur Rahman, Mahzabin Tamanna, Greg Tystahl, Nusrat Zahan, Patrick Morrison, Yasemin Acar, Michel Cukier, Christian Kästner, Alexandros Kapravelos, Dominik Wermke, and William Enck. 2025. Research Directions in Software Supply Chain Security. *ACM Trans. Softw. Eng. Methodol.* 34, 5, Article 146 (May 2025), 38 pages. https://doi.org/10.1145/3714464