# Open Cybersecurity Education: Five Years of pwn.college

Connor Nelson
Arizona State University
Tempe, AZ, USA
connor.d.nelson@asu.edu

Robert Wasinger
Arizona State University
Tempe, AZ, USA
rwasinger@asu.edu

Adam Doupé
Arizona State University
Tempe, AZ, USA
doupe@asu.edu

Yan Shoshitaishvili
Arizona State University
Tempe, AZ, USA
yans@asu.edu

## Abstract

Over five years, pwn.college evolved from a demanding upper-division cybersecurity elective into a global, continuously running learning ecosystem—free and open to the world—with more than 50,000 learners having solved at least one challenge. As participation expanded beyond a single university cohort, the curriculum itself stopped functioning as a semester-bounded artifact and became a continuously lived experience, with learners engaging year-round and improvements propagating immediately. At this scale, thousands of learners effectively "playtest" the platform and its curriculum in real time, surfacing issues invisible in conventional courses and creating a feedback loop that improves the material our university students use. Voluntary global participants often persisted longer than enrolled students, became the most active mentors, and contributed significantly to refining both content and infrastructure. This ecosystem is anchored by incremental, education-first CTF challenges delivered through DOJO and supported by Twitch instruction, YouTube archives, and near-real-time peer help on Discord. Yet an always-on, openly archived curriculum also introduces tensions, including a form of "digital archaeology" in which past debugging sessions become both learning scaffolds and tempting shortcuts. Opening a CTF-based cybersecurity course to the world did not merely scale enrollment—it fundamentally changed the curriculum, how students learned, and how instructors taught.

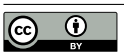
## CCS Concepts

• **Applied computing → Education**.

## Keywords

Open Cybersecurity Education, Challenge-Based Learning, Community Building

**ACM Reference Format:**
Connor Nelson, Robert Wasinger, Adam Doupé, and Yan Shoshitaishvili. 2026. Open Cybersecurity Education: Five Years of pwn.college. In *Proceedings of the 57th ACM Technical Symposium on Computer Science Education V.1 (SIGCSE TS 2026), February 18–21, 2026, St. Louis, MO, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3770762.3772636




*SIGCSE TS 2026, St. Louis, MO, USA*

ACM ISBN 979-8-4007-2256-1/2026/02
https://doi.org/10.1145/3770762.3772636

## 1 Introduction

Hands-on cybersecurity courses are notoriously difficult to scale. Effective instruction requires far more than conveying abstract concepts: students must configure fragile environments, interpret obscure program behavior, and iteratively debug complex exploits. Capture The Flag (CTF) challenges—widely regarded as a gold standard for authentic practice—intensify these demands, often requiring sustained mentorship and technical support that do not readily scale to large enrollments. As institutions seek to integrate cybersecurity into both elective and required curricula, they face a tension between providing deep, practice-oriented learning experiences and sustaining the staffing, infrastructure, and support such courses require.

Traditional challenge-based courses attempt to address these issues by introducing a small number of hands-on activities to complement lectures. In practice, however, these challenges are often large, loosely scoped, and only lightly connected to one another. Students touch each concept briefly, hopping between topics without sufficient scaffolding, deliberate practice, or opportunities for intermediate feedback. Our prior work introduced two pillars that address these limitations: an *incremental, education-first* challenge design that decomposes complex skills into tightly scoped steps [9], and a browser-based hacking environment that removes setup barriers and ensures consistent tooling across learners [8]. What remained underexplored was the third pillar needed to sustain such a system at scale: the surrounding community, instructional model, and social infrastructure that support learners before, during, and after each challenge.

This paper examines pwn.college, our attempt to reconceptualize what a cybersecurity course becomes when these three pillars—incremental challenges, an accessible hacking environment, and a wide-open learning community—are combined. Across five years, pwn.college evolved from a demanding upper-division elective at a large public R1 university into a global, continuously operating learning ecosystem, free and open to the world, with more than 50,000 learners having solved at least one challenge as of November 2025. Opening the course to the world did not merely scale enrollment—it fundamentally changed the curriculum, how students learned, and how instructors taught. Surprisingly, it also proved to be worth it.

Central to this transformation is DOJO, a browser-based Linux workspace that provides instant access to a fully configured hacking environment [8]. By eliminating setup friction and ensuring tool

parity across learners, DOJO allows students to begin experimenting within seconds. Incremental, education-first challenges [9] provide a fine-grained progression of tasks, each targeting a specific concept or skill and offering clear, gradual progression toward complex objectives. Live instruction on Twitch blends a one-to-many broadcast with many-to-many chat interaction, while YouTube archives preserve both polished conceptual explanations and spontaneous debugging sessions. Discord serves as the backbone of the community, providing asynchronous support, peer mentorship, and a persistent record of collective problem-solving.

As participation expanded, we observed network effects that dramatically reshaped the learning experience. Thousands of learners effectively "playtested" the curriculum and platform in real time, surfacing conceptual bottlenecks and usability issues that would be invisible in a smaller, single-semester course. Volunteer Global Learners—with no grade or credential at stake—quickly became some of the most active mentors, often outpacing our teaching assistants. We refer to the most dedicated among them as *HANTOs* (Helpful And Nice To Others). Their contributions extended far beyond answering questions: many submitted pull requests fixing bugs, smoothing rough edges, and adding new features. Instructors rarely encounter such organic, highly skilled mentorship networks in traditional courses; openness made them possible.

Yet openness also introduced tensions. Recorded debugging sessions and Discord discussions became a form of "digital archaeology": powerful learning scaffolds for motivated students, but tempting shortcuts for others. Leaderboards—initially motivating—became less meaningful as long-running global participation made them static. Balancing these forces became a central concern in how we design challenges, structure help, and decide what to archive or remove.

Crucially, this same ecosystem underpins our university's *required* introductory cybersecurity course. Unlike the intrinsically motivated students who self-select into upper-division electives, many novices have limited prior exposure, less interest, and little ability to opt out when the material becomes difficult. Intrinsic motivation—so abundant among voluntary Global Learners—is far more fragile here. Designing for this population required revisiting our challenge progression, adjusting scaffolding, moderating the influence of archived materials, and carefully calibrating gamification. A long-running open ecosystem can support both elective and required courses, but only when motivation is treated as a core design constraint.

Across five years of operating `pwn.college` for both voluntary global learners and required university students, several insights emerge. The combination of incremental CTF challenges, an accessible hacking environment, and **an open community does more than scale learning—it *improves* it, producing network effects that accelerate iteration, amplify mentorship, and enhance the curriculum itself.** At the same time, always-on, openly archived courses introduce novel pedagogical tensions, especially around digital archaeology and the risk of superficial solution-seeking. The challenge for educators is not whether to open their courses, but how to do so responsibly; in this paper, we explore how these benefits and tensions played out in practice within `pwn.college` and what design principles emerged as a result.

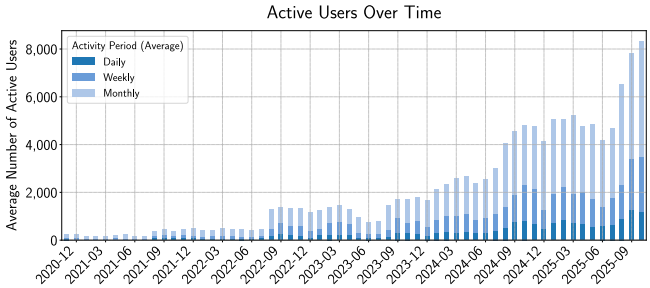



**Figure 1: Active users over time. A user is counted as active in a period if they solved at least one challenge.**

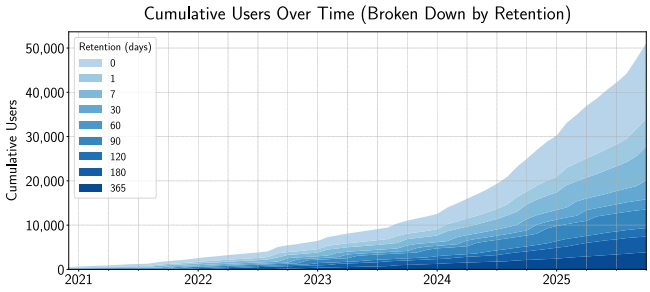



**Figure 2: Cumulative users over time, broken down by retention. Retention is defined as the time between a user's first and most recent solved challenge. As of November 2025, the plot reflects more than 50,000 learners who have solved at least one challenge.**

## 2 Background

Before discussing the design of `pwn.college` in Section 4, we first provide a high-level overview of its history, evolution, and organizational structure.

**An upper-division elective course.** The original `pwn.college` course was offered in 2018 as an upper-division elective in computer systems security at a large public R1 university and has continued to run each Fall with a typical enrollment of 100–150 students. The course teaches cybersecurity through hands-on, challenge-based learning: students progress through modules on topics such as memory corruption, shellcoding, reverse engineering, and sandboxing, each containing on the order of 15–25 CTF-style challenges. Students' entire grade is based on these challenges, with no traditional exams; lectures provide context and guidance but the core is independent experimentation, debugging, and problem solving. Many students report spending 20–30 hours per week on the material. We attribute the course's positive reception, despite this intensity, to its *elective* nature: students self-select into the course, bringing strong intrinsic motivation and a willingness to invest substantial time. Many students also view the course as their first opportunity to engage directly with core systems concepts—memory, processes, files, and binaries—that earlier courses often discuss in the abstract but rarely let students engage with them in a hands-on way, and

they consistently report finding this depth of exploration incredibly rewarding.

**Opening to the world.** In 2020, prompted by the COVID-19 pandemic and recognizing that our online, challenge-based infrastructure was inherently scalable, we opened pwn.college to the public at no cost. Fundamentally, all that was required was to give interested learners access to the existing educational challenges and workspace environment. Learners worldwide could now register, launch workspaces, and solve the same challenges as our enrolled students. As shown in Figure 2, participation quickly expanded beyond our university, and even in semesters without a local course, the platform maintained a steady baseline of daily and monthly active users. This shift marked a transition from pwn.college as a single, semester-bounded course to pwn.college as a continuously running, open-access learning ecosystem.

**An introductory required course.** In Fall 2022, building on the success of the pwn.college paradigm and the positive feedback from elective students, our university adopted this approach for a revamped introductory cybersecurity course. Unlike the elective, this course is *required* for all undergraduate computer science majors and is generally taken in the second or third year. It is offered every semester, with approximately 600–1000 students enrolled in total each term, and serves as a broad introduction to cybersecurity concepts ranging from web security to cryptography and binary exploitation. While structured similarly to the elective, this population requires a more deliberate pedagogical approach: the content is more introductory, students may be less initially interested, and the course cannot be easily dropped without affecting degree progress. These differences led us to provide more structured guidance, additional support resources, and a more gradual introduction to the material. An important outcome of this new course is that it dramatically expands the global audience able to meaningfully engage with pwn.college. By raising the floor of prerequisite knowledge for thousands of learners each year, the introductory course makes our broader content far more approachable to newcomers while also preparing many global participants to continue into the more advanced material—substantially enlarging the pipeline of long-term contributors, mentors, and expert learners.

**A global learning ecosystem.** As shown in Figures 1 and 2, pwn.college's learner population is now dominated by global participants. Since inception, more than 50,000 learners have solved at least one challenge, compared to roughly 5,000 students enrolled in our university courses over the same period—meaning that only about 10% of all learners originate from our institution. Long-term engagement also skews heavily toward global learners. More than 13,500 participants have a retention period exceeding 90 days, even though a single university semester lasts fewer than 120 days. Some portion of learners in the 90–120 day range are likely our own students who chose to continue past the required material, or who later enrolled in the upper-division elective. However, even after accounting for this, the majority of sustained engagement at every threshold (e.g., 3,572 learners retained 180–365 days and 3,833 learners retained over 365 days) necessarily comes from outside the university. This global population forms a large, stable pool of long-lasting learners—many of whom later become volunteer mentors, contributors, or module authors. Our local classes therefore operate within, and benefit from, a much larger open learning ecosystem, a relationship that underpins many of the observations in this paper.

## 3 Related Work

**CTF-Based Cybersecurity Learning.** Capture the Flag (CTF) challenges have long been used as hands-on tools for cybersecurity education. Platforms such as EDURange provide cloud-based exercises through guided scenarios [12]. Research has shown that novices can effectively engage with simplified CTF-style tasks integrated into introductory courses [5]. Large-scale events like picoCTF introduce thousands of younger learners to cybersecurity through structured puzzles and hints [3]. However, traditional CTF exercises often frustrate newcomers with steep difficulty curves and limited intermediate guidance [3, 4, 11]. Recent work on incremental, "education-first" challenge design proposes smoothing difficulty by decomposing exploits into smaller conceptual steps and embedding them in a browser-based workspace that eliminates setup barriers [8, 9]. Our work extends this literature by examining how education-first challenges embedded in a fully provisioned, browser-based workspace behave in a long-running, open-access ecosystem where thousands of learners progress asynchronously, continuously surface conceptual bottlenecks, and directly shape iterative curriculum refinement.

**Gamification and Student Motivation.** Gamified elements such as points, badges, and leaderboards are commonly used in cybersecurity education. Leaderboards can motivate competitive learners but often discourage students who see the gap to top performers as insurmountable [6]. Prior work also shows that voluntary, enjoyable CTF-style activities increase engagement and encourage continued participation beyond required tasks [5]. Our platform includes challenge points and digital "belts," but our findings indicate that these mechanisms function differently across learner populations. Intrinsically motivated global participants often pursue long-term mastery goals, while students in required courses benefit more from frequent, lightweight incentives. By comparing these populations within a single ecosystem, our work highlights the limits of global leaderboards at scale and the value of mastery-oriented signals in open learning environments.

**Online Communities and Collaborative Learning.** Online platforms have increasingly been incorporated into cybersecurity education. Twitch has been shown to support interactive teaching through real-time chat engagement [10]. Communication platforms such as Discord enable scalable, student-driven assistance and collaborative problem solving [1, 2, 7]. These studies emphasize the pedagogical value of real-time chat, peer support, and persistent public Q&A. Our setting differs in scale, duration, and openness. Instead of supporting a single course offering, our Twitch and Discord communities operate continuously across semesters and time zones, mixing enrolled students with global learners. This persistence gives rise to dynamics not captured in prior work, including sustained volunteer mentorship, large-scale community contributions, and the "digital archaeology" created when years of recorded debugging sessions and help threads become instructional resources—and potential shortcuts—for future learners.

## 4 Design

The design of `pwn.college` reflects a convergence of educational best practices and tools drawn from hacker culture. We emphasize learning-by-doing through CTF-style challenges, provide immediate access to a fully equipped hacking environment, and layer streaming and community platforms on top to create a continuously running, open course ecosystem. These design choices are central to our later observations about motivation, network effects, and digital archaeology.

**Challenge-Based Learning.** At the heart of `pwn.college` is a repository of hands-on challenges that serve as both the primary learning activity and the assessment mechanism. These challenges follow the familiar Capture The Flag (CTF) model: each task is a small puzzle or hacking exercise in which students must "capture" a secret flag by exploiting a vulnerability, solving a problem, or passing some verification test. This flag-based, auto-graded format provides immediate feedback—students know exactly when they have succeeded—and scales to thousands of learners without human grading. However, we found that traditional CTF challenges are poorly aligned with novices' needs. They typically provide only binary feedback (solved or not), with little intermediate guidance when students get stuck and no sense of what to try next. Building on prior work on "education-first" challenges [9], we redesigned problems as sequences of smaller, incremental tasks, each focused on a single concept or skill. We monitor completion rates and question patterns; when a challenge shows a steep drop-off or recurring confusion, we decompose it further or insert intermediate scaffolding tasks. This approach preserves the sense of discovery and experimentation while smoothing difficulty curves and making learning trajectories legible at scale.

**`DOJO` as a Laboratory.** A major barrier to hands-on cybersecurity work is simply getting an appropriate environment running. Rather than ask each student to configure virtual machines, install toolchains, and debug platform differences, we use `DOJO` [8], a browser-based hacking environment tightly integrated with the challenges. With a single click, learners receive a fresh Linux workspace—accessible via web terminal, in-browser editor, or `SSH`—with all dependencies pre-installed. This architecture, built on containerization technologies (e.g., Docker), provides two key advantages for our experience report. First, it dramatically lowers activation energy for new learners: they can focus on the conceptual task rather than on setup. Second, it allows instructors and contributors to iterate rapidly on challenges and tooling; updates to images or problem files propagate instantly to all learners. As we discuss later, this transparency and ease of modification are crucial preconditions for community contributions and network-effect improvements.

**Twitch as a Lecture Hall.** For synchronous instruction, we use Twitch as a virtual lecture hall instead of conventional video-based meeting platforms like Zoom. Twitch follows a "broadcast + chat" interaction model: the instructor streams video and screen-share to many viewers, while a shared text chat scrolls alongside the stream. Students can ask questions in real time without interrupting the lecture flow, and the instructor can choose when and how to respond—immediately, at planned pauses, or by synthesizing common themes. This model blends one-to-many delivery with many-to-many dialogue. Because Twitch's culture normalizes rapid, informal chat interactions, students engage more freely than in conventional classrooms or video meetings. In practice, Twitch sessions have been consistently more lively than equivalent Zoom sessions or even in-person lectures. The fast-moving chat provides a lightweight read on learner confusion and interest, and peers or teaching assistants can often answer questions directly in chat. Twitch usernames also introduce a degree of pseudo-anonymity, which we found reduced the social cost of asking questions and encouraged participation from quieter students. In later sections, we return to how this format, while highly engaging, also feeds into the long-lived archives that make digital archaeology possible.

**YouTube as a Library.** Twitch's built-in video-on-demand window (14–60 days) is too short for a course ecosystem that spans semesters and years. To preserve instructional moments, we automatically export every stream to YouTube, creating a permanent record of live debugging sessions, tangents, and chat-driven exploration. Over time, these recordings form a rich corpus of worked examples and thought processes that learners can revisit long after the original lecture. In parallel, we produce more polished, *evergreen* videos focused on the core concepts of each module. These shorter, chapter-marked recordings function as structured lectures that students can skim at higher playback speeds or consult as a conceptual refresher before returning to the hands-on challenges. Together, the live archives and evergreen videos create a layered media library that supports both just-in-time help and deeper review. As we discuss in later sections, however, the same permanence that makes these resources powerful also creates opportunities for shortcut-seeking.

**Discord as a Study Space.** To support learners between live sessions, we use Discord as our primary platform for asynchronous communication. Discord's persistent text channels, voice rooms, and lightweight threading make it well suited for ongoing help: students can post questions, share guidance, and receive feedback from peers, alumni, volunteer mentors, or instructors. Because `pwn.college` is open-access, these conversations naturally mix local students with Global Learners working on the same challenges at different times and from different time zones. In practice, this has produced near-real-time support for many active learners, with questions often answered within minutes. Compared to traditional discussion boards, where response times are often measured in hours or days, Discord's real-time chat enables iterative dialogue that sustains momentum and increases the likelihood of resolution. The public, searchable nature of channels turns one learner's difficulty into a reusable resource for others, and the low barrier to participation encourages incremental contributions such as small hints or clarifications. Later in the discussion, we examine how this community support both exemplifies positive network effects and contributes to the growing archive that underpins—and complicates—learning in a long-running open course ecosystem.

## 5 Discussion

Drawing on five years of experience with over 50,000 learners, we have the following high-level ideas that we believe are worth sharing with the community. These observations are not controlled experiments, but practice-driven insights about how openness, motivation, and archival practices interact in a long-running cybersecurity course ecosystem.

## 5.1 Network Effects in Learning Communities

pwn.college's open-access model produced an unexpectedly powerful form of mentorship and contribution. Although we initially encouraged peer support—offering extra credit to enrolled students who helped others on Discord—the most active mentors quickly became *Global Learners* with no grade incentive and no formal affiliation with our institution. Across nearly every semester we tracked participation, the single most helpful Discord member was a volunteer from the broader community. Their help was not only prolific; it was high quality, focusing on conceptual understanding rather than quick, grade-driven fixes. We refer to the most dedicated contributors as *HANTOs* (Helpful And Nice To Others), whose consistent, kind, technically deep engagement effectively extended the teaching staff at no cost.

A common concern about open educational environments is that they may attract spam, low-quality help, or toxic behavior—especially when participants are not tied to a course grade or institutional code of conduct. In practice, we found the opposite. Incidents of spam or inappropriate behavior were exceedingly rare and easily managed, while constructive participation became the norm. The visibility of questions, the persistent public record of discussions, and the presence of skilled volunteers appeared to encourage prosocial norms. Learners who invested time in helping others often developed a sense of ownership and responsibility toward the platform. This sharply reduced the moderation burden on instructors and enabled the emergence of a sustained, high-quality mentoring culture.

Openness also catalyzed continuous technical and curricular improvement. Over five years, we received 447 pull requests from 125 contributors. Many addressed subtle issues—UI papercuts, confusing challenge text, slow workspace launches—that instructors rarely notice but that meaningfully affect the learner experience. Curriculum improvements followed: top learners authored entire public modules, experimented with new ideas, and remained engaged long after completing the core material. These contributions were possible because both the workspace environment and the challenges themselves were fully transparent and easy to modify.

Scale amplified these effects. In online systems research, a *network effect* describes a setting where a product becomes more valuable as more people use it. We observed a striking pedagogical analogue: as the number of learners increased, the quality of the learning ecosystem improved. With thousands of learners progressing at different speeds, conceptual bottlenecks surfaced rapidly. If a challenge caused disproportionate frustration or if learners repeatedly misunderstood a concept, we would see a spike of Discord questions, GitHub issues, or pull requests within hours. This real-time "playtesting" of the curriculum helped us identify and repair rough edges far faster than in a traditional, semester-bounded course. When we deployed fixes—new intermediate challenges, clarified instructions, or environment improvements—they became immediately available to all learners, including those in our university courses.

Importantly, the benefits were bidirectional. While Global Learners received guidance from enrolled students and volunteers, our local students benefited substantially from mentoring others. Explaining concepts to novices deepened their own understanding, exposed misconceptions that had gone unnoticed, and built confidence in their developing expertise. A larger community created more opportunities for these mentoring interactions, effectively turning every student into a potential assistant instructor and reinforcing the mastery-oriented learning we aimed to cultivate.

Taken together, these dynamics demonstrate that openness is not merely a public service; it yields substantial *local* benefit. The volunteer community did not compete with our instructional staff—they expanded and enhanced it. Their mentoring reduced demand on teaching assistants, their contributions improved the platform for our enrolled students, and their collective feedback created a rapid iteration loop that strengthened the curriculum itself. For educators, this suggests that open-access course ecosystems can be advantageous not only for global learners, but also for the home institution—provided the environment is designed to lower the activation energy for constructive participation, visibly value contributors, and provide structured opportunities for learners to mentor each other.

## 5.2 Gamification, Motivation, and Their Limits

Gamification has long been used in cybersecurity education, and we incorporated several game-like elements into pwn.college. Early on, guided by our CTF background, we created global *Leaderboards* ranking learners by the number of challenges they solved. These initially motivated some students to progress quickly; however, the value of leaderboards diminished sharply at scale.

A key reason is that pwn.college evolves through continuous refinement rather than dramatic content releases. Most of our development effort goes into clarifying challenge text, improving scaffolding, and occasionally adding intermediate challenges—progressive changes that improve learning but do not meaningfully "reset" the ecosystem. As a result, global leaderboards quickly stabilized: early adopters accumulated large point leads, and new learners could never catch up. When new challenges did appear, Global Learners typically solved them within hours, briefly reshuffling the top of the board before it settled again. Leaderboard position therefore became a reflection of timing and historical participation rather than ongoing learning, reducing its value as a motivational signal.

One notable exception was the release of new material. Each new challenge or module triggered an immediate surge of interest, with learners developing scripts and tools to detect updates in real time. This behavior stress-tested new content and helped surface bugs or conceptual bottlenecks, but it also demonstrated how tightly some learners optimized for extrinsic indicators such as points, timestamps, or first-solve status—regardless of topic or difficulty.

In contrast, a more mastery-oriented form of gamification proved far more effective. We grouped sequences of modules into *belts*, echoing the martial-arts metaphor underlying the DOJO environment. Digital belts appear on Discord, and learners who complete all content can request a physical belt. Originally introduced as a thematic extension of the dojo concept rather than a deliberate pedagogical device, belts turned out to be deeply motivating: students in our university courses frequently continued solving challenges *after the course ended and for no credit*, solely to qualify for a belt. Where leaderboards encouraged competition, belts encouraged sustained, self-directed mastery.

However, belts are a large-grained reward, and the threshold to earn one is high. Global Learners—who opt in voluntarily—often embraced these long-term mastery goals. In contrast, students in our required introductory course, who may only be seeking course credit, often benefit more from frequent, lightweight incentives that reward consistency or incremental progress. We introduced a *streak* system to encourage regular engagement, but our experience suggests that additional, more approachable forms of recognition—such as small-cohort leaderboards, micro-achievements, or progress badges—may better support this population.

Overall, our experience shows that gamification is most effective when it reinforces intrinsic motivation rather than substitutes for it. Mastery-oriented signals such as belts pair well with open communities and voluntary learners, while static, global leaderboards lose meaning at scale and risk unintended consequences. For required-course students, who differ in motivation and available time, future work should explore finer-grained, non-competitive rewards that celebrate incremental progress and sustained effort. For educators considering gamification in large or open courses, we recommend emphasizing progress markers that celebrate mastery and improvement—rather than permanent competitive rankings—and carefully attending to how these signals interact with the motivational profile of different student populations.

## 5.3 Digital Archaeology

As `pwn.college` grew, we observed the emergence of what we call *digital archaeology*: the practice of using archived materials—past debugging sessions, Discord discussions, posted hints, or historical solutions—as a resource for navigating current challenges. This phenomenon arose naturally from the structure of a long-running, openly archived course. Every livestream on Twitch was recorded and exported to YouTube; every help request on Discord remained visible to future learners; every conceptual misstep corrected in public became part of a permanently accessible trace. Over time, these interactions accumulated into a rich corpus of community-generated instructional content.

Digital archaeology has important pedagogical benefits. Archived discussions provide concrete examples of misconceptions, thought processes, and debugging strategies, giving students insights comparable to those who participated in the original exchange. Highly motivated learners often revisit these archives, replaying earlier misconceptions and comparing them to their own—extending the reach and lifespan of instructional moments that would otherwise disappear.

However, these same archives introduce significant challenges. Because many debugging sessions culminate in a successful exploit or reveal key ideas behind a challenge, the archived record often contains partial or complete solutions. Required-course students—who often have less intrinsic motivation and less confidence in their own reasoning—were especially prone to treating these materials not as opportunities to engage with the underlying thought process, but as shortcuts that bypass it altogether. Instead of grappling with their own misconceptions or following the instructor's probing questions, they could simply scrub through a recording or scroll a thread to jump directly to the "punchline," losing the pedagogical value of the original exchange. These shortcuts produce superficial understanding and a brittle grasp of concepts, and repeated reliance can lead to a cycle of dependency on archival help rather than the development of genuine debugging skill.

This tension forced us to think carefully about what we choose to record, retain, and expose. In several cases, we removed or trimmed archived videos that were "too helpful," especially when a livestream inadvertently walked through the full solution path to a challenge intended to foster independent problem solving. We also revised our streaming and help-giving practices: instructors became more deliberate in framing hints conceptually rather than procedurally, and we set clearer community norms around avoiding spoilers in public discussion. On the design side, we experimented with templated challenge generation to produce multiple variants of certain tasks, but this approach has natural limits. Because each challenge is intended to teach a specific concept, only superficial variation is possible without introducing unfair difficulty differences or obscuring the intended learning outcome. Templating can reduce direct solution transfer in some cases, but it cannot fully prevent shortcut-seeking when archives reveal the core idea behind a challenge. In many cybersecurity tasks, the educational value lies not in executing a known sequence of steps, but in *discovering* the strategy itself—identifying the vulnerability, forming a hypothesis, experimenting creatively, and reasoning through the exploit. When a learner substitutes this exploratory process with an archived explanation, the challenge ceases to cultivate the critical thinking and self-directed problem-solving that it was designed to foster.

Digital archaeology thus presents a core design trade-off in open, long-running courses. The archival practices that make help scalable and accessible also create opportunities for unproductive shortcut-seeking, particularly for learners under extrinsic pressure. For educators considering similar open or recorded formats, we recommend treating archives as a first-class pedagogical variable: plan what to record, how long to keep it, how to label it, and how to guide learners' use of it. In our experience, intentional curation, conceptual scaffolding, and challenge designs resilient to partial solution exposure are essential for preserving the benefits of openness while mitigating its risks.

# 6 Conclusion

Over five years, `pwn.college` demonstrated that a long-running, open-access course ecosystem can meaningfully reshape cybersecurity education. When curriculum, infrastructure, and community are designed to evolve continuously—and when learners beyond the institution are invited to participate—openness generates powerful benefits: deeper engagement, rapid refinement, and sustained mentoring cultures that exceed what traditional semester-bounded classes can provide. Our experience suggests that this model is not merely scalable, but genuinely enriching for both global learners and local students. We encourage educators and researchers to take seriously this emerging paradigm of continuously operating, openly archived courses as a promising direction for expanding the reach and quality of hands-on computing education.

**Acknowledgments.** This work would not have been possible without the vibrant enthusiasm of the `pwn.college` community, and the generous support of the Department of Defense, including DARPA (HR001124C0362); thank you.

## References


[1] Kathryn Bridson, Jeffrey Atkinson, and Scott D Fleming. 2022. Delivering round-the-clock help to software engineering students using discord: An experience report. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education-Volume 1*. 759–765.

[2] Cameron Brown and Laura Cruz Castro. 2025. Coordinate: A Virtual Classroom Management Tool For Large Computer Science Courses Using Discord. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*. 165–171.

[3] Peter Chapman, Jonathan Burket, and David Brumley. 2014. PicoCTF: A Game-Based Computer Security Competition for High School Students. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)*.

[4] Kevin Chung and Julian Cohen. 2014. Learning Obstacles in the Capture The Flag Model. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)*.

[5] Zack Kaplan, Ning Zhang, and Stephen V Cole. 2022. A Capture The Flag (CTF) platform and exercises for an intro to computer security class. In *Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education Vol. 2*. 597–598.

[6] Mac Malone, Yicheng Wang, Kedrian James, Murray Anderegg, Jan Werner, and Fabian Monrose. 2021. To Gamify or Not?: On Leaderboard Effects, Student Engagement and Learning Outcomes in a Cybersecurity Intervention. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*. 1135–1141.

[7] Kenrick Mock. 2019. Experiences using discord as platform for online tutoring and building a CS community. In *Proceedings of the 50th ACM technical symposium on computer science education*. 1284–1284.

[8] Connor Nelson and Yan Shoshitaishvili. 2024. DOJO: Applied Cybersecurity Education In The Browser. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education (SIGCSE)*.

[9] Connor Nelson and Yan Shoshitaishvili. 2024. PWN The Learning Curve: Education-First CTF Challenges. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education (SIGCSE)*.

[10] Johanna Pirker, Alexander Steinmaurer, and Aleksandar Karakas. 2021. Beyond gaming: The potential of twitch for online learning and teaching. In *Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1*. 74–80.

[11] Jan Vykopal, Valdemar Švábenskỳ, and Ee-Chien Chang. 2020. Benefits and Pitfalls of Using Capture the Flag Games in University Courses. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. 752–758.

[12] Richard Weiss, Franklyn Turbak, Jens Mache, and Michael E Locasto. 2017. Cybersecurity education and assessment in EDURange. *IEEE Security & Privacy* 15, 03 (2017), 90–95.