# From Module Consumers to Module Producers

## A Mentored Summer Cohort for AI-Generation, Data-Driven Cybersecurity Education

Anonymous Author(s)

## Abstract

For students, hands-on exposure to large-scale Internet measurement data is rare because the data are too big, too sensitive, and/or too operationally awkward to fit on a laptop. We describe the design of a ten-week mentored summer hybrid program that treats course module production as both a research-mentorship experience and a curriculum pipeline. Participants are paired with measurement-community mentors, and work on national cyberinfrastructure (the National Research Platform and SDSC Expanse) with real datasets (BGP, active topology, network telescope traffic). Our goal is that each participant leaves with an open-source teaching module they built and peer-tested on real Internet measurement data, and with first-hand experience of the AI-era data-science workflow we want future students to learn. The ten-week program ends with an in-person hackathon where modules are integrated, tested, and peer-reviewed before release as open educational resources. We position the design within a central tension of CS education—integrating AI so that it *deepens* rather than *shortcuts* learning—and report here as a work-in-progress. We describe the program's motivation, the intended module catalog, the assessment infrastructure that makes the modules adoptable at scale, and invite discussion on how to maximize utility of the results for educational use.

## Keywords

networking education, cybersecurity education, Internet measurement, open educational resources, generative AI, cyberinfrastructure, mentored research

## 1 Introduction

A first course in computer networking still tends to end where the interesting part of the modern Internet begins. Traditional undergraduate networking and security courses are heavy on protocol theory (the layered model, the TCP/IP stack, some queuing math) but light on empirical contact with the messy, opaque, planet-scale system those protocols run on. Understanding how the real Internet works comes best through analyzing real, large-scale measurement data (interdomain routing, DNS, the certificate ecosystem, core traffic) that almost never make it into a classroom. Amplifying the challenge, generative AI has become a default tool for students, who will use it to read documentation, generate starter code, and explain errors whether or not an instructor sanctions it [4, 9]. These two realities result in curriculum that is simultaneously too easy (AI dispatches the toy assignments) and too shallow (students never touch the data that would make the concepts real).

The community has good reasons for this state of affairs. Real Internet measurement datasets are unwieldy to teach with, due to their size and privacy sensitivities [6]. The three common ways of sharing measurement data with a class—download-from-a-web-API, "bring code to data," and commercial cloud—each break down at the scale of a real course: APIs buckle under deadline-driven flash crowds, code review does not scale to a hundred students, and cloud bills are unpredictable for inexperienced programmers [3]. Instructors fall back on small, curated data samples that are pedagogically safe but strip away the messiness that makes measurement worth teaching.

**The curricular gap.** The motivating observation behind the ESCALATE (Engaging Scholars in Cybersecurity Analysis: A Laboratory for Teaching and Education) project [3] is concrete: a steady stream of student researchers arrives at the Center for Applied Internet Data Analysis (CAIDA) able to write code, but unable to answer simple questions about how the modern Internet is structured, secured, and financed. We built a 2023 graduate course, *Internet Data Science for Cybersecurity* [12], to address this gap.

**The infrastructure gap.** The data needed to teach the real Internet is large and sometimes sensitive. CAIDA's network telescope alone produces terabytes of packet traces per day as of 2026 [6], and multi-decade routing archives run to tens of terabytes. Under-resourced institutions, which often lack both campus compute and the capabilities to build their own, are hit hardest, precisely the populations broadening participation efforts aim to reach.

The premise of the ESCALATE project is that the now scarce resource in AI-generation networking education is not lectures or datasets; it is *adoptable modules*: self-contained, AI-aware, data-grounded teaching units that run on shared high performance cyberinfrastructure, and can be dropped into an existing course. Such modules age quickly and are time-consuming to build, and the people best equipped to build them—early-career researchers who are *currently* doing measurement—are rarely given the

time, mentorship, or infrastructure to do so. The ESCALATE summer training program inverts the usual consumer relationship: instead of recruiting students to *take* a course, it recruits them to *produce* the course, learning the material by building a teachable, gradable, openly licensed module for a networking or cybersecurity course.

Concretely, we contribute:

- A framing of open educational resource (OER) creation as a joint *research-mentorship* and *curriculum-pipeline* activity, and an argument for why a mentored cohort of practitioner-producers is the right unit for AI-generation networking education (§2, §3).
- The design of a ten-week, milestone-structured, mentored summer cohort that produces modules on real cyberinfrastructure and converges on an in-person integration hackathon (§3).
- A description of the AI-integration stance that distinguishes AI-as-scaffold from AI-as-shortcut, and how it is encoded in module and assessment design (§4).
- The technical pipeline: catalog, national compute, notebook environments, and eventually autograding—that turns a summer program's output into modules other instructors can easily adopt (§5, §6).
- An evaluation plan oriented around *adoption* (§7).

We present this program as work in progress. Because the workshop's submission deadline precedes the start of this year's program cohort, and the cohort's culminating hackathon immediately follows the A4NE workshop, this paper focuses on the *design and rationale* of the program. The workshop talk will add the first qualitative observations from a cohort that will at that time be days from completion. This timing lets the community shape the program's interpretation of its own pilot.

## 2 Background and Motivation

ESCALATE's first set of modules are seeded from CAIDA's 2023 graduate course [12], which taught Internet measurement through a data-science, security-focused lens. We studied three primary systems supporting Internet operation: interdomain routing (BGP), naming (DNS), and certificate management. This course addressed knowledge gaps from undergraduate curricula, but surfaced another limitation that ESCALATE now targets: general lack of campus infrastructure for Internet-scale data.

Traditional undergraduate networking courses present protocols and performance in the abstract, with little attention to the security vulnerabilities that those protocols carry and almost none to empirical questions that security analysts and policy makers face today. Answering those questions requires confronting interdomain routing and its lack of origin authentication, the operational realities of the DNS, the certificate authority ecosystem, and the background radiation visible to a network telescope. These systems are best understood through data, not just diagrams. Prior community reports on undergraduate computing and big-data education have repeatedly flagged the gap between what students are taught and the data-intensive reality of the field [5], but the gap has been addressed only sporadically in the cybersecurity context.

**Table 1: Why common data-sharing approaches break in a classroom. "Module pipeline" is the shared-CI model this program targets.**

| Approach | Scales to many users | Scales to big data | Cost to instructor |
|---|---|---|---|
| Web/API download | Low | Low | Low |
| Bring code to data | Low | High | Low |
| Commercial cloud | High | High | High |
| **Shared-CI module pipeline** | High | High | Low |

The deeper obstacle is logistical. Table 1 summarizes why the three prevailing data-sharing approaches do not survive contact with a real class. The lesson from our own 2023 graduate course was blunt: to let students work on their laptops, we shrank the datasets, and in doing so removed the scale and noise that make measurement instructive. Teaching Internet-scale data science requires *shared* infrastructure that brings students to the data rather than the reverse, and it requires that the cost of using that infrastructure, for setup, enrollment, and grading, be low enough to not deter an instructor at an under-resourced institution.

### 2.1 Producers, not just a platform

A catalog of modules and a shared (subsidized) compute substrate are necessary but not sufficient. Modules embody current research practice, and research practice in measurement evolves quickly. What sustains a curriculum is a *community of producers* who keep building and revising. Mentored production cohorts have precedent in adjacent areas of computing: Google Summer of Code pairs students with open-source maintainers to produce working software, and the Summer of Reproducibility pairs them with researchers to produce reproducibility artifacts [14]. ESCALATE's producers differ from both in what they produce and for whom: an assignment that an adopting educator is expected to adapt to their own students and course. The question this program asks is how to manufacture such producers deliberately—and the answer we test is a mentored summer cohort in which production is part of the pedagogy.

## 3 The ESCALATE Training Program (ETP)

The ETP program is a ten-week summer experience for up to 24 participants, spanning undergraduates, graduate students, postdocs, and experienced research collaborators. The Summer 2026 pilot accepted 20 participants. Nine weeks are remote and the final week is an in-person hackathon at a national supercomputer center, held alongside Internet measurement research experts. Participants are paired with Internet measurement mentors at a 4:1 ratio, with regular small group meetings, a weekly plenary lecture/discussion, chat room for hands-on help, and supplementary sessions that bridge gaps for participants at different skill levels. The cohort spans undergraduates through experienced research collaborators, and prior networking and measurement exposure varies accordingly.

*Production as pedagogy.* We seed this summer training program with six modules that we create; our goal is that each participant leaves with a *module* they built—a self-contained teaching unit (prerequisites, lecture material, and a data-science programming assignment) intended for adoption in a university course. The pedagogical bet is that building a teachable artifact on a dataset forces a depth of understanding that taking an assignment does not: to write a good autograder for a routing-relationship analysis, you must understand the failure modes of the analysis, which requires some understanding of the routing system. Production reframes "learning the material" as "making the material learnable for someone else." It also disciplines the work: a module that only its author can run is a failed module, which forces documentation, reproducibility, and sane minimum-resource specifications from the start.

*Milestone structure, not seat time.* The program is milestone-structured, accommodating participants who can give 10 hours a week and those who can give 40 in the same cohort: a lighter commitment moves through the same milestones on a longer personal clock. This matters for breadth of participation—it lets a full-time undergraduate and a part-time research collaborator contribute side by side—and it aligns the program's deliverable (a finished module) with its schedule (milestones toward that module) rather than with attendance.

*Mentorship and a path to adoption.* Admissions give priority to applicants whose advisor or department has a credible plan to adopt the resulting module in a networking, data science, or cybersecurity course [8]. We weighed the applicant's (and, if relevant, their advisor's) research interest against the domain of the module they would build. This approach binds each module to a first real classroom experience. A module's value is realized only when it is taught, so building the teaching relationship into the intake makes downstream classroom use the default, not the exception.

*Instructional quality, not only technical correctness.* Most participants are researchers-in-training rather than experienced instructors, so the program does not assume they arrive with pedagogical judgment. However, being students themselves lets them more readily adopt the module consumer's perspective, e.g., identifying the knowledge gaps a learner would have, and how that learner would approach the material. The instructor-authored seed modules (§6) serve as worked examples, so participants see examples of existing pedagogy before designing their own. A common module template encodes structural expectations a novice author would otherwise have to discover: stated learning objectives, a background introduction, dataset guides with access instructions, a task-structured notebook assignment, and lecture slides. Mentors review module drafts for teachability as well as correctness: are the objectives assessable, and can a newcomer run the instructions? The hackathon's structured peer testing (§7) puts each module in the hands of cohort members acting as proxy students before any classroom sees it.

### 3.1 Cyberinfrastructure Training

The cohort works on NRP/Nautilus, and SDSC Expanse [8], using large-scale Internet measurement datasets. Participants learn to stand up the JupyterHub-on-NRP workflow, access community data sets via the Open Science Data Federation (OSDF) [1], and stage and run batch jobs on SDSC's high performace computing system Expanse.

*The hackathon as integration crucible.* The on-site week is where student-built modules are peer-tested, and revised under the eye of mentors. It functions as integration testing, peer review, and onboarding rehearsal at once: if a fellow participant cannot run your module on the cluster from your instructions, neither can a future instructor.

*Open licensing, maintenance, and a feedback loop.* Modules and tooling are released open-source for free academic use, and the hosting project commits to maintaining them after the cohort disperses, closing the loop that usually dooms summer-built educational artifacts to bit-rot. Module developers, often students, move on, so revision cannot depend on the original author. Each module lives in a public repository. Instructors who adopt one can report problems, confusions, and local adaptations through its issue tracker. Later cohorts can take up adopted modules, informed by that classroom feedback, alongside proposals for new ones. Not every module will warrant this care: relevance shifts quickly in measurement, and retiring a stale module is maintenance too.

## 4 Integrating AI So It Deepens Learning

The workshop's framing question—how to integrate AI so that it *deepens* rather than *shortcuts* learning—is, for a data-driven measurement curriculum, unusually tractable, because the hard part of measurement is not writing code but *deciding whether an answer is trustworthy*. That is where today's generative models are weakest and where the learning objective lives. We encode three commitments into module design.

*AI as a navigation aid, not an answer oracle.* The biggest barrier the 2023 course targeted was navigational: the sheer complexity of which dataset, which field, which tool answers a given question [3, 12]. A generative assistant legitimately helps here, surfacing the right dataset, explaining a cryptic field, scaffolding a first query, essentially lowering the on-ramp without doing the thinking. ESCALATE's *recipes* (documented, runnable example code for a dataset [3]) are the human-authored complement: together they get a novice unstuck without handing over the conclusion.

*Assessment that probes what AI cannot do well.* Modules are designed so that the graded objective is interpretation under uncertainty: *is this spike an attack or a measurement artifact? does this inference hold given how the data were collected? what would falsify it?* An autograder can check that a student correctly computed a routing relationship according to a known algorithm; the assignment's writing and analysis components ask the student to defend the *validity* of the result against the dataset's known biases—a task where an unguided model tends to produce confident, wrong, generic answers. The aim is not to police AI use but to make the assessment robust to it. This turns the AI generation's ML fluency into an asset—students apply real techniques—while making domain reasoning what earns the grade. ETP participants build modules by performing this workflow themselves: a module author has to decide, concretely, where an AI assistant belongs in their own assignment and where it must be designed out.

*ML/AI on real data as first-class content.* Beyond AI as a study aid, modules could treat ML/AI as the object of study on real measurement data—for example, applying text-based learning to infer router ownership and geolocation across millions of interfaces, or distinguishing security events from background radiation in telescope traffic.

*Tooling, access, and guardrails.* Student feedback in the weekly surveys suggested it would be valuable to add mid-program training in agentic, repository-aware AI coding workflows, rather than leaving participants to default to general-purpose chat interfaces. Data governance is handled by construction. The NRP treats staged data as public, so the 2026 pilot used only datasets that are already public, in some cases older releases of data whose recent versions carry restrictions. No privacy-sensitive data is therefore in scope for AI tools.

## 5 Infrastructure and the Adoption Pipeline

For a module to be adoptable, the friction an instructor faces between "interesting idea" and "running in my class next term" must be near zero. The program builds on national cyberinfrastructure to remove that friction.

We prototype a sample pipeline that unifies the structure of modules developed throughout the program. Fig. 1 illustrates the roles of stakeholders and the processes and infrastructure involved. The pipeline integrates resources from the National Research Platform (NRP)—an NSF- funded, geographically distributed system with petabyte-scale storage and thousands of CPU/GPU resources—directly addressing the scaling failure identified in Table 1.

**Researchers** often disseminate data using different methods and infrastructure. For example, CAIDA publishes data through its own web and object storage servers. To prevent flash-crowd effects from overwhelming researchers' infrastructure, learning module creators and faculty can collaborate with researchers to stage data into large-scale national storage, such as Ceph S3 object storage on the NRP [16] and the Open Science Data Federation (OSDF) [1]. These systems interconnect via high-speed networks, mitigating data-transfer bottlenecks.

**Learning module creators** follow our module template to create GitHub repositories containing instructions, deployment code, optional autograders, and completed Jupyter Notebooks. Creators may also publish a student-facing version as a GitHub Classroom template repository, allowing educators to adopt modules directly into their existing GitHub Classrooms.[1] Each notebook provides a reproducible environment [13] preloaded with the libraries and data access the assignment requires, along with a *recipe*: well-documented example code that demystifies a dataset's fields and encodes best practices for working with it.

**Educators** can discover modules through CAIDA's resource catalog. They can fork a module into their GitHub account to use it as is or customize it before adoption. Three operational costs often deter instructor adoption: manually enrolling students on the cyberinfrastructure, configuring

[1] GitHub announced in May 2026 that it is retiring GitHub Classroom, with existing classrooms decommissioned on August 28, 2026 [11]. We are evaluating replacement platforms, including GitHub's named partner Codio and the open-source Classroom 50, for the template-repository and roster-based enrollment workflow described here.

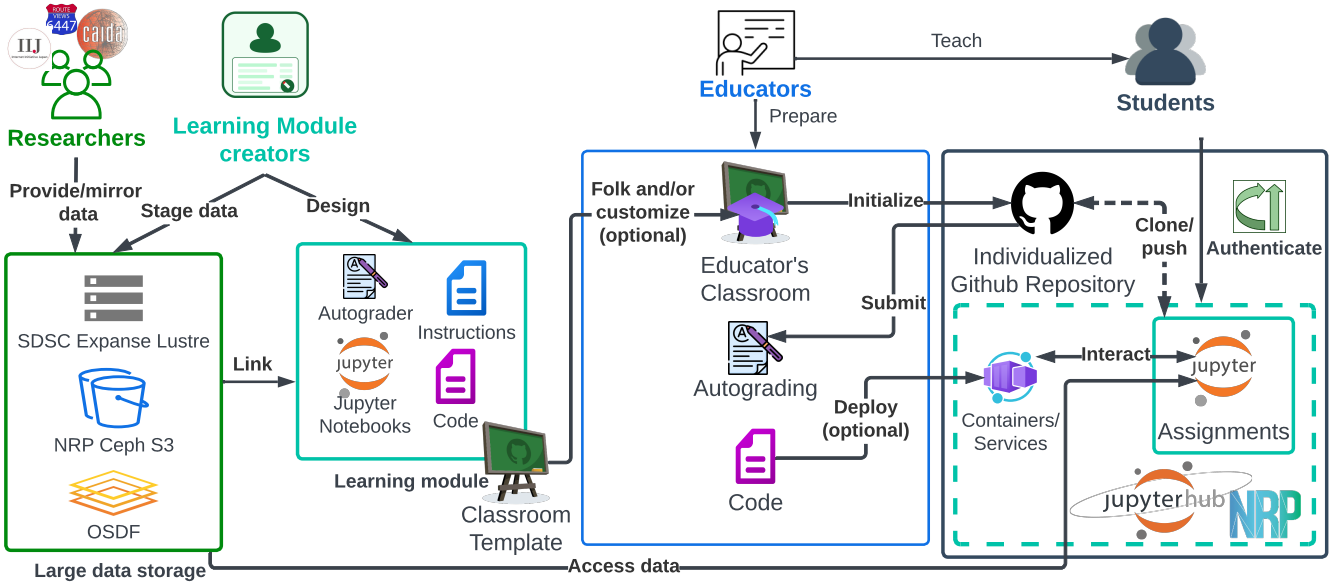


**Figure 1: Overview of infrastructure and adoption pipeline.**

complex software environments, and hand-grading data-analysis assignments. The pipeline addresses each of these. It provides batch enrollment instructions to streamline student onboarding. Our goal is to have modules ship with an autograder built on standard notebook-grading tooling [18], enabling instructors and teaching assistants to smoothly adopt the module.

**Students** securely access the NRP using their university's credential via CILogon, an authentication solution widely supported across U.S. institutions. When an instructor sets up a GitHub Classroom, students who accepted the assignments automatically clone the starter code into individualized student repositories. Students execute notebooks on NRP's JupyterHub, complete assignments using large-scale datasets, and submit their work to instructors for grading by pushing to their personal repositories.

In this summer program, we will collect input from the participants and refine the pipeline as needed to streamline the generation and adoption of modules.

## 6 The Seed Module Catalog

The pilot seeds the catalog with modules drawn from, and scaled up from, the prior graduate course [12]. Each module pairs a security concept with a real dataset and an analysis task (Table 2). Bridging modules (Internet fundamentals via hands-on tools such as `traceroute` and packet capture; and a data-science-on-CI primer covering notebooks, common Python libraries, and the cluster environment) lower the entry bar for students without a networking, cybersecurity, data science, or cyberinfrastructure background.

Although some modules draw on skills learned in previous modules, each module is designed for independent adoption: self-contained, with documented prerequisites, sized to drop into an existing course. Several modules integrate multiple datasets (e.g., correlating certificate anomalies with DNS, or joining topology with routing), because integration is where measurement gets hard and where a knowledge-graph view of Internet data [10] pays off. The bridging modules exist so a single module can travel without the rest of the sequence, which matters for under-resourced institutions, the least likely to offer a dedicated Internet-measurement course.

## 7 Evaluation Plan

**What success looks like.** Following the ESCALATE evaluation framework [3], our metrics are: (1) *Production*: how many classroom-ready modules does the cohort complete, and do they pass peer testing? (2) *Adoption*: how many have a committed instructor and a scheduled course? (3) *Learning*: do participants report increased confidence and competence with CI and with measurement-data reasoning? (4) *Inclusion*: did the milestone structure broaden participation?

**Evaluation Instruments** include pre/post participant surveys; mentor logs from one-on-ones; structured peer testing of each module, and hackathon artifacts. Longer term, the ESCALATE project will track downstream classroom use and research products enabled by the materials.

**Table 2: Seed modules: a security concept, a real dataset, and a data-analysis task.**

| Module | Data & task |
|---|---|
| Internet addressing | IP-to-organization mapping; address-space and AS basics |
| IP spoofing | Multiple; track source-validation deployment by region/network |
| Interdomain routing | Global routing data; inference of business relationships |
| Routing security threats | Labeled ASN topology; reason about security risks [15] |
| Active topology | router-level topology; ML to infer router ownership/geolocation [7] |
| DNS attack surface | active DNS [17]; explore vulnerabilities |
| TLS/CA ecosystem | CT logs; detect suspicious issuance |
| Internet Background radiation (IBR) | UCSD Telescope traffic; separate scanning/attacks from noise [6] |

*Status and early observations.* : At this writing the pilot cohort is in week 4 of 10; the workshop falls the day before its hackathon begins. The early weeks already show the intended feedback loops. Weekly participant surveys drove an expansion to two lectures per week, with the second slot going to topics the cohort votes onto the agenda (packet capture, AI tooling, assignment-specific deep dives, agentic-AI training. To counter the anonymity of a 20-person remote cohort, we partitioned participants into six groups of about four (some joined more than one), each with its own chat channel and a short regular meeting of 15–30 minutes. By week six these groups turn to choosing each participant's module project ahead of the hackathon. One early signal on documentation quality: attendance at the weekly office hours (§3) has been low, and participants report working through infrastructure setup from the seed modules' own instructions instead—a tentative but encouraging sign that we are documenting modules well enough.

## 8 Discussion and Open Questions

*Does production scale as pedagogy?* The central hypothesis—that building a gradable module teaches its subject better than completing an assignment—is unproven, and the cohort is small. A fair test must separate the effect of *production* from the effects of mentorship and national-CI access that come bundled with it. The hackathon's "can a peer run your module" check will be a near-term signal of utility and classroom-readiness of the modules.

*Where is the line between scaffold and shortcut?* Our approach aspires to make assessment robust to generative AI by moving the graded objective to interpretation and validity. This is an empirical claim about a moving target; as models improve at hedged, data-aware reasoning, the line will move, and modules will need to move with it. We see this as an argument *for* a producer community rather than a static module repository.

*Generalizability.* The model leans on assets not every program has: a measurement-research community to supply mentors and datasets, and access to national CI. The transferable core is narrower and more portable: pair learners with practitioners, make the deliverable an adoptable artifact, and remove instructor friction with shared infrastructure and autograding. Whether that core generalizes to other "Networking+X" subfields is an open and, we hope, discussable question.

*Is the barrier really materials?* A fair challenge to the premise is that the scarce resource is less the availability of modules than the instructor time, expertise, infrastructure, and curricular flexibility needed to adapt and use them. Autograding, batch enrollment, shared CI, and standalone module design (§6) target that friction directly. The adoption metrics will test whether removing it is *sufficient*, across course structures, institutional contexts, and levels of instructor expertise.

*Sustainability.* The design explicitly courts under-resourced institutions—through shared CI that needs no local cluster, automated enrollment and grading, and travel support for the hackathon. Whether the modules are adopted *there*, and whether the maintenance commitment survives past the 2-year funded project, are open questions.

## 9 Conclusion

The AI generation of networking and security students needs more than new lectures about AI; it needs a sustainable way to produce and share course materials that are AI-aware, grounded in real measurement data, runnable on shared infrastructure, and gradable at scale. We have developed and described a mentored summer cohort that makes module production the pedagogy and an integration hackathon the initial proving ground for this approach. The pilot runs as this workshop convenes. We offer this description of its design for community review in the window before its first results arrive.

## 10 Acknowlegments

This material is based on work sponsored by NSF grant OAC-2519416. The views expressed are those of the authors and do not necessarily represent the official policies or endorsements, either expressed or implied, of NSF.

## References


[1] 2026. Open Science Data Federation. https://osg-htc.org/services/osdf.

[2] ACCESS Support. 2025. Knowledge Base: Resources tagged "cybersecurity". https://support.access-ci.org/knowledge-base/resources.

[3] Anonymized for review. 2025. ESCALATE: Engaging Scholars in Cybersecurity Analysis—A Laboratory for Teaching and Education (Project). NSF award; citation anonymized for submission.

[4] Brett A. Becker, Paul Denny, James Finnie-Ansley, Andrew Luxton-Reilly, James Prather, and Eddie Antonio Santos. 2023. Programming Is Hard—Or at Least It Used to Be: Educational Opportunities and Challenges of AI Code Generation. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education (SIGCSE)*.

[5] L. Birnbaum, S. Hambrusch, and C. Lewis. 2020. Report on the CUE.NEXT Workshops.

[6] CAIDA. 2018. The UCSD Network Telescope. https://www.caida.org/projects/network_telescope/. Accessed 2026-05-31.

[7] CAIDA. 2024. Macroscopic Internet Topology Data Kit (ITDK). https://www.caida.org/data/active/internet-topology-data-kit/.

[8] CAIDA. 2026. ESCALATE Training Program (ETP), Summer 2026 pilot. https://www.caida.org/workshops/training/2606/.

[9] Paul Denny, James Prather, Brett A. Becker, James Finnie-Ansley, Arto Hellas, Juho Leinonen, Andrew Luxton-Reilly, Brent N. Reeves, Eddie Antonio Santos, and Sami Sarsa. 2024. Computing Education in the Era of Generative AI. *Commun. ACM* 67, 2 (2024).

[10] Romain Fontugne, Malte Tashiro, Raffaele Sommese, Mattijs Jonker, Zachary S. Bischof, and Emile Aben. 2024. The Wisdom of the Measurement Crowd: Building the Internet Yellow Pages, a Knowledge Graph for the Internet. In *Proceedings of the ACM Internet Measurement Conference (IMC)*.

[11] GitHub. 2026. Important announcement for educators: GitHub Classroom functionality will be transitioning to partners. https://github.com/orgs/community/discussions/196615. Accessed 2026-07-18.

[12] kc claffy. 2023. CSE 291(e): Internet Data Science for Cybersecurity (Syllabus). https://cseweb.ucsd.edu/classes/wi23/cse291-e/syllabus.html.

[13] Thomas Kluyver et al. 2016. Jupyter Notebooks—a publishing format for reproducible computational workflows. In *Positioning and Power in Academic Publishing*. IOS Press.

[14] Stephanie Lieggi, Fraida Fund, Kate Keahey, and Marc Richardson. 2025. Summer of Reproducibility: Building Global Capacity for Practical Reproducibility through Hands-On Mentorship. In *Proceedings of the 3rd ACM Conference on Reproducibility and Replicability (ACM REP)*. https://doi.org/10.1145/3736731.3746149

[15] MANRS. 2024. Mutually Agreed Norms for Routing Security (MANRS). https://manrs.org/. Accessed 2026-05-31.

[16] National Research Platform. 2024. Nautilus. https://nationalresearchplatform.org/nautilus/. Accessed 2026-05-31.

[17] OpenINTEL. 2024. OpenINTEL: Active DNS Measurement. https://openintel.nl.

[18] Project Jupyter. 2024. nbgrader Documentation. https://nbgrader.readthedocs.io/.