CRUSHING THE CANNED CYBERSECURITY COURSE
HOW TODAY'S SCRIPTED CURRICULUM COULD INFLUENCE TOMORROW'S INCIDENT RESPONSE

By Anthony W. Rose
Miami University

In this paper, I argue that scripted curricula yield very different outcomes than unscripted ones. That claim is not merely a matter of pedagogical style, but a question about the kinds of habits students develop when they are taught to move through learning environments that privilege compliance, repetition, and predefined outcomes over inquiry, interpretation, and collaborative meaning-making. This argument is framed around my own experiences as a student and as an educator teaching cybersecurity online at a public university. I argue that, in fields where ambiguity is unavoidable, those habits do matter. They may shape not only how students learn but also how they later respond when their responses to experiences in the real world exceed the reach of the script.

Cybersecurity is a profession often described through frameworks, standards, best practices, playbooks, and checklists. These tools are valuable as they offer structure, shared language, and a means of coordinating response. Yet cybersecurity is also a domain defined by incomplete information, rapidly shifting conditions, and problems that do not always announce themselves in familiar ways. A zero-day exploit, by its very definition, emerges before documentation, before standardization, before the polished language of the certification blueprint. If educational experiences train students primarily to recognize and repeat pre-modeled patterns, then the professional consequences of that training deserve closer scrutiny.

The literature on scripted curriculum provides a strong foundation for that scrutiny. Fitz and Nikolaidis (2020) argue that scripted curriculum is both a technical intervention and a democratic problem, because it narrows the intellectual agency of teachers and students. Timberlake et al. (2017) show how the appeal of scripted curricula often rests in simplicity and promises of equity but that such simplicity flattens learners' differences and reduces responsiveness. Powell et al. (2017) show that teachers working inside scripted programs often experience significant tensions around power and agency. In response, Vaughn et al. (2022) argue for adaptive teaching as a challenge to rigid curricular scripts, while Valdez (2020) illustrates how inquiry can interrupt the closures that scripted approaches impose. Across these works, a common theme emerges: scripts may produce efficiency, consistency, and measurable outcomes, but they also risk constraining the very forms of judgment and dialogue that complex learning requires.

That concern becomes sharper when illuminated by *currere*. Pinar (2019) and Pinar et al. (2006) invite us to understand curriculum not merely as a sequence of objectives and materials but as lived experience. Poetter (2024) describes curriculum fragments as a means of tracing life processes and educational experience through pivotal moments. The question is not simply what curriculum contains but what it *does* to the person moving through it. Through a series of regressive and progressive fragments, I explore how educational conditioning can preclude alternate realities. Beginning with learning scripted behavior in first grade, diverting momentarily to a non-scripted nightmare in the gifted and talented program, and back to the script in an undergraduate business applications course, and finally projecting forward into cybersecurity

teaching and incident response, I argue that scripted curricula can normalize a relationship to knowledge in which students learn to search for permission, procedure, and pre-validation before acting. By contrast, unscripted learning experiences can cultivate a greater tolerance for ambiguity, shared inquiry, and adaptive judgment.

REGRESSIVE FRAGMENT ONE: A PEDAGOGY OF FEAR

I was incredibly happy to begin school. My brother, who was almost two years older, got on the big yellow bus every morning to go to school, and I wanted desperately to go, too. I was already four years old when he started, but I would cry every day because I wanted to ride that bus and enter that world. School, from a distance, looked magical. It looked important. It looked like belonging. Eighty acres of woods is great fun if there is someone to play with but lonely when you are on your own. I was finally able to start first grade the very next year.

On the first day of school, Mrs. Johnson had been paged over the intercom and asked to come to the office. She told us to sit quietly and left. After a few minutes, the silence was overtaken by whispering, and the girl behind me said something that I had not heard. I turned around. Just then, Mrs. Johnson walked in. She called out several names, including mine, as she pulled a giant board from her desk. It was dark red, shiny wood with dozens of holes and a carved handle. As she held it, she casually mentioned that she "often lifted kids off the floor with this paddle," but she was going to "take it easy on us this time." I do not remember whether it really hurt, but I remember the fear it instilled.

Poetter's (2024) understanding of curriculum fragments helps illuminate why such a moment remains educationally significant 50 years later. The fragment is not important because it summarizes an entire schooling history, but because it condenses a relation to authority, uncertainty, and self-presentation that can echo across later experiences (Poetter, 2024). Pinar's (2019)*currere* asks us to return to such moments, not for nostalgia, but to understand how educational experience is *lived and carried forward*. In my case, the first-grade fragment marks an early conditioning toward caution. Seen through the lens of the literature on scripted curriculum, this early moment also reveals how behavior and curriculum are intertwined. Golann's (2021) analysis of scripting in a no-excuses charter school shows how educational environments can train not just academic performance but bodily comportment, speech, and acceptable forms of participation. The student is not simply learning content; the student is learning how to inhabit a script (Golann, 2021). My first-grade experience was not a formal, scripted curriculum in the contemporary, packaged sense, but it functioned similarly in terms of habit formation. It taught me to associate learning with surveillance, correction, and preemption of error. Looking back, I can see that the incident did more than frighten me. It taught me a relationship to authority that remained surprisingly durable. It was a curricular lesson about risk.

REGRESSIVE FRAGMENT TWO: WHAT AM I SUPPOSED TO DO?

After this very bumpy start, I quickly acclimated to the task at hand. I got good at following the script. Head down, do the work, and don't ever talk to anyone. I loved school and learning. Unlike Dr. Poetter, I even loved SRA (Poetter, 2026). My parents were part of a strict religious sect that didn't approve of watching TV, so no matter how boring others might have perceived the

content, I read everything I could get my hands on. In second grade, I somehow scored in the 97th percentile on a test that asked me to determine that "A is to B as B is to C." I didn't know that at the time because my parents didn't share it with me to keep me humble.

The unfortunate result of this "A is to C" situation was that I was sent to the gifted-and-talented program. Six or seven of us were marched down to the library with, as I recall, a large box of crap—rubber bands, straws, pipe cleaners, paper, etc.—and no further instructions. Crap was my favorite word until I left home, rejected the strict religious sect, and allowed myself to say "shit" on such occasions. The principal's son, Stephen, had been anointed to lead the group in her absence by Ms. Shackelford, the gifted-and-talented teacher who moved among the county's elementary schools. I wanted to ace this task like I had everything else up to this point, but there wasn't anyone to tell me what to do. This should have been an opportunity for me to create something that showcased my individual creativity, but I didn't have a frame of reference for self-direction at school, and Stephen was no help. The possibilities of what I could have done during this time were obscured by my fear of not producing the correct answer. For someone who had so completely mastered following the script, this was devastating. I hated this time every week. The longer this went on, the more I hated school. I ended up becoming depressed and even regressing to bedwetting. Luckily, the torture only lasted for a quarter before my grades dropped just enough for me to escape this ambiguous hellhole. I happily returned to the regular classroom and once again thrived by mastering the script. I loved school once again. How could I have been so unprepared for this experience? Because the concept of real learning was alien to me, even after a year and a half of successfully navigating school.

REGRESSIVE FRAGMENT THREE: PUNCTUAL PERFECTION

Years later, as an undergraduate at Eastern Kentucky University, I found myself in a Computer Information Systems course in the business school. The class covered basic business applications, including WordPerfect, Lotus 1-2-3, and R-base. If you are younger, think Word, Excel, and Access. The course was practical, workforce-oriented, and clearly designed around procedural mastery. By then, I had developed the habit of sitting in the front row. Part of that came from learning in the third grade that I was nearsighted. If I squinted and held my eyelid just right, what was on the board would pop into focus. I had asked teachers if I could sit in the front, and no one questioned it. Even after getting glasses so that I could pass my driver's test, front-row seating remained a habit. Plus, sitting in the front row usually got me brownie points, which I loved.

On the first day of this class, though, I decided to sit in the back. I think I might have wanted to feel like I was doing something slightly rebellious, "living on the edge" as a college freshman. The classroom was set up with rows of computers all facing the front. The professor sat at a workstation in the middle row facing the back of the room. She brought out a small monochromatic projector panel, set it atop an overhead projector, and aimed it at the front wall. The amber text on the black background was barely visible in the fluorescent light of the brightly lit classroom in the Bert T. Combs building. Almost immediately, I was rethinking my decision to sit in the back.

We had also been told that we would need to keep the same seat the entire semester, and here I was writing my name in the tiny block the professor would use to take attendance for the rest of the term. Two cute girls scurried in at the last second and took the remaining seats in the back row. I could feel my face getting red, my default behavior in that era of my life when

confronted with members of the opposite gender, especially if they were attractive. I busied myself by staring intently at the syllabus as the professor discussed the assignments and projects due that semester. We were lectured on the importance of being on time with all assignments and ensuring that we did everything exactly as listed. Failure to do so, we were told, would result in significant penalties. We were being prepared for the workforce, and we needed to get used to "punctual perfection." Her phrase.

That phrase has stayed with me: punctual perfection. It captured the ethos of the course. There was a proper way to proceed, and the goal was to replicate it efficiently and accurately. The professor pointed us to Chapter 1 in a three-inch-thick spiral-bound textbook with brightly colored tabs marking the different applications. She began working through Section 1-1. The first program we would learn was WordPerfect. This was before the "What You See Is What You Get" interface design became standard. We began by opening the code window at the bottom of the screen using ALT+F3. Text was typed in the top pane, and when you wanted to make a formatting change like italicizing a word, you would issue a specific keystroke from memory, such as CTRL+F8, type the word, then press the right arrow to move the cursor on the right side of the italics operator in the bottom screen. There was little indication on the top screen that anything had changed. At the bottom, in the Reveal Codes window, you could see the codes listed in brackets, one before and one after the text. I found this totally intuitive, but many people did not, including one of the cute girls sitting next to me. She had accidentally pressed the right arrow key before starting to type, so the codes were showing up in the bottom window, but she wasn't seeing the expected change in the top window. The professor obviously needed to get through the whole chapter, so we were moving through the material very quickly. The girl next to me was starting to get upset.

She was really cute. So, I took a deep breath and pointed to the left arrow key on her keyboard, but by then she was too confused to understand the cryptic gesture. She slid her keyboard over to me. I clicked the left arrow, quickly typed the required word, then typed the next sentence, adding bold, underline, or whatever was needed to get her caught up with the professor. I pointed to the codes in the bottom window along with the text, both in front of and behind the words. Then I looked up. The professor had stopped talking and was looking at me. She did not look angry, but I saw that she noticed. Crap. I had felt so cool helping the cute girl, but now I was having a flashback to the first day of first grade with Mrs. Johnson. I put my head down and caught back up with the document we were creating as the professor made her way to the back of the room. I am not sure if my face was even redder now than when the girls first sat down, but I was still having difficulty breathing. I was surprised when she leaned over and said something like, "Thank you for helping. Feel free to do that if it doesn't result in you getting behind." I remember the relief washing over me.

That moment matters because it reveals two things at once. First, the course was efficient—the goals were clear and the procedures could be modeled and repeated. In many institutional settings, that clarity is synonymous with quality. The assignment could be assessed. The pace could be managed. The outcomes could be verified. Second, it exposes the narrowing effect of such clarity when it becomes totalizing. The student who fell behind became a problem for the script's pacing. My brief act of helping her interrupted the smooth flow of the predefined progress. It was a human response to confusion inside a system organized to move forward regardless. The professor's response made room for that, but the course's overall logic still leaned toward coverage, synchronization, and correct replication.

The literature helps clarify the significance of this moment. As previously noted, Timberlake et al. (2017) point out that the allure of scripted curriculum lies in its simplicity and

predictability. Powell et al., though, show how teachers working within scripted programs often experience tensions between fidelity to program design and responsiveness to real learners (Powell et al., 2017). In the WordPerfect classroom, that tension surfaced in miniature. The script wanted coverage. The situation needed responsiveness. My professor's response briefly privileged the latter, but the course itself was still governed by the former.

Currere, Lived Experience, and the Problem of Coverage

Curriculum theory asks questions that standardization often tries to quiet. Pinar (2019) resists the reduction of curriculum to delivery mechanisms and objective sequences. Pinar et al. (2006) frame curriculum as a contested conversation, shaped by historical and contemporary discourses rather than neutral technical design. Poetter (2024) similarly treats curriculum as something lived, narrated, and fragmented across time. This theoretical orientation challenges the assumption that a course can be evaluated solely by whether its parts are aligned, efficient, and replicable. From this perspective, the problem with canned curriculum is not that it is boring. The problem is that it often construes learning as coverage rather than experience. It privileges what can be standardized over what can be transformed. The learner becomes a recipient of the sequence rather than a participant in meaning. This is where scripted curriculum intersects with larger political and ethical concerns. Fitz and Nikolaidis (2020) argue that scripted curriculum undermines democratic education by reducing the interpretive and deliberative roles of teachers and students. Love (2020) pushes this further by suggesting that education must aim at freedom rather than mere survival within managed systems. If students are habituated to perform within pre-authored scripts, then they may become proficient at sanctioned participation while remaining underprepared for situations that require critique, imagination, or collective reinterpretation.

This concern resonates with debates over online course design. "Master" courses can support consistency and quality, but they can also erode instructor agency and narrow the lived texture of a course if they are treated as immutable products. Nehrbass et al. (2022) show that faculty often value flexibility within structure, a phrase that captures the tension well. The question is not whether the curriculum should have structure but whether the structure leaves room for educational life. Huun and Hughes (2014) and Grincewicz and Simunich (2021) suggest that templates and master courses can support engagement when designed thoughtfully. That possibility should not be dismissed. Yet Moon (2019) and Lieberman (2019) remind us that, when design becomes too centralized and transactional, courses can begin to feel detached from the communities that inhabit them.

Cybersecurity education, particularly when taught online, sits squarely inside this tension. It is a field with strong incentives toward standardization: workforce alignment, certifications, scalable labs, competency matrices, and employer expectations. That does not make standardization wrong. It does make it powerful. And power demands scrutiny.

Scripted Curriculum in Cybersecurity Education

Cybersecurity courses are especially vulnerable to over-scripting because the field lends itself to modularization. A course can be broken neatly into reconnaissance, scanning, phishing, malware analysis, incident response, SIEM, threat intelligence, firewall rules, ACLs, VLANs,

policy frameworks, and compliance mapping. Each segment can be paired with a lab, each lab with a rubric, and each rubric with screenshots or command outputs that verify correctness. This is manageable. It scales. It aligns beautifully with certifications and program assessment.

But it also risks teaching students that every authentic problem looks like a lab. In such environments, students may come to see tools as answers rather than instruments, frameworks as recipes rather than lenses, and uncertainty as evidence that something has gone wrong in the course design. The classroom becomes a place where ambiguity is systematically removed, enabling students to demonstrate clear mastery. Yet cybersecurity work, particularly at the levels of policy, systems thinking, and incident response, often requires students to dwell in ambiguity rather than escape it. The appeal of canned curriculum is understandable. Faculty members are busy. Institutions want consistency. Students often say they prefer clarity. I understand that preference deeply. As the earlier fragments suggest, I have often loved efficiency, and I have taught courses from this position myself. Structured courses can reduce anxiety, help novice learners orient themselves, and make online education more navigable. The problem is not the structure itself. The problem is when the structure leaves no room for genuine thought.

Mulcahy and Irwin (2008) warn that a standardized curriculum can obstruct critical pedagogy. Serrano et al. (2018) argue for authenticity in learning and assessment that cannot be reduced to prepackaged performance. Young (2018) goes so far as to suggest that subversion may be necessary when canned curriculum overwhelms educational life. These arguments matter in cybersecurity because the field often celebrates the lone expert who knows the answer. That myth pairs too easily with scripted pedagogy: the expert writes the course, the student follows it, and the correct outputs prove competence. What gets lost is the social and interpretive character of actual cybersecurity practice.

PROGRESSIVE FRAGMENT ONE: THE PLAYBOOK

The alert arrived at 2:17 a.m., not as a siren, or flashing red banners, but as a quiet anomaly—an outbound connection pattern that did not match anything in the rulebook. No known signature. No known exploit. Just an odd behavior. By 7:45 a.m., the Security Operations Center (SOC) was full. Twenty-four analysts sat at identical desks, monitors glowing with dashboards they knew intimately. They had trained for this room. They had passed certifications that promised they were ready for this room. Their screens displayed alerts, packet captures, and incident response checklists. Everything looked familiar. And yet, no one spoke. Evan was the first to break the silence. “Which playbook applies?” he asked, his voice careful. Maya scrolled through the incident classification flowchart projected on the wall. “It doesn’t map,” she said. “There’s no known exploit chain. No malware family. No Indicator of Compromise we can validate.” Across the room, someone laughed nervously. “That’s not possible,” another analyst said. “Everything maps. It must.” They had been taught that. If not explicitly, then implicitly. Their education had been thorough, structured, and efficient. Every lab had a goal. Every exercise had a correct answer. They memorized frameworks, mastered tools, and executed response plans flawlessly, so long as the problem matched the example. This problem did not.

The attacker moved slowly. By midmorning, the anomaly had spread laterally, not through privilege escalation or credential theft, but through something subtler: trust relationships embedded deep within automated workflows. The attack did not exploit a vulnerability so much as it relied on the assumption that internal systems behaved predictably because they always had.

"Is this a misconfiguration?" someone asked. "Run the standard triage," said the incident lead, already flipping through the laminated response binder. "Step one: identify known indicators." "There aren't any," Maya said again. The binder did not have a page for "there aren't any." They escalated to containment anyway, isolating endpoints based on thresholds defined in training. Systems went offline. Critical services followed. Business units began calling. "What are you seeing?" a director asked over the conference bridge. "We're still classifying the incident," the lead replied. "How bad is it?" A pause. Then, "We're not sure."

In the classroom, uncertainty had been implicitly framed as failure. Students were rewarded for reproducing procedures rather than questioning them. Success meant following instructions exactly. Deviations, even thoughtful ones, were marked incorrect. Frameworks had been presented as recipes rather than lenses. When asked why a control worked, the answer was usually, "Because that's best practice." Now, in the SOC, those habits surfaced like muscle memory. "Let's wait for vendor guidance," someone suggested. "Or a signature update." "Or a threat intel bulletin." They waited. The attacker did not. By early afternoon, the breach was undeniable. Data exfiltration patterns emerged, small, encrypted, deliberately indistinct. The analysts watched it happen in real time, graphs inching upward, alarms chiming softly. "Should we block the traffic?" Evan asked. "Based on what justification?" the lead replied. No one answered. Blocking it might disrupt operations costing millions of dollars. Not blocking it allowed the unknown to continue. The playbook required evidence. Evidence required known categories. Known categories did not exist. Leadership joined the call. "Why hasn't this been contained?" someone asked, not angrily, but with confusion. "We're following protocol," the lead said. The silence that followed was heavier this time.

At 4:03 p.m., the system failed on its own. Automated processes began rejecting each other's outputs. Integrity checks failed. Logging pipelines choked. The SOC dashboards flickered, then froze. The attacker had not destroyed the system. They had simply revealed how interdependent and unquestioned it was. The room filled with sound: chairs scraping, keyboards clattering, voices overlapping. "What do we do now?" For the first time that day, no one reached for the binder. Maya stood. "We stop treating this like a checklist problem," she said, surprising herself as much as the room. "This isn't malware. It's not ransomware. It's not anything we were shown. It's abusing how our systems assume trust." The lead looked at her. "What's your recommendation?" Maya hesitated. There was no rubric for this. No guarantee of partial credit. "We model the system," she said. "Not the attack. The system. We trace how trust moves. Where automation hands off without verification. We break those chains, manually if we must." Evan nodded slowly. "That's not in the playbook." "No," Maya said. "But it's in the system architecture." They began sketching on the whiteboard: arrows, assumptions, dependencies. Not steps, but relationships. Not answers, but questions. What trusts what? Why? What happens if it doesn't? The work was slow. Messy. Uncomfortable. It worked. The breach was contained by evening, not through a signature or patch, but by deliberately interrupting automated trust paths. Systems were rebuilt. Data loss was assessed. The zero-day was eventually named, documented, and taught, flattened into a diagram that fit neatly onto a slide.

Weeks later, the incident was added to the curriculum. A new lab appeared. A new checklist. A new answer. But the story the analysts remembered was not the exploit. It was the moment when the scripts stopped working, and no one had shown them how to think without them.

This fragment is not an argument against playbooks. Playbooks matter. Cybersecurity without procedures would be irresponsible. The point is different: the habit of looking first for the authorized script may become dangerous when the situation is unprecedented. If education has

taught students that uncertainty signals failure rather than the beginning of inquiry, then the moment of the zero-day becomes not only a technical crisis, but a pedagogical reckoning.

PROGRESSIVE FRAGMENT TWO: BUILDING MEANING TOGETHER

A group of students who take an unscripted cybersecurity course. The final exam is not a multiple-choice test, nor even a narrowly prescribed lab. Students are tasked with writing a comprehensive security policy for a regional healthcare organization and then translating it into technical enforcement in a real network. They must design VLAN segmentation for clinical systems, craft ACLs to protect patient data, implement firewall rules that reflect least privilege principles, and justify every line of configuration in policy language. The final exam is individual but not isolated work. By design, the Slack workspace remains open. Channels pulse with names: #policy-questions, #vlan-logic, #acl-debug, #firewall-strategy, #infrastructure-help. The structure mirrors the course itself. Policy and implementation are not separate worlds; they are recursive, entangled, and always informing one another. At 8:02 a.m. on the day the exam opens, a message appears: "Does anyone else struggle with how specific to make the acceptable use section? I don't want it to be vague, but I also don't want to write a novel." Within seconds, replies begin: "I framed mine around risk categories. High, medium, low. Then the technical controls reference those." "That helped me, too. Remember when we talked about alignment between policy language and enforcement?" A snippet follows—three short paragraphs defining role-based access for nursing staff, tied to a VLAN labeled VLAN20-CLINICAL.

The conversation is not about copying. It is about remembering—remembering that policy is not decoration, remembering that VLAN segmentation is not simply an exercise in configuration mode, remembering that writing "Clinical devices shall not access administrative financial systems" must translate into an ACL that denies traffic between 10.20.0.0/24 and 10.50.0.0/24. The Slack stream continues. "I'm trying to block inter-VLAN traffic except for DNS and HTTPS. My ACL keeps breaking everything." "Post your logic, not the whole config." A pause, then a cryptic display of code. "Check your DNS line. DNS is usually UDP 53." "And you deny statement placement matters." No one hands over a finished answer. Instead, they offer questions, nudges, and reminders. The architecture reflects weeks of conversation about trust boundaries and data classification.

Progressive curriculum work is not about efficiency alone, but about engagement with complexity. It resists the banking model of education, where knowledge is deposited and withdrawn (Freire, 1968/2017). Instead, it invites students into uncertainty, into the unfinishedness of thought. That uncertainty is visible. The Slack messages bring together multiple voices weaving through policy citations, Cisco syntax, firewall logic, and ethical reflection. Students reference NIST principles. They debate whether "deny ip any any log" should be the final line in their ACLs. They share short code fragments, careful not to cross into wholesale duplication but generous in conceptual clarity. "Remember that order matters." "Test from the source VLAN." "Use show access-lists to verify hit counts." In these exchanges, technical fluency becomes inseparable from relational accountability. Students are not merely implementing VLANs and ACLs; they are protecting a network on which lives depend. They are enacting the belief that security is not an individual heroic act but a collective responsibility.

Earlier in the semester, many of them approached networking as a series of isolated commands to memorize. Configure. Save. Repeat. But through sustained dialogue in Slack threads

that stretched late into the evening, they began to see infrastructure as narrative. A firewall rule tells a story about trust. An ACL encodes a value judgment. A VLAN boundary reflects an ethical claim about data separation. By the final exam, that narrative thinking has become habitual.

Near the end of the day, a message appears in **#policy-questions**: "Anyone else feeling like this is hard but also kind of amazing?" "Yes." "I've never had a final where I actually felt like I was building something real." "Same. It's stressful, but it feels authentic." Authenticity. That word lingers. The configurations must work. The policies must align with technical enforcement. The grading rubric is detailed and demanding. But the rigor is distributed. Knowledge circulates horizontally, not just vertically, from instructor to student. Slack does not replace thinking. It amplifies it. The instructor watches, occasionally interjecting with clarifying responses, but mostly witnessing collective problem-solving unfold. What emerges is not a perfect network, nor a flawless policy document. What emerges is a community capable of thinking together.

As students complete their topologies, the messages slow. Students upload screenshots of successful pings blocked between unauthorized VLANs. They share brief affirmations. "Good luck, everyone." "Whatever happens, we learned a lot." After submission, the channels remain. Not silent, but reflective. "I realized I understand ACL logic way more than I thought." "I used to be scared of firewall rules." "I still am," someone replies, "but less now."

Reflection as Bridge: From Precision and Control Toward Dialogue and Inquiry

I am sitting at my desk late in the evening, the house quiet except for the clicking of my mechanical keyboard, rereading the fragments I have written so far. The documents are open on the screen. I scroll slowly. Reading them together, I begin to see them less as isolated scenes and more as parts of a longer educational journey that continues to shape my understanding of curriculum, teaching, and leadership. As I consider how these experiences continue to influence my educational journey, I find myself increasingly skeptical of curriculum models that treat knowledge as something to be delivered in fixed, linear sequences. Reflecting on these fragments considering Pinar's question, "What is the nature of your educational experience?" I see that my own journey has moved between two curricular worlds: one shaped by clarity, efficiency, and predictability and another shaped by uncertainty, dialogue, and shared inquiry. Scripted curricula offer structure and measurable outcomes. I understand their appeal. I love efficiency. Yet I have also seen how rigid structures can narrow learner agency, discouraging the exploration and collaboration that complex problems require. My shift away from scripted approaches emerged directly from growing disillusionment with the outcomes of integrating canned curricula into our program. In my role as program director for Cybersecurity, Networking, and Platform Engineering (that's a mouthful), I have developed more than a dozen courses, though as a program director, I can only teach three of them each semester. Canned curriculum provides a consistent floor across sections, but in most respects, the results are underwhelming. Students who complete our introductory courses *are* prepared for multiple-choice certification exams; however, without additional experience, many struggle to complete the comprehensive final projects. Struggling here suggests students might struggle in real-life scenarios, so I understand the need for a different possibility.

The Slack-based cybersecurity course represents this different possibility. Students are still working within a curriculum, and expectations remain rigorous, but the learning environment

invites uncertainty rather than suppressing it. Students are not simply carrying out commands; they are wrestling with their meaning. Rather than memorizing procedures, they are constructing understanding together. The implications for cybersecurity education are significant. A field defined by rapidly changing threats cannot rely exclusively on pedagogies that teach students to wait for recognition before acting. Certifications, frameworks, playbooks, and standardized labs all have their place. However, if they become the dominant shape of the learning environment, they may produce graduates who are proficient at following scripts but who may encounter difficulty when the scripts fail. Scripted curricula and unscripted learning experiences do not simply organize content differently; they cultivate different dispositions toward knowledge, authority, uncertainty, and action. Through the regressive fragments, I have shown how early conditioning toward compliance can make ambiguity feel risky and deviation feel suspect. Through the progressive fragments, I have argued that those dispositions can carry forward into professional life.

To "crush" the canned cybersecurity course is not to reject standards, rigor, or structure. It is to reject the assumption that the best preparation for complexity is the removal of complexity from learning. Students need opportunities to wrestle with the unfinished, the ambiguous, and the relational. They need to connect policy and practice, test assumptions, collaborate under conditions of uncertainty, and build meaning together. If we continue to teach cybersecurity as though every future incident already has a page in the binder, we may help create professionals who search too long for the correct playbook while the breach unfolds. If, however, we design courses around central problems and collective inquiry, we may cultivate something more valuable: responders who know how to think, not only when the answer is known, but more importantly, when it is not.

References

Fitz, J. A., & Nikolaidis, A. C. (2020). A democratic critique of scripted curriculum. *Journal of Curriculum Studies*, *52*(2), 195–213. https://doi.org/10.1080/00220272.2019.1661524

Freire, P. (2017). *Pedagogy of the oppressed* (M. B. Ramos, Trans.; 30th anniversary ed.). Bloomsbury Academic. (Original work published 1968)

Golann, J. W. (2021). *Scripting the moves: Culture and control in a "no-excuses" charter school*. Princeton University Press.

Grincewicz, A. M., & Simunich, B. (2021). Designing master courses that promote significant, engaged learning. In B. Hokanson, M. Exter, A. Grincewicz, M. Schmidt, & A. A. Tawfik (Eds.), *Learning: Design, engagement and definition* (pp. 69–86). Springer International. https://doi.org/10.1007/978-3-030-85078-4_6

Huun, K., & Hughes, L. (2014). Autonomy among thieves: Template course design for student and faculty success. *The Journal of Educators Online*, *11*(2). https://doi.org/10.9743/JEO.2014.2.4

Lieberman, M. (2019, May 21). Plug-and-play online courses: Innovation or dystopia? *Inside Higher Ed.* https://www.insidehighered.com/digital-learning/article/2019/05/22/predesigned-courses-help-institutions-stay-efficiency-can-be

Love, B. L. (2020). *We want to do more than survive: Abolitionist teaching and the pursuit of educational freedom.* Beacon Press.

Moon, P. A. (2019, March 26). Building online courses communally. *Inside Higher Ed.* https://www.insidehighered.com/digital-learning/views/2019/03/27/we-can-do-better-master-model-online-course-development-lets-build

Mulcahy, E. D., & Irwin, J. (2008). The standardized curriculum and delocalization: Obstacles to critical pedagogy. *Radical History Review*, *2008*(102), 201–213. https://doi.org/10.1215/01636545-2008-024

Nehrbass, K., Murcray, T., & Morris, E. (2022). Flexibility within structure: Factors contributing to faculty perceptions of autonomy and standardization in course design and delivery. *The Journal of Applied Instructional Design*, *11*(3). 25–40.https://doi.org/10.59668/378.10316

Pinar, W. F. (2019). *What is curriculum theory?* (3rd ed.). Routledge.

Pinar, W. F., Reynolds, W. M., Slattery, P., & Taubman, P. M. (2006). *Understanding curriculum*. Peter Lang.

Poetter, T. S. (2024). *Curriculum fragments: A currere journey through life processes.* Routledge.

Poetter, T. S. (2026, April 8). *Scripted curriculum, past and present: "Now what?"* [Conference presentation]. SSCH Annual Meeting, Los Angeles, CA.

Powell, R., Cantrell, S. C., & Correll, P. (2017). Power and agency in a high poverty elementary school: How teachers experienced a scripted reading program. *Journal of Language and Literacy Education*, *13*(1), 93–124. https://eric.ed.gov/?id=EJ1141492

Serrano, M. M., O'Brien, M., Roberts, K., & Whyte, D. (2018). Critical pedagogy and assessment in higher education: The ideal of 'authenticity' in learning. *Active Learning in Higher Education*, *19*(1), 9–21. https://doi.org/10.1177/1469787417723244

Timberlake, M. T., Burns Thomas, A., & Barrett, B. (2017). The allure of simplicity: Scripted curricula and equity. *Teaching and Teacher Education*, *67*, 46–52. https://doi.org/10.1016/j.tate.2017.05.017

Valdez, C. (2020). Flippin' the scripted curriculum: Ethnic studies inquiry in elementary education. *Race Ethnicity and Education*, *23*(4), 581–597. https://doi.org/10.1080/13613324.2018.1497959

Vaughn, M., Parsons, S. A., & Gallagher, M. A. (2022). Challenging scripted curricula with adaptive teaching. *Educational Researcher, 51(3)*, 186–196. https://doi.org/10.3102/0013189X211065752

Young, J. J. (2018). How to subvert your university's canned curriculum. *Radical Teacher*, *111*, 134–135. https://doi.org/10.5195/rt.2018.523