

# RADICL CTF: Low-Cost Capture the Flag Platform for Industrial Control Systems Education

Taegan Williams  
*Dept. of Computer Science*  
*University of Idaho*  
Idaho Falls, USA  
ORCID: 0000-0003-0434-1040

Tiffany Fhurmann  
*Dept. of Computer Science*  
*University of Idaho*  
Idaho Falls, USA  
ORCID: 0000-0003-0196-4787

Michael Haney  
*Dept. of Computer Science*  
*University of Idaho*  
Idaho Falls, USA  
ORCID: 0000-0003-2359-2478

**Abstract**—To address the nationwide workforce shortage of skilled and educated cyber-informed engineers, we must develop low-cost and highly effective resources for industrial control systems education and training. College curricula in technology management, cybersecurity, and computer science aim to build students’ computational and adversarial thinking abilities but are often done only through theory and abstracted concepts [1]. To better a student’s understanding of industrial control system applications, post-secondary institutions can use gamification to increase student interest through an interactive, user-friendly, hands-on experience. RADICL CTF can provide post-secondary institutions with new opportunities for low-cost, guided exercises for industrial control system (ICS) education to help students master adversarial thinking. Based on an extension to PicoCTF, RADICL CTF is a platform for students to design, implement and evaluate exercises that test their understanding of core concepts in industrial control systems cybersecurity, answering the need for more interactive education methods. The main contributions of this paper are the improvement of the cybersecurity curriculum through extending the PicoCTF platform to promote the gamification of industrial control system concepts with consideration to the Purdue Reference Architecture.

**Index Terms**—Cybersecurity, Capture the Flag, Gamification, Cyber-Physical Systems, ICS Education, RADICL, PicoCTF

## I. INTRODUCTION

Less than ten years ago, the computer science curriculum was largely absent for students in K-12 education, with most of their interaction happening in enrichment opportunities [2]. Our education system is adapting to a growing demand for cybersecurity professionals. Today’s STEM education has more consideration for computer science, focusing on developing students’ computational thinking [3]. Computational thinking encompasses problem-solving, and system design through logical steps and algorithms with consideration of human behavior [3]. In post-secondary institutions, computer science courses should now aim to mature students’ computational thinking to consider the adversarial mindset [4]. Applying the adversarial mindset encourages the development of security concepts within the implementation of the computational process.

In August of 2022, the Pentagon tested the security of microgrids at DEFCON in Las Vegas before they were to be deployed to 134 bases across the United States. Their goal was to test the reliability and security of their design

as a realistic evaluation of potential cyberattacks against our nation’s public infrastructure. With over 1,700 in attendance at DEFCON, multiple vulnerabilities brought down the mock infrastructure. One of the attacks was a trivial integer overflow of the temperature sensor stopping the availability of wind, and solar power [5]. The inherent problems of putting engineering before security proves that we should introduce students of all disciplines to adversarial thinking. Adversarial thinking can strengthen our reliance on information and services - including industrial control systems.

There is a high demand to support the development of resources that improve adversarial thinking in education. Educational courses need hands-on activities to complement the analysis of theories and principles [4]. One of the ways educators are meeting the demand for hands-on activities is through gamification. Gamification implements learning concepts as game elements to increase social interaction, user-engagement and enhance positive learning patterns [6]. Gamification, which began as a method for understanding marketing and customer engagement, is being used across educational disciplines. A literature review of game-ifying learning concepts in 2014 proved the acceptance and use in the industry to produce positive results in retention [6].

This paper presents a means for gamification to extend to computer science and cybersecurity specializations to further educational resources around industrial control systems, operations technology (OT), and critical infrastructure resilience. Education in these disciplines is still in its infant stages due to its complexity and lack of educational resources. With the importance of preparing the next cyber-resilient workforce, partnerships have been developing to provide post-secondary education students access to current trends, and modern problems in critical infrastructure security [7]. While these partnerships improve education, they are not widely available across educational institutions and target students with existing interests or experience. RADICL CTF aims to bridge the academic gap through the gamification of these specialized topics targeting a broader range of students with all experience levels.

### A. Need for Game-based Cybersecurity Education

The cybersecurity curriculum is evolving with the constant state of risk, which causes a lack of technical training and materials [1]. Educational resources for teaching cybersecurity affect the next generation of cybersecurity professionals who will help research and develop countermeasures against attacks on ICS networks [8]. Current methods of teaching cybersecurity education focus on analyzing abstraction and principles about specific case studies to teach adversarial thinking [4]. The curriculum needs to improve adversarial thinking for students by applying these concepts to design, and implementation [4]. This paper presents RADICL CTF as a platform for students to design, implement, and evaluate small-scale physical systems to master adversarial thinking, build students' understanding of operations technology, and improve the national cybersecurity curriculum.

### B. RADICL CTF Objectives

The objectives of the RADICL CTF include the following:

- 1) Integrate a guided learning experience for interacting with CPS/ICS/IoT devices in mock operations networks;
- 2) Maintain a realistic network architecture backed by the Purdue Reference Architecture; and
- 3) Increase the availability of resources by using educationally available or open-source software.

## II. BACKGROUND

### A. Reconfigurable Attack Defend Instructional Laboratory (RADICL)

The Idaho Falls RADICL Lab aims to provide a resilient cyber range for students to practice using reconfigurable defense-in-depth networks to leverage multiple systems in a realistic learning environment. It has evolved since its adoption in 2004 to assist as an educational tool against the evolving cybersecurity curriculum [9] [10]. Funding from the state of Idaho and grants from NSF have helped develop an isolated and controlled network for students to engage in realistic attack and defense exercises. The lab also produces data sets for additional research and learning. Applying hands-on cyber-physical activities working with malware samples in real-time makes RADICL a unique environment to help fill knowledge gaps between enterprise and operations networks in current cybersecurity education.

1) *The Goals of Hands-on Cybersecurity*: For students to master adversarial thinking through hands-on activities and for the mission of RADICL to be effective, students must engage in the five freedoms of play. The US Naval Postgraduate School, in collaboration with the University of Washington, published an article titled *Security through Play*. This article showcases two tabletop exercises developed around learning concepts in cybersecurity. Their work is a successful example of the gamification of security concepts considering five elements essential to children's playground games [11]. Similarly, RADICL's hands-on exercises work to meet the following four freedoms.

- **Freedom to Experiment** - the freedom for students to experiment with defenses against adversaries.
- **Freedom to Fail** - the freedom to observe adversarial attacks.
- **Freedom to be an Adversary** - the freedom to assume the adversary's mindset and explore motivations.
- **Freedom for Reinforcement Learn** - the freedom to improve strategy and learn lessons about the game through playing the game.

These freedoms allow students to consider multiple perspectives to master the adversarial mindset. Within this project, the above freedoms are met by students by building challenges, observing packet captures, or completing challenges. Multiple platforms engage students to learn cybersecurity concepts through gamification with similar freedoms. Capture the flags (CTF) is a more serious gaming experience inspired by military training with realistic game mechanics. In cybersecurity, gamification and CTFs help introduce students to a breadth of specialization topics.

### B. Open Source CTF Platforms

Capture the flag (CTF) platforms are one of the many competitive educational tools that can guide students to understanding learning concepts through gamification. Using a CTF platform, we provide a learning experience for students to engage in multiple challenges that evaluate realistic vulnerabilities found in the industry.

When evaluating CTF engines for the RADICL lab, we considered the underlying services, community support, and flexibility in challenge deployment. Our team's solution should be a full stack with a user-web interface, admin control interface, database, and documentation for challenge creation and deployment. The CTF engine should also interact with docker containers for ephemeral, controlled environmental connections. Research in 2020 analyzed twenty-five well-known cyber ranges, including the NATO Cyber Range, Virginia Tech, and Georgia Cyber Range, then interviewed ten diverse ranges to inform the University of West Attica about the essential components to run and manage a secure environment [12]. Educational cyber ranges provide a wide variety of gamification around the web, forensics, and malware analysis challenges. Their research revealed that most cyber ranges have open-stack solutions with options for remote or on-premise access, ansible is primarily used to configure infrastructure, and the environment's gamification scoring often involves YAML or JSON formatted documents. The following are a few notable CTF platforms that we took into consideration when deciding on our CTF engine.

Another consideration for the CTF platform is to educate students in multiple stages. Students can first learn through a jeopardy style CTF and then participate in a red blue exercise with the physical platforms. The CTF engine should be able to manage scoring the jeopardy style competition, but additional tools should be utilized to evaluate the availability of ICS services. In 2017, a two stage CTF was performed internationally. The SwaT Security Showdown (s3) competition was created to

address the problems of education around ICS systems through a CTF platform [13]. Researchers in Singapore conducted the CTF in two phases around a water treatment testbed. In their red blue exercise they virtualized the PLC and considered the architecture similar to an ICS honeypot - a valuable solution to implementing diverse challenges within our cyber range.

- **KCTF** - A versatile, Kubernetes-based CTF infrastructure created by Google [14]. With security in mind for hosting vulnerable applications, kCTF offers local and remote deployments through gcloud [14].
- **CTFd** - A CTF framework that uses python flask to host challenges in a Docker image [15]. CTFd offers easy management for administrators and simple use for its users; however, there are additional costs to hosting different style challenge questions.
- **PicoCTF** - A full-stack jeopardy-style CTF with a supportive community. The available code for PicoCTF is based on their 2019 competition and uses multiple systems to host different types of challenges [16].
- **Facebook CTF** - A full-stack jeopardy or king of the hill style CTF that uses Vagrant or Docker to host challenges. The platform uses an interactive world map where each country can represent separate challenges. The project was last active in 2018 [17].

With multiple frameworks for an open-source CTF engine available, our team decided on working with the PicoCTF framework as it provides the most diversity in challenge provisioning with options for local and remote deployments. It is particularly adept at managing multiple Docker containers, and multiple simultaneous users can spawn simultaneous and different challenge types. By leveraging the existing code base for Docker challenge deployments, we can extend the platform to interact with a networked hardware environment, including mock industrial control systems.

### C. PicoCTF

Researchers at Carnegie Mellon University first published a paper about their 2013 PicoCTF competition's success after recognizing the shortage of computer security experts [18]. Their initial competition introduced a game-based competition with over 2,000 high-school teams. They released their source code for the competition online, which quickly became adapted as a part of the curriculum in 40 high schools. The code base continued to be supported and saw major improvements after the 2019 PicoCTF competition. As dependencies within the infrastructure become no longer supported and interest in students maintaining the code base slowly diminished, the code base became archived on GitHub at the end of 2022.

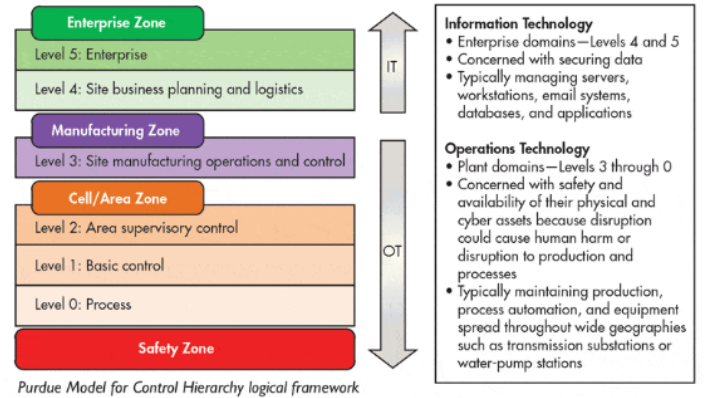
While RADICL CTF is unique in its goals and methodology, some existing successful CTFs and workshops interact with physical devices or virtualize the interaction. CTFs like Microcorruption teach adversarial thinking around virtualized systems. RADICL CTF is different from Microcorruption, Hack the Box, and other web-based red-team exercises as RADICL CTF creates and destroys virtual connections to

physical systems allowing for a wider opportunity for visual gratification.

### D. Purdue Enterprise Reference Architecture

The Purdue Reference Architecture has been long-established in modern ICS networks. The model includes six levels separating physical assets and services. Level zero is the lowest level of the model. It includes physical processes, sensors, and actuators, while level four and five are the highest and represents the information technology/enterprise side of the organization for ICS networks [19]. Figure one shows the official separation of the Purdue Enterprise Reference Architecture.

Fig. 1: Purdue Enterprise Reference Architecture



[20]

Understanding the different levels of the Purdue Enterprise Architecture helps build a reference for deploying mitigations against adversary techniques. To cause physical damage using an actuator, they must move through the levels of the Purdue Reference Architecture to conceal and carry out their attack. This reference architecture provides a blueprint for systems engineers to build a more defensible system, wherein the components of the system's various detective and preventive controls can be implemented to have the greatest impact. At a most basic level, the architecture describes how to segment a network and key business and operations functions based on logical and functional groupings.

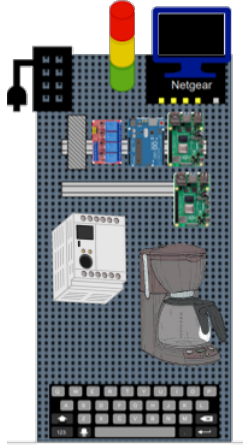
## III. RADICL CTF ARCHITECTURE

### A. Cyber-Physical Platform

The Cyber-Physical Platform contains 2 Raspberry Pis, an Arduino, a PLC, and a physical process. The physical process can be a wide variety of solutions around a relay, led, or actuator. The platforms will contain challenges representing each level of the Purdue Reference Model. The physical devices are separated by two din rails, the top din rail is for a gateway raspberry pi with connects a platform to the cloud infrastructure. All student communication travels through a physical switch to the gateway pi. The gateway pi is also responsible for observing the challenging environment and

communicating to a status light controlled by a relay to inform the instructor about the current status of a challenge. When the tower light is green, it indicates that the challenge is ready for a student to interact with the platform. A student successfully interacts with the platform when the tower light turns yellow. And when the tower light turns red, a student successfully interacted with the physical process. The bottom dim rail is for raspberry pi containing docker challenge for level 4, level 3, and level 2 of the Purdue Reference Model. As students connect further with the environment, they will complete new challenges and eventually interact with physical processes. Figure two shows an example platform with a coffee pot as the physical device.

Fig. 2: ICS Target Learning Platform



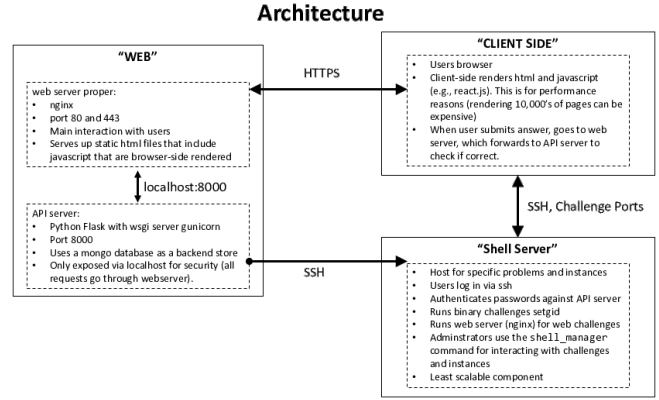
## B. Cloud Infrastructure

Our research aims to extend the PicoCTF method for local deployment to include configuration and communication with a physical ICS network. We first configured Ovirt as a virtualization solution on a rocky 8 Operating system to provision the cloud infrastructure. We then created two Ubuntu 20 virtual machines and used ansible scripts from the PicoCTF GitHub repository. We had to modify the repository to work with Ubuntu 20 servers and update many of the python dependencies in the ansible script. When the ansible script finished, we were able to use successfully use SSH for management and Nginx, python-flask, and MongoDB for web hosting and challenge deployment. The PicoCTF community created the architecture in figure three, which shows the communication between two deployed virtual machines. [16].

For containerized challenges to work on PicoCTF, the shell\_manager command with support of the hacksploit library and the flask web API with support of the service gunicorn must be operational. The following diagram is on the PicoCTF's GitHub repository for On-Demand Challenges. Figure four shows the relationship between the client, shell, web, and docker using shell\_manager and the API.

Our extension of the PicoCTF platform adds additional functionality to the API and on-demand docker challenges.

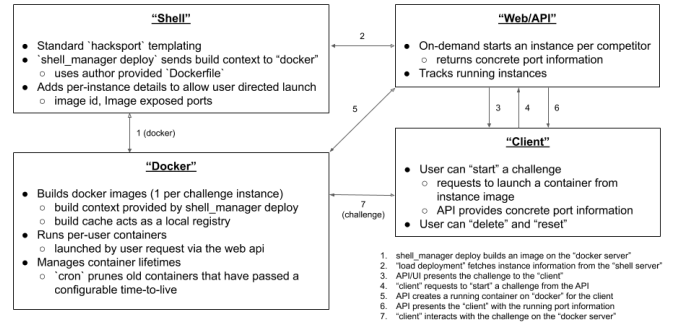
Fig. 3: PicoCTF Platform



[16]

Fig. 4: PicoCTF Platform

## `DockerChallenge` Architecture



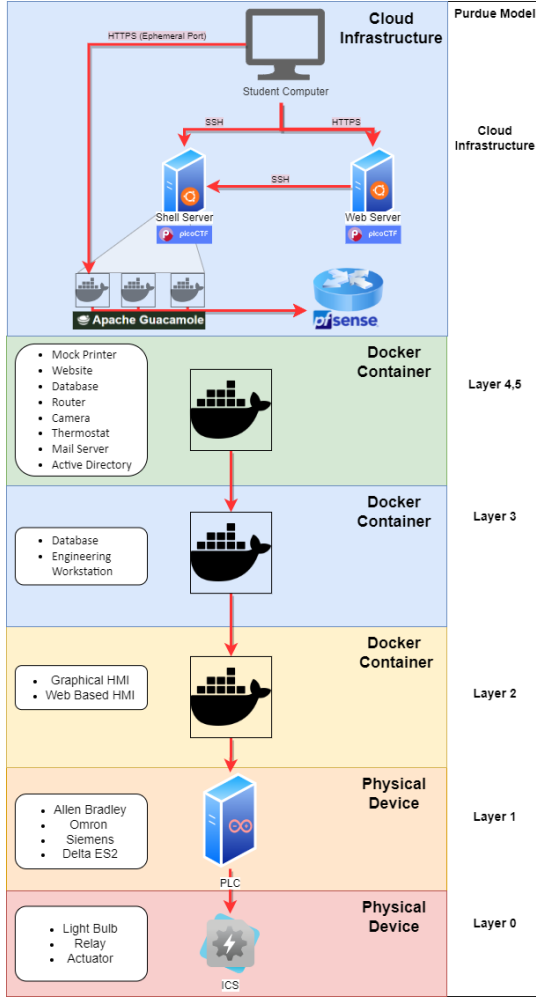
[16]

When a student accesses the web interface, the student can start a challenge and instantiate a containerized jump box web accessible from an ephemeral port on the pico web server. The docker jump box uses Apache guacamole, allowing VNC and SSH over HTTP to complete containerized challenges and interact with a physical platform. After the student connects to the ephemeral port on a web browser, the student is greeted by a desktop with all the necessary installed tools and isolated network connectivity to a learning platform. After reviewing the problem description on the pico web server, the student will connect to different docker containers residing on the raspberry pi on the second din rail until they can interact with the physical process.

Communication travels through a PFSense firewall to

Each level of the Purdue Reference Model will have virtual challenges that reside on separate docker-generated networks. The goal is for students to recognize the importance of network segmentation, recognize misconfigurations, and understand where devices and services should reside. A new container is generated when a student starts a challenge from the pico web server, and the Docker containers for each level are generated from a static configuration image. The following are example containers our team would like to see implemented for each

Fig. 5: RADICL CTF Architecture



level of the Purdue Reference Model.

### C. Enterprise Containers

Enterprise Docker containers represent devices at levels four and five, such as enterprise or ICS services, that companies may typically expose to the Internet. These services may include web servers, VPN servers, IoT devices, cameras, and database management interfaces.

### D. Operations and Control Services

Level three of the Purdue model contains engineering workstations, historians, manufacturing execution systems (MES), Manufacturing Operations Management (MOM) systems, and databases. These services would exist in a realistic environment to manage plant production and aggregate data to devices at level four. Students can access the physical system on level three using existing services from the enterprise container. After completing the challenge, the student can access a network containing level-two devices.

### E. Control Systems

Level two of the Purdue model consists of supervisory control and data acquisition systems (SCADA), programmable logic controllers (PLCs), remote transmission units (RTUs), and human-machine interfaces (HMIs). These devices would exist on the plant floor to control and operate physical processes and systems seen at level one. In our target platforms, these devices are represented by a combination of real PLCs.

### F. Physical Processes

Purdue levels one and zero are physical systems with sensors, actuators, and running processes. Students will be able to see the physical effects of these systems from their manipulation of control systems at level two. Currently, platforms are hung on the walls of the laboratory and classroom. Future work for the facility is to include a separate network of IP-based cameras that will allow remote students to interact with the physical systems.

The gateway raspberry pi has access to each docker-generated network, allowing the pico shell server to fully deploy, configure, and reset each challenge.

## IV. PROBLEM DESIGN

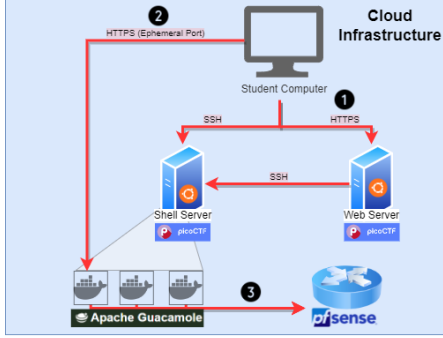
Our first problem was created to test our ideas as a proof of concept and recognize further challenges that may prevent progress. When considering the design of the problem, we did not take into consideration the effectiveness in conveying learning concepts through problem design; however the 2014 USENIX conference published a paper on the learning obstacles in CTFs. Their findings showed some of the downfalls in education through gamification is the difficulty of playing [21]. For future problems, we will consider the learning objective for the challenge relative to the difficulty of the challenge.

To exercise our architecture, We connected the pico shell and web servers on the Ovirt cloud infrastructure to an Cyber-Physical Platform. An on-demand problem was added to the CTF engine for the student. The problem asks for students to start brewing coffee for the DVCP company. ① The student starts the challenge by clicking a button to instantiate a docker jump box on the Pico shell server that's connected to an ephemeral port. ② The student can then access the web desktop and connect to the Cyber-Physical Platform. ③ To communicate to the platform associated with the challenge, a Pfsense router facilitates the communication the correct Cyber-Physical Platform.

④ The Pfsense router then communicates to the gateway pi which forwards the connection to a Docker container. ⑤ The first problem presents a website that would exist at level four or five of the Purdue Reference Architecture. The initial webpage is simple, with only a login interface. Exercising a known vulnerability, students solve the problem by inputting default credentials. Once logged in, the student can access the website's interior pages. A tab in the sidebar reads "Ping the OT environment". The second challenge is for the student to perform a cross-site scripting attack. In the input field, the student can traverse the directories of the Docker container

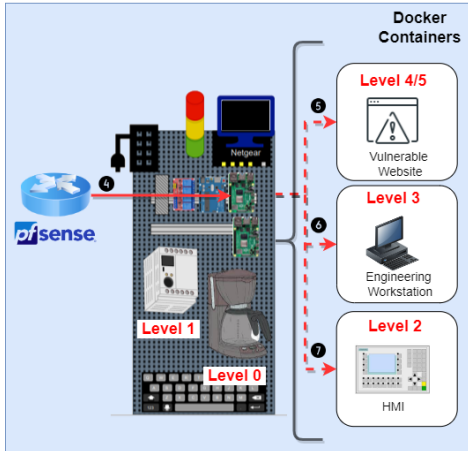


Fig. 6: Initial Steps to Interact with a Platform



to discover an engineering account with a text document containing the engineering workstations IP address. ⑥ The student can then access the engineering workstation using SSH and complete a simple forensics based problem. Once the student solves the forensics challenge they can then use RealVNC viewer on the jump box to connect to a container running a python Tkinter HMI at level two of the Purdue model. ⑦ After the student connects to the container running the HMI, they can interact with a Click PLC. By interacting with the PLC, the student can turn on the coffee pot and see the flag written on the platform display.

Fig. 7: Interacting with the Physical Process



While in this example, we do not have a challenge with a service that would exist at level one of the Purdue model, the problem could be further expanded to include more Docker containers to create a realistic environment. Because of OS limitations, there are plans to expand the Cyber-Physical Platforms to include challenges on Microsoft Windows to interact with PLCs that are Operating system dependent such as the click PLC.

## V. CHALLENGES

When developing RADICL CTF, we expect challenges with multiple students trying to access a single platform, resetting the physical environment, and separating student communication from management. The following is a breakdown of our

team's concerns and solutions to the challenges during the deployment and evolution of RADICL CTF.

### A. C1: Resetting the Physical Environment

One of the challenges that is necessary for students to practice on the cyber-physical platforms is the ability to reset the environment to a starting point. The Picoweb server has buttons for students to start, stop and reset challenges that we believe exists as an API that can be modified. Another method for operators and teachers to reset the environment is to implement an administration page that allows for the same reset functionality for any active platform. Additionally, after a student completes a training exercise on the physical network, destroying the student's jump box container should successfully trigger a reset of the cyber-physical Platform. We believe utilizing existing remote code execution that is currently available through PicoCTF as well as Docker and Ansible will orchestrate the commands to reduce the management overhead of RADICL CTF.

Likewise, we continue to explore robust and resilient methods of resetting physical processes and devices as the student sessions during challenges are abandoned or dropped after being partially completed.

### B. C2: Separating Management and Student Communication

The physical network design involves devices at each level of the Purdue model. While the devices exist to promote a learning experience about industrial control system networks, the small-scale network does not allow the devices to be dual-homed or for students to perform jumps from one level to the next or follow multiple different paths through the environment as would be possible in the real world. Figure 4 shows how students interact within the environment to control the physical process at level zero.

Separating management and student communication is important for students to be able to review PCAP files from an exercise and observe an adversarial attack. The hardware on each Cyber-Physical Platform is limited in the number of physical connections, and management interactions are critical to reset a platform from an unplanned state. To solve this problem, all communication from the students will travel as described in figure 6 and figure 7; however communication between the management network, PFSense, and each platform, are VLAN tagged on the same interface - effectively separating communication between students and operators on the platforms.

### C. C3: Multi-Use Lockout

When a student creates a solution to pivot from an enterprise container to the physical network, RADICL CTF should be able to identify if a current student is already communicating with the physical device to prevent two students from working on the same exercise. If two students connect to the same device, the actions of one student could affect the solution of another and the overall learning experience. Therefore, we implement a load balancer that forwards communication

from a Docker jumpbox to a Cyber-Physical Platform. Load balancing is a system that distributes network or application traffic across a group of servers.

We initially believed that implementing an Nginx load balancer would be the best solution to forward student connections; however, with our current implementation, we can perform load balancing on the PfSense router. To maintain direct connections between problems in the web interface as well as offer a challenge that performs multi-user lockout, the shell server should establish a new ipvlan docker network to communicate with the PfSense router. Currently all communication from the Docker container jump box's are on the default docker bridge network; therefore all communication from the jump box appears as the Pico shell server. Future work should expand the segmentation of problem availability by creating a new ipvlan network for each problem - only allowing the desired communication.

## VI. RESULTS

### A. Cyber-Physical Platforms

At the end of the project, we successfully created two cyber-physical platforms that integrate with RADICL CTF. Students are able to practice basic web exploitation and common methods of access to disrupt the Coffee Pot Company or Light Company. The following images show the assembled platforms in a functional state.

The coffee pot Cyber-Physical platform is the same elements as described in section 3A, while the light bulb cyber-physical platform makes use of an Arduino to communicate to the status light tower and relays to control 20 volts to the light bulb and the status light tower. Several design choices were considered and the platform design is further explained in *Cyber-physical Shooting Gallery: Gamification to Address the IT-OT Gap in Cybersecurity Education*

### B. Our Improvements to PicoCTF

While using the platform, our team updated 20+ dependencies defined in ansible playbooks for services that relied on python2, python3, and cryptography. We also updated the support for the infrastructure to work on Ubuntu 20 systems. After utilizing the PicoCTF framework for local competitions unrelated to this research, we extended the capabilities of PicoCTF to allow multiple correct answers for a more user-friendly experience. Our work on improving the user experience and extending the functionality led us to remove features such as the PicoCTF anti-cheating mechanism. Because of our contributions, RADICL CTF is a branch of the primary PicoCTF code base and has been made available on GitHub at <https://github.com/taeganw/RadiclCTF>

### C. Our Solution to Challenges

One of the critical functions of the user experience is the ability to reset the environment after an exercise for another student to use. To reset an Cyber-Physical Platform, our team modified the web API to recognize a student's interaction with the pico web server. When a student creates or destroys

a jump box, the Pico web server first queries the mongo database for the problem the student is interacting with. It then communicates to the pico shell server over SSH. The pico shell server then executes an ansible playbook that communicates to the Raspberry Pi on the platform to restart the docker containers and the Raspberry pi on the top din rail to reset the tower light and platform status checks. Additionally, after a student completes a training exercise on the physical network, the CTF engine destroys the student's jump box container and resets the physical environment. While we were able to successfully reset the environment without intervention between the student and the Cyber-Physical Platform, future work should expand the administration interface to allow for an administrator to start, stop, and reset any active device.

We did not reach a point to work on challenge two or three, but we believe its possible to monitor communication between a student and the platform by capturing communication on the pfsense router and recognizing the automatic commands that start, stop, and reset the Cyber-Physical Platforms. By mirroring the communication to a new interface on the PfSense router and using python, we can create pcap files for each interaction with the physical platforms; further work will need to plan for decrypting the communication by knowing the encryption key. Challenge three is another modification to the pfsense router that would load balance communication to individual platforms.

## VII. FUTURE WORK

Through the ongoing implementation of RADICL CTF and with the interaction and gathered feedback of local high school students and collegiate cyber-defense students, we expect to expand the number of challenges available and improve the game platform. Beyond addressing the challenges anticipated and discussed in the previous section, we plan to work with local industry partners in designing and deploying a more comprehensive selection of realistic cyber-physical systems platforms that represent systems in each of the 16 DHS Critical Infrastructure Sectors. Current platforms include logical/physical access control measures representing Government Facilities and multiple Energy and Manufacturing sector systems.

Additionally, as we improve and expand the capabilities of the RADICL CTF platform, the code developed to support our interactions with cyber-physical systems platforms and enhancements to the PicoCTF engine will be made available on Github.

## VIII. CONCLUSION

To prepare the next generation of cybersecurity professionals for building and defending resilient critical infrastructure, students need to build adversarial thinking [4]. Adversarial thinking considers how computational thinking applies considering modern protocols and security principles. Students can better prepare for future careers in critical infrastructure by improving adversarial thinking through access to more hands-on exercises focused on industrial control systems. RADICL CTF is a hands-on guided learning experience that promotes

all levels of challenges to students. RADICL CTF aims to promote critical and adversarial thinking through the gamification of operations networks to build an understanding of industrial control systems. Using the PicoCTF 2019 framework, Docker, PFsense, and several cyber-physical system platform designs, students can traverse the Purdue Enterprise Architecture and complete challenges to take control of physical processes. To better understand adversarial thinking, students must apply abstract ideas to the design and implementation [4]. With the reconfigurable design of the cyber-physical system platform, students can implement defensible industrial control system networks with different software at levels three and lower. The platform switch also segments management and student traffic, allowing for realistic packet captures of an artificial attack on ICS to be further used within academia. The RADICL CTF platform, paired with the selection of small-scale cyber-physical systems, allows students to gain hands-on experience with operations technology networks and expands and improves the cybersecurity curriculum.

#### REFERENCES

- [1] M. A. Khan and others, "Game-based learning platform to enhance cybersecurity education," *Education and Information Technologies*, pp. 1–25, 2022.
- [2] A. Repenning *et al.*, "Scalable game design: A strategy to bring systemic computer science education to schools through game design and simulation creation," *ACM Trans. Comput. Educ.*, vol. 15, no. 2, Apr. 2015. DOI: 10.1145/2700517. [Online]. Available: <https://doi.org/10.1145/2700517>.
- [3] S. Grover and R. Pea, "Computational thinking in k–12: A review of the state of the field," *Educational Researcher*, vol. 42, no. 1, pp. 38–43, 2013. DOI: 10.3102/0013189X12463051. [Online]. Available: <https://doi.org/10.3102/0013189X12463051>.
- [4] F. B. Schneider, "Cybersecurity education in universities," *IEEE Security & Privacy*, vol. 11, no. 4, pp. 3–4, 2013.
- [5] S. Smalley, "Pentagon put microgrid technology to the test at def con, drawing on hackers' ingenuity," *Cyberscoop*, 2022. [Online]. Available: <https://www.cyberscoop.com/pentagon-hackers-secure-the-microgrid/>.
- [6] J. Hamari, J. Koivisto, and H. Sarsa, "Does gamification work? – a literature review of empirical studies on gamification," in *2014 47th Hawaii International Conference on System Sciences*, 2014, pp. 3025–3034. DOI: 10.1109/HICSS.2014.377.
- [7] D. Oliver and M. Haney, "Preparing the next cyber-resilient workforce through cross-pollination education," in *2017 Resilience Week (RWS)*, 2017, pp. 44–49. DOI: 10.1109/RWEEK.2017.8088646.
- [8] W. Newhouse *et al.*, "National initiative for cybersecurity education (nice) cybersecurity workforce framework," *NIST special publication*, vol. 800, no. 2017, p. 181, 2017.
- [9] S. Caltagirone *et al.*, "Design and implementation of a multi-use attack-defend computer security lab," in *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06)*, vol. 9, 2006, pp. 220c–220c. DOI: 10.1109/HICSS.2006.115.
- [10] D. C. de Leon, C. E. Goes, M. A. Haney, and A. W. Krings, "Adles: Specifying, deploying, and sharing hands-on cyber-exercises," *Computers & Security*, vol. 74, pp. 12–40, 2018.
- [11] M. Gondree, Z. N. Peterson, and T. Denning, "Security through play," *eng, IEEE security privacy*, vol. 11, no. 3, pp. 64–67, 2013, ISSN: 1540-7993.
- [12] N. Chouliaras, G. Kittes, I. Kantzavelou, L. Maglaras, G. Pantziou, and M. A. Ferrag, "Cyber ranges and testbeds for education, training, and research," *Applied Sciences*, vol. 11, no. 4, p. 1809, 2021.
- [13] D. Antonioli, H. R. Ghaeini, S. Adepou, M. Ochoa, and N. O. Tippenhauer, "Gamifying ics security training and research: Design, implementation, and results of s3," in *Proceedings of the 2017 Workshop on Cyber-Physical Systems Security and Privacy*, 2017, pp. 93–102.
- [14] "Kctf." (), [Online]. Available: <https://google.github.io/kctf/>.
- [15] K. Chung. "Ctfd : The easiest capture the flag framework." (), [Online]. Available: <https://ctfd.io/>.
- [16] picoCTF. "Picoctf." (2022), [Online]. Available: <https://github.com/picoCTF/picoCTF>.
- [17] "Fbctf." (), [Online]. Available: <https://github.com/facebookarchive/fbctf/>.
- [18] P. Chapman, J. Burket, and D. Brumley, "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)*, San Diego, CA: USENIX Association, Aug. 2014. [Online]. Available: <https://www.usenix.org/conference/3gse14/summit-program/presentation/chapman>.
- [19] J. Searle, *ICS Overview* (ICS/SCADA Security Essentials). Secure Anchor Consulting, vol. 1.
- [20] S. Balogh, *Top 10 automation trends in 2018: Innovative manufacturing technology*, Mar. 2020. [Online]. Available: <https://www.encompass-inc.com/top-10-automation-trends-in-2018/>.
- [21] K. Chung and J. Cohen, "Learning obstacles in the capture the flag model," in *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14)*, 2014.