Create and Host Cyber Competitions Using Preliminary Persistent Cyber Training Environment (PCTE)

Christopher Thompson, Gabriel A. Bearden, Ty Sloan, Roy Laurens, Cliff Zou, Bruce Caulkins
University of Central Florida
Orlando, FL
cthompson@hackucf.org, <gbearden, rlaurens, tysloan>@knights.ucf.edu, bcaulkin@ist.ucf.edu, czou@cs.ucf.edu

ABSTRACT

As the world becomes more interconnected and our lives increasingly dependent on the cyber world, the increasing threat of cyberattacks and cybercrimes makes it critical for us to provide better and practical training of the cybersecurity workforce. In recent years, cybersecurity competitions have become one of the most effective ways of educating and training both college students and professionals alike. In this paper, we first systematically introduce in detail the step-by-step procedure and technical knowledge on how we made use of the ongoing Department of Defense Persistent Cyber Training Environment (PCTE) to set up a cyber-competition virtualization environment, configure and install operating systems and popular services with various exploitable vulnerabilities, and set up the participants' access to the event and the scoring system. We then introduce the cybersecurity competition successfully organized by us at the I/ITSEC 2019 conference, as well as the experience and lessons learned from this real-world competition event. The technical details and knowledge presented in this paper could help other researchers and educators to set up their own cyber competition environment or event to better train the future cybersecurity workforce.

ABOUT THE AUTHORS

Mr. Christopher Thompson is a Digital Media major undergraduate student in the University of Central Florida (UCF) with an interest in cybersecurity.

Mr. Gabriel A. Bearden is a UCF Information Technology undergraduate pursuing a career in the cybersecurity sector.

Mr. Ty Sloan is a UCF undergraduate Psychology student with an interest in software development.

Mr. Roy Laurens is a distance-learning lecturer in the Department of Computer Engineering, Dinamika University, Surabaya, Indonesia. He is also currently pursuing a Ph.D. in computer science from UCF. His main research area includes payment card fraud and network security.

Dr. Cliff C. Zou is an associate professor in the Department of Computer Science, University of Central Florida. He received his Ph.D. degree from the University of Massachusetts, Amherst, MA, in 2005. His research interests include computer and network security, computer networking, and performance evaluation. He is a senior member of IEEE.

Dr. Bruce Caulkins is an assistant professor at the School for Modeling, Simulation & Training at UCF. He received his Ph.D. degree from the University of Central Florida, Orlando, FL, in 2005. He is a retired Army colonel with over 28 years of active-duty service. His research interests include behavioral cybersecurity, cyber analytics, cyber workforce development and certification, and network security.

Create and Host Cyber Competitions Using Preliminary Persistent Cyber Training Environment (PCTE)

Christopher Thompson, Gabriel A. Bearden, Ty Sloan, Roy Laurens, Cliff Zou, Bruce Caulkins
University of Central Florida
Orlando, FL
cthompson@hackucf.org, <gbearden, rlaurens, tysloan>@knights.ucf.edu, bcaulkin@ist.ucf.edu, czou@cs.ucf.edu

1 INTRODUCTION

The dependency on the Internet by businesses and governments has led to a new frontier for technological innovation, intercommunication, and an unintentional stage for malicious actors. As the world becomes more interconnected and our lives increasingly automated, the threat of cybercrime has proven to be an ongoing and evolving force. Subsequently, a properly trained information security team is critical for ensuring the proper preventative layers are in place in the event of a cyber-attack. In recent years, there has been an increasing shortage of qualified security professionals going from academia into industry, this void left mostly unfilled due to lax and outdated cyber-curricula (Martini & Choo, 2014). The expansive breadth and fast pace nature of cybersecurity has made procuring this effective training material difficult. However, with the introduction of cybersecurity competitions, students are getting hands-on experience that can better prepare them for their future careers in information security as well as introduce them to the different aspects of the security field (Katsantonis, Fouliras, & Mavridis, 2017).

The Persistent Cyber Training Environment (PCTE) is a realistic cloud-based training environment created for DoD Cyber Mission Forces (CMF) ("Persistent Cyber Training"). It allows customizable skill-based and scenario-based assessments to be deployed and played by individuals concurrently. One important objective of the project is to help facilitate cybersecurity operations and training, an essential function of which is the ability to generate practical cyber-oriented competitions. PCTE's ability to be managed and customized according to a set of specifications proves effective when hosting these cybersecurity events, as such the platform is currently being used to host small to mid-sized cyber competitions. This provides unlimited ways of teaching students security-oriented skills while also optimizing the functionality, stability, and malleability of the platform. We have a funded project from the creators of PCTE to test both software and infrastructural stability based on a real-world, practical competition scenario. Thanks to this funding we have created and organized a cyber-competition event using the PCTE platform and presented at the I/ITSEC 2019 conference.

This paper will present a detailed introduction on how to create a cyber-competition scenario and virtual networking environment, as well as how to create and host a cyber-competition event. This will greatly help other cybersecurity researchers and practitioners to create their own cybersecurity competition and train and educate future cybersecurity workforce. Some of the most beneficial contributions of this paper include:

- The lessons and experiences learned from creating our event will help others avoid many pitfalls and obstacles,
- Concrete, step-by-step instructions on how multiple aspects of a pen-testing/cyber training competition are made,
- A description of the most common design conventions used in a modern cybersecurity competition, including terminology, technologies, and structures,
- Tangible competition data and observations made from a real, deployed cyber competition based on PCTE with participation by eight cyber competition teams from multiple universities, and
- Accurate reflection on how the PCTE platform can be used to help automate a cybersecurity competition.

Contained in this paper, section 2 provides the reader with a broad, birds-eye view about what encompasses a cyber-competition, what the PCTE platform is, and how it relates to the creation of cyber competitions. Section 3 goes into how PCTE has been designed to ease the process of Capture the Flag (CTF) deployment and how the competition will keep score. Section 4 showcases the application of PCTE at the 2019 I/ITSEC conference, with a capture-the-flag competition created, hosted, and deployed using PCTE resources. Towards the end of the competition we showcased

performance and score data cumulatively. Section 5 gives our own personal observations about PCTE through extensive use of the platform over the course of roughly a year. Finally, section 6 is a conclusion.

2 BACKGROUND

2.1 Cyber Competition

A cybersecurity competition is a hosted event, formal or informal, hosted online and/or in-person which allows students, enthusiasts and security professionals to compete amongst each other for prizes, notoriety, experience, and more (Katsantonis, et al., 2017). The formats of each competition can differ in their content, theme, category, and length depending on who is hosting the event and its intended audience. The most popular format of cyber competition is popularly referred to as a capture-the-flag (CTF) (McDaniel, Talvi, &. Hay). The main objective of a CTF is to acquire the most points given a security-oriented problem to solve; the team or individual with the most points usually wins. Similar to most other sports, there are generally first, second, and third place winners of these events. Rules, essential information, and any necessary procedure are conducted before the start of a game. A capture-the-flag competition possesses numerous sub-categories that dictate how a game will be played. The main formats of CTF include:

- **Jeopardy-style Format**: A team or individual must try to complete as many cybersecurity challenges as possible given a time constraint and set of categories. Challenges will usually contain problems that span a plethora of fields and skillsets, including but not limited to cryptography, forensics, reverse engineering, network security, web application security, and programming. Upon successfully solving a problem, a flag can be exchanged for several points corresponding to the difficulty of that challenge.
- **Attack-Defend Format:** Each team is allocated their own server which will possess intentionally vulnerable or misconfigured services and a unique flag. They're responsible for defending their machine's flag from the other competitors by patching these vulnerabilities, while also actively attacking another team's server and making it their own. The goal is to keep their initial servers running, prevent captured machines from being compromised, and concurrently attack the other teams. Points are determined based on defense performance as well as attack performance.
- **Mixed Format**: This is a combination of both Jeopardy-style and attack-defend which can be executed in numerous ways. While attempting to defend their machines and attack competitors, they must also complete challenges which have been integrated into competition somehow. This format requires numerous effective intercommunications of team members, vast knowledge of various security concepts, and the ability to multitask and do well under pressure. This gives challengers a unique experience that may never be the same among different competitions.

A competition will usually be put on by some entity, whether it be a company, university, or group of individuals. Each competition will present content in formats previously mentioned over the Internet or in-person. Most CTF competitions are not constrained to physical barriers, and as such give everyone with a stable Internet connection an equal opportunity to play. Some well-known competitions held in the United States include:

- **PicoCTF** ("CMU Cybersecurity Competition"): A beginner-oriented capture-the-flag competition created by Carnegie-Mellon and aimed towards middle and high-school students. Players will gain essential skills in a variety of topics through solving Jeopardy-style questions.
- **Sunshine CTF** (Hack@UCF): Hosted by HackUCF students and presented at the B-Sides Orlando Security Conference, Sunshine is a Jeopardy-style CTF that covers reverse engineering, web, forensics, cryptography, and scripting challenges. A theme is picked every year that gets incorporated into the challenges.
- **National Collegiate Cyber Defense Competition (NCCDC)** ("National Cyber League")**:** The largest college-level cyber attack-defense competition in the United States, teams from different Universities compete amongst one another to test their operational effectiveness when given a network of purposely vulnerable machines. Rather than focusing on both offensive and defensive skills, NCCDC gauges teams based on their ability to defend the integrity of their network while also performing operational/business-centric tasks.

2.2 Cyber Competition Environments

The vast amount of configuration and content that goes into a cyber competition is what makes it such a useful learning experience for those breaking into information security. While the methods of creating a CTF may differ from host to host, there have been successful implementations of automating this process through the use of dedicated platforms.

2.2.1 Persistent Cyber Training Environment (PCTE)

PCTE is a conjoined effort between the United States Military, the private sector and academia to create a centralized, cloud-based cyber-simulation platform to facilitate cybersecurity operations and education for DoD Cyber Mission Forces ("Persistent Cyber Training"). The platform integrates a suite of tools that allow you to create reusable network schematics, manage virtual machines with pre-made templates, as well as deploy one or more instances in the form of events. It also includes on-the-fly network monitoring facilities for administering an event in progress. Utilizing these tools, we created a deployed instance of the PCTE software in the form of the I/ITSEC 2019 CAC[1] event.

2.2.2 SimSpace (SimSpace's CyberRange)

SimSpace's CyberRange provides a creation, configuration, and hosting platform for fully simulated networks that is aimed towards cybersecurity training and education ("SimSpace"). It allows for the instantiation of virtual organizations with common infrastructure assets such as file servers and active directory. Additionally, it introduces a simulated Internet with generated network traffic for an extra layer of realism. These networks can be easily created using the Network Wizard, then configured manually, cloned, and deployed, allowing for a working environment to be operational in less than an hour. The content created within CyberRange can be reused as all parts are packaged individually, allowing for an event to utilize multiple components from separate events at once. While a scoring and survey system are built into this platform for guided exercises and labs, we did not use them for our competition.

3 COMPETITION VIRTUAL ENVIRONMENT CREATION

In this section, we introduce a general-purpose system for creating a virtualized cyber competition environment, including the necessary background knowledge and thought processes required. The considerations outlined below were used in conjunction with the PCTE platform to host our competition at the I/ITSEC 2019 conference. While our specific event was hosted with PCTE assets, the steps and technologies presented can be applied by educators to a variety of personal solutions such as VMware vSphere or Oracle VirtualBox.

3.1 Overview of Capture-the-Flag (CTF) Competition Environment Creation

The types of challenges present in a CTF are entirely dependent on the organization hosting it, providing unique learning resources for students and professionals alike.

Jeopardy-style problems have numerous categories of challenges requiring different areas of expertise to create. The level of difficulty is equivalent to its own weight in points. A common Jeopardy lineup could consist of cryptography, steganography, reverse engineering, forensics, web, and exploitation challenges. The major benefit of this style is that it promotes learning by doing, enabling individuals to find their niche and specialize in certain areas if they wish. This is one of the more malleable formats—a challenge can be anything from a linear set of steps to having multiple layers of abstraction, completed as a group or as an individual in some cases. As such, challenges are created in different ways depending on the category.

Attack-defend is a red-vs-blue team dynamic, a common convention in the field of information security. The goal of the blue team is to secure the network as fast as possible, as well as thwart any oncoming attacks. The red team attempts to exploit machines in order to compromise network integrity and pivot into other resources. A team can be either red, blue, or even both depending on the competition. They will have little to no knowledge about the network prior to the beginning of the competition—the services and machines that are on the network need to be discovered manually by each team. When designing problems for this format of CTF, we are looking at creating vulnerable

[1] CAC – Cyber Academy Competition

machines in ways that would pose a threat if left undiscovered. This includes making certain protocols out of date, misconfiguring Active Directory, hosting vulnerable web apps, and more.

3.2 Steps and Procedures

The process of creating a capture-the-flag competition depends on careful planning, teamwork, and an organized methodology. In the following we list in detail the steps and procedures used in creating a typical CTF competition.

3.2.1 Interpersonal Variables

a. Identify the target audience: The level of difficulty and conceptualization can be catered towards a set group of people. Students and beginners will generally know less than experienced CTF'ers.

b. Create a list of constraints: Depending on the formality of the CTF, you must cater to the environment it will be hosted at. While a student-oriented competition may have more leeway, more serious competitions may limit certain situational contexts, phrases, and technologies.

c. Teams versus Individuals: Determining how the competitors will play amongst one another can dictate some different design schemes in scoring or layout of challenges.

3.2.2 Event Material

a. Generate a theme: This makes the capture-the-flag fun and engaging. Normally a theme can be used as a narrative to support an overarching story and even supply hints for challenges. It also defines color schemes, references, etc.

b. Settle on a format: This influences the entire design phase of an event. A jeopardy-style CTF has individual challenges within categories, while an attack-defend requires a network and virtual machines; hybrids also bring new challenges. Thus, this must be established early on to keep on schedule.

c. Design the Content

i. Jeopardy-style competitions will need a certain number of challenges per category. The ways in which these challenges are made differ widely, but the two most important rules are that they must be solvable, and they must possess a flag that equates to points.

ii. Attack-defend competitions require remote-exploitation vulnerabilities and privilege-escalation vulnerabilities. These must be patchable by the blue team, and exploitable by the red team. The use of out-of-date services, vulnerable software, etc., can be used to create these loopholes.

3.2.3 Infrastructure

a. Hosting the Challenges

i. You must determine how people will be accessing your CTF challenges. If you're using the cloud, overhead from large amounts of traffic must be taken into consideration.

ii. The Amazon Web Services, commonly known as AWS, is a great cloud platform that many CTF's use.

b. Network Design and Virtual Machine (VM) Templating

i. When using VM's, designing an effective network schema is important for making the competition less convoluted and more approachable from both sides.

ii. The types of operating systems and their underlying vulnerabilities are planned out according to difficulty, tied assets, and where they exist within the network topology.

iii. Hybrid competitions can leverage both a network and individual challenges residing on virtual machines—how this is executed depends on the event goals.

c. Presenting the Scoreboard

i. This is one of the most important aspects of any CTF. You must assure that when a team has successfully submitted a flag, they get the appropriate points. It is of utmost importance to correctly configure each flag for each challenge.

ii. Using scoreboard software like CTFd is highly recommended due to its reliability.

3.3 Creating Virtual Network and Virtual Machine Environment

Within our event, the initial network map was created using the New Spec Wizard within the PCTE platform. There were three organizations representing three fictitious countries: Arstotzka, Borduria, and Franchia. Along with providing plot devices for the CTF, each country possesses a corresponding level of difficulty. All organizations possessed five servers and five workstations (room was left for future expansion). Four Red Team Kali Linux VMs

were created, and a simulated Internet was employed to connect them together. After the network creation, we removed unnecessary assets from the simulated Internet other than the central router that interconnected the organizations to the Red Team. Each country was then assigned a VM template specific to that country, with a Windows OS for the client machines and a Linux OS for the servers. Three of the Red Team VMs were then given access to one country's network each. The last remaining machine is the Red Team home platform which had SSH access to each Red Team tunnel. This host machine was cloned as needed to match the number of participants per team. (See Figure 1.)

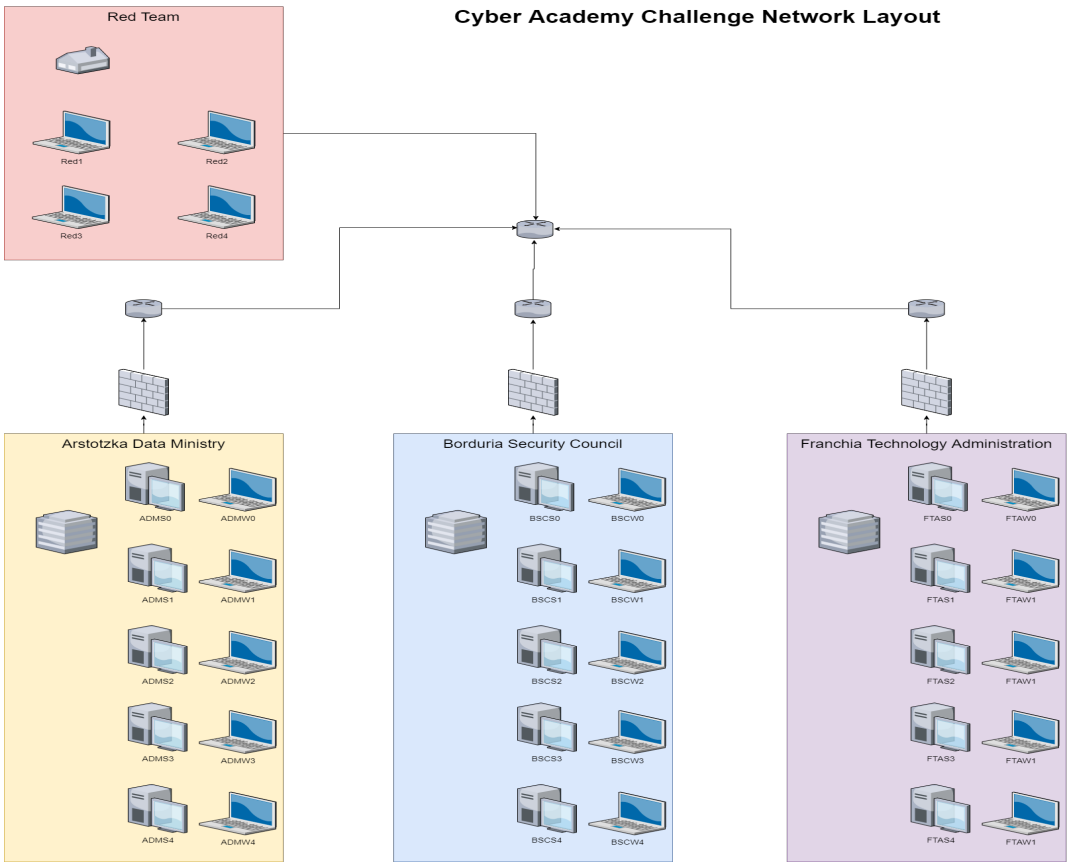


Figure 1: Network diagram of the CFT virtual network environment used in 2019 Cyber Academy Competition Event held in I/ITSEC conference

3.4 Creating Vulnerable Services and Computers

For this competition, three fictitious countries were created to represent three levels of difficulty for gaining access to their machines, as well as provide a narrative for the story. The operating systems present on the simulated networks were deemed appropriate for their technological status. Each machine had three stages: initial entry, privilege escalation if necessary, and challenge solving. The background of the three countries are as follows:

- **Arstotzka - Easy:** Modeled after a country that has very limited resources and outdated infrastructure.
- **Bordura - Medium:** Represents a country which has actively upgraded, middle of the road network of computers
- **Franchia - Hard:** Presents an incredibly durable, up-to-date network of modern operating systems that are difficult to break into

In order to access each challenge present on the virtual computers, the competition team (i.e., red team) had to first find a way into the computer through exploiting various points of entry, as shown in Figure 1. These are referred to as vulnerabilities, which are an unintended, exploitable outlet where attackers can leverage some form of control over

the machine. The act of creating vulnerabilities for initial access, privilege escalation, and even for progression in the CTF challenges themselves are dependent on multiple factors, such as technologies being used, the type of vulnerability, the level of access that can be obtained, ease of use, and other things. For the sake of our competition, rather than developing from scratch, we used exploit-DB, which is a trusted resource for samples of vulnerable software as well as the code needed to exploit this software ("Offensive Security"). We list in the following procedure how to create and test vulnerable services as well as virtual computers in our competition:

- Identify the type of vulnerability that you need. For our purposes we separated them into two categories, but this can be expanded upon if needed.
    - **Remote Access Vulnerabilities:** There are vulnerable versions of built-in protocols or applications which allow the attacker to gain a foothold on the machine. We generally excluded any remote vulnerabilities which gave the player administrative access upon exploitation.
    - **Privilege Escalation Vulnerabilities:** After gaining a presence on the competition box, if the challenge requires administrative privileges, we utilize these vulnerabilities to allow players the ability to upgrade their user to admin.
- Find the vulnerability on exploit-DB and download it onto the host VM.
    - After looking over some information about a potential vulnerability, including what it is, how it's to be present on a machine, its severity upon exploitation, and most importantly its intended operating system, we download the file onto the virtual machine and configure it if needed.
    - The most crucial step is reading over the documentation about the vulnerability. If you're not careful, it might crash your system.
- Test the vulnerability by exploiting it.
    - Once this vulnerability is running, whether it be a web server or some application in the background, a very important step is to test it. This will resolve any post-setup issues that will arrive otherwise.

Arstotzka Workstations - Windows XP: The Arstotzka workstation machines used the Windows XP operating system, which had intentional firewall misconfigurations. Due to Windows XP's inherent vulnerability to MS17-010, it provided another option for initial access. MS17-010, popularly termed Eternal Blue, allows for arbitrary code injection through specially crafted SMB packets ("Microsoft Security Bulletin"). No installations are needed for this vulnerability due to this being a service level vulnerability rather than an application vulnerability.

Arstotzka Servers - Centos 7: The Arstotzka server machines used the CentOS 7 Linux operating system, and were used primarily for web servers that hosted CTF web challenges. As such, web-application vulnerabilities were used over machine-based vulnerabilities. This was done by simply downloading vulnerable web-server programs such as Apache Syncope 2.0.7. (Kuo, 2018) from exploit-DB and making the applications launch on startup.

Borduria Workstations - Windows 7: The Borduria workstation machines used the Windows 7 operating system, and had either a remote exploitation, a privilege escalation, or both depending on the challenge. The exploits were found on a database like exploit-DB. After finding the exploits, which were usually in the form of software applications versus vulnerable services, they were then downloaded directly onto the machines. Examples of some of the exploits used include WinRAR v5.61 ("RARLAB") for privilege escalation and EasyFile Web Sharing 7.2 (Chako, 2017) with remote exploitation.

Borduria Servers - Ubuntu 14.04: The single Borduria server machine used the Ubuntu 14.04 operating system. This machine hosted a web challenge with a simpleHTTP 2.2 rc2 server which was vulnerable to a remote buffer overflow exploit ("Simple HTTP").

Franchia Workstations - Windows 10: The Franchia workstation machines used the Windows 10 operating system. All machines had remote exploitation vulnerabilities in the form of various web applications, as well as privilege escalation vulnerabilities from vulnerable software when necessary. Because the Windows 10 machines were newer and more refined, they proved relatively difficult to make vulnerable.

3.5 Scoreboard Creation

The importance and role of a scoreboard for cyber competition are to help distinguish each of the participant's total points earned by completing each of the challenges, and where each team stands in regarding the other teams. Using the scoreboard, the time it took for each team to complete challenges can also be logged and be used to generate statistical data.

We used the CTFd framework (Chung) to create our independent scoreboard by renting a server and accessing the CAC CTFd instance at *cac2019.ctfd.io/*. Using the services CTFd provided, we created and hosted challenges with a corresponding number of points and categorizing them based on the countries (Arstotzka, Borduria, and Franchia). Each challenge also had a description which gave a hint regarding how to complete the challenge.

4 HOSTING CYBER COMPETITION EVENT

4.1 Pre-Event Testing and Trial by Competition Teams

For the initial test, red team members from Metova ("Metova") and Simspace ("Simspace"), both vendors on the PCTE project, competed in an asynchronous simulated competition. The CTFd scoreboard was deployed with all the finalized challenges, and two independent events were deployed for each team. The credentials for the red team machines and the steps for accessing them were emailed out, and participants were assigned the Red Team roles in their respective events. They competed until the two-hour mark simultaneously, at which point their score was recorded and they could continue until all machines were compromised. This test and its results led to us adjusting the CTF challenge component to lean away from a rigid penetration testing angle and more towards something less linear, such as additional categories of questions.

The second test was performed with only a single red teamer from both Metova and Simspace. Two events were deployed, similar as before, but instead of running a full competition the two testers were asked to compare the content on their networks with the entries of our work tracker, a document containing the content which should be present as well as their solutions. The goal was to verify that we haven't missed any content and the solutions were valid.

Finally, a pre-qualifying event was made with a simplified structure to test the challenge process and get competitors familiar with the environment. Eight separate virtual networks were deployed with five basic Windows XP machines containing five simple steganographic challenges[2]. These deployments failed, and as we were unable to fix them, we instead focused on making those repairs to the main event and scrapped the pre-qual scoring section.

4.2 Hosting Cyber Academy Competition Event in I/ITSEC 2019 Conference

For the I/ITSEC 2019 CAC event, a copy of the network was deployed for each of the eight teams and they were given credentials for their virtual Red Team VM's, as well as a packet on the competition and network. The CTFd scoreboard was configured to launch at the competition start time. Three teams played locally at our booth at I/ITSEC 2019, and the other five teams played remotely. The competition was paused twice for an hour to allow the local teams to travel to and from the convention center. It was then concluded the next day, scores were tallied, and prizes were awarded. The eight teams that competed are shown in Table 1.

For the three local teams, denoted by an asterisk in Table 1, we had to set up a physical ethernet connection as the Wi-Fi at the convention center was non-functional. We configured a camera feed that displayed both the onsite and offsite teams, as well as presented a live scoreboard throughout the duration of the event. (See Figure 2.)

During the event, the on-site teams had repeated troubles with consistent access to the platform as the Internet for the convention center was unreliable (there were hundreds of companies doing exhibition in the center). A few challenges were also found to have the wrong answer or not be working fully—these challenges were quickly repaired and redeployed. Eventually, by the end all but a few challenges were solved, and all scores were finalized. UCF C^3 team won by a large margin, namely due to their early compromise of the Franchia machines. (See Figure 3.)

[2] Where some form of data or piece of a message is hidden inside another image, file, message, or video.

Table 1. CAC Competition Teams

Team Institution	Competition Name
*** University of Central Florida**	UCF C^3
University of Florida	KernelSanders
***University of Southern Florida**	WCSC
United States Military Academy West Point	BitsForEveryone
United States Naval Academy	JohnPwnJones
United States Coast Guard Academy	Alternative Success
United States Air Force Academy	Delograd
*** UCF Reserve Officer's Training Corps**	Cyber Knights

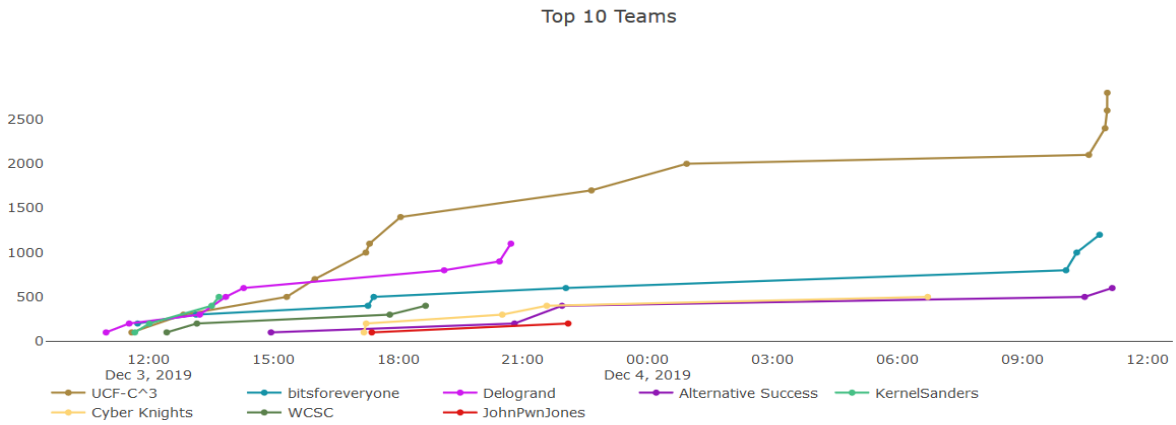
Figure 2: Two large monitors used in the competition event held at the I/ITSEC 2019 conference. The left monitor shows the real-time onsite and remote competition teams by using Zoom Meeting software; the right monitor shows the real-time scoreboard for the 8 competition teams during the event.

5 LESSONS LEARNED FROM CREATING AND HOSTING CYBER COMPETITION EVENT

5.1 Improvement on PCTE

The main thing we noticed lacking from PCTE initially was a library of vulnerable VMs and services or a straightforward way to add custom content to machines. While not strictly necessary, a vulnerability library or a way

to add vulnerabilities easily would have made event creation hassle free and less time-consuming. The unreliability of the platform at the time of event creation was also a major time factor, but the reliability of the platform has improved and is expected to improve further as development continues. The current documentation for the platform is also unclear and confusing, especially since the PCTE system works in a tiered construction system that is not well explained.



Place	Team	Score
1	UCF-C^3	2800
2	bitsforeveryone	1200
3	Delograd	1100
4	Alternative Success	600
5	KernelSanders	500
6	Cyber Knights	500
7	WCSC	400
8	JohnPwnJones	200

Figure 3: Scoreboard statistics showcasing each team and their placement

5.2 Lessons and Improvement on Hosting Cyber Competition Event

By the end of the PCTE CAC event, an abundance of observations was made on what things to improve on and what we should worry about in future event organizing. Planning for appropriate on-site network infrastructure is something that must be prepared ahead of time and is critical to the success of a competition. The on-site teams experienced multiple Wi-Fi issues due to the massive amount of radio traffic from other booths in the convention, causing us to switch from Wi-Fi to ethernet in the middle of the competition. If physical teams will be a part of your competition, be sure to provide adequate and reliable access to the Internet.

Time management is another important hurdle that you may encounter. Having a solid design and implementation process and a way of scheduling who does what will greatly expedite the creation process. Software like Asana ("Asana"), a multi-purpose team organization and task scheduling application, greatly increased the overall productivity of the project. Communication platforms like Slack ("Slack) and Discord ("Discord") facilitated team intercommunication and kept all members on the same page. This aspect of task accountability, deadlines, and

effective communication channels were an absolute necessity, especially towards the end of our challenge development.

Lastly, exploring different avenues for creating challenges is essential for making an engaging experience. There's a near endless array of content that can be presented conventionally within a competition environment or a classroom. Figuring out which concepts to utilize is one thing, however presenting them in a manner that provides a need for critical thinking as well as personal enjoyment upon getting the answer is an important part of a CTF. A good method for achieving both of those things is to think outside-the-box and not stick with mundane, unfulfilling formats. Providing a good backstory behind the competition is an excellent way to provide a feeling of immersion, such as why the participant is doing what they're doing. Color schemes, naming conventions, the customization is limited only by imagination.

6 CONCLUSION

The overarching possibility of cyberattacks impacts not just large-scale companies or well-known individuals, but also the lives of every person connected to the Internet. Collateral damage in the form of digital assets and personal information is a huge risk that requires an active, mendable, and dependable solution. One of the best ways of ensuring digital integrity is the employment of well-trained cybersecurity professionals, particularly ones that have had exposure to the field early on in their careers. One of the most effective methods of training students for these high-stakes roles is to allow them to apply their skills in cybersecurity competitions, where they apply theoretical approaches to solve practical problems. Not only does it facilitate the exchange of skills and promote teamwork, it also gives students and industry professionals alike the ability to learn and adapt to changing technologies. Using the Persistent Cyber Training Environment to conduct a real-world cyber competition provided a solid base for generating a complex range of challenges with ease. However, educators can create their personal cyber training environment using the procedures we have outlined without a specialized all-in-one platform. Creating content and challenges is a matter of research and creative thinking. Virtualization technologies like VMware or cloud hosting services like AWS give individuals the ability to host this content on vulnerable virtual machines, simulated networks, or as individual challenges. An instance of CTFd can be easily run to help keep score and provide everything needed to administer a capture-the-flag competition. While the process of generating a cyber-competition can vary, they provide an incredibly effective and enjoyable means of learning hands-on security skills at nearly any level, student or professionals alike.

ACKNOWLEDGEMENTS

This work was supported in part by NSF grants DGE-1915780. Any opinions, findings, conclusions, and recommendations in this paper are those of the authors and do not necessarily reflect the views of the funding agency.

REFERENCES

Asana. (n.d.). Manage your team's work, projects, & tasks online · Asana. Retrieved from https://asana.com/

Chako. (2017, June 28). Easy File Sharing Web Server 7.2 - Unrestricted File Upload. Retrieved from https://www.exploit-db.com/exploits/42268

Chung, K. (n.d.). CTFd : The Easiest Capture The Flag Framework. Retrieved from https://ctfd.io/about/

CMU Cybersecurity Competition. (n.d.). Retrieved from https://picoctf.com/

Discord (n.d). Retrieved from https://discord.com/

Hack@UCF. (n.d.). SunshineCTF 2020 Retrieved from https://sunshinectf.org/

Katsantonis, M., Fouliras, P., & Mavridis, I. (2017). Conceptual Analysis of Cyber Security Education based on Live Competitions. *2017 IEEE Global Engineering Education Conference (EDUCON)* .doi: 10.1109/educon.2017.7942934

Kuo, C.-C. (2018, September 13). Apache Syncope 2.0.7 - Remote Code Execution. Retrieved from https://www.exploit-db.com/exploits/45400

Martini, B., & Choo, K.R. (2014). Building the Next Generation of Cyber Security Professionals. *ECIS.*

McDaniel, L., Talvi, E., &. Hay, B., Capture the Flag as Cyber Security Introduction. *2016* Hawaii International Conference on System Sciences (HICSS), doi: 10.1109/HICSS.2016.677.

Metasploit. (2019, April 25). RARLAB WinRAR 5.61 - ACE Format Input Validation Remote Code Execution (Metasploit). Retrieved from https://www.exploit-db.com/exploits/46756
Metova Software Development. (2020, May 12). Retrieved from https://metova.com/
Microsoft Security Bulletin MS17-010 - Critical. (n.d.). Retrieved from https://docs.microsoft.com/en-us/security-updates/securitybulletins/2017/ms17-010
Mr.pr0n. (2012, July 19). Simple Web Server 2.2 rc2 - Remote Buffer Overflow. Retrieved from https://www.exploit-db.com/exploits/19937
National Cyber League. (n.d.). Retrieved from https://nationalcyberleague.org/
NCI Information Systems, Inc. (n.d.). Persistent Cyber Training Environment (PCTE). Retrieved from https://www.peostri.army.mil/persistent-cyber-training-environment-pcte
Offensive Security's Exploit Database Archive. (n.d.). Retrieved from https://www.exploit-db.com/about-exploit-db
SimSpace Corporation. (n.d.). Retrieved from https://simspace.com/
Slack. (n.d.). Retrieved from https://slack.com/