Open Access License Notice

Interactive Cyber-Physical System Hacking: Engaging Students Early Using Scalextric

Jonathan White
Computer Science Research Centre
University of the West of England
Bristol, UK
Jonathan6.white@uwe.ac.uk
0000-0002-7931-658X

Phil Legg
Computer Science Research Centre
University of the West of England
Bristol, UK
phil.legg@uwe.ac.uk
0000-0003-3460-5609

Alan Mills
Computer Science Research Centre
University of the West of England
Bristol, UK
alan.mills@uwe.ac.uk
0000-0003-4187-2270

***Abstract*—Cyber Security as an education discipline covers a variety of topics that can be challenging and complex for students who are new to the subject domain. With this in mind, it is crucial that new students are motivated by understanding both the technical aspects of computing and networking, and the real-world implications of compromising these systems. In this paper we approach this task to create an engaging outreach experience, on the concept of cyber-physical systems, using a Scalextric slot-car racetrack. In the activity, students seek to compromise the underlying computer system that is linked to the track and updates the scoreboard system, in order to inflate their own score and to sabotage their opponent. Our investigation with this technique shows high levels of engagement whilst providing an excellent platform for teaching basic concepts of enumeration, brute forcing, and privilege escalation. It also provokes discussion on how this activity relates to real-world cases of cyber-physical systems security in the sports domain and beyond.**

Keywords*—cyber security education, cyber-physical systems, outreach

I. INTRODUCTION

Since the formation of the UK Government National Cyber Security Centre (NCSC) in 2016, a growing outreach community has developed to bring together schools, industry, and academia, engaging young people with computer science and applied cyber security skills through their CyberFirst programme [1]. Across the wider West of England region, Unlock Cyber is a regional consortium, supported by the NCSC, bringing together cyber representatives from employers, professional bodies, delivery partners, education providers and academia to raise awareness of cyber security, to excite and engage young people about cyber, and to showcase the opportunities that exist for young people on local and national levels [2]. Since 2018, Unlock Cyber have hosted an annual cyber taster day at the University of the West of England, attended by approximately 300 students aged between 12-14 from schools across the West of England. The most recent event in June 2022 was supported by 12 industry workshops and reached over 20 regional schools. Students rotated around a set of 40-minute workshops throughout the day, covering a variety of cyber security topics such as Capture-The-Flag competitions, a digital forensics crime scene investigation, introduction to encryption, and a physical wiretapping exercise.

Our objective was to provide an exciting and immersive experience for students, that would both capture their attention to draw them in, whilst learning fundamental concepts of cyber security and relating these to real-world scenarios. As a result, our team developed a cyber security challenge, integrated with a Scalextric slot-car racing set (Fig. 1), where two teams compete to achieve the highest possible score, which ultimately could be achieved by compromising the scoreboard system, either to inflate their own score or to sabotage their opponent. The challenge was designed to guide students through the cyber kill chain model, starting with basic reconnaissance and moving through to exploitation, privilege escalation, and actions on objective. The environment was configured with multiple vulnerabilities ranging in complexity, allowing for deeper exploration by more experienced students, whilst also being accessible for those less experienced.

Fig. 1. Scalextric slot-car racing challenge where students can hack the scoreboard to reset the opponents score and artificially increment their own

In this paper, we detail the process of developing a cyber-physical system for cyber security education, balancing the need for an engaging environment that also encourages deeper learning. We report on initial pilot studies of running the activity, discuss lessons learnt and propose possible improvements and future work.

II. RELATED WORK

We focus our review on the current trends and techniques for cyber security education, and the pedagogic aspects of cyber security. There exists a wealth of online content for learning about cyber security, including services such as TryHackMe [3] and HackTheBox [4], where virtual machines are deployed for experimentation and practice. One major limitation with many introductory resources is the abstract nature of cyber security - students who are relatively new to the discipline may not grasp the impact of an attack on a virtual machine given this is an intangible asset. In terms of cyber-physical security, simulation training tools have also been developed, such as the Graphical Realistic Framework for Industrial Control Systems (GRFICS) [5]. This suite of tutorials progresses through a simulated attack on the Modbus networking protocol, and on Programmable Logic Controllers (PLCs), resulting in the explosion of a chemical plant. Whilst this provides an excellent learning platform for understanding various concepts, the simulated nature may still fail to capture the imagination of students compared to that of a physical setup. In [6], we set out to overcome some of the limitations of simulation by linking home IoT devices to an online CTF submission system in a way to engage students remotely, and to continue our outreach activity during the Covid-19 pandemic. By connecting via a video conferencing platform, students could complete tasks and submit flags as per traditional methods, however they could then observe physical interactions such as turning devices on and off, through to controlling a robotic vacuum or manipulating a light display, giving a sense of "hacking" the home IoT.

Pencheva *et al*. [7] held discussion groups with teachers on how cyber security can be brought into classrooms, and around student engagement when teaching cyber security. They conclude that students and parents need to be aware of the range of opportunities that cyber security careers can bring, and teachers need to be supported in how they can bring practice-based learning to the classroom in a manner that they feel confident to run the class, whilst ensuring that practical demonstrations work first time so as to not disengage students. Crick et al. [8] also highlight challenges related to university teaching of Cyber Security within the UK. Similarly to other works, many of their challenges relate to having up-to-date practice-based learning that helps to enthuse and excite students, whilst clearly highlighting the relevance of this within the wider organisational and societal issues of cyber security.

Our approach couples a hands-on practice-based exercise with multiple networked systems, including a scoreboard system, and a motion sensor to provide connectivity with the physical environment. Working in teams, students can immediately observe the consequences of their actions on the local scoreboard, thus fostering a sense of competition. Studies have shown [9] [10] that peer-based competition has a constructive effect on participation and learning that will result in higher learning through social pressure to achieve and better conceptual knowledge. Furthermore, our objective is to create a memorable, yet educational, environment that will entice students to learn more about cyber security, whilst offering a deeper level of competition and exploration for those with greater domain knowledge.

III. SYSTEM DESIGN

Fig. 2 shows the overall configuration of our system, that comprises of the following components:

- One Scalextric track with two cars
- Two infrared (IR) sensors
- One Raspberry Pi Model 3b+ to act as a combined DHCP server and lap counter
- One pi-TopCEED [11] incorporating a second Raspberry Pi Model 3b+ to display the scoreboard via a web server
- Two laptops running Kali Linux, one for each team
- One network switch to connect all devices together on a local network

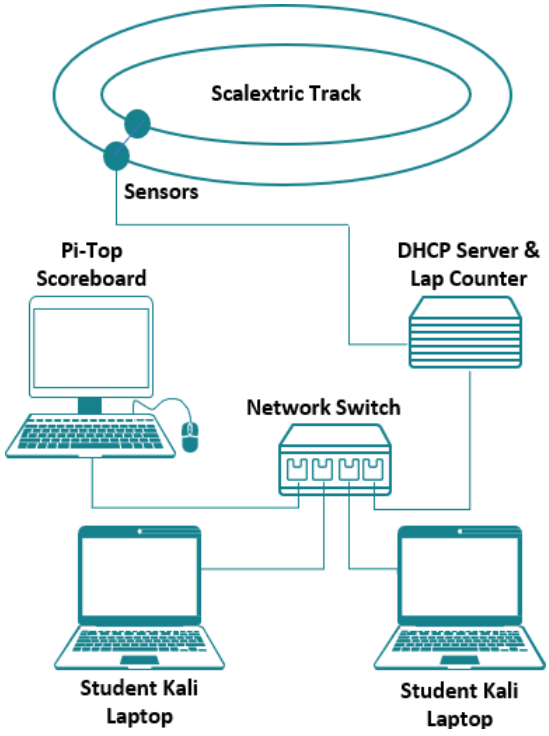

Fig. 2. Equipment setup

A. IR Sensors

The connectivity of the Scalextric to our scoreboard was achieved through two infrared sensors mount either side of the race track start position. The IR sensors used were HW-201 object detection sensors [12] which have an on-board adjustable sensing range of 2cm to 30cm. This enabled each sensors to be adjusted so that they could detect the passing car on each respective side of the track, without unintended detection of the opposing car. The sensor 3-pin connector consists of VCC, GND and OUT that allows for connection

to the Raspberry Pi GPIO. We developed a small Python script to detect the car motion and created a 3D printed housing to mount the sensors to the track (Fig. 3).mount the sensors to the track (Fig. 3).

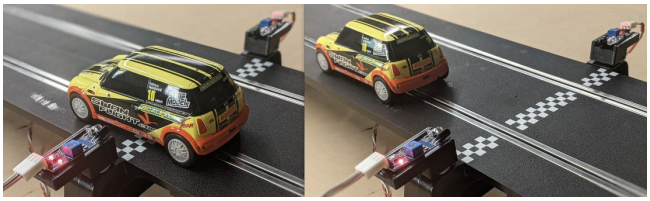

Fig. 3. Lap counter mounted to the track to detect each car passing

B. *Lap Counter and DHCP server*

The lap counter and DHCP functionality was combined on a single headless Raspberry Pi Model 3b+ to reduce the amount of equipment required. Since this device is out of scope for the student exercise, we hardened the device so that it would not contain any known vulnerabilities or easily crackable account passwords. As mentioned previously, the GPIO pins connect to the motion sensors, providing a simple means to develop a lap counter using Python. The Python script was configured to run on boot to allow for a headless Pi configuration, requiring minimal user interaction to start the activity. When the GPIO pin status changes to indicate that a car has passed, the lap counter triggers an HTTP POST request to the scoreboard web server that includes the team number so that the web server can increment the team score.

This device and the web server device were pre-configured to use static IP addresses, primarily to enable staff to easily identify machines and provide consistency between sessions. Nevertheless, a DHCP server was also used on this device so that student laptops can connect to the system and be issued a dynamic IP address. This enables multiple students to be able to connect to the system, beyond the intended two teams, where this may be useful in supporting the learning outcomes of a session.

C. *Scoreboard Web Server*

The scoreboard web server is the target device that students attack. This device is intentionally designed to have several vulnerabilities, facilitating multiple ways to compromise the service and manipulate the scoreboard. The vulnerabilities exhibit a range of complexities which allow the activity to be used for a wide age range of students with differing experience and abilities. The following services are exposed externally: HTTP, FTP, SSH, and VNC. The main HTTP web server is a Python Flask application which interfaces to a local SQLite database that stores the team names, lap count and lap times. The default /index.html homepage displays the two teams scores. Within the web server there exists additional “hidden” pages that are not linked from the home page, containing additional information that aid students in compromising the system: /admin, /test and /robots.txt. Various Flask endpoints are defined that support GET and POST methods to create, show, update and delete teams, and display and add laps. Listing 1 shows the code extract that receives the lap update and updates the local database.

```
# Endpoint to add lap
@app.route("/team/<id>/lap", methods=["POST"])
def lap_update(id):
    now = time.time()
    team = Team.query.get(id)
    lap = team.lapcount
    if lap != 0:
         then = team.time
         laptime = now - then
         laptime2 = round(laptime,2)
         new_lap = Lap(team.teamname, laptime2)
         db.session.add(new_lap)

    lap += 1
    team.lapcount = lap
    team.time = now

db.session.commit()
return team_schema.jsonify(team)
```

Listing 1. Python example

D. *Laptops*

Each team received a laptop with Kali OS installed. This allows easy access to common penetration tools such as:

- NMap - Service Discovery and Reconnaissance
- DIRB - Web Directory and File Enumeration
- Hashcat - Password Hash Cracking
- rockyou.txt - Text file with 14 million common passwords

Such tools are required for the students to progress through the challenge, along with common Linux utilities such as curl, FTP and Python. By giving the students a Kali laptop rather than virtual machine setup complexity is reduced and limits use case situations where students incorrectly interact with the host OS.

E. *Worksheets*

Different worksheets have been designed with the target age range and experience in mind. Students that have no Linux or cyber security experience can receive worksheets that explain the process in detail and give command walkthroughs. Those who were more confident can use worksheets as a basic steer without relying on the hints, whilst those who were more unsure could make full use of the hints, suggested tools, and example commands, filling in the gaps where required.

TABLE I. EXAMPLE ENTRY – TOOLS WORKSHEET

Name	**Examples**	**Description**	**Useful for**
Hashcat	hashcat –a 0 –m **mode hash-file.txt** /usr/share/wordlists/rockyou.txt –force	Hash cracker	Uses word lists to crack a hash. You need to get the right mode (hashcat–help). The hash needs to be in a txt file

In our activity, each team was provided with two worksheets. One that had a series of questions, the second, a series of useful tools and some hints on their usage and use case. The target age range for the cyber taster day was 12-14 years old so their previous cyber experience was low, and since the time available was short, the worksheets had a guided structure. For example, the first question was "What is your IP address?" and the first tool listed was `ifconfig`. This was done so that students could sequentially work through the questions and associated tools in tandem. Table I is an example entry of one of the more complex tools and commands, hashcat. Students are given a brief description of hashcat, its use case and a hint about how to find one of the missing parameters (the mode). Students working at a more advanced level would be able to fill in the missing sections themselves.

The questions themselves were designed to guide the students through the activity and the logical sequence of actions; from getting their own IP, mapping the network, enumeration etc. Each question also had an associated hint that would point them to the second worksheet which listed the tools or a particular action they might be able to take, such as using a web page on the server to reset an opposing team's score.

Students were split into two teams of 3-5 students and allocated one Kali laptop each which they could use to perform all activities.

IV. ACTIVITY

The premise of the activity itself was relatively simple. Students were told that they will be split into two teams who are competing to beat the other team's score. Students were informed that they can increase their score using the cars and race track itself and given a rundown of the potential actions they could take through cyber engagement, such as increasing their own score or reducing their opponent team's score. After a brief demonstration of the sensors and scoreboard functionality by having the cars complete a lap of the racetrack, students were split into competing teams.

The exercise was designed to walk students through the common steps of any CTF. They were given no starting information about the network other than to know that all devices required were connected to the same network. From there (and with the help of the worksheets and / or supervisors) they would work through initial reconnaissance, first by finding their own IP address and then by mapping all active IPs (and their associated services) on the network using NMap.

Once the web server was located, the students moved onto enumeration of the scoreboard page using DIRB. This would highlight four accessible pages:

- admin - Where they could easily reset the other teams score
- test - A "debugging" curl command that can be used to manually increase their score (by 1)
- teams - The target URI for increasing their own score
- console - The Python Werkzeug console, this could be used by teams to achieve a root reverse shell

Teams were encouraged to visit each of these pages to explore their options, with the 'admin' page being the first suggestion followed by the 'test' page. Once they had reset the opposing teams score and manually increased their own, they were encouraged to revisit some of the other available services on the server. Using FTP students were able to access a password hash in the download directory, which when cracked using hashcat, would allow SSH access to the web server.

With terminal access, students can explore the user's home directory and find a .txt file that contains a Python script which automates the process of increasing the team's score. For many teams this was the final stage of the exercise.

It is also possible for teams to go through the system logs on the web server and identify a debugging pin that would give them access to the console page. From there a simple Python reverse shell would give root access to the web server where further actions on can be conducted, such as shutting down the web server at a point where one team had more points than the other, effectively declaring themselves the winners. Other possible exploit avenues include using a Metasploit exploit against the Werkzeug console or analysing the local network traffic using Wireshark to determine the format of the POST request so that a manual curl command can be constructed to increment the lap count. This illustrates just some of the various methods that students may be able to uncover to gain access to the system.

A. Engagement

Initial engagement would often be focused on the physical aspects of the setup (the Scalextric cars and racetrack), often with one or two students taking the lead and working through the provided cyber activity sheets, with assistance when required from those supervising the teams. By the end of the event usually 80-100% of the students were

engaged in the cyber activities and the slot-cars became irrelevant.

A common factor was an increase in cyber engagement when a team was able to reset the opposing teams score to 0. Even if teams had been slower to engage with the cyber aspects of the exercise and setup this would quickly draw attention by having a "real world impact", and in some cases teams would move away from physically racing the cars at this point as it was now deemed as "pointless".

This stage was often a key moment in the team's engagement, when they realise that physical movement is no longer required to trigger the update of the scoreboard. Later actions such as increasing their own team's score, either through manual or scripted curl commands were observed as key milestones in how the students perceived the activity. Yet, a common question was how to script or automate the ability to reduce the other team's score. This reveals a strong motivation as to why the student may choose to proceed with the activity, so not only to improve their own position, but ensuring that the other team is unable to do so.

This was not universally the case however, in one or two instances, students from within the groups would gravitate to and stick with the physical engagement (racing the cars on the track) regardless of the impact this had overall on the scoreboard. In this way, it was possible to observe students who had grasped the purpose of the activity and had a greater understanding of, and interest in, the cyber security aspects of the exercise.

V. DISCUSSION

The primary motivation for this project was to develop an engaging, hands-on activity that actively involves students in the learning process and to continue our outreach work with regional schools. Such an activity enables the development of a pipeline of enthusiastic young students who may be encouraged to follow a career path in cyber security. By bringing cyber security education into secondary schools, we can inspire students to pursue further study in this subject area, helping to address the skills gap.

Further work would seek to examine the psychological motivators for students whilst engaging with the activity and our initial feedback suggests that students particularly enjoyed the Scalextric activity during the outreach workshop day. One teacher commented that they "had to literally tear the last group away from the Scalextric activity as they were glued to it.". Another school stated "The pupils were totally engaged and couldn't stop talking about it on the way home."; suggesting that engagement was particularly successful for this kind of immersive and captivating activity. The event had a wide impact, with one school commenting "It was really impressive for the students to experience so many different events in such an inspiring space. They got an awful lot out of it and I hear from their teachers they are still talking about it today. It has also inspired us to think about how we can develop our curriculum."; showing that with support, we can enable teachers to deploy and develop their own activities.

One of the main challenges of this proposed approach that we recognise is scalability. Each set of Scalextric kit can only comfortably support a maximum of 8 to 10 students at a time and requires a reasonably significant sized footprint to set up (approximately 2m x 1.5m). To support an averaged size class, three sets of kit would be required, and each set of kit needs at least two student ambassadors or staff members to support the students. Significant effort has been made in the development of the two Raspberry Pi images to remove the need for any configuration, which should allow the activity to be run by supporting teachers who may not be familiar with the equipment. If the students cause any permanent damage to the web server device, then a new SD Card can be quickly burnt from the provided image and the Raspberry Pi rebooted back into the default configuration. The environment has also been taken to schools and used in after-school clubs. In supporting schools and teachers by providing appropriate training to upskill and enable them to potentially deploy these activities themselves, teachers can reach a wider audience, providing inspiration to students and exploring further discussion and learning opportunities.

VI. CONCLUSIONS AND FUTURE WORK

In this paper we have demonstrated an approach for a self-contained active learning environment to create a novel teaching experience for students. We combine a physical Scalextric circuit as an initial hook to engage students, with a small local network containing two Raspberry Pis and two Kali laptops. Students work through various aspects of the cyber kill chain to compromise the scoreboard and modify the scores. Early feedback from students and teachers suggests that this is an engaging and exciting outreach activity for students, and the packaged activity enables teachers, with support, to upskill, own and run their own similar sessions. Students have shown creativity in their attacks against the system, with some teams performing unexpected attacks such as performing a denial-of-service attack against the web server once their team were in front, or even using physical attacks against the infrastructure by ripping pieces of paper from the worksheet and blocking the opposing team's sensor.

Future work will explore the containerisation of the application using Docker so that the environment can be reset without the need to reimage the SD card. By restarting the container, the environment will be reset to its default configuration, thus reducing the maintenance complexities and overhead of having to burn backup SD Cards. Furthermore, we will explore how other cyber physical systems could be used as an engagement mechanism in a similar fashion, such as Industrial Control Systems; and how the system can include actuation control from the Raspberry Pi devices, as well as the motion sensing capabilities, allowing for a greater range of parameters that could be manipulated programmatically.

The Raspberry Pi images and worksheets have been made available at http://go.uwe.ac.uk/scalextric.

REFERENCES

[1] "CyberFirst Schools / Colleges." [Online]. Available: https://www.ncsc.gov.uk/cyberfirst/cyberfirst-schools

[2] "Mission." [Online]. Available: https://www.unlockcyber.com/mission/

[3] "TryHackMe." [Online]. Available: https://tryhackme.com/

[4] "HackTheBox." [Online]. Available: https://www.hackthebox.com/

[5] "GRFICSv2 - Graphical Realism Framework for Industrial Control Simulation." [Online]. Available: https://github.com/djformby/GRFICS/

[6] P. Legg, T. Higgs, P. Spruhan, J. White, and I. Johnson, "'Hacking an IoT Home': New opportunities for cyber security education combining remote learning with cyber-physical systems," in *2021 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)*. Dublin, Ireland: IEEE, Jun. 2021, pp. 1–4. [Online]. Available: https://doi.org/10.1109/CyberSA52016.2021.9478251

[7] D. Pencheva, J. Hallett, and A. Rashid, "Bringing cyber to school: Integrating cybersecurity into secondary school education," *IEEE Security & Privacy*, vol. 18, no. 2, pp. 68–74, 2020. [Online]. Available: https://doi.org/10.1109/MSEC.2020.2969409

[8] T. Crick, J. H. Davenport, P. Hanna, A. Irons, and T. Prickett, "Overcoming the challenges of teaching cybersecurity in uk computer science degree programmes," in *2020 IEEE Frontiers in Education Conference (FIE)*, 2020, pp. 1–9. [Online]. Available: https://doi.org/10.1109/FIE44824.2020.9274033

[9] C.-H. Chen, "The impacts of peer competition-based science gameplay on conceptual knowledge, intrinsic motivation, and learning behavioral patterns," *Educational Technology Research and Development*, vol. 67, no. 1, pp. 179–198, Feb. 2019. [Online]. Available: https://doi.org/10.1007/s11423-018-9635-5

[10] J. C. Burguillo, "Using game theory and competition-based learning to stimulate student motivation and performance," *Computers & education*, vol. 55, no. 2, pp. 566–575, 2010. [Online]. Available: https://doi.org/10.1016/j.compedu.2010.02.018

[11] "Pi Top CEED computer for the Raspberry Pi" [Online] Available: https://uk.pi-supply.com/products/pi-top-ceed

[12] "Detecting Objects Using the IR Obstacle Sensor" [Online} Available: https://arduinointro.com/projects/detecting-objects-using-the-infrared-ir-obstacle-sensor