Teaching Offensive and Defensive Cyber Security in Schools using a Raspberry Pi Cyber Range

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

Ian Johnson
Computer Science Research Centre
University of the West of England
Bristol, UK
Ian.Johnson@uwe.ac.uk
0000-0002-1383-1079

***Abstract*—Computer Science as a subject is now appearing in more school curricula for GCSE and A level, with a growing demand for cyber security to be embedded within this teaching. Yet, teachers face challenges with limited time and resource for preparing practical materials to effectively convey the subject matter. We hosted a series of workshops designed to understand the challenges that teachers face in delivering cyber security education. We then worked with teachers to co-create practical learning resources that could be further developed as tailored lesson plans, as required for their students. In this paper, we report on the challenges highlighted by teachers, and we present a portable and isolated infrastructure for teaching the basics of offensive and defensive cyber security, as a co-created activity based on the teacher workshops. Whilst we present an example case study for red and blue team student engagement, we also reflect on the wide scope of topics and tools that students would be exposed to through this activity, and how this platform could then be generalised for further cyber security teaching.**

Keywords—Cyber Security Education, Raspberry Pi, CyberRange

I. INTRODUCTION

Cyber security education as embedded within Computer Science programmes continues to pose a challenging discipline for many schools. Teachers require the confidence and domain knowledge to educate on the topics related to cyber security, whilst also aligning to the imposed curriculum standards, and also providing engaging and exciting opportunities for learners. Whilst Universities offer a variety of courses on Cyber Security and Computer Science based on their independent judgement and the domain expertise of their academic staff, school teachers are mandated by national curriculum and exam boards. There is a tension of how school teachers can develop practice-based learning to educate and inspire students, whilst also covering fundamental knowledge as set out in curriculum standards to help students achieve their full potential in GCSE and A level examinations, and doing so in a way that does not deter students away from further study in the subject.

Working with a number of schools across the West of England region, we developed a skills workshop on cyber security that was supported by the UK National Cyber Security Centre (NCSC). This workshop set out to achieve the following objectives: 1) To help upskill teachers in their cyber security knowledge, 2) To develop their confidence in understanding and debating cyber security issues, and 3) To help them develop practice-based learning that aligns with curriculum needs and that would excite and inspire their own students. Key challenges highlighted by the schools included: 1) Constraints imposed by the school IT systems, 2) Time available in the teaching schedule for practice-based learning, and 3) Time available to them for developing practice-based resources. Whilst we could direct teachers to the many excellent resources that are available online, there were reported cases where this would not suffice within the school IT environment (e.g., services such as TryHackMe are blocked within schools simply because of the word 'hack'). Teachers expressed interest in existing kit that they had available within their schools, including Raspberry Pi devices, however did not feel they had the appropriate knowledge for how to utilise this equipment effectively. As a result, we worked to create an isolated computing infrastructure that could be used for a range of cyber security education activities, that we refer to as the Pi Lab (Fig. 1). As a series of networked machines that are isolated from broader networks (e.g., school, Internet), this provides a safe and controlled environment for students to operate, where they can defend and attack systems to develop their knowledge and understanding of cyber security. Built upon the Kali linux environment, this platform can be easily extended through further development to suit the teacher's needs, whilst also being pre-configured with a variety of applications and scenarios to inspire students and help teachers build their confidence on practical teaching of cyber security.

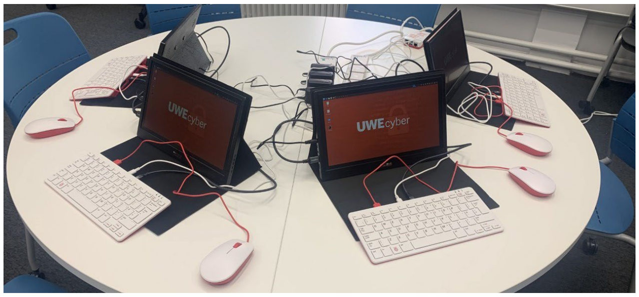


Fig. 1. UWEcyber Pi Lab portable setup, consisting of 4 Raspberry Pi 400 devices with portable monitors and a Raspberry Pi 4 access point.

II. RELATED WORKS

In recent years, there have been a number of initiatives for developing and sharing cyber security education resources online. In regard to virtualised systems, services such as VulnHub have provided virtual machine images for download for many years. Whilst this is a fantastic resource, it requires knowledge and time to utilise and deploy these systems. TryHackMe has helped by taking a similar approach, however integrating this with cloud services so that virtual machines are deployed within the browser. Furthermore, services such as their AttackBox mean that the client machine can also be accessed in the browser, making this much easier for users to get started and utilise. Whilst we utilise TryHackMe, and we promote its use, some schools have experienced issues with their IT teams for being able to deploy and connect to the AttackBox, making this a challenging hurdle for teachers to overcome. Hence, our approach is one of avoiding any reliance on Internet connectivity and existing school computing resources. This also helps to segregate between our resources used for teaching cyber security, versus school infrastructure that also needs to be utilised for other subject areas.

In [1], the authors present what they describe as a 'Cyber Range' using Raspberry Pi. However, more specifically is that they use a Raspberry Pi 3 cluster to host Docker container applications. They also then demonstrate the concept using the DVWA (Damn Vulnerable Web Application). This provides a single application, or set of applications, that all students can then utilise centrally. Whilst this offers an excellent resource, it does require some configuration and setup by the teachers, as well as requiring an Internet connection or access point for communication between devices. In contrast, our approach is intended to work out-of-the-box, with zero configuration beyond the creation of the SD card media. Previously, the Raspberry Pi setup was for the creation of a Docker cluster for centralised applications, however our approach is decentralised across the Pi devices, that are also utilised by students. In this way, students can not only access resources on the local Pi network, but furthermore, they can learn offensive and defensive techniques for defending their own assets, including web applications and sites that are hosted on their own device.

In [2], we previously presented an online approach for delivering an engaging student experience for cyber security education, on the theme of controlling IoT devices remotely. As a contingency outreach activity during the covid-19 pandemic, we linked IoT device controllers to a flag submission web service. Through structured activities, students could then "hack" devices by submitting flags and observing their remote operation via video conferencing.

In [3], factors relating to the choice of post-16 study of computer science and related disciplines are considered. In the UK, key subject decisions are made at 14 years old that determine which GCSEs a student will study for, as well as at 16 years old where students will decide whether to pursue Advanced Levels (A-levels), or alternative study routes including the more recent technical levels. Whilst post-16 study is key for whether students decide to pursue higher education such as University, GCSE is a pivotal moment and therefore it is vital that how the subject is taught is indicative of what knowledge, skills, and potential career options it may present to a student in the future.

Pencheva *et al.* [4] held discussion groups with teachers to identify key challenges in how cyber security is to be brought into classrooms, and around student engagement. Students and parents need to be aware of cyber security careers opportunities, and teachers need to be supported in how they can bring practice-based learning to the classroom in a manner that they feel comfortable teaching that inspires students. Swire describes a pedagogic cybersecurity framework [5] for teaching the organizational, legal, and international aspects of cybersecurity. This approach extends the traditional 7-layer OSI model to account for organizational issues (layer 8), government (layer 9), and international (layer 10). The proposal supports a greater connection between the technological issues, and their relevance in the wider societial context.

Crick *et al.* [6] highlight many of the challenges related to University teaching of Cyber Security within the UK. Whilst they discuss technical aspects relating to Computer Science, they also consider the development of a wider spectrum of skills that align with the role of a Chief Information Security Officer. As also observed by other works in this review, 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.

Karjalainen *et al.* [7] study the pedagogical aspects of cyber security exercises from the perspective of behaviorist, cognitivist and constructivist design principles. This ties in closely with other works, and draws attention on the initial learning and presentation, the ability to understand and assimilate, and being able to develop and build new habits through experience. Planning, implementation and feedback of exercises is crucial to support effective learning throughout the process. Workman *et al.* [8] study the effectiveness of various learning styles for cyber security education. They compare traditional classroom and lab instruction to simulation, live competitions, and a combination of these two, using a Present-Test-Practice-Assess model of learning. They found that simulation improved learning performance over traditional classroom and lab instruction alone, whereas live competition did not. The greatest improvement in learning performance came from a combination of simulation and live competition.

Our literature review further evidences that cyber security education does not fit well with traditional methods of learning, and that practice-based learning is fundamental to develop both the knowledge, and the skill of how to apply this knowledge, in a way that can help further understanding, whilst also further enthusiasm and engagement so as to build the next generation of cyber security professionals. However, the literature accounts do also reinforce the pressure and challenges faced by educators in developing their own knowledge and understanding of cyber security issues, and how this in turn is used to construct practice-based examples to support their teaching. School teachers often do not have the time, resource, or capability, to develop such resources personally which therefore can restrict their offering for students.

III. TEACHER WORKSHOP

Our early investigation started through a teacher's workshop that we hosted at the University, where we offered teacher's the opportunity to attend in-person workshop activities between January and March 2022. The workshop was centred around three core objectives: 1) Firstly to understand the challenges that current teaching staff face within secondary schools for delivering cyber security education, 2) Secondly to provide practical examples and demonstrations of cyber security activities that could be adapted for delivery in their classrooms, and 3) Thirdly, to support teachers in developing their own practical teaching experiences for improving cyber security education in secondary schools.

Firstly, teachers flagged a number of issues in the initial workshop. The challenges include the need to make lessons fun and interactive whilst also aligning to the rigid and overwhelming curriculum. Teachers are keen to inspire students, however in some cases the syllabus expects students to be able to *describe* various concepts, including SQL injection, however do not ask to do practical examples to help understand these concepts fully. Where the syllabus covers a number of *describe* topics, and when considering the limited time available that teachers have students for in the school timetable, teachers feel that they do not have sufficient time to work through complicated examples. Another challenge identified was with school computer networks, that are so heavily restricted in some cases that students are unable to access many "cyber security" resources that may talk about hacking, as an example key word that is often blocked. Even then, installing applications are heavily restricted, and likewise, networking ports and services are often blocked that even prohibit services such as TryHackMe from accessing the remote AttackBox service that they offer. Finally, some teachers still report varying levels of confidence around delivery of cyber security, and their understanding of technical concepts. It is likely that teachers have learnt what they need to know for the curriculum, rather than having the time to fully develop their knowledge in a practical sense, which therefore in reflected in how students are taught about such concepts.

Secondly, we showcased activities including TryHackMe with teachers to illustrate the potential of these platforms. Whilst teachers appreciated these resources for personal learning, they would not necessarily be usable in schools simply because of the word 'hack' being blocked by their networks.

As a result, we demonstrated how a Raspberry Pi cyber lab could be deployed to alleviate against the challenges identified by teachers. Furthermore, this provides a highly adaptive and flexible platform that teachers could extend as their confidence in delivery increases. Initial feedback with schools and related stakeholders found the concept useful for introducing practical-based cyber security. There would be a need to tailor the delivery based on the time available, the prior knowledge, and the age group in question. We will walk through a possible scenario of how the environment can be used over a session in Section IV-B.

IV. RASPBERRY PI LAB – CYBER RANGE

The Raspberry Pi is a low-cost single-board computer that was first launched in 2012. Over the last 10 years, the device has evolved to increase the computing capabilities whilst maintaining a low cost. The Raspberry Pi 4 device was released in 2019 and features a 1.5 GHz 64-bit quad core ARM Cortex-A72 processor, on-board 802.11ac Wi-Fi, Bluetooth 5, full gigabit Ethernet, two USB 2.0 ports, two USB 3.0 ports, between 1–8 GB of RAM (depending on model), and dual-monitor support via a pair of micro HDMI (HDMI Type D) ports for up to 4K resolution. It can run a variety of Unix-based operating systems, including Ubuntu, Debian, and Kali Linux, by imaging an SD card with the required OS.

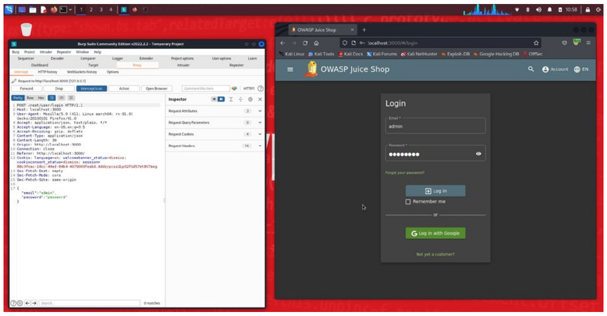
We instrumented an OS image that can be downloaded and used to create an SD card for use with the Raspberry Pi to create an "out-of-the-box" cyber range. The approach removes requirements on the school IT infrastructure, and once the image is downloaded, it would not require any further Internet connection as the system is designed to be fully isolated, thus also providing a safe environment for student's to use. Whilst this approach is designed to be highly flexible to a variety of use cases, we will describe an example process that makes use of three Raspberry Pi devices, and can be scaled up based on the number of seats required in the classroom.

We use one Pi to act an access point, for which all remaining Pis will automatically connect to, based on the pre-configured SSID network name. This is achieved by running a predefined script that is available on the device. We do also offer an image with this command already triggered for further simplicity - however, the script approach means that only a single image need be downloaded where bandwidth and download speeds are limited. We use the Raspberry Pi edition of Kali Linux as the base image for our UWEcyber client image. We have customised this with additional software such as Burp Suite Community Edition (which Kali Linux for Raspberry Pi does not typically ship with), as well as Docker for the deployment of containerised instances of OWASP Juice Shop, DVWA, and CTFd. By connecting to services such as DockerHub as part of a configuration stage, the containerisation approach easily allows additional applications to be deployed on the device should a teacher wish to do so.

A. Activity 1: OWASP Juice Shop

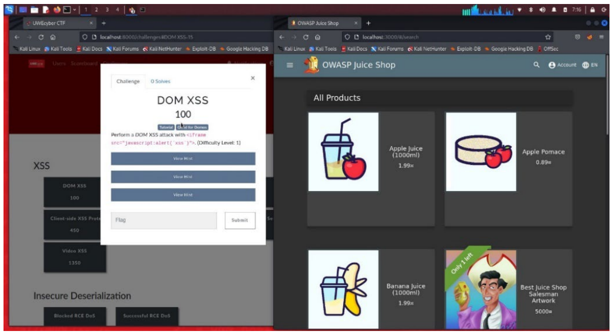
OWASP Juice Shop is a modern insecure e-commerce web application, that is used for security training and

Capture-The-Flag (CTF) activities. It incorporates all vulnerabilities detailed in the OWASP Top Ten along with many other security flaws found in real-world applications that cover varying difficulty levels, making it well-suited for both beginners and advanced users alike. Fig. 2 shows the UWEcyber Pi Lab with the OWASP Juice Shop, along with the use of Burp Suite for brute force password attacks.



```
POST /rest/user/login HTTP/1.1
Host: localhost:3000
User-Agent: Mozilla/5.0 (X11; Linux aarch64; rv:91.0)
Gecko/20100101 Firefox/91.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/json
Content-Length: 39
Origin: http://localhost:3000
Connection: close
Referer: http://localhost:3000/
Cookie: language=en; welcomebanner_status=dismiss;
cookieconsent_status=dismiss; session=
80c3feac-18cc-44ed-84b4-4079003feab4.Addzypcsz2Lp02fGd57etBX7aeg
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin

{
  "email":"admin",
  "password":"password"
}
```

Fig. 2. UWEcyber Pi Lab with Burp Suite (left) and OWASP Juice Shop (right). The screenshot demonstrates the use of Burp Suite to perform a web-based Brute Force password attack on the account login page.



DOM XSS

100

Perform a *DOM* XSS attack with `<iframe src="javascript:alert(`xss`)">`. (Difficulty Level: 1)

Fig. 3. UWEcyber Pi Lab with CTFd (left) and OWASP Juice Shop (right). The screenshot illustrates a typical CTF question to perform a DOM XSS attack on the Juice Shop web page. As an early question for learners, hints and guidance for completing the task can also be provided.

Our image also comes pre-configured with CTFd, a Capture-The-Flag platform for hosting competitions. Students would typically complete tasks to obtain a unique string (known as a flag), that when entered into the CTF platform awards the student with points against the specified task. It provides an interactive front-end and scoring view, so that students can compete in real-time to achieve the most points. Fig. 3 shows the UWEcyber Pi Lab with the OWASP Juice Shop, along with the CTFd platform showing a DOM XSS (Document Object Model Cross Site Scripting) challenge, for code injection into the search text entry. In this example, a command is provided for the student to use, and hints are also available for further help, meaning that students can begin on easy challenges and progress through to more difficult challenges.

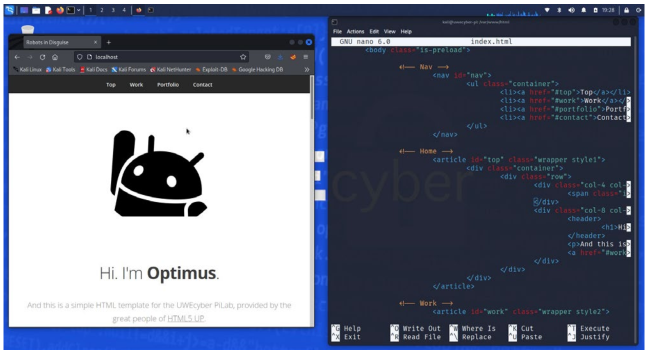
Whilst OWASP Juice Shop is offered on other training platforms such as TryHackMe (as well as being a standalone application that can easily be downloaded), with our approach, students can utilise this offline and without reliance on school networking, or the concern of installing additional tools such as Burp Suite or the Kali Linux environment. Some schools were unable to access TryHackMe due to the name of the website. The addition of the CTFd platform means that teachers can deploy a Capture-The-Flag activity for their students with minimal effort, whilst also giving them the ability to extend this activity in their own custom way should they wish. The use of the Raspberry Pi Labs means that all students can access a single CTFd instance, to submit flags and observe real-time changes to the scoreboard, incentivising their activity to achieve the highest score and to help maximise their learning.

B. *Activity 2: Red / Blue Team*

As seen previously, the device can provide containerised deployment of existing services such as OWASP Juice Shop and CTFd, however where online platforms such as TryHackMe are available this may only replicate functionality. In this next section, we demonstrate the unique capability that our approach provides compared against typical online platforms and services, that lends itself to physical equipment that students have a sense of ownership of, and therefore a desire to protect. We will walk through a simple red/blue team exercise that could be conducted with a class using the Raspberry Pi Labs. In this, students will learn about a variety of tools and their usage, but they will also learn about defensive and offensive security in the context of hardware that they each have in front of them. Cloud computing often abstracts the nature of a physical computing device, as does accessing a web server remotely, however in this example students work directly with the networked Pi that is assigned to their group. As we walk through, we will focus on the topics covered and the structuring of the task as a lesson, where additional learning opportunities are presented for students to think about and discuss. The activity would be designed to work as a practical-based table top exercise, where red and blue teams are open in the discussion so that all members of the class can understand and learn.

By default, all devices connect to the pre-configured network access point. With this in mind, rather than interacting only with a centralised application, students can actually interact directly between devices. Given the disposable nature of creating a new copy of the SD card image, students can therefore modify and manipulate these within the isolated network, where at the end of the session it is easy to revert each device back to its original stage. We present the following case study as one possible activity that students can be guided through with the aid of a teacher. For this, students can work in groups or individually, where we consider an offensive red team, and a defensive blue team.

Step 1: Blue - Create the webpage resource: In the first instance, the blue team are tasked with defending their system from attackers. For this to have some value, there needs to be an asset and some reason for the blue team to defend. Each Raspberry Pi is pre-configured with a web server running a default web page. The teacher can guide the blue team to develop their own web page (as shown in Fig. 4), which is often covered in the wider Computer Science curriculum. This web page would be accessible to all devices within the network by navigating to http://192.168.99.XXX (the IP address of the designated blue team device). In completing this stage, students learn the basics of web development and aspects of HTML coding. Students could of course develop more sophisticated webpages at this stage, however for the purpose of our demonstration we will work with a simple webpage. The core objective is to have the students create something that has some intrinsic value to them, given that they have spent time and effort on modifying the template with their own text and imagery.

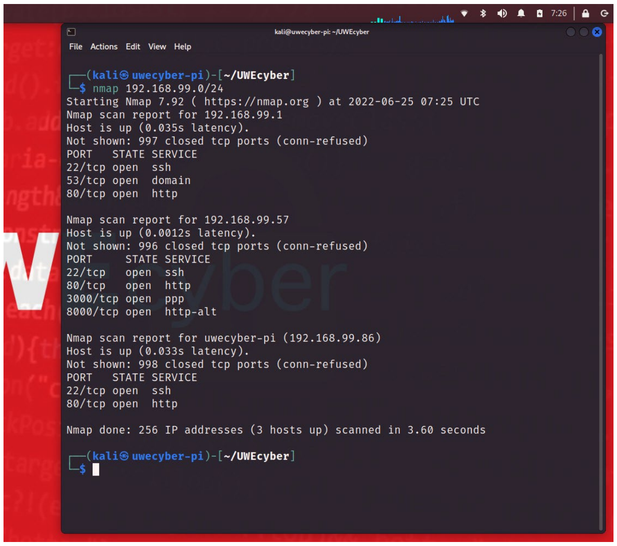


Hi. I'm **Optimus**.

And this is a simple HTML template for the UWEcyber PiLab, provided by the great people of HTML5 UP.

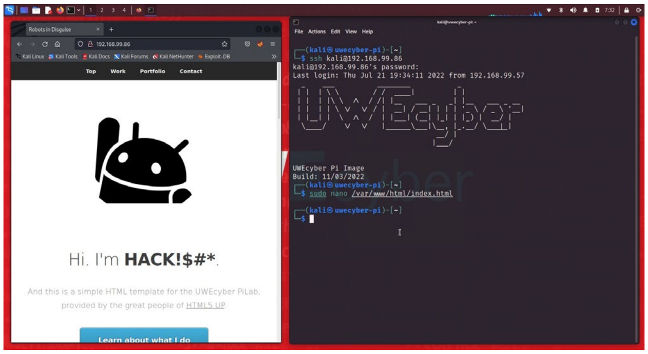
```
GNU nano 6.0                    index.html
    <body class="is-preload">

        <!—— Nav ——>
            <nav id="nav">
                <ul class="container">
                    <li><a href="#top">Top</a></li>
                    <li><a href="#work">Work</a></>
                    <li><a href="#portfolio">Portf>
                    <li><a href="#contact">Contact>
                </ul>
            </nav>

        <!—— Home ——>
            <article id="top" class="wrapper style1">
                <div class="container">
                    <div class="row">
                        <div class="col-4 col->
                            <span class="i>
                        </div>
                        <div class="col-8 col->
                            <header>
                                <h1>Hi>
                            </header>
                            <p>And this is>
                            <a href="#work>
                        </div>
                    </div>
                </div>
            </article>

        <!—— Work ——>
            <article id="work" class="wrapper style2">

^G Help      ^O Write Out  ^W Where Is  ^K Cut      ^T Execute
^X Exit      ^R Read File  ^\ Replace   ^U Paste    ^J Justify
```

Fig. 4. Blue team create a simple web page on their device.



```
┌──(kali㉿uwecyber-pi)-[~/UWEcyber]
└─$ nmap 192.168.99.0/24
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-25 07:25 UTC
Nmap scan report for 192.168.99.1
Host is up (0.035s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT    STATE SERVICE
22/tcp  open  ssh
53/tcp  open  domain
80/tcp  open  http

Nmap scan report for 192.168.99.57
Host is up (0.0012s latency).
Not shown: 996 closed tcp ports (conn-refused)
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
3000/tcp open  ppp
8000/tcp open  http-alt

Nmap scan report for uwecyber-pi (192.168.99.86)
Host is up (0.033s latency).
Not shown: 998 closed tcp ports (conn-refused)
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap done: 256 IP addresses (3 hosts up) scanned in 3.60 seconds

┌──(kali㉿uwecyber-pi)-[~/UWEcyber]
└─$ 
```

Fig. 5. Red team scans the network and finds the target machine and their running services and port numbers.



Hi. I'm **HACK!$#***.

And this is a simple HTML template for the UWEcyber PiLab, provided by the great people of HTML5 UP.

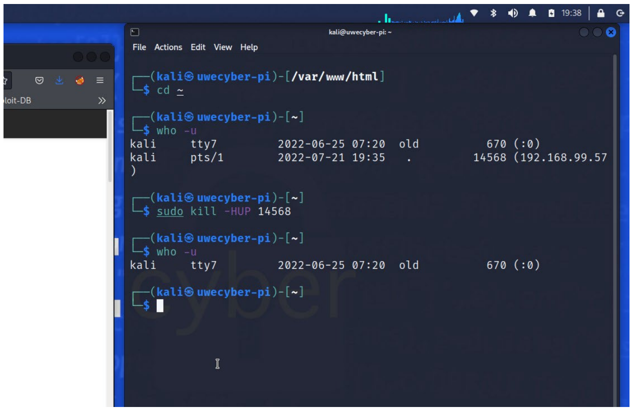
```
┌──(kali㉿uwecyber-pi)-[~]
└─$ ssh kali@192.168.99.86
kali@192.168.99.86's password:
Last login: Thu Jul 21 19:34:11 2022 from 192.168.99.57

UWEcyber Pi Image
Build: 11/03/2022
┌──(kali㉿uwecyber-pi)-[~]
└─$ sudo nano /var/www/html/index.html

┌──(kali㉿uwecyber-pi)-[~]
└─$ 
```

Fig. 6. Red team remotely connects to the target, using known credentials, and modifies the target web page.

Step 2: Red - Gain access to the vulnerable machine: Once the asset of interest is created, the red team are then tasked with compromising the target machine. At the start of the engagement they are not provided any details about the target machine, and so they need to scan the network to acquire the IP of the device. Working in the Terminal, they can type `ifconfig` to find out their own IP address, and then they can use the popular `nmap` tool to scan the entire network for other connected devices: `nmap -sS 192.168.99.0/24`, as shown in Fig. 5. This would give a list of all IP addresses connected to the network access point, as well as their services and the associated port numbers. The list of services will reveal which devices are currently hosting a web service (where port 80 is open). At this stage, the teacher can invite students to discuss what they have learnt about the network. Having identified that port 80 is open, the red team may recognise that this is a web server, and so could use a web browser to navigate to the IP address. They may also see that SSH is running, and so could attempt to connect to the device using `ssh kali@192.168.99.86`. The teacher may need to guide at this stage, but since all devices are identical at this stage, all devices have a default account called *kali* and also has the password `kali`.

Step 3: Red - Tamper with the Web Site and the Desktop files: Having gained access to the machine, the red team can conduct some malicious action. As a simple example, the teacher may notify the students of the path to the web server, such that the red team can access this and modify the content. By typing, `cd \var\www\html`, this will change directory to the web server. The web page can then be modified using `sudo nano index.html`, which will open the source file within the nano text editor. The red team can then take their action, for example, they may change the web page title to read `<h1>HACK</h1>`, as shown in Fig. 6. They can then save the document by pressing Ctrl+S and exit the editor by pressing Ctrl+X.

Step 4: Blue - Harden our security: In Fig. 7, the blue team can scan their system to identify users that are logged in. It can be seen initially that there are two kali users, with one having a different IP address, suggesting that this is a remote connection. Using the command `sudo kill -HUP PROCESS_ID` we can disconnect the session that the remote user is accessing. Having observed a malicious user on the system, the blue team may decide to change their password. The teacher guiding the exercise could have it so that students 'request' a password change - for example, they choose a new password from the teacher (draw this from a raffle). In this way, the password is new but drawn from a set of known passwords (for which we will investigate in the next stage). The blue team can change their password by typing `sudo passwd kali`, and entering this when prompted.



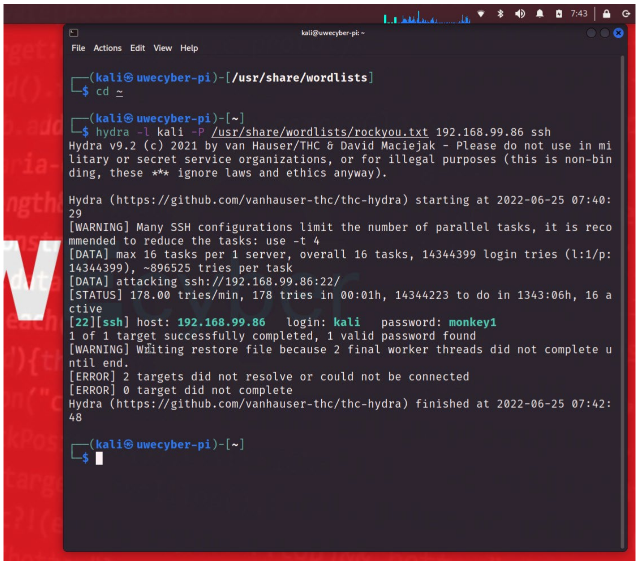
```
┌──(kali㉿uwecyber-pi)-[/var/www/html]
└─$ cd ~

┌──(kali㉿uwecyber-pi)-[~]
└─$ who -u
kali     tty7         2022-06-25 07:20   old          670 (:0)
kali     pts/1        2022-07-21 19:35    .         14568 (192.168.99.57
)

┌──(kali㉿uwecyber-pi)-[~]
└─$ sudo kill -HUP 14568

┌──(kali㉿uwecyber-pi)-[~]
└─$ who -u
kali     tty7         2022-06-25 07:20   old          670 (:0)

┌──(kali㉿uwecyber-pi)-[~]
└─$ 
```

Fig. 7. Blue team scans connected users, and can disconnect the attacking machine.

Step 5: Red - Brute force the password: Once the blue team have changed their password, the question is whether the red team can gain access to this machine again? Fig. 8 shows the use of a brute force password attack on the SSH service. The command to perform this is:

```
hydra -l kali -P
/usr/share/wordlists/rockyou.txt
<IP_ADDRESS> ssh.
```

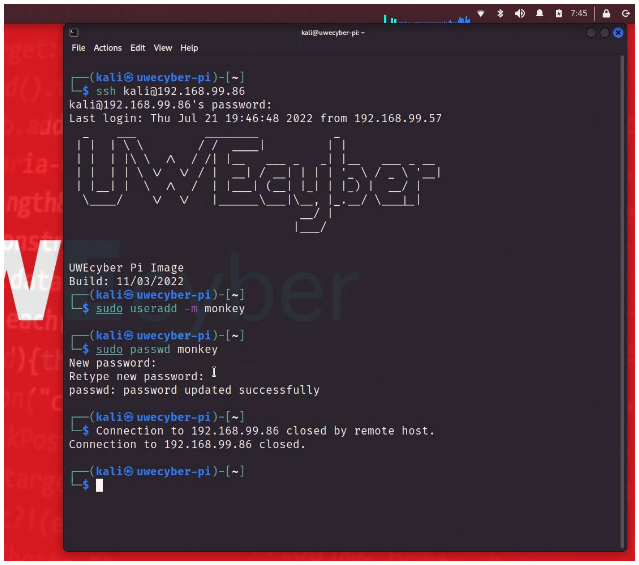
In this step, the red team are using all passwords available to them in the rockyou database to attempt to log in to the blue team machine. The teacher may ask students to comment on the time taken to perform this activity. A password drawn from the raffle that appears earlier in the rockyou file will be found sooner than one that is later on in the file. In the interest of time when running a live exercise, we would suggest that the teacher limits their password selection to the first 500 entries of the rockyou database (which consists of 14 million passwords in total).



```
┌──(kali㉿uwecyber-pi)-[/usr/share/wordlists]
└─$ cd ~

┌──(kali㉿uwecyber-pi)-[~]
└─$ hydra -l kali -P /usr/share/wordlists/rockyou.txt 192.168.99.86 ssh
Hydra v9.2 (c) 2021 by van Hauser/THC & David Maciejak - Please do not use in mi
litary or secret service organizations, or for illegal purposes (this is non-bin
ding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-06-25 07:40:
29
[WARNING] Many SSH configurations limit the number of parallel tasks, it is reco
mmended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:
14344399), ~896525 tries per task
[DATA] attacking ssh://192.168.99.86:22/
[STATUS] 178.00 tries/min, 178 tries in 00:01h, 14344223 to do in 1343:06h, 16 a
ctive
[22][ssh] host: 192.168.99.86   login: kali   password: monkey1
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 2 final worker threads did not complete u
ntil end.
[ERROR] 2 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-06-25 07:42:
48

┌──(kali㉿uwecyber-pi)-[~]
└─$ 
```

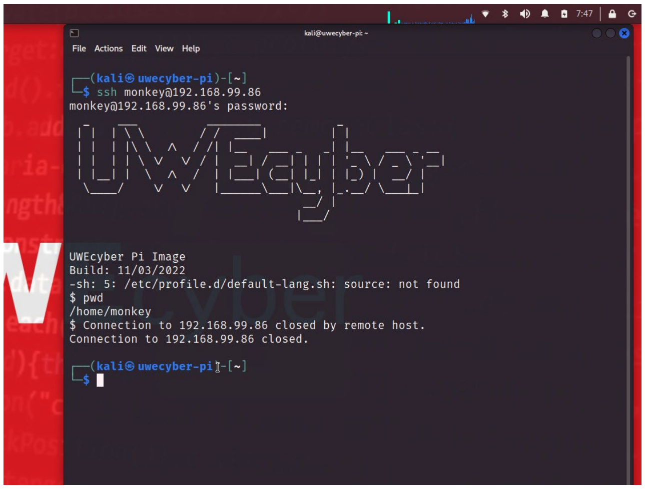
Fig. 8. Red team performs a brute force password attack on the known *kali* user account and uncovers the new password through a dictionary attack.

Step 6: Red - Gain access and create additional user: Having previous cracked the new blue team password, the red team can gain access to the machine again via SSH. A key stage in any attack is maintaining access, and so the red team could therefore create an additional user that the blue team are not aware of. As shown in Fig. 9, for this the red team would type: `sudo useradd <USERNAME>`. Here, we create an additional user called *monkey*. We can then disconnect from SSH, and reconnect using the new account name, as shown in Fig. 10.



```
┌──(kali㉿uwecyber-pi)-[~]
└─$ ssh kali@192.168.99.86
kali@192.168.99.86's password:
Last login: Thu Jul 21 19:46:48 2022 from 192.168.99.57
 _   ___         _______         _
| | | \ \       / / ____|       | |
| | | |\ \  ^  / /| |__  ___ _  | |__   ___ _ __
| | | | \ \ v v / |  __|/ __| | | | '_ \ / _ \ '__|
| |_| |  \  ^  /  | |___| (__| |_| | |_) |  __/ |
 \___/    v   v   |______\___|\__, |_.__/ \___|_|
                                __/ |
                               |___/

UWEcyber Pi Image
Build: 11/03/2022
┌──(kali㉿uwecyber-pi)-[~]
└─$ sudo useradd -m monkey

┌──(kali㉿uwecyber-pi)-[~]
└─$ sudo passwd monkey
New password:
Retype new password:
passwd: password updated successfully

┌──(kali㉿uwecyber-pi)-[~]
└─$ Connection to 192.168.99.86 closed by remote host.
Connection to 192.168.99.86 closed.

┌──(kali㉿uwecyber-pi)-[~]
└─$ 
```

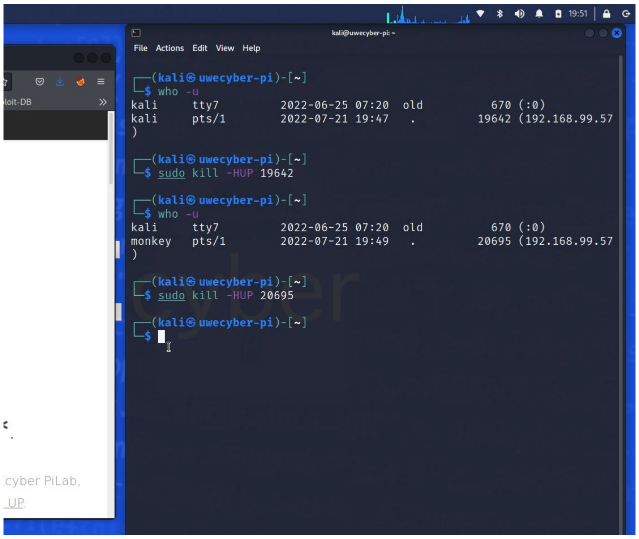
Fig. 9. Red team remotely access the target and create a new user account before being disconnected by the blue team.



```
┌──(kali㉿uwecyber-pi)-[~]
└─$ ssh monkey@192.168.99.86
monkey@192.168.99.86's password:
 _   ___         _______         _
| | | \ \       / / ____|       | |
| | | |\ \  ^  / /| |__  ___ _  | |__   ___ _ __
| | | | \ \ v v / |  __|/ __| | | | '_ \ / _ \ '__|
| |_| |  \  ^  /  | |___| (__| |_| | |_) |  __/ |
 \___/    v   v   |______\___|\__, |_.__/ \___|_|
                                __/ |
                               |___/

UWEcyber Pi Image
Build: 11/03/2022
-sh: 5: /etc/profile.d/default-lang.sh: source: not found
$ pwd
/home/monkey
$ Connection to 192.168.99.86 closed by remote host.
Connection to 192.168.99.86 closed.

┌──(kali㉿uwecyber-pi)-[~]
└─$ 
```

Fig. 10. Red team remotely connects to the target machine using the new account credentials.

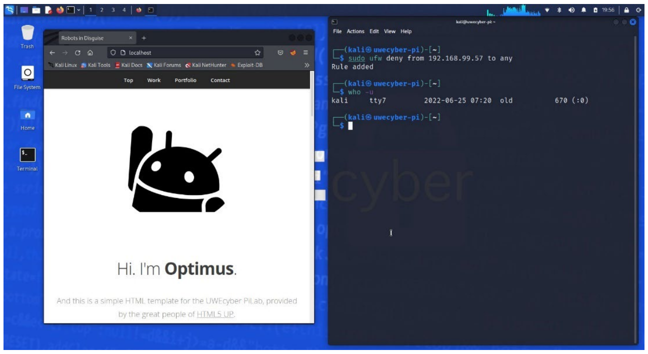
Step 7: Blue - Disconnect the new user: Once again, the blue team can observe the connected users, and identify this unknown new user. Fig. 11 shows the identification of the user *monkey*, and killing the process by which they are connected. As the teacher, you may at this stage have students consider the challenge where there are many users across an organisation. Whilst this may be trivial for our small scale lab environment, if an organisation was managing hundreds or thousands of users, some of whom may have multiple accounts, or where some accounts may be obsolete or of employees who have left the organisation, the task of managing connected users becomes more challenging.



```
┌──(kali㉿uwecyber-pi)-[~]
└─$ who -u
kali     tty7         2022-06-25 07:20  old          670 (:0)
kali     pts/1        2022-07-21 19:47   .         19642 (192.168.99.57
)

┌──(kali㉿uwecyber-pi)-[~]
└─$ sudo kill -HUP 19642

┌──(kali㉿uwecyber-pi)-[~]
└─$ who -u
kali     tty7         2022-06-25 07:20  old          670 (:0)
monkey   pts/1        2022-07-21 19:49   .         20695 (192.168.99.57
)

┌──(kali㉿uwecyber-pi)-[~]
└─$ sudo kill -HUP 20695

┌──(kali㉿uwecyber-pi)-[~]
└─$
```

Fig. 11. Blue team scans connected users, and can disconnect the new user.

Step 8: Blue - Block access using firewall rules: Whilst the blue team is successfully terminating the red team connections, this is not practical for denying their access. This is where the concept of firewalls can be introduced. The blue team can deploy a firewall rule to block this malicious user from connecting to their machine. Fig. 12 shows the blue team using the commend `sudo ufw deny from 192.168.99.57 to any`. UFW stands for *uncomplicated firewall* and is pre-installed on the device. It provides a simple rules-based engine for allowing and denying IP and port connections. Depending on the environment, if there are multiple red team actors in the class, the blue team may wish to block multiple IP addresses, however for the purpose of illustration we show a single machine that is blocked. Finally, Fig. 13 illustrates how the red team have now been blocked from the target device, and therefore can not longer connect via the SSH service.

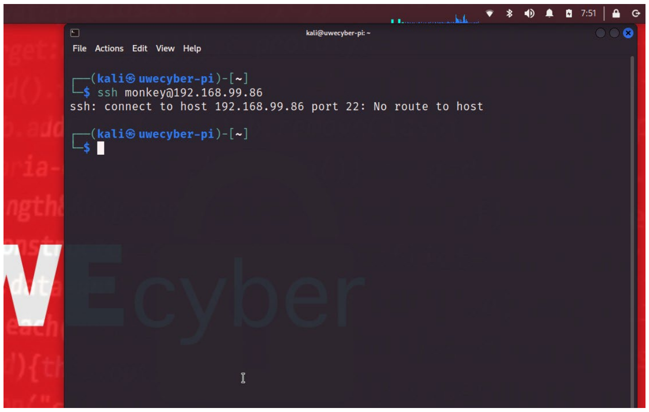


```
┌──(kali㉿uwecyber-pi)-[~]
└─$ sudo ufw deny from 192.168.99.57 to any
Rule added

┌──(kali㉿uwecyber-pi)-[~]
└─$ who -u
kali     tty7         2022-06-25 07:20  old          670 (:0)

┌──(kali㉿uwecyber-pi)-[~]
└─$
```

Hi. I'm **Optimus**.

And this is a simple HTML template for the UWEcyber PiLab, provided by the great people of HTML5 UP.

Fig. 12. Blue team adds firewall rule using UFW and blocks any access from the attacking machine, and can now resolve the defaced website.



```
┌──(kali㉿uwecyber-pi)-[~]
└─$ ssh monkey@192.168.99.86
ssh: connect to host 192.168.99.86 port 22: No route to host

┌──(kali㉿uwecyber-pi)-[~]
└─$
```

Fig. 13. Red team attempts to remotely connect, however the connection is refused for this machine.

This activity could easily be extended. For example, having blocked the attacking IP address, maybe the offensive red team deploy a new Raspberry Pi to continue the attack from. This is akin to a Distributed Denial of Service, whereby rather than relying on a single source to perform an attack, multiple machines are orchestrated to work together to attack the target.

We presented this example methodology with teachers, and found that they were engaged with the walkthrough process. Concepts such as SSH to remote access other Raspberry Pi devices were particularly intriguing for the teachers, and given the physicality of the devices made it much easier to comprehend how one devices was accessing another. Furthermore, the blue team activities of being able to identify connected users and disconnect them was particularly well received, as this is something that they felt they had not seen on other training platforms. Teachers wanted to consider what aspects of the full walkthrough would work best with different year groups, between the ages of 11-18. Through the use of guided walkthroughs to more open-ended questioning, teachers wanted to explore how they would incorporate this within their own teaching. As mentioned previously, teachers felt that treating this as a full class table-top exercise, where groups can explore the practical aspects but then the class hold group discussions at each key stage, would work best. This way, students get to explore the practical aspects themselves, but then all students get to benefit and understand the stages conducted by both red and blue teams. Overall, the teachers found the process to be of interest that would maintain the attention of their students, whilst also appreciating that the level of technical difficulty was important to manage. If the task was too easy, many students would complete this quickly and disengage with the remainder of the classroom session, whilst if the task was too difficult, it would put students off trying to work through and understand the problem.

Other extensions could include the use of a Security Information and Event Management (SIEM) platform, using tools such as the Elastic suite (ElasticSearch, Logstash, and Kibana). Our containerisation approach means that these can be easily deployed from DockerHub.

Whilst on one hand this activity may come across as simplistic, it is fundamental to reflect on the steps covered through this task, and the learning that underpins the process. Students are introduced to a variety of concepts in a relatively short time frame, starting with the notion of an access point and how computing devices can connect to this to communicate, and how devices are assigned an IP address to allow for this communication to take place. Furthermore, devices connected to the same access point can potentially communicate with each other, which as demonstrated opens up the potential of attack from a malicious actor. Students then are introduced to nmap for the purpose of scanning the network to identify connected devices, the services they are hosting and the ports they are hosted from, introducing students to the idea that a single machine can run multiple services, each from a single port. They learn about common ports and their services, such as 80 for HTTP and 22 for SSH. They learn about HTML coding and how this can be modified to alter the appearance of a web page. They learn about remote access through the SSH protocol, and how they can therefore log in to another user's computer using known credentials. They learn about system monitoring, such as observing which users are logged in to a given machine. They then learn about how to disconnect a remote user by terminating the process that hosts the SSH session. They learn about changing passwords, and how 'simple' passwords can be easily targeted through brute forcing techniques using a dictionary of known passwords. They also learn about how attackers may maintain access within a system by creating additional users, which in a large corporate environment could easily go unnoticed. They then further learn about the steps taken by the defensive team, including firewall rule configuration for blocking requests from a given IP address. Even as a walkthrough activity, students will have covered a significant wealth of topic areas that would enrich their understanding of cyber security practice. Furthermore, this activity offers a unique perspective that is not so readily-available through other learning tools, in that as a group-based exercise involving both red and blue teams, students get to observe both the defensive and offensive processes in tandem.

V. CONCLUSION

In this paper, we draw together our experiences of working with school teachers who want to develop cyber security education materials to further their teaching and learning, along with our experience of supporting this development to facilitate practice-based learning against the current obstacles that teachers face within the school environment.

Much curriculum guidance, including GCSE Computer Science, currently asks for students to be able to 'describe' particular details, such as 'describe what is mean by penetration testing' or 'describe what an SQL injection attack is'. Here in lies some of the challenges, such that students may be able to describe this activity, however may not have any practical knowledge of how one may actual do this in practice. Further examples include being able to describe an SQL injection attack, yet not having capacity to actual experiment with such an attack in a safe and controlled environment. When presented with existing resources such as OWASP Juice Shop, teachers found this overwhelming and also struggled to imagine how such practical aspects could be taught within the lesson timetable when they are already stretched for available teaching time. Unfortunately then, such practicals fall to the likes of after school clubs, meaning that only those that are truly enthused get to experience this, and also requiring additional resource from teachers and schools for covering this time. In addition, this does nothing for how to inspire students in the classroom - those with a genuine interest will be able to study materials at home anyway - however, how do teachers capture the imagination of those who are undecided or not fully aware of the topic area, at the time when they actually see the student (i.e., during lessons).

Whilst the workshop has helped us to work closer with our regional school partners, and has also helped teachers to develop their own confidence in delivering practical-based learning on subjects for cyber security, as well as related areas including Open Source Intelligence and Artificial Intelligence. Our ongoing outreach work will help support schools in the development of resources and teaching of practical cyber security. Importantly, we see our role not of 'parachuting' in for occasional workshop activities, but to inspire teachers as well as students so that teachers can improve their confidence in the subject area, and therefore can be of greater support to students in exploring new subject matter. Our ambition remains to develop the pipeline of cyber security education, and therefore, bringing teachers along in this journey is vital. The workshop materials are available to download from http://go.uwe.ac.uk/uwecyber, and we would like to encourage practitioners and educators to use our Raspberry Pi Labs and provide feedback on their experience.

ACKNOWLEDGEMENT

We thank the National Cyber Security Centre for funding this project through the Academic Centre of Excellence in Cyber Security Education (ACE-CSE) grant scheme.

REFERENCES

[1] S. K. Oh, N. Stickney, D. Hawthorne, and S. J. Matthews, "Teaching web-attacks on a raspberry pi cyber range," in *Proceedings of the 21st Annual Conference on Information Technology Education*, ser. SIGITE '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 324–329. [Online]. Available: https://doi.org/10.1145/3368308.3415364

[2] 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

[3] J. Allison, "The who, how and why of choosing post-16 computing curricula: a case study of english further education colleges," *Journal of Further and Higher Education*, vol. 0, no. 0, pp. 1–18, 2022. [Online]. Available: https://doi.org/10.1080/0309877X.2022.2088269

[4] 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

[5] P. Swire, “A pedagogic cybersecurity framework,” *Commun. ACM*, vol. 61, no. 10, p. 23–26, sep 2018. [Online]. Available: https://doi.org/10.1145/3267354

[6] 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

[7] M. Karjalainen, T. Kokkonen, and S. Puuska, “Pedagogical aspects of cyber security exercises,” in *2019 IEEE European Symposium on Security and Privacy Workshops (EuroSPW)*, 2019, pp. 103–108. [Online]. Available: https://doi.org/10.1109/EuroSPW.2019.00018

[8] M. D. Workman, J. A. Lue´vanos, and B. Mai, “A study of cybersecurity education using a present-test-practice-assess model,” *IEEE Transactions on Education*, vol. 65, no. 1, pp. 40–45, 2022. [Online]. Available: https://doi.org/10.1109/TE.2021.3086025