Applied Cyber Security for Applied Software Engineering Undergraduate Program

Yulia Cherdantseva, PhD
Cardiff University
UK
cherdantsevayv@cardiff.ac.uk

Phil Smart, PhD
Jisc
UK
philip.smart@jisc.ac.uk

***Abstract*—In the current landscape where a constantly growing number of cyber threats is accompanied by the increasing shortage of cyber security professionals, it is essential to provide a well thought-out hands-on cyber security education as a part of all Computer Science and Software Engineering degrees. This paper described the experience of designing and delivering a Cyber Security module to Level 5 students on a three-year BSc Applied Software Engineering program. The key goal of the module is to instil the importance of cyber security in software development, and to teach in practice modern security techniques. While being predominantly focused on web-application security, the module also covers foundational cyber security concepts, cryptography and network security, and discusses non-technical topics including security frameworks and security economics. The paper presents the outline of the module, the configuration of the virtual machine used, the structure and content of sessions, and student feedback.**

Keywords—cyber security education, software engineering, web-application security

I. INTRODUCTION

"Knowledge isn't power until it is applied."

Dale Carnegie

Modern society at large and each of us individually need secure and trustworthy information systems. A workforce educated in cyber security is the key to building such systems. However, for the past several years reports on cyber security workforce have been showing a significant shortage of cyber security professionals as well as the lack of cyber security skills among IT professionals. At the end of 2016, 82% of employers globally admitted a shortage of cyber security skills; the global cyber security workforce shortfall is predicted to hit 1.8 million positions by 2022 [1].

A cyber security capability gap is recognized by academia, professional societies, and governments across the world, and there are various initiatives to address this gap. In 2017, professional societies including the Association for Computing Machinery (ACM) and the IEEE Computer Society (IEEE-CS) released university curriculum guidelines for cybersecurity degrees – CSEC 2017 [2]. In 2018, UK government published an Initial National Cyber Security Skills Strategy for public review and is currently considering the views collected before publishing the final strategy in the near future [3] .

In the US, the National Security Agency support and recognise Centers of Academic Excellence in Cyber Defense and Cyber Operations in order to advance cybersecurity skills [4]. In the UK, National Cyber Security Centre (NCSC) provides certification for cyber security degrees in Higher Education (HE), and has recently launched a new program for the Academic Centres of Excellence in Cyber Security Education encouraging broader commitment to cyber security teaching [5].

In the current landscape when a constantly growing number of cyber threats is accompanied by an increasing shortage of cyber security professionals, it is essential to provide a well thought-out hands-on cyber security education as a part of all Computer Science and Software Engineering degrees in HE. In this paper, we describe our experience of designing and delivering a Cyber Security module to Level 5 students on a three-year BSc Applied Software Engineering program at the National Software Academy, Cardiff University, UK (NSA-CU). The BSc program commenced in 2016 with the first cohort of students graduated in 2018.

The NSA-CU is a part of the School of Computer Science & Informatics at Cardiff University and is a center of excellence for software engineering education in Wales, UK. We adopted innovative industry-oriented teaching approach throughout curriculum. The NSA-CU works in partnership with the Welsh Government and industry leaders producing a supply of skilled, workplace-ready software engineering graduates.

The NSA-CU is driven by a "*learning by doing*" ethos strongly relying on project-based learning with continuous support from a wide range of industry partners from different domains. Every semester during this three-year program students participate in industry-led educational software development projects of progressing complexity. The projects are carefully chosen by academic staff to align with the learning objectives of modules in each semester. Industry partners set project requirements, run regular meetings with students in small groups to gauge progress and finally provide feedback for a final product which contributes to assessments. Industry experts regularly deliver guest sessions to our students. For example, for the cyber security module over the past three years guest sessions were delivered on the

topics including, but not limited to GDPR, malware analysis, DDoS attacks, law and regulations, and penetration testing.

The rest of this paper is organized as follows. Section II provides an outline of the module. Section III describes the configuration of a virtual machine used in practical sessions. Section IV covers a set of hand-on exercises included in the module. Section V presents student feedback for the module, while Section VI contains concluding remarks and sketches the direction for future work.

II. Module Outline

The module design was guided by Biggs' "Constructive Alignment" where all teaching components are integrated and tuned to support high-level learning [6]. The Intended Learning Outcomes (ILOs), which are listed below, are defined using active verbs and refer to action that could be verified empirically:

- ILO1: Appropriately use the key security concepts and terminology associated with the covered security topics in discussions and writing
- ILO2: Implement a range of countermeasures to secure a web-application
- ILO3: Implement appropriate database systems security countermeasures
- ILO4: Encrypt data in transit
- ILO5: Perform penetration testing of a web-application and produce a penetration testing report
- ILO6: Employ a range of techniques to secure network communications
- ILO7: Independently research a known security vulnerability and implement an appropriate solution for it
- ILO8: Be aware of cyber security standards and regulations, and understand the role of non-technical factors in cyber security

Teaching activities and assessments are closely aligned with the ILOs and designed to achieve them. ILOs are explained to the students. Each assessment has a set of ILOs associated with it. Related ILOs are explained to students at the beginning of each new topic/session helping students to gauge their progress better.

The Cyber Security module is delivered over the period of 11 weeks during 22 sessions, each being 2.5 hours long. In each session, a range of teaching methods takes place including traditional lectures; group and individual practical coding exercises, discussions, brainstorming, presentations; independent student research; planning; reflection; knowledge tests, and Q&A sessions. During each 2.5-hour session, we typically use a mix of activities switching approximately every 20 minutes. The "change up" approach [7] avoids interest loss that students typically experience as a lecture progresses, allows splitting material into manageable chunks, and gives them a chance to practice thinking about new concepts [8]. As an example, one of the activities is for students to research and present - while working in small groups - a recent cyber security incident (students are offered options to choose from). A discussion is then held about what could be learnt from each incident and how it could be prevented.

As the initial part of this module, students conduct a range of labs on the topics of database security, cryptography, and network security. The knowledge is then assessed by a class test. After that, students learn to pen-test and secure web-applications. The progress is assessed via a web-application security portfolio.

In this Cyber Security module, which is delivered during the Spring (second) semester, students work on securing web-applications they have developed for industry clients in the previous semester. The examples of projects include a token system for volunteers, an attendance monitoring system for sport clubs, a system to monitor behaviour of people with mild anxiety problems, a pilot training booking system, etc. Students work on projects in small groups of 3 or 4. Each software development project is a web-based system with a three-tier architecture: a MySQL database at the back end, a Java-based middle-tier developed using the Spring Boot framework which simplifies the creation of stand-alone enterprise-ready Spring-based applications and a presentation tier implemented with HTML/CSS/Java Script/JQuery. Students have solid programming skills in Java, which they developed during Year 1 and in the Autumn semester of Year 2. Students also undertake a Database Systems module focusing on relational databases in the Autumn semester preceding the Cyber Security module. Business logic may be implemented either in the database layer, or in the application layer based on a group's decision and the projects requirements. The front-end implementation varies from project to project as it is not dictated by assessment requirements, but is typically based on Thymeleaf, a server-side Java template engine for web environments. Students work on individual laptops which they receive for the duration of each academic year. All required software is preinstalled and configured in preparation for teaching. Students work in modern facilities with a start-up feel and look, providing flexibility and allowing students to work comfortably in small group.

The module is assessed via an automated class test in Week 5 and an individual web-application security portfolio due at the end of the module. A detailed discussion of the assessment methods is out of the scope of this paper.

The module is delivered by the authors of the paper, one of whom is an academic member of staff and a researcher in the field of Cyber Security, and another is a Trust and Identity Technical Expert at Jisc – a not-for-profit organisation providing digital services and solutions for higher, further education and skills sectors in the UK. The involvement of an industry expert in the development and delivery of the module further contributes to the practice-informed learning model [9] followed by the NSA-CU and provides multiple

benefits to students that include, but not limited to developing a better understanding of cyber security practices, improving transferable skills and building professional relationships. The benefits of practice-informed learning for students, organisations and HE are extensively discussed in [9].

III. Virtual Lab

Virtualization technologies are actively used for teaching Cyber Security. The most prominent example of a freely available pre-configured educational virtual cyber security lab environment accompanied by a set of practical exercises covering a wide range of cyber security topics is the SEED project that is active since 2002 [10].

In our approach to a virtual lab, we have been inspired by the SEED project. We adopted and extended SEED Ubuntu 16.04 VM with additional software in order to mimic the development environment that our students have on their laptops and to support practical exercises that we developed at Cardiff University. This approach allows our students to benefit from all exercises freely offered within the SEED project which are based on the SEED Ubuntu VM, in addition to the labs developed at Cardiff University that we offer to students. Students could develop, run and pen-test their web-applications within the VM as well as on their host machines.

The following additional software packages were installed on the VM: IntelliJ IDEA (Community Edition), Firefox with Live HTTP Headers and Cookies Manager extensions for examining and managing cookies, Burp suite for conducting brute force attacks, Fail2Ban used to ban IP addresses and limit login attempts for locally deployed web-applications, Ettercap to simulate a DDoS attack on a web-application on one of the local VMs, SQLmap to perform SQL injections attacks automatically, MySQL Workbench for a database user management exercise. All software installed is open-source and free. Oracle VM VirtualBox open-source hosted hypervisor is used for virtualization.

For most exercises on web-application security, a single VM is sufficient. For the exercises on network security and database replication multiple VMs are required. The VM(s) are deployed using a host-only network so that VMs can communicate between themselves, but are isolated from the campus network or the public Internet ensuring that no damage is accidentally caused. Students are instructed on the secure use of VMs and of all software provided, and their responsibility according to the Computer Misuse Act 1990.

IV. Sessions Content and Hands-on Exercises

Each session includes practical learning activities that turn declarative knowledge into functional knowledge and ensure that students acquire essential technical skills. We developed detailed step-by-step student lab manuals for most sessions, but the work is still ongoing to ensure their relevance in the current cyber security landscape. Each manual contains (a) background information required for an exercise, (b) explains ILOs addressed, (c) provides a step-by-step walkthrough for the exercise accompanied by screenshots and explanations, (d) outlines a set of revision questions, and (e) suggests alternative experiments.

The module begins with the introduction to cyber security, where the key concepts are covered and the foundational principles of the growing ISO/IEC 27000 family of standards. Students are encouraged to self-study ISO/IEC 27001 (ILO8).

To ensure continuity between modules within the program, we pick up from the Database Systems module taught in the previous semester and cover such aspects of database security as user management, secure storage of passwords, and database backup and replication. The practical exercises on encryption algorithms and hashing are demonstrated within a DBMS. During the Database Replication lab students configure a MySQL replica set with one master and one slave. The slave server is setup to replicate the master server's database by connecting to the IP address of the master. Once configured the slave server asynchronously checks the master server for updates to the database/s, as soon as there is a change the slave copies the change to its own database(s). This lab addresses ILO3 (Section II).

In the sessions on Network Security we cover the basics of network connectivity, core protocols and concepts from the TCP/IP stack, subnetting, and typical tools including ifconfig, nmcli, netstat, arp utility. In one of the exercises, students set up two VMs to communicate via ping messages, and then modify the Address Resolution Protocol (ARP) table on one VM to investigate the relationship between the IP and MAC addresses on a machine. Students also use Wireshark for network communication analysis examining all embedded layers of network frames, including IP packets, TCP segments and HTTP messages. It is invaluable to understand how network communication is achieved in order to learn how network traffic could be manipulated and protected. Firewall configuration is examined too. The above listed sessions address ILO6.

For the web-application security sessions (ILO2, ILO5) we developed an insecure vulnerable Java web-application for an imaginary company called BetterBuy who sells stationery and furniture online. The BetterBuy web-app is implemented using a three-tier architecture similar to the students' projects described in Section II. We have two versions of the application one implemented using JSP technologies and another as a Spring Boot application with JSF technologies. The latter BetterBuy application utilises Spring Boot for easy configuration of libraries, e.g. Tomcat and JSF mainly for the front-end representation and expression language it offers. In the development of sessions on web-app security we were guided by the Open Web Application Security Project (OWASP) Top Ten vulnerabilities [11].

The app is deployed locally within the VM and available using its own hostname www.betterbuy.com creating an illusion that students access it on the Internet. Our setting ensure that it is safe for students to pen-test the app and

explore the inbuilt vulnerabilities. The code of the BeterBuy app is also available on the VMs and students can run the app on localhost and explore the runtime audit logs and error messages in IntellyJ IDEA, as well as observe the changes propagated to the MySQL database using MySQL Workbench. For example, among other vulnerabilities, we use the app for demonstrating SQL and JavaScript injections, direct-object reference vulnerabilities, insufficient user input validation, information leakage, and insufficient logging and monitoring.

During the session on securing data in transit, students employ pen-testing techniques using Wireshark to sniff unencrypted network traffic, including usernames and passwords, between their user-agent (browsers) and the web-server. In mitigation, students, then, learn how to configure an encrypted HTTPS connection for the application. Further, using Wireshark (or tshark/tcpdump), students examine traffic between the web-application server and a database server (each located on a different VMs) establishing that all commands sent to the database server are transmitted in plain text. Students then learn to set up an encrypted connection between the web-server and the database server as follows: enabling SSL Connection on MySQL Server with OpenSSL, creating a MySQL account that will require SSL, and configuring an SSL connection to MySQL in the Spring Boot Application.

In the summer 2018, we run an 8-week student project as one of the Cardiff University Student Education Innovation Projects (CUSEIP). Under supervision, a year 1 student was involved in setting up the VM adhering to the requirements of the NSA-CU and in the development of lab manuals for four sessions on web-application security including the following topics: user input validation, securing data in transit, authentication and authorization. The involvement of a student helped us to ensure that the material in the lab manual presented at the level accessible for the target audience. This project further contributed to student-led teaching when the student provided help and support to peers.

V. Student Satisfaction and Feedback

At the end of each semester, the NSA-CU runs module evaluation for each module. The Cyber Security module has received high scores of satisfactions from our students in 2017/18 and 2018/19 academic years. The feedback is provided by students using a Likert scale "Definitely Disagree" – "Mostly Disagree" – "Neither Agree nor Disagree" – "Mostly Agree" – "Definitely Agree" with the values 1 to 5 assigned to the answers respectively. Figure 1 shows consistently high average score of 4.41.

Fig. 1. Summary of Module Evaluation Data

In the module evaluation survey, students answer questions about Teaching and Learning, Assessment and Feedback, Organisation and Management, Learning Resources, Learning Community, Student Voice and rate their Overall Satisfaction. According to the student feedback in 2017/18 regarding Teaching and Academic Support 4 students choose "Mostly Agree" and 2 students - "Defiantly Agree" when rating their attitude towards the statement "*The module met my expectations in terms of the knowledge I have gained*". Figure 2 shows the answers in the section Teaching and Academic Support for 2017/18. Figure 3 shows the answers in the section Teaching and Learning provided in 2018/19.

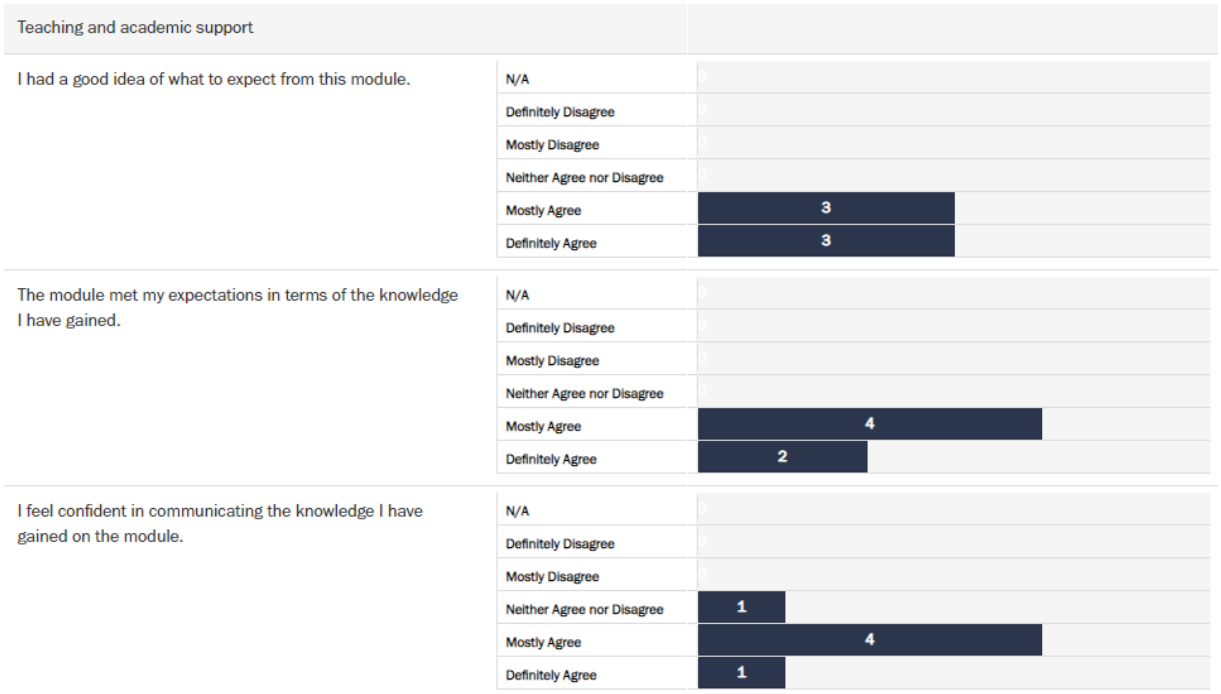

Fig. 2. Student Feedback on Teaching and Academic Support 2017/18

Fig. 3. Student Feedback on Teaching and Learning 2018/19

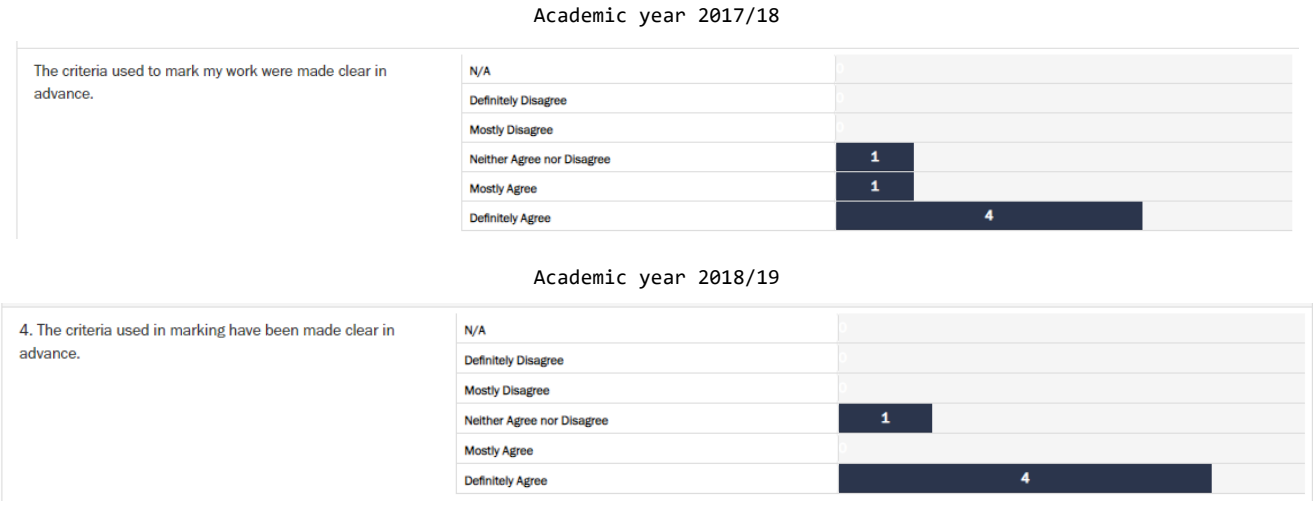

Fig. 4. Clarity of Marking Criteria

Figure 4 shows a consistent level of agreement with the statement "*The criteria used to mark my work were made clear in advance*" for 2017/18 and 2018/19.

In 2017/18 and 2018/19, in their feedback students indicated that they liked the following about the module:

- ✓ The teaching material was very clear and the guest speakers gave very good insight;
- ✓ Lots of practical exercises;
- ✓ Fun to learn about how to secure an application. The powerpoints were very detailed and great for referring back to and helped me a lot with my coursework;
- ✓ Made very interesting, hands-on and relevant. Well taught and lots of resources available;
- ✓ I liked the use of recordings to help students re-look over past lessons to relearn key things;
- ✓ Good range of guest speakers for related topics;
- ✓ Good feedback;
- ✓ The lectures were fun and taught me a lot. this was a mostly perfect module for me, the courseworks are perfect and well explained;
- ✓ The lecturer is engaging, which helps when covering particularly dry topics.

Among the things that the students would like to change about the modules the following suggestions have been listed:

- ✓ More help sessions;
- ✓ It would be nice to have more lab sessions using pen test tools like Metasploit;
- ✓ More time to cover the material;
- ✓ More time given for final project portfolio from lecture content delivery required to complete assessment;
- ✓ Other web application security examples for non-server-side templating applications.

All suggestions are considered during annual module review and reasonable adjustments are made.

Unfortunately, we do not have quantitative evaluation data in the same format for 2019-2020. However, one of our students have provided a video feedback for the module where the student said: "*This module was definitely one of my favorite in the second year of this course, I learnt so much about security which helped me to realize the effort that is going into securing a web-application, and above all, the benefits and the importance of implementing web-security. Having learnt some of the vulnerabilities in cyber security, it has made me more aware and now I believe I have sufficient knowledge and understanding and the skills into implementing and strengthening the security of a web-application.*" The video is available on request.

We hope that the feedback discussed above including the evaluation comments from our students will aid the reader in determining the value of reproduction and adoption of the module structure and material.

VI. Concluding Remarks and Way Forward

In this paper, we shared our experience with developing and delivering the Cyber Security module. In future, we plan to work further on the automated deployment of security VMs using system container technologies, such as the Docker platform and Vagrant. These technologies are already taught at the NSA-CU in the DevOps module, and using this valuable knowledge and skills within the Cyber Security module will be beneficial for students. This will also reduce

time spent by students on performing VMs restoration and network configuration.

From the beginning, we aimed to develop detailed and complete lab manuals for every lab, however, we feel that lab manuals will benefit from going through multiple development cycles for improving clarity and accessibility of the material. We anticipate that the practical exercises we developed may be used in other security modules within Cardiff University and beyond. We hope that producing more hands-on cyber security teaching material will contribute to tackling the cyber security skills gap and the worldwide shortage of cybers security professionals.

ACKNOWLEDGEMENTS

We would like to thank our former student Ieuan Jones for help with the development of the lab manuals for web-application security sessions, and with setting up the VM, and our colleague Carl Jones for help with the development of the BetterBuy web-application.

REFERENCES

[1] W. Crumpler and J. A. Lewis, "The Cybersecurity Workforce Gap," CSIS, 2019.

[2] ACM, IEEE-CS, AIS SIGSEC, IFIP WG 11.8, "Curriculum Guidelines for Post-Secondary. Version 1.0 Report," 2017. [Online] Available: https://www.acm.org/binaries/content/assets/education/curricula-recommendations/csec2017.pdf

[3] H. Govenment, "INITIAL NATIONAL CYBER SECURITY SKILLS STRATEGY," A CALL FOR VIEWS. [Online]. Available: https://www.gov.uk/government/publications/cyber-security-skills-strategy. [Accessed 29 01 2020].

[4] NSA|CSS, "National Centers of Academic Excellence," [Online]. Available: https://www.nsa.gov/resources/students-educators/centers-academic-excellence/. [Accessed 29 01 2020].

[5] C. E., "Launch of the Academic Centres of Excellence in Cyber Security Education.," 23 January 2020. [Online]. Available: https://www.ncsc.gov.uk/blog-post/launch-of-the-academic-centres-of-excellence-in-cyber-security-education. [Accessed 24 01 2020].

[6] J. B. Biggs, Teaching for quality learning at university: What the student does., McGraw-Hill Education (UK)., 2011.

[7] Middendorf, J., & Kalish, A., "The "change-up" in lectures.," *In Natl. Teach. Learn. Forum*, vol. 5, no. 2, pp. 1-5, 1996.

[8] Verner, C., & Dickinson, "The lecture, an analysis and review of research.," *Adult Education*, , vol. 17, no. 2, pp. 90-91, 1967.

[9] GuildHE, "Practice-Informed Learning: The Rise of the Dual Professional," 2018. https://guildhe.ac.uk/wp-content/uploads/2018/11/Practice-Informed_Learning-_Final_Nov_18.pdf [Accessed 29 01 2020].

[10] Wenliang Du, "SEED: Hands-On Lab Exercises for Computer Security Education," *IEEE Security & Privacy*, vol. 9, no. 5, pp. 70-73, 2011.

[11] OWASP, "OWASP Top Ten," [Online]. Available: https://owasp.org/www-project-top-ten/. [Accessed 29 01 2020].