Teaching Secure Software Development Through an Online Course

Christopher Theisen[1], Ted Zhu[2], Kevin Oliver[1], and Laurie Williams[1]

[1] North Carolina State University, Raleigh, North Carolina, United States
[2] Duke University, Durham, North Carolina, United States

Abstract. With an increasing number of cybersecurity attacks threatening consumers, organizations, and governments, the need for trained software security professionals is greater than ever. However, the industry is experiencing a shortage in security professionals for roles at all levels of cybersecurity. Massively Open Online Courses (MOOCs) offer educators an opportunity to retrain current professionals on cybersecurity topics to meet new and ongoing threats. The goal of this paper is to assist instructors of online software security courses in making their courses engaging and effective. In this paper, we present the details of our online software security course, including the technologies used and the material presented. We conducted a pre- and post-survey of course participants and report information on their backgrounds, motivations, and learning objectives. Based on our reflection on the course, we recommend that future instructors of online security courses seed peer discussion on online discussion forums, carefully choose their course platform, and have professionally shot lecture videos.

Keywords: security, education, MOOC, online

1 Introduction

A 2015 report on the shortage of security professionals worldwide by Frost and Sullivan and (ISC)^2 indicated that the cybersecurity industry faces a shortage of 1.5 million security professionals by 2020[1]. The gap in security professionals cannot be filled through new graduates alone. Retraining current members of the workforce on cybersecurity skills can provide immediate relief for the security professional shortage, while also opening a new, lucrative career path for those who retrain.

In: M.G. Jaatun, D.S. Cruzes (eds.): Proceedings of the International Workshop on Secure Software Engineering in DevOps and Agile Development (SecSE 2017), published at http://ceur-ws.org

[1] http://blog.isc2.org/isc2_blog/2015/04/isc-study-workforce-shortfall-due-to-hiring-difficulties-despite-rising-salaries-increased-budgets-a.html

One common method used for retraining professionals is the use of Massively Open Online Courses, or MOOCs. MOOCs are typically run by universities or professional organizations, and provide students a flexible, online platform for learning. However, research has shown that effective execution of a MOOC can be difficult [1,2]. MOOCs typically have thousands or tens of thousands of participants, and having instructors interact with all of the participants is not feasible.

In an effort to assist in these retraining efforts, we ran an online course on the topic of software security during the Spring of 2017. The course builds on lessons learned from a similar online course [3] we conducted in Fall 2014 as well as suggestions for running online courses from the larger scientific community [4,5]. The class featured nine weeks of content, with two lectures per week, a weekly discussion on the latest security news, an episode of the Silver Bullet Podcast from Synopsys [6], and an exercise for the students. The goal of this paper is to assist instructors of online software security courses in making their courses engaging and effective. By presenting the results of our course and the effect of our improvements from a previous iteration of the course, we hope to help instructors of future software security courses improve their offerings.

In this paper, we examine the following research questions in relation to our Software Security MOOC:

RQ1 How did students respond to the presentation of the course and the course content?

RQ2 How did previously suggested improvements help the course, and what additional lessons were learned during the execution of the latest course?

To answer these questions, we asked students in the course to reply to pre- and post-surveys about a variety of topics, including their reason for taking the course, what their professional background was, what their goals were for taking the course, and their knowledge about software security subjects. We compared the results of the pre- and post-surveys to determine how students met their goals. Additionally, each of the instructors or support staff for the course reflected on their experience running the course and provided a list of lessons learned for instructors of future online courses to benefit from.

The rest of the paper is structured as follows. Section 2 covers related work to software security education and online courses. Section 3 describes our online course including the software used and the material presented. Section 4 presents the demographic information for the students who signed up for the course. Section 5 details the questions presented during our pre- and post-surveys and the student responses. Section 6 presents the authors' lessons learned while running the course, so instructors running future online courses can benefit. Section 7 describes the limitations of the conclusions of our study.

2 Related Work

MOOCs are not uniform in their construction. Some MOOCs are offered free to the public, while others have fees or organization membership requirements for

registration [7]. MOOCs have suffered from exceptionally high dropout rates, with up to 97% of registered students dropping out by the end of the course [8]. However, MOOCs have been highlighted for their ability to reach populations that would not otherwise have access to educational opportunities on advanced subjects [9].

We draw on previously constructed security courses to strengthen our own course. Thesien et al [3] provided a list of lessons learned from their own experience in running software security courses. Specifically, they spoke on the challenges of peer evaluation, the time consuming nature of running an online course, and the positive reception of informal roundtable discussions on the weekly topic.

We also drew on general MOOC pedagogy suggestions from the wider online learning community when we constructed our course. Bruff et al. [10] suggested that discussion forums be seeded with open-ended questions to encourage student discussion. Fournier et al. [11] recommended that tools be provided to students to encourage learning outside of the course itself. Pardos et al. [12] recommended multiple sources of information on the same topic, so students with different learning styles can choose their preferred delivery method.

3 Course Description

In this section, we describe the content and structure of the course. We present the learning objectives for students taking the course, the course syllabus, and the structure of each lecture, along with other parts of the course content. Our course was hosted on Amazon AWS using the OpenEDX open source software package, at `https://www.learnsoftwaresecurity.com/`.

3.1 Learning Objectives

For our course, we used the following four learning objectives:

- *Security Risk Management*
  Students will be able to assess the security risk of a system under development. Risk management will include the development of formal and informal misuse case and threat models. Risk management will also involve the utilization of security metrics.
- *Security Testing*
  Students will be able to perform all types of security testing, including fuzz testing at each of these levels: white box, grey box, and black box/penetration testing.
- *Secure coding techniques*
  Students will understand secure coding practices to prevent common vulnerabilities from being injected into software.

– *Security Requirements, Validation, and Verification*
  Students will be able to write security requirements (which include privacy requirements). They will be able to validate these requirements and to perform additional verification practices of static analysis and security inspection.

These four learning objectives inform the selection of materials taught in the course, and were presented to students when they signed up for the course.

3.2 Syllabus

The online course took place over a period of nine weeks, from March 27th to May 29th, during the Spring of 2017. The course ran as an independent MOOC (i.e. not associated with a MOOC company, such as Coursera or Udacity) offered by North Carolina State University. McGraw [13] states that 50% of security errors are implementation bugs, while 50% are design flaws. McGraw's assertion about the ratio of implementation bugs to design flaws informs our split of implementation bug coverage and design flaw coverage in our course.

The first four weeks covered the Open Web Application Security Project (OWASP) [14] Top 10 Vulnerabilities. OWASP periodically releases a list detailing the top 10 types of vulnerabilities that are most commonly seen by software developers and security professionals, and detail how the type of exploit works, how widespread the exploit is, what the possible negative effects are, and some mitigation techniques. The OWASP Top 10 has a particular focus on implementation bugs. We use the OWASP Top 10 as an introduction to security vulnerabilities, and how to think about them from a defensive perspective.

The next three weeks introduced the IEEE Center for Secure Design's (CSD) Top 10 security design flaws to give a background on design decisions that could result in vulnerabilities [15]. Certain design flaws, such as the failure to authenticate users, may not be picked up by static analysis or other automatic tools designed for security. Understanding common security design pitfalls teaches students to avoid making similar mistakes in their own programs, or spot deficiencies in programs that they are reviewing or testing.

Finally, after making students aware of potential security issues, we conclude the final three weeks by presenting security mitigation techniques: attack trees, abuse cases, threat modeling, the STRIDE threat model, security requirements, usability issues in security, and security risk analysis.

– Week 1 (Available March 27th)
  - OWASP Top 10: A1 Injection
  - OWASP Top 10: A2 Broken Authentication and Session Management
– Week 2 (Available April 3rd)
  - OWASP Top 10: A3 Cross Site Scripting (XSS)
  - OWASP Top 10: A4 Insecure Direct Object References

- Week 3 (Available April 10th)
  - OWASP Top 10: A5 Security Misconfiguration
  - OWASP Top 10: A6 Sensitive Data Exposure
- Week 4 (Available April 17th)
  - OWASP Top 10: A7 Missing Function Level Access Control and A8 Cross-Site Request Forgery (CSRF)
  - OWASP Top 10: A9 Using Components with Known Vulnerabilities and A10 Unvalidated Redirects and Forwards
- Week 5 (Available April 24th)
  - IEEE CSD D1-D3: Trust, Authenticate, Authorize
  - IEEE CSD D4-D5: Separate Data, Validate Data
- Week 6 (Available May 1st)
  - IEEE CSD D6-D7: Use Cryptography Correctly, Sensitive Data
  - IEEE CSD D8-D10: Consider Users, Attack Surface, Flexibility
- Week 7 (Available May 8th)
  - Attack Trees
  - Abuse Cases
- Week 8 (Available May 15th)
  - Threat Modeling and STRIDE
  - Security Requirements
- Week 9 (Available May 22nd)
  - Usability
  - Security Risk Analysis

3.3 Lectures

We partnered with local startup Stembrite[2] who specialize in assisting educators with the video portion of their online course offering. We limited our lectures to 5-15 minutes each, as recommended by Aiken et al [16]. Each lecture was shot using a Lightboard[3], an Open Source Hardware "chalkboard" specifically designed for video lectures. During each lecture, PowerPoint slides were superimposed onto the video, and the lecturer wrote additional items on the Lightboard. Lecture slides were adapted from a set of slides use for a software security course at North Carolina State University.

After each lecture, students were asked at least five multiple choice questions on the presented topics. Students who scored at least an 80% average on these quizzes earned a Certificate of Completion for the course.

[2] `http://stembrite.org/index.html`
[3] `http://lightboard.info/`

3.4 Silver Bullet Podcast

The Silver Bullet Podcast with Gary McGraw [6] is a security podcast that discusses a variety of topics related to software security. Each episode features an interview with a prominent security professional on a specific topic in cybersecurity. Each week, we selected an episode of the podcast that was relevant to that week's topics or current events in software security.

Similar to the lectures, students were asked at least five multiple choice questions on the topic of the podcast. These quizzes were also included in the average for the Certificate of Completion.

3.5 Weekly Security News Discussion

During the course, the three instructors taped a 10-15 minute panel discussion each week about the latest news in software security. The goal of these panel discussions was to familiarize students with recent developments in software security news, and also to get the reaction of a group of knowledgeable security professionals to these news items. Topics included the WannaCry attack on the National Health Service (NHS), the Mirai botnet using Internet of Things devices, and a large breach into the Cloudflare service. Each week, 2-4 recent events in software security were discussed, and the panel members would reflect on the event and discuss the reasons why the attack occurred, the consequences of the attack, and possible mitigation techniques. Students were also prompted to reflect on each topic on the course discussion boards.

4 Course Context

Students enrolling in our online course were invited to take a pre-survey at the start of the course and a post-survey upon the completion of the course. In total, surveys were received from 98 unique persons, including 49 pre-surveys, and 70 post-surveys. Only 21 students completed both a pre-survey and a post-survey. In this section, we describe the responses to these surveys and draw conclusions for further improvement of the course.

4.1 Who Enrolled?

A total of 1048 students signed up for the online course. Of those students, 372 completed at least one quiz. At the end of the course, 191 students were issued a certificate of completion, which was issued if the student averaged at least an 80% on the quizzes for the entire nine weeks of the course. This is a total retention rate of 18.2%, which compares favorably against the typical MOOC retention rate of 3% [8].

Demographics of the 98 students who completed a survey(s) provide a snapshot of the type of student enrolling in the MOOC. The full demographic information about enrollees can be found in Table 1. For the education fields, primary

refers to students who have completed the compulsory education in their country, and associate's refers to students who have completed a two year university or community college degree. These students were largely American, male, and Caucasian. Their age ranges varied widely, most commonly between 30-59. Most held a bachelors or masters degree and were currently employed full-time in the private sector in the areas of software engineering or computer security and cryptography. Students also reported the number of years of experience they had in their areas of computer science responsibility with a mean of 15.5 years reported, and a widely divergent standard deviation of 11.9.

Table 1. Demographic information about enrollees in the course.

Nationality		**Gender**		**Race**		**Age**	
USA	83.7%	Male	74.2%	Caucasian	66.3%	40-49	25.8%
Canada	5.4%	Female	25.8%	Asian	19.6%	50-59	24.7%
India	3.3%			Black/Af-Am	6.5%	30-39	23.7%
				Hispanic/Latino	4.3%	22-29	15.1%
						60+	10.8%
Education		**Employment**		**Sector**		**Responsibilities**	
bachelor's	54.3%	full-time	91.4%	private sector	60.2%	software engineering	54.5%
masters	28.3%	student	5.4%	education	17.0%	computer security	33.8%
doctoral	5.4%	part-time	3.2%	government	15.9%	databases	19.5%
associate's	4.3%			health care	3.4%	computer networks	15.6%
primary	4.3%			non-profit%	3.4%	information science	15.6%
						architecture	15.6%
						performance analysis	11.7%

As self-directedness is a trait known to impact online course performance, students were asked to self-rate their ability to self-direct academic work without direction or external motivation. Not surprisingly for students signing up for a self-directed MOOC, 68.1% of students reported they were very self-directed, 30.9% moderately self-directed, and only 1.1% lacking in self-direction.

4.2 Enrollment and Participation

Surveyed students indicated how they heard about the MOOC with a majority hearing about it through a colleague (69.9%) or social media outlet like LinkedIn or Twitter (19.4%). A small percentage (10.8%) heard about the MOOC through a professional association or other news outlet.

When asked to reflect on different reasons for enrolling in a MOOC, students indicated the reasons that applied to them or did not apply to them. The reasons students were most likely to enroll in our MOOC included: general interest in topic (96.8%), personal growth and enrichment (96.7%), relevance to job (85.9%), for fun and challenge (80.5%), for resources applicable to the practice of software security (72.4%), to earn a certificate or statement of accomplish-

ment (71.3%), to share what they learned with colleagues/peers (67.4%), and to become a better coach or mentor to colleagues/peers (60.2%).

Conversely, students indicated a number of reasons that did not factor into their enrollment in the MOOC, including: to improve English skills (89.2%), to receive incentives from employer such as a promotion (83.3%), relevance to academic research (83.1%), for a career change (76.2%), relevance to school/degree (69.9%), to take the course with colleagues/friends (68.6%), or to connect/network with other software security professionals (67.9%).

When asked what could be done to make students more active in the course, a few comments were received. Seven students suggested more discussion/interaction among course participants, perhaps in the form of an online meet up or group chat, and perhaps as a required/mandatory part of the course since some students did not pay attention to it as a non-required element. One student suggested posting recommended activity requirements and a timeline to help students stay on pace.

Students reported the number of hours spent on the course per week, with most spending from 1-2 hours per week (70.4%) or 3-4 hours per week (25.4%). Only 4.2% reported spending 5-6 hours per week on the course.

5 RQ1 -Student Response

In this section, we report the student survey responses to the course, along with feedback received directly from students enrolled in the course.

5.1 Course Quality

Students were asked on the post-survey (n=70) if they agreed or disagreed that the MOOC effectively followed a set of common instructional design principles. The most well-received instructional design elements in the MOOC were elements dealing with content authenticity and in particular how the MOOC covered real-world problems relevant to the workplace. The least well-received instructional design elements in the MOOC were elements dealing with interaction and collaboration, with 26.1% of students stating that the course required collaboration outside of the course, and 25.7% stating the course required collaboration within the course. The full survey responses from participants' about the overall course quality can be found in Table 2.

Students were asked on the post-survey (n=70) if they agreed or disagreed that certain course elements aided their learning in the MOOC, on a five-point Likert scale [17] from strongly disagree to strongly agree. A strong majority of students agreed or strongly agreed that course lectures (87.3%), linked resources (80.0%), readings (78.9%), and assessments (77.5%) aided their learning. When asked what additional non-human supports students would recommend to aid their learning, 20 students provided written comments. Twelve of these students recommended more labs and exercises and a virtual machine or Web site test bed to complete these exercises in an authentic setting. Four students commented on

Table 2. Survey responses from participants about the quality of the course.

Percentage of Post-Survey Participants Who Agreed or Strongly Agreed that the MOOC Effectively Followed Certain Instructional Design Principles		*Percentage of Post-Survey Participants Who Disagreed or Strongly Disagreed that the MOOC Effectively Followed Certain Instructional Design Principles*	
objectives related to real-world problems	95.8%	problems ill-structured (more than one solution)	29.6%
problems typical of those in real world	88.7%	activities require building on others work	26.1%
activities relate to real workplace problems	85.9%	activities require collaboration outside course	26.1%
resources re-used from real-world settings	81.7%	activities require collaboration in course	25.7%
provides example problem solutions	77.5%	activities require contributing to collective knowledge, not merely consuming knowledge	25.7%
activates relevant prior knowledge/experience	66.7%	activities require divergent peer interaction groups	25.7%
activities build upon one another	66.2%	activities require learning from one another	25.0%
activities require application of knowledge/skill	64.3%	problems divergent	22.5%

course content, requesting more examples and reading material, suggesting that content be provided earlier in the course, and suggesting more in-depth reading material since one person deemed existing materials to be too light weight. Other suggestions by individuals included: providing a glossary of technical terminology, adding minimum browser requirements since one person said their browser would not handle the course Web page, and partnering with an external entity to provide more extensive guides on how to complete some of the tasks discussed.

Students were also asked on the post-survey (n=70) if certain feedback loops aided their learning in the course, on a five-point Likert scale from strongly disagree to strongly agree. When asked whether the feedback they received from the instructors was sufficient, 65.7% agreed or strongly agreed that it was. Ten students provided written comments about course feedback loops, with five of these comments reflecting on quizzing. Students recommended more authentic assessment beyond multiple choice questions that could be retaken until correct. Students also suggested providing corrective feedback or explaining why any quiz answers were marked incorrect. Two students commented on peer review with one suggesting deadlines were needed and another that discussion threads might be a better way to conduct peer review. Finally, individual students made a few recommendations, including the inclusion of information on the use of artificial intelligence for tracing security vulnerabilities.

Students were asked on the post-survey (n=70) if certain student strategies aided their learning in the course on a five-point Likert scale from strongly disagree to strongly agree. A majority (67.1%) of students agreed or strongly

agreed that study skills strategies aided their learning and that time management strategies aided their learning (e.g., setting a schedule and working on the course consistently) (67.1%). Students responded more neutrally to collaboration strategies with peers, however, with only 37.1% agreeing or strongly agreeing that these aided their learning. Only a few comments were registered about student strategies with one student requesting more readings and another more opportunities to collaborate with peers.

Finally, students were asked on the post-survey (n=70) if the technical learning curve and MOOC platform were manageable on a five-point Likert scale from strongly disagree to strongly agree. Students largely agreed that the technical learning curve (91.0% agreed or strongly agreed) and MOOC platform were manageable (88.4%).

5.2 Student Beliefs About Software Security

Students were asked on both the pre- and post-survey about their software security beliefs, such as its relevance in the workplace. Results were generally similar on both the pre- and post-survey with a strong majority of respondents indicating software security was an applicable topic in the workplace, was a current problem with unsolved components, was relevant and applicable to their work, had severe consequences if not tended to, and was an important priority in their work. On one item about software security getting worse over time, it is noteworthy that more respondents agreed with this statement on the post-survey than on the pre-survey, perhaps suggesting they learned from the course about the escalating nature of threats to software security. Any pre-post comparisons should be made cautiously, however, as these two survey groups were largely different persons who may have simply held different beliefs about software security. The full survey responses from participants about their beliefs about software security can be found in Table 3.

Table 3. Student beliefs about software security.

Student Beliefs About Security	Pre-Survey	Post-Survey
Software security is a highly applicable topic in the workplace.	87.00%	98.50%
Software security is a current problem with unsolved components.	80.40%	88.10%
Software security is getting worse over time.	39.10%	61.20%
Software security is relevant and applicable to the work I conduct or will conduct in the near future.	84.80%	89.60%
The consequences of not tending to software security are severe.	89.10%	97.00%
Software security is an important priority in my work or future work.	84.80%	86.60%

For the 21 students who did complete both pre- and post-surveys, results were compared to determine if any changes in beliefs had occurred by the end of the MOOC. For these 21 students, there were no significant differences pre-to-post for these question items, except for one item: software security is a current problem with unsolved components, for which the pre-survey agreement (M=3.9, SD=1.37) was significantly less than the post-survey agreement (M=4.6, SD=.68), t(18) = -2.42, p = .026. This finding might suggest students became slightly more aware of the unsolved nature of software security components through the MOOC.

5.3 Student Understanding of Software Security Course Topics

Students were asked to rate their current understanding of course topics on both the pre- and post-survey, on a five-point scale (no understanding, minimal, moderate, good, and strong). The percent of participants who reported good or strong understanding of course topics at pre-survey and at post-survey is reported in the table below. Students on average did report stronger understanding on the post-survey, but again it is difficult to compare between these two survey groups who were largely different persons. The improvement in scores may suggest the course did improve some participants understanding, but this result cannot be confirmed and is generally not supported by what little statistical data is available, presented in Table 4.

Table 4. Student understanding of software security course topis - pre and post surveys.

Items	Pre-Survey - Good or Strong Understanding	Post-Survey - Good or Strong Understanding
Security risk management	39.1%	63.6%
Security testing	37.0%	53.0%
Secure coding techniques	26.1%	65.2%
Security requirements, validation and verification	34.8%	62.1%

For the 21 students who did complete both pre- and post-surveys, results were compared to determine if self-reported understanding of course topics improved from pre-to-post. There was no significant difference in these students self-reported understanding of security risk management, security testing, or security requirements from pre-to-post. There was a significant difference in pre-post understanding for secure coding techniques, with these 21 students reporting significantly less understanding at pre-survey (M=3.0, SD=.94), than at post-survey (M=3.5, SD=1.0), t(18)=-2.73, p=.014. This finding might suggest students became slightly more knowledgeable about secure coding techniques through the MOOC.

5.4 Student Importance Placed on Course Topics

Students were asked to rank the aforementioned four course topics in terms of their order of importance to ones area of employment responsibility. Security risk management was ranked slightly higher among pre-survey takers relative to post-survey takers, while secure coding techniques and security requirements were ranked slightly lower among pre-survey takers relative to post-survey takers. These results could suggest some shifting in importance placed on course topics, but again it is difficult to compare between these two survey groups. The importance rankings for each topic are listed in Table 5.

Table 5. Student importance placed on course topics - pre and post surveys.

Items	Pre-Survey Average Rank	Post-Survey Average Rank
Security risk management	2.1	2.5
Security testing	2.9	2.9
Secure coding techniques	2.7	2.5
Security requirements, validation and verification	2.6	2.3

For the 21 students who did complete both pre-post surveys, results were compared to determine if average rank order for a given course topic shifted from pre-to-post. However, no significant differences were found.

5.5 Student Recommendations

Students taking the software security MOOC in spring 2017 appreciated that the content and problems reflected real-world issues, and most students agreed that the different course content elements (lectures, linked resources, readings, and assessments) aided their learning. To complement this content, students recommended the MOOC incorporate further labs or virtual test bed exercises, more authentic assessment beyond multiple choice questions, and further guidance to monitor ones progress through these materials and activities (e.g., activity requirements, timelines, pacing schedule).

To improve the MOOC, the number one recommendation across four different question sets (recurring theme) was the need to incorporate more student-student discussion and interaction into the course (e.g., online hangouts, chats, forums). Only a few students requested more student-instructor interaction, perhaps acknowledging this is a challenge in a large enrollment MOOC course. Students, however, did expect and ask for further student-student interaction. Students noted that optional discussions were not likely to lead to any meaningful interaction, thus it may be necessary to make discussions/interactions mandatory to get credit for the MOOC.

6 RQ2 - Lessons Learned

In this section, we discuss the lessons we learned during the execution of the course, so future instructors may benefit from our experience. In no particular order:

1. **Peer discussion does not happen organically.** The amount of student-to-student interaction on our discussion boards was lower than we expected, and as discussed above, students would have liked more opportunities to interact with their peers. We saw spikes in activity on the discussion boards when our weekly panel discussions prompted them to provide their opinions in a specific way, such as describing how they would prevent a particular attack if it targeted their organization.

2. **Carefully choose your course platform.** Two of the instructors of this course have previous experience running a software security MOOC, and the previous course was severely hampered both in quality and time spent because of the previously used platform. OpenEDX on Amazon AWS have a straightfoward implementation process with better performance than the previous course platform, and we would recommend this combination to instructors looking to run their own independent courses.

We did receive feedback from one student that OpenEDX was the reason they stopped participating in the course, as they felt the interface was clunky and not intuitive. While we felt the interface was an overall improvement over the previous platform, continued improvement in streamlining the course for participants is important for student retention.

3. **Professional video editing improves lecture quality.** The support of Stembrite's video editor improved the quality of the lecture videos significantly compared to the previous course. While consumer video editing solutions can work for creating lecture videos, a student who participated in both courses commented that the current lecture videos were more authoritative and were easier to follow, thanks to the Lightboard technologies and the improvements in video quality and editing.

4. **Having instructor office hours.** Based on student feedback, we recognized the need for more instructor-student interaction where possible. To that end, halfway through the course we made an instructor available for "office hours," or a set time that the instructor could be reached on a web conferencing service. However, this service was used minimally by students, with only one student taking advantage of the service over the last four weeks of the course. However, if this service was available from the beginning of the course, more students may have taken advantage of it, or we might have retained students that we lost during the course. For large courses with 10,000+ students, having webinars like this may be unfeasible without additional moderators or other pre-planning activities.

7 Limitations

Results from the pre- and post-survey are only from students who self-selected to provide results to the researchers. Students who opt to provide demographic information may represent a different population than the whole body of students. Additionally, students who opt to provide feedback on the course may represent a different population than the whole body of students. Students who were satisfied with the course may be more likely to respond to a survey asking for feedback.

Results from the pre- and post-survey do not represent the same students. While there is some overlap between the two groups, the majority of the responders only took one of the two surveys. A lack of data from the same students in the pre- and post-surveys could result in a different understanding of the students' satisfaction with the course.

Parts of the lessons learned represents the opinions of the authors, and is not necessarily grounded in feedback received from the students unless otherwise noted. Different instructors may have different lessons learned from the authors.

8 Acknowledgements

We thank the participants for their attention and feedback on the course. We thank the Realsearch group for their important feedback on the paper. The work in this paper was funded under National Science Foundation grant number 4900-1318428.

References

1. Liyanagunawardena, T.R., Adams, A.A., Williams, S.A.: Moocs: A systematic study of the published literature 2008-2012. The International Review of Research in Open and Distributed Learning **14**(3) (2013) 202–227
2. Kay, J., Reimann, P., Diebold, E., Kummerfeld, B.: Moocs: So many learners, so much potential... IEEE Intelligent Systems **28**(3) (2013) 70–77
3. Theisen, C., Williams, L., Oliver, K., Murphy-Hill, E.: Software security education at scale. In: Software Engineering Companion (ICSE-C), IEEE/ACM International Conference on, IEEE (2016) 346–355
4. Bali, M.: Mooc pedagogy: gleaning good practice from existing moocs. Journal of Online Learning and Teaching **10**(1) (2014) 44
5. Breslow, L., Pritchard, D.E., DeBoer, J., Stump, G.S., Ho, A.D., Seaton, D.T.: Studying learning in the worldwide classroom: Research into edx's first mooc. Research & Practice in Assessment **8** (2013)
6. Synopsys: Silver bullet podcast - https://www.cigital.com/podcast/ (2017)
7. Vardi, M.Y.: Will moocs destroy academia? Communications of the ACM **55**(11) (2012) 5–5
8. Rivard, R.: Measuring the mooc dropout rate. Inside Higher Ed **8** (2013) 2013
9. Hyman, P.: In the year of disruptive education. Communications of the ACM **55**(12) (2012) 20–22

10. Bruff, D.: Lessons learned from vanderbilts first moocs (2013)
11. Fournier, H., Kop, R., Sitlia, H.: The value of learning analytics to networked learning on a personal learning environment. In: Proceedings of the 1st International Conference on Learning Analytics and Knowledge, ACM (2011) 104–109
12. Pardos, Z.A., Schneider, E.: First annual workshop on massive open online courses. In: International Conference on Artificial Intelligence in Education, Springer (2013) 950–950
13. McGraw, G.: On bricks and walls: Why building secure software is hard. Computers & Security **21**(3) (2002) 229–238
14. OWASP: Top 10 - https://www.owasp.org, accessed 2017-06-25 (2013)
15. IEEE: Avoiding the top 10 software security design flaws - http://cybersecurity.ieee.org/center-for-secure-design/, accessed 2017-06-25 (2017)
16. Aiken, J.M., Lin, S.Y., Douglas, S.S., Greco, E.F., Thoms, B.D., Schatz, M.F., Caballero, M.D.: The initial state of students taking an introductory physics mooc. arXiv preprint arXiv:1307.2533 (2013)
17. Likert, R.: A technique for the measurement of attitudes. Archives of psychology (1932)