Understanding challenges to adoption of the Microsoft elevation of privilege game

Inger Anne Tøndel, Tosin Daniel Oyetoyan, Martin Gilje Jaatun, Daniela Cruzes

Citation:
Inger Anne Tøndel, Tosin Daniel Oyetoyan, Martin Gilje Jaatun, Daniela Cruzes. Understanding challenges to adoption of the Microsoft elevation of privilege game. HoTSoS '18,Proceedings of the 5th Annual Symposium and Bootcamp on Hot Topics in the Science of Security, Raleigh, North Carolina, April 10-11, 2018 DOI: https:/doi.org/10.1145/3190619.3190633

Understanding challenges to adoption of the Microsoft Elevation of Privilege game

Inger Anne Tøndel
Department of Computer Science, Norwegian University of Science and Technology (NTNU)
Trondheim, Norway

Tosin Daniel Oyetoyan
Martin Gilje Jaatun
Daniela Cruzes
SINTEF Digital
Trondheim, Norway

ABSTRACT

The goal of secure software engineering is to create software that keeps performing as intended even when exposed to an active attacker. Threat modelling is considered to be a key activity, but can be challenging to perform for developers. Microsoft has tried to lower the bar through creating a threat modelling game called Elevation of Privilege (EoP), but anecdotal evidence suggests that it has seen little use in actual development projects. To learn more about challenges facing adoption of EoP, we performed a case study in a university setting comprising several agile development projects. The results show that the game aided in discussing and learning about software security, but the impact on development seems to have been limited. In addition, challenges related to game dynamics, relevance of hints on the cards, and the time needed to play the game, limits the acceptance of the game.

CCS CONCEPTS

• **Security and privacy → Software security engineering**;

ACM Reference Format:
Inger Anne Tøndel, Tosin Daniel Oyetoyan, Martin Gilje Jaatun, and Daniela Cruzes. 2018. Understanding challenges to adoption of the Microsoft Elevation of Privilege game . In *HoTSoS '18: Hot Topics in the Science of Security: Symposium and Bootcamp, April 10–11, 2018, Raleigh, NC, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3190619.3190633

1 INTRODUCTION

Contrary to popular belief, software security is important for most software development efforts, not just for software with specific security requirements [9]. To achieve effective software security at reasonable cost, it is necessary to start with understanding the risks and threats. This implies a need for threat modeling, which has been a core component of secure software development lifecycles (SSDLs) such as Microsoft SDL [14]. Unfortunately, we have found that threat modeling is seldom performed in the small and medium-sized organisations that dominate the Norwegian market [10].

HoTSoS '18, April 10–11, 2018, Raleigh, NC, USA

In many countries, agile methods are now dominating software development efforts [5], and traditional SSDLs are difficult to align with an agile development approach [10]. It is therefore a challenge how to ensure that threat modeling is not forgotten in the agile process. We have investigated how games can help in this context, by studying the use of the Microsoft Elevation of Privilege (EoP) [17] game. Our initial motivation for studying this game was that one of our partner organisations had discovered it, and approached us for more information and training. On the face of it, having a game would be an attractive way of sneaking threat modelling into the development lifecycle, but based on our interaction with Norwegian development organisations, it appears that "nobody" uses EoP in the real world. Even Microsoft admit [17] that they have abandoned EoP as a regular development activity, now using it solely as an educational tool. We thus wanted to scrape a little bit under the surface to determine why this is so.

Use of EoP was studied in a capstone development project with six development groups. The goal of the study was to assess to what extent the EoP game would be accepted as a technique in agile teams, and if possible to determine obstacles to adoption of the game. Our investigation is centered on the following research questions:

RQ1: To what extent is EoP accepted by the players, both short-term and longer term?
RQ2: What lessons learned and improvements are identified by the players?

The rest of this paper is structured as follows: In Section 2 we provide an overview of threat modeling with more details on the Elevation of Privilege game. In Section 3 we describe the study that we conducted. We present the results in Section 4, and discuss in Section 5. We conclude the paper in Section 6.

2 THREAT MODELING AND EOP

Threat modelling is an activity used to identify security defects in a system [18]. All major software development lifecycles and frameworks (e.g. BSIMM [13], OpenSAMM [16], Software Security Touchpoints [12], Microsoft's Security Development Lifecycle (SDL) [14]) contain activities directly related to assessment of risk and threats. Threat modelling is even stated to be *"The Cornerstone of the SDL"*, and the threat model *"the major SDL artifact"* that *"must be used as a baseline for the product"* [15].

Threat modeling was an integral part of the Microsoft Secure Development Lifecycle as described by Howard and Lipner [8], and further detailed by Shostack [18]. At the risk of over-simplifying, the Microsoft approach to threat modeling can be summarized

as follows: 1) Draw data flow diagram of system; 2) Add trust boundaries; 3) Apply STRIDE threat types (see below); 4) Develop attacks using attack trees. Shull and Mead [19] compare three threat modelling approaches, of which one is STRIDE. The other two are Security Cards from the University of Wisconsin[1] and the Persona non-Grata (PnG) from DePaul University[2]. Even though they claim that the other two can perform better in certain circumstances, they still confirm that STRIDE is the state of the art.

To the best of our knowledge, there exist three card games intended for use in development to address software security risks and threats. In the following, we describe EoP. A similar game can be found in OWASP Cornucopia[3] that is played in a similar fashion as EoP, only with different hints on the cards. Another alternative game is the Protection Poker game [22] which is a technique for risk assessment particularly suited for agile development teams.

Elevation of privilege (EoP) [17] is a game suitable for 3-6 players. The EoP game requires the model or architectural diagram of the system before play starts. As such, EoP is suitable during the design phase. There are 74 playing cards, divided into six suits based on the STRIDE threat mnemonic[4]. Each suit consists of cards numbered in similar way to normal playing cards; 2-10, Jack, Queen, King, Ace. Each card lists which suit it belongs to, a number, and a threat represented by the suit. An example threat is *"An attacker can replay data without detection because your code doesn't provide timestamps or sequence numbers"* (5 of Tampering). The ace cards are open threat cards and a player must identify threats not listed on another card when they are played.

Before the game starts, all cards are dealt out. The play starts with the 3 of Tampering and each player plays in turn in the suit (Tampering). However, if they do not have the suit, they can play another suit. The rule is that the highest card takes the trick unless someone has a card from the elevation of privilege suit. This suit trump all other suits and take the trick. For every card that is played, the card is read out and the threat on the card is discussed. If a player cannot link the threat to the system, play proceeds. One point is awarded for a *relevant* threat on the card played. In addition, the player that takes the trick gets one point, and starts the next round. When all the cards have been played, the winner is the player with the most points.

3 METHODOLOGY

Regarding the *case context*, the study was performed in the *Customer Driven Project* course (TDT4290) at the Norwegian University for Science and Technology (NTNU), autumn 2016. This course is mandatory for all 4th year computer science students at this university. In this course, the students are divided into development teams (5-8 students per team), and every team is given a development project from an external customer. Customers can be private companies, public organisations or research institutes. During this course the students are expected to investigate the needs of the customer, develop software, do some testing of this software and document everything in a report and a presentation given to the customer. In general, all student groups use agile methodologies to some extent. Six groups, consisting of 36 students in total, were required to use EoP for their project. This was the first year software security was included as part of the course.

An overview of data collection activities can be found in Figure 1. As most students had received limited formal training on software security before this course[5], we arranged a lecture where all students were given a short plenary introduction to software security in general, and an introduction to the EoP game. They played the game on an example project, and responded to a questionnaire that covered the students' acceptance of the technique. Data collection proceeded through facilitation and observations of students playing EoP in their group, and the observations were followed by group interviews towards the end of the course, allowing detailed student feedback on the technique. Additionally, the main author of this paper acted as supervisor for one of the student groups[6] and took part in project leader and supervisor meetings throughout the course. The questionnaire on acceptance was repeated towards the end of the course. The study has been reported to the Norwegian Data Protection Official for research (NSD). In the following we explain the data collection methods in more detail; the questionnaire, the observations and the group interviews.

The main motivation for using a *questionnaire* was to capture students' immediate and longer term acceptance of the EoP technique (RQ1). A questionnaire could easily reach a large number of the students, and could easily be repeated. We decided to base the questionnaire on the Technology Acceptance Model (TAM) for two reasons. First, TAM, although being criticized [11], is considered a highly influential and commonly employed theory for describing an individual's acceptance of information systems. TAM, adapted from the Theory of Reasoned Action [1] and originally proposed by Davis [3], suggests that when users are presented with a new technology, a number of factors influence their decision about how and when they will use it (see Figure 2), notably:

- *Perceived usefulness:* this was defined by Davis as *"the degree to which a person believes that using a particular system would enhance his or her job performance"* [4]
- *Perceived ease of use:* Davis defined this as *"the degree to which a person believes that using a particular system would be free from effort"* [4].
- *External variables:* include *"system characteristics, training, user involvement in design, and the nature of the implementation process"* [21]

Thus, we believed TAM could help us understand the different reasons for acceptance of EoP by the students, and that TAM-based questions could trigger comments from the students related to acceptance. Second, we were able to adapt questions from an existing

[1] http://securitycards.cs.washington.edu/
[2] https://www.computer.org/csdl/mags/so/2014/04/mso2014040028.pdf
[3] https://www.owasp.org/index.php/OWASP_Cornucopia
[4] STRIDE stands for S-Spoofing, T-Tampering, R-Repudiation, I-Integrity, D-Denial of Service, and E-Elevation of Privilege

[5] No mandatory training in security, except security being a minor part of some courses that mainly covered other topics.
[6] In addition to EoP, one other technique (Protection Poker) was studied in the course. Groups were assigned to use either Protection Poker or EoP by two researchers in cooperation based on name of the project and name of the customer. In deciding which group should use which technique, the researchers aimed for a balance in size and type of customer and in the type of systems developed so that both games had a mixture of different project types. The student group where the first author acted as supervisor used Protection Poker, not EoP. However, the overall context of the groups were similar, and the project leader forum included all groups, also the EoP groups.

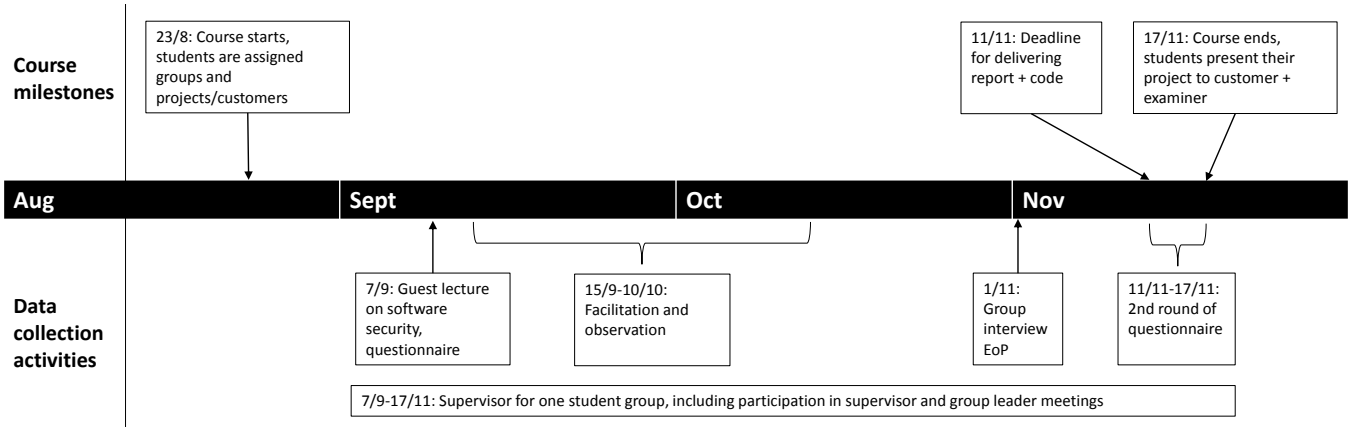

Figure 1: Overview of data collection activities

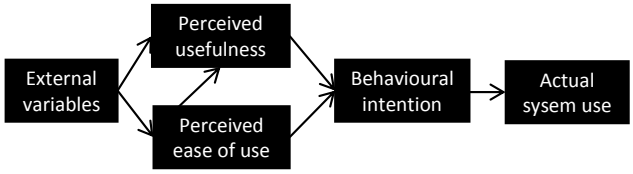

Figure 2: Final version of TAM [21]

questionnaire [6] to the phenomena we are studying (the questions used are shown in Figure 3).

For the *observations*, we created a rota where one of the authors served as facilitator, and at least one other author participated as observer. After each observation session, both the facilitator and the observer filled in reflection notes in a template that contained the following topics: group information; questions from the students on the technique; suggested changes to the game; participation; mood; topics discussed; what worked well with the game; challenges with the game, and; reflections on the observation and how the researchers may have influenced the process. After playing one session of EoP, all groups were encouraged to keep on playing by themselves during the project, and we offered to return and offer support and/or facilitation at a later time, according to their needs.

Towards the end of the course, all groups were invited via email to send two to three participants to an event where the technique would be discussed in more detail. This event was organised as a *group interview* and was scheduled to last for two hours. The following topics were covered: students' expectations to the event; use of the game in the group; brainstorming and discussion on the 4Ls (Liked, Lacked, Learned, Longed for) [2]; suggestions for improvements to the technique; suggestions for improvements to how software security was handled in the course, and; feedback on the event. To encourage participation, all participants were served pizza and they had the opportunity to win cinema gift cards. Non-responding groups were reminded via email. To promote active participation by all participants in the group interviews, each event was split in two parallel sessions to limit the number of participants in each session. The participants were split so that each session had participants from as many groups as possible.

4 RESULTS

In the following we give an overview of results from the study. The section is structured as follows. First we explain how a typical EoP session proceeded, so as to be clear of how the game was played and thus what is evaluated in this study. Then we give an overview of results related to the two research questions of this study, i.e., adoption (RQ1) and improvements and lessons learned (RQ2).

4.1 What was a typical session like? (observation)

The EoP sessions lasted between 45 and 75 minutes, of which the first 20 minutes were spent on having the students explain their system to the facilitator, and in reminding the students about how to play the game. All groups had either made a sketch of their system to use as a basis for discussion, or were able to draw one in the first part of the session. In the discussions in the beginning, security of the system was discussed in an overall fashion, and in general the student groups did not believe security to be an important consideration for their project. Typical arguments were that the system is behind a firewall, other parts of the system (that is, parts that are not developed by the students) are responsible for collecting the data they make use of in their program, and limited connectivity. Still, many groups were able to already at this stage identify attack vectors, e.g. that .doc files they use as input is a possible attack vector, and reflect about the sensitivity of the data their software were to handle.

After the initial discussion, we moved on to the main part: playing the EoP game. We did not have the ambition to play the full deck. Thus students handed out 5-7 cards for each player. The exception is one group, where it was decided to only play with two suits (Information Disclosure and DoS) because of the limited attack surface of the system this group was developing. None of the groups used the Aces. In most groups and for most of the cards what happened was that the student read the card out loud, stated that this is not relevant, or that if it was relevant it would happen in

this part of the system (pointing to the sketch), and then making an entry on the score sheet. But cards could also trigger discussions, either by the students themselves or by the facilitator explaining or asking questions to the students.

The students played from 20 to 45 minutes, depending on how much time they had available. The groups varied in how quickly the game progressed. The fastest group spent on average 0,8 minutes per card. This group only played two suits, and thus many cards were very similar and were considered already discussed by the group. One group spent 1,4 minutes per card, while two groups spent the double (2,7 or 2,8 minutes per card). The reason for this difference is not clear, but may be related to the following:

- *Familiarity with the game rules:* The group that spent 1,4 minutes per card was the first group we facilitated, and the group that spent 2,7 minutes per card was the last group - they were almost a month apart. Thus the faster group were more likely to remember the introduction to the game given in the lecture.
- *Facilitation:* In all groups, facilitation included giving input to the security discussion in the group. The facilitator was active in this regard in all the groups, but observation notes from the 2,7 minute group suggests this slowed down the playing of the game in this group more than in the other ones.
- *Enthusiasm:* The group that spent 2,8 minutes per card was the group in which we experienced the highest burst in interest for security during the play.

All students participated in the game, and this came quite naturally since everybody took turns in putting out a card and assessing if it was relevant. However, most groups had one or two persons that were more skilled in security and that were more into the discussions than the others. Only in one group was there a large part of the team that did not participate much; three out of six people in this group hardly participated in discussions, but just put out cards that then were discussed by the more active participants.

At the end of each session we had some time for reflection in the group where we asked the students what they thought about playing the game. This took between 3 and 15 minutes.

4.2 Acceptance of EoP (questionnaire, observations, group interview)

Acceptance of EoP (RQ1) was mainly studied through the TAM-based questionnaire in the beginning and the end of the course, however, acceptance of the technique was to some extent covered in observations and group interviews as well. In this section we give an overview of the questionnaire results and explain how observations and group interview responses add to the findings from the questionnaire.

Figure 3 gives an overview of the questionnaire results on the TAM variables *future use intention*, *perceived usefulness* and *perceived ease of use*. The results marked *before* refer to responses at the end of the introductory lecture, and the results marked *after* refer to responses at the end of the course. 31 out of 36 students provided questionnaire responses on both occasions.

Four questions together cover the variable *future use intention*, and overall this intention seem to be low and declining throughout the course. One obvious reason for the decline is the requirement to use EoP in the course, something that is not the case for any future projects the students encounter. This is in particular likely to influence the students' responses to the first two questions (question 1 and 2) - what being most surprising in the responses to those questions being the rather low intention to use EoP in the beginning despite this being a requirement. Questions 3 and 4, being less tied to the requirement to use EoP, are thus more important than questions 1 an 2 in order to understand how the students' future use intentions progress throughout the course. In these two questions, one can observe a slight increase in preferences towards EoP in question 3, and a stronger decrease in question 4. Though about 15 percent (between 3 to 10 students based on the question) respond that they are positive/not negative to use EoP also after the project, there are more students that are negative to use, giving the impression that overall students do not want to use EoP in the future.

Four questions together cover the variable *perceived usefulness*. Based on the two most general of these questions (question 5 and question 8) the usefulness seem to be perceived as roughly the same throughout the course, with some people becoming more positive (more agree in question 5) while others are becoming more negative (more strongly disagree in question 5 and also slightly less positive results in question 8). And in the end, more students agree (9 students) than disagree (7 students) that the advantages of using EoP outweigh the disadvantages (question 8), something that can be considered a slightly positive results for EoP. However, EoP is in the end not considered to be particularly useful for what was believed to be the main advantage of using EoP, namely improved security - question 6 and 7.

Six questions together cover the variable *perceived ease of use*. Overall the responses to these questions are more positive than for the previous variables, and they also improve towards the end of the course. More than half of the students ended up finding EoP easy to learn (question 9, 19 students) and about half of the students ended up finding the game easy and understandable (question 10, 15 students). More students than not found EoP easy to use (question 12, 12 vs. 9 students), and there is a decline in students finding the EoP cumbersome to use (question 13). However, there is, also towards the end of the course, quite a large portion of the students that seem to find the game to be difficult, i.e. not easy to learn (8 students), not easy to use (9 students), cumbersome to use (10 students), not clear and understandable (11 students), require a lot of mental effort (6 students). When it comes to time (question 14), the students become less positive throughout the course, with almost half (13 students) agreeing that EoP takes too much time to play.

The questionnaires, both at the beginning and end of the course, invited students to provide open ended comments to supplement their responses. In the beginning of the course, students were asked the two open ended questions: *How do you think playing EoP will influence the product?* and *How do you believe software security is important to your project?* In line with the responses to the TAM questions, those students that provided a response to these questions varied between being positive to the technique and expecting security to improve, and not seeing security, and thus EoP, to be particularly relevant for their project. Almost no students responded to the open-ended question on the questionnaire at the end of the

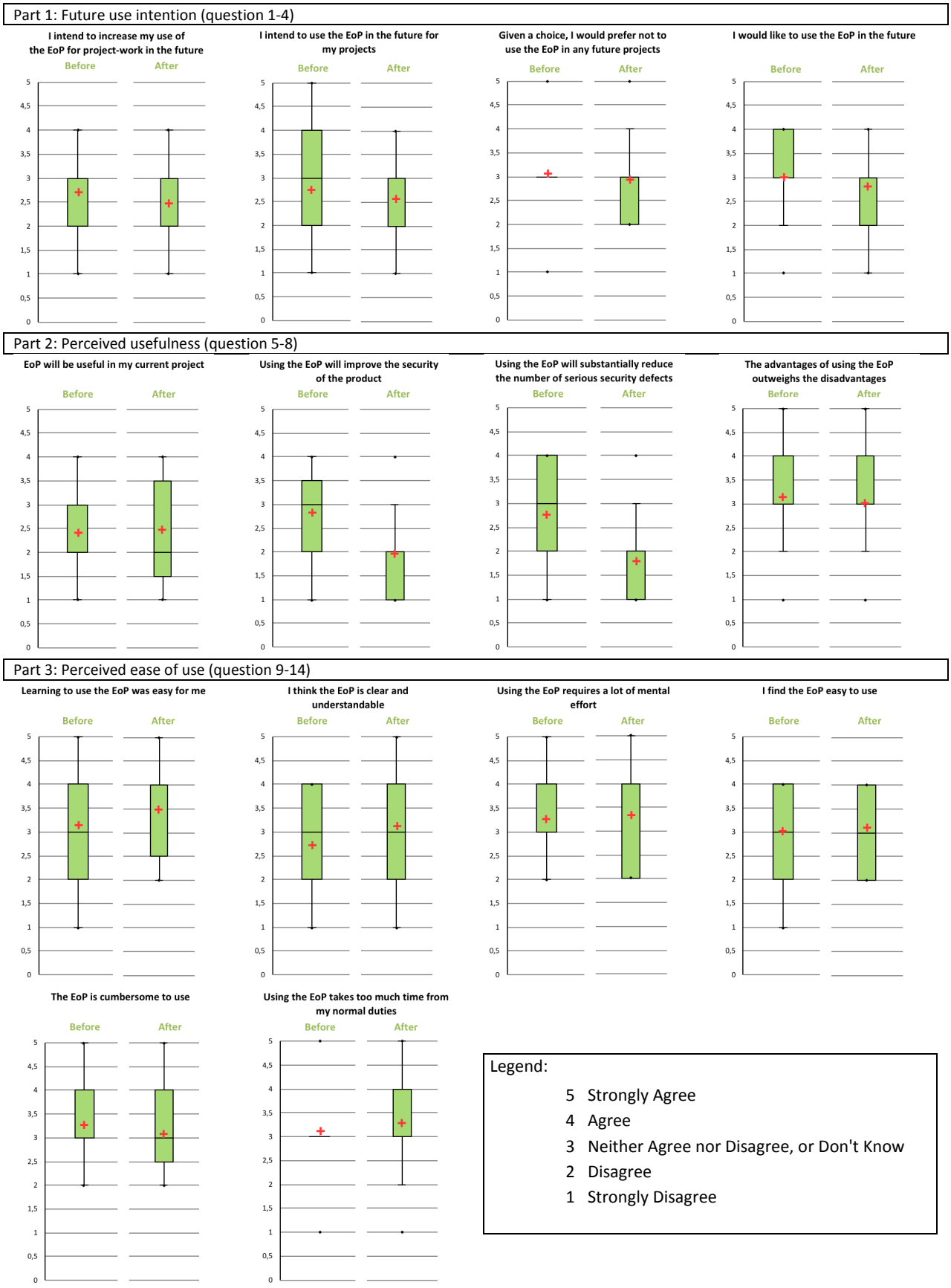

Figure 3: Results from TAM-based questionnaire

course (*Any other comments on software security?*), and responses were not directly related to acceptance.

To sum up, the questionnaire responses point to acceptance of EoP to be rather low. Based on the factors covered in the questionnaire responses, as well as open-ended responses, potential reasons for this may be: limited improvement of security as a result of using EoP; EoP taking too much time; some finding EoP to be difficult to use, and; limited perceived need for security in the projects. Overall, this is in line with what we found in the observations and in the group interviews where the students' acceptance of the EoP game varied with time. In the observations, many of the students were reluctant at first in spending time on security (and thus the game) since security was not considered important for their project. We did not do a full analysis of their projects to determine if this was in fact true, but in our initial discussion with the group we in most of these cases had to agree that based on our superficial understanding of the system they were developing, security was not a major issue for them. Despite the initial reluctance, students however seemed happy in general throughout the session. In all the observation sessions, the mood in the groups was considered to be good. However, in two of the groups discussion was slower, some students seemed a bit bored, and some lacked motivation. The differences in enthusiasm can be illustrated by the following observation notes taken after two of the sessions: *"Felt like a really good session! It seemed fun and there was a lot of discussion, also parallel discussions. And they played for longer than they had planned."*, and *"The atmosphere was kind of semi-formal. One of the students talked the most. But in general, the students seemed happy."*

The influence on the development from playing the game seems to have been very limited. In the observations, although groups were reluctant at playing the game at first, they in general found that they had some security concerns after all. In the group where enthusiasm increased the most, the following was noted by one observer: *"They started with the impression that their system does not need any security. But by the end of the exercise, they were surprised to have identified important security issues. They mentioned that it was an eye opener and gave them insight into the security issues in their system."* For other groups, the playing seemed not to be as useful despite some security issues being identified: *"After the session I think that for this type of system, with one very clear entry point, EoP seems a little too much, and they end up discussing a lot of unimportant issues. (…) Maybe a game is not needed at all - they were already aware of what their main entry point was."*

Our impression from the students in the group interview was that they were in general positive to playing the game again, but then in a project with more security issues. There is however a risk that they appeared more positive than they actually were to please the researchers. Also, it is important to note that none of the groups continued playing the game on their own for their project. Based on the feedback we got from the students in the group interviews it seems that this type of revelation that we had in some of the groups did not really result in security being taken care of in the projects. Reasons mentioned was that the customer did not think security was important in the delivery, and that they were behind firewalls. Additionally, when playing the game and in considering everything that can go wrong, there is a risk that security is given too high a priority. It seems that the sudden burst in security awareness that we experienced in some of the groups wore off when considering the realities of the full project.

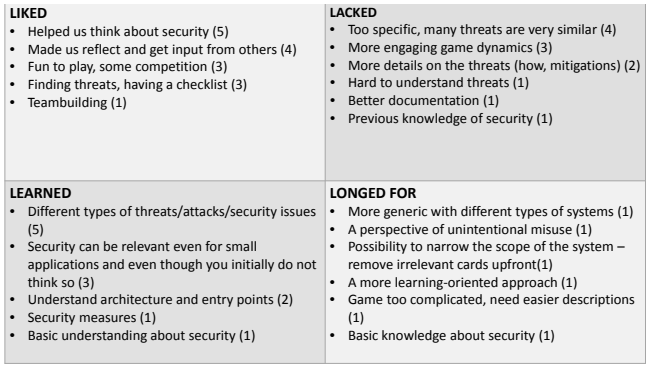

Figure 4: Result of 4L brainstorming on EoP

4.3 Lessons learned and improvements (observations, group interviews)

Lessons learned and improvements (RQ2) was studied through observations and group interview, though the questionnaire responses also gave some indications as to potential areas where improvement is needed, namely some finding EoP difficult to use and a general opinion that playing EoP takes too much time. Below we give an overview of the main areas where observations and group interview responses suggest that improvements are needed. These are grouped into two main areas: the hints on the cards and the flow of the game. For these areas, results from both observations and group interviews are presented. In addition Table 1 gives an overview of the *observer notes* related to what worked well and what was challenging. The aspects reported on were part of the observation protocol that had to be filled out after the session by both the observer and the facilitator. The count behind each statement shows for how many groups an aspect was noted by the observer/facilitator. Note that a missing count for a group does not necessarlly mean that the statement is not relevant for the group, but that none of the observers made notes on this issue for that session. Figure 4 gives an overview of the EoP students' feedback on the technique. Ten students from five out of the six groups participated in the *group interview*.

4.3.1 Hints on the card. In the sessions where we facilitated and observed the students in playing EoP, all groups managed to make use of the hints on the card, and often one or two people understood more than the others and were able to explain to the rest of the group. However, we as facilitators received many questions on the cards, and we played an important role in explaining the cards to the students and helping them relate the threat on the card to their system. In dead, we got the impression that the game would not have worked in the student groups without an external facilitator with knowledge about software security. Most students did not seem to have the necessary security background to understand the threats and apply them to their system without extensive support. When we, at the end of the session, asked students about any suggestions for changes to the game, some gave feedback that they thought

Issue	Worked well	Challenge
Understand and discuss the hints on the cards	• As a group they managed to understand the hints (2) • Some hints were useful and relevant to start discussion (2) • Managed to relate to their system (1)	• Difficult to understand the hints on the cards (4) • Many of the hints are not relevant (4) • Similar hints on the cards (1)
The model of the system	• They were able to create a model of the system (6) • They were able to use the model throughout (2)	• Did not use the details of the model (2) • Sometimes they forgot to use the model (2)
The score sheet	• Students were able to fill it out (6)	• Unclear how to assign points (4) • Filling in the score sheet breaks the flow of the game (2) • Sometimes forgot and had to be reminded (1)
Facilitation	• Facilitator helped with understanding the game, and made them reflect on threats (6)	• Input from facilitator can slow down the game (1) • Unsure if the students would have managed to play the game without an external security expert (1)
Keep track of important parts of the discussion	• Noted key things on the score sheet (6)	• Some key things seem not to be noted on the score sheet - risk missing important aspects from the discussion (2) • Not much to keep track of (1)

Table 1: Observation notes on what worked well and what was challenging in the EoP sessions

the hints on the cards were too detailed or too technical and that many cards were too similar. They suggested using simpler terms on the cards. Though students responded that they found it useful to discuss security in the group, and some liked to have hints and examples on the cards so that they did not just brainstorm, the hints on the cards wereoften perceived not to be relevant for the type of systems the students were developing.

In the group interviews students spent a lot of time providing feedback on the cards, and gave the response that a checklist-based approach is useful, but that the hints on the EoP cards were too difficult to use for their project and their knowledge level. The following quotes from the interviews illustrate this common opinion:

- *"We lacked a lot of knowledge about security in the group to be able to really play it. If you had not been there, we would not have gain much from it."*
- *"We had a lot of trouble understanding the actual threats on the cards. And needed explanations for many of them. And many of them were so similar that we thought they were the same, it was just one word difference maybe."*
- *"It was hard to try to find out where in the system that could happen, when you did not know what it was."*

Despite problems with the cards, the students responded that they had a learning outcome from playing EoP. Though learning about security threats from the hint of the cards was an important part of this learning outcome from some of the students, others stated that the learning was not so much about particular security issues, but more general; they learned that security issues can be easily overlooked and that security can be important although this may not be obvious to begin with. They also told us that they learned from the process diagram (DFD) and about the system architecture of their project.

To improve the hints on the cards, students in the group interview suggested having more concrete examples for each threats and better and more thorough explanations of each card. They also suggested including threat mitigation and the perspective of unintentional misuse, to make the cards more directly useful for learning and more relevant.

4.3.2 Flow of the game. In the sessions where we facilitated and observed the students in playing EoP, we found that the general flow of the game was usually grasped quite easily by the group, however we received some questions related to this, especially on how to assign points. There were several misunderstandings, e.g. where students were only assigned points for relevant threats, and not for winning the round with the highest card (especially this was hard to understand if the highest card was not relevant). Some groups assigned relevance-points to all cards, although the discussion indicated that some cards were not relevant. And in general, it was unclear how relevant a card should be to deserve a point. Students also asked if they had to come up with a solution to get a point. The flow of the game was easier to understand than the point-giving, but still many students had questions on rules for what card to play and some asked when to write on the score sheet etc. In general, the students did not seem interested in who won the game, and after the session some students gave the feedback that we could just as well have used the cards as a checklist.

The students were able to fill out the score sheet, though they had some questions on this as well. But the main challenge on the score sheet was the lack of notes from the discussion. The resulting score sheets looked like those in Figure 5. In none of the groups did we notice any students taking more notes from the discussion than what was put on the sheet. Still, although little information was put on the score sheet, the filling out of the score sheet was observed in some groups to break the flow of the game. In the

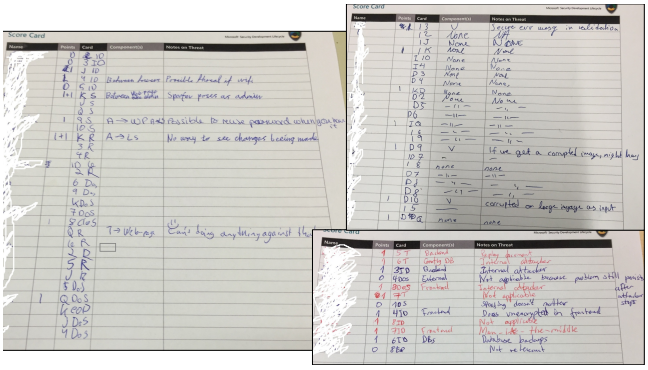

Figure 5: Example score sheets EoP

reflection part after the session, some responded that the need to discuss every card resulted in slow progress; *"The purpose of the game is to have fun, but it is too specific"* (student response as noted in the observation notes).

In the group interviews, game dynamics was discussed at length. We found that the game was considered fun by some students, it being *"something new"* and providing *"team building"*. One student also explained the gamification as a motivation for putting effort into finding security issues: *"because you want to find threats where you think there aren't any."* For the majority of the students in the group interview, the enjoyment of the game was however reduced due to game-dynamics problems. The gamification was described as *"arbitrary"* and that it gave the feeling that a side-game was happening in the background of the security discussion. Some stated that the game was almost distracting, and suggested having just a check-list instead (no game). The following quotes from the discussion illustrate the kind of feedback the students provided:

- *"It feels like you're sort of playing a random card game at the same time as you are discussing security issues."*
- *"I think that when you gamify something, (. . .) you are trying to motivate someone to do something that you think they think are boring or something by winning or playing a game, but now the motivation for playing the game was still to discuss the issues. And the game was like nothing important. So, that was kind of an issue."*
- *"I feel like the gamification aspect kind of sucks, but I don't have a suggestion as to how to improve it. I would just suggest to drop it. Replace the cards with a checklist. Throw away maybe half of them because most of them are, it seems it's just a bad attempt at filling up with cards. (. . .) I see why you think [a checklist] is boring, and to an extent I agree, but I feel like the game in its current form isn't much better, because it is essentially what you do; you go through a checklist."*

All students did however not agree that a checklist-based approach was preferable to the game.

In addition to the gamification issues, other factors impacted the flow of the game in a negative way, and made the game less interesting. In short, students wanted the game to be faster and more relevant to their systems. A high amount of what was considered irrelevant cards was a main source of frustration. Student stated: *"We felt like we often explained the same issue over and over again,"* and; *"I would think in most scenarios this game scope is too large. Not that useful because half of the cards are not relevant. (. . .) It would be like, why, we could do something better with that time probably."* Students also found the notes on the score sheets to be insufficient to remember the discussion when writing their report.

Improvements suggested by the students in the group interviews included, in addition to a checklist approach and card decks better tailored to different types of systems, to have a board game instead, to provide points based on contribution to the discussion rather than the cards on your hand, and to do notes on a computer instead because this was considered faster and because you could more easily refer to the cards that way.

5 DISCUSSION

The goal of this study, as explained in the introduction, was to assess to what extent the EoP game would be accepted as a technique in agile teams, and if possible to determine obstacles to adoption of the game. We found that acceptance of EoP in the student teams was rather low, and identified lessons learned and improvements within two main areas: the hints on the cards and the flow of the game. In the following we discuss implications of the findings regarding acceptance of EoP in agile teams. Additionally we discuss lessons learned from the research design used in this study, and identify and discuss threats to validity.

5.1 Adoption of EoP in agile teams

Although this study revealed challenges with the EoP game, it did identify some positive effects (e.g., learning about security, team building) that suggest that playing the EoP game can be useful in some settings. In the following we discuss when the playing of EoP could be useful, and when it would probably be better to use another approach to identifying security needs in a project. We also briefly address how results from playing EoP could be integrated into the project.

We started out with the premise that software security is something that should be considered for all software security projects. We still believe this is the case, but at the same time we found that in this study, security was considered of little importance in most of the projects the students got from their customers. We would claim that, although security may be *assessed* to be of little importance in a given project, it is still important to make this assessment. Playing EoP is however not necessary to make such an overall assessment, and from our experience this can be done with less effort than what is required for playing a full EoP deck. Thus, EoP is probably too much for many projects. For efficient use of EoP, there is a need for criteria to determine the level of risk and threat analysis necessary, and whether EoP or equivalent methods should be used for the project. Based on this study, EoP should then only be used if one expects that the threats towards the system are sufficiently numerous to warrant a thorough threat modeling effort.

There is a need to make the game more fun and engaging. However, there is reason to believe that the game is more fun to play if security is more important in the project. The groups that experienced a burst in security awareness during the playing of the game were the ones that seemed to enjoy the game the most. Still, this

is more the discussion than the game itself. As of now, the game dynamics do not seem to be engaging when combined with the discussions and the score sheet. The discussions and the score sheet slows the game down, but if these are cut down on, the usefulness of playing is reduced.

If choosing to use EoP or a corresponding threat modelling technique for a software development project, there is the question of when and how often to play or model. With agile development there can be new requirements in each iteration. Playing EoP in every iteration is however not an option, as this would take quite a lot of time. The number of cards in the EoP deck is quite high (74). If the time spent by the students in our study is representative for how long it takes to play the game, teams have to expect to spend between one hour and three and a half hours to play the full deck. With the students, most groups had a basic overview of their system and had started implementing when we played EoP. Although there was about a month between the time the first and the last group played EoP, we did not spot any major differences in usefulness of the technique that can be based on time of play.

The direct output of plaing EoP is a list of threats and their potential relevance to the system under development, but not any evaluation of the threats' importance and priority, though the importance may come up in the discussions. As a result, a project risks ending up with too many security requirements that only address minor security issues in the project. This is not cost effective. Students dealt with this by dropping security altogether (stating it was not that important anyway), thus going to the other extreme. However, what is needed is a process to pick the key issues to address in the project after playing EoP. This is a non-trivial task, but corresponds to what you would do for software security risk assessments. EoP can thus be input to a risk assessment activity.

A large portion of the students' feedback on the technique was related to the hints on the EoP cards. The students appreciated the support from the hints, but highly criticized their relevance and granularity. Criticism of the hints on the EoP cards is nothing new. In fact, critique that the EoP hints are not that relevant for web applications is a main motivation for the development of the OWASP Cornucopia game. However, this critique points at a more general challenge: how to trade off the need to give concrete hints with achieving relevance for a broad range of systems. Students clearly found the hints on the EoP cards to be too detailed, but we do not know how they would have responded to more general hints. As many of the students lacked knowledge about software security, the detailed hints were difficult to understand and relate to their system. However, more generic hints may also be difficult to relate to the system without expert knowledge on security. In its current state, development projects that would like to use the gamification and checklist approach to threat modeling as offered by EoP and Cornucopia should be sure to consider which game offers the hints most relevant to the type of system that is developed.

Experiences from Microsoft show that EoP may end up being used more for training in threat modelling than for doing threat modelling in real development projects [17]. In this study EoP has been used in the development projects, but with training of the students in software security as a side effect. We have not assessed the technique's usefulness for training, but it seems that the hints on the cards are too detailed for training at this level. When using EoP for training, it however reduces the implications of the relatively long time needed to play the full deck and the need to assess when to play and how to make most use of the results of playing the game in development. Instead, the increased knowledge of threats gained from playing EoP can be used to improve risk assessment and threat modelling activities performed in the development projects.

It is important to note that the positive feedback we got from students on the effects of playing the EoP game need not be specific for EoP, as illustrated by the following quote from the group interviews: *"I do not think it matters much what game you play, or what is on that list, but that you start thinking about [security]."* EoP is one very concrete way to start thinking about security, and as such can be useful for development teams that need a concrete tool to get started. This would however probably be the case for many other techniques as well.

5.2 Reflections on study design

This study uses students instead of professional software developers. By performing the study in a university course setting instead of in a professional setting we drastically reduced the effort needed to get participants to the study, we were able to control the setup of the study much more than we would have been able to in a company, and we had easy access to collect more data since the time spent on data collection by study participants was less of an issue than would be the case for professional developers. This allowed us to use several data collection methods to increase confidence in the results. In the following we discuss the usefulness of the different data collection methods, to provide recommendations for similar future studies.

Observation was time consuming, but essential for the success of the study. We found that students would probably not have managed to play EoP on their own. By facilitating and observing the groups, we additionally got first-hand experience with how the game were used and thus know what we are studying. However, with researchers acting as facilitator and observer we potentially influence the students quite a lot. Additionally we add a role to the EoP game that is not there initially, that of the facilitator.

Though our opinion of the adoption and challenges of the EoP game from the observations was very much in line with what the students expressed in the *group interviews*, these interviews gave the added benefit of hearing students explain their experiences in their own words. Additionally, the students provided more suggestions for improvements than in the session when we did observations. The key benefit of doing the group interview, however, came in getting access to how the EoP game and its output were used after the session we facilitated. With only observations as our data collection method, we would have risked to believe acceptance of EoP was higher than what ended up being the case, as they were more positive in the end of the observation session than they were in the group interviews.

Questionnaire results were less useful than the observation and group interview results, since they did not add much to what we had already collected. However, they came with the benefit of getting responses from most of the students (group interviews may have included the more motivated students from each group), and

allowed us to see the rather low acceptance, and the decline in acceptance. The effort needed to design the questionnaire and collect data was limited, compared to the effort needed for observations and group interviews, since we could base the questions on an existing questionnaire used in other studies.

Based on our experiences from this study, we recommend that similar studies use observations, but that observations are combined with other data collection methods (e.g. group interviews or questionnaire) to increase confidence in the results.

5.3 Threats to validity

In this study it is difficult to separate the effect of the technique itself from other factors, such as motivation, skills, group dynamics, and our influence as researchers. Therefore we have aimed to be aware of the impact of the context throughout the study. One way we did this is by having the first author be supervisor of one student group. Additionally, we made sure we reflected on our role as researchers and took this into account in the analysis (reflection on our influence as researchers was part of the template for observation notes). As part of this, we made it clear for students that their opinion on the security techniques would not have any impact on their grade in the course. We as researchers did not have any influence on the grades the students got.

This study involves students, and thus not professional software developers. There are studies available that show that students in the later parts of their studies can be used with success in studies instead of professional software developers in some cases, namely for understanding dependencies and relationships in software engineering [7] and for requirements selection [20]. The topic of this study is related to, but not identical to, those studies. We do not claim that the results from our study can be generalised to software developers in general, but believe it to be likely that many of the same issues that we found would apply also in professional settings, in particular since many professionals in small and medium sized development organisations (as those dominating the Norwegian market) would also be considered novices when it comes to EoP and threat modelling, and have limited software security training [10]. However, the context would be different. Although the students in our study did have an external customer and the aim of the course is to have a setting that is as similar as possible to a real development project, the students had some concerns that professionals would not have (e.g. the report and getting a good grade) and this may have impacted the results.

6 CONCLUSION

We have studied the use of the Microsoft EoP game in capstone development projects with 4th year computer science studies. In this study we identified positive effects from playing the game, especially related to learning about software security and aiding discussions about software security. These are important effects if wanting to improve software security in projects, as well as software security competence among developers. However, the game itself has weaknesses that may impact its adoption in development projects. In particular, the game dynamics are not considered engaging, the game may take a long time to play, and the hints on the cards are not suited for many projects. We conclude that EoP is probably most suited for projects with high security concerns, and for training purposes.

ACKNOWLEDGMENT

This work was supported by the SoS-Agile: Science of Security in Agile Software Development project, funded by the Research Council of Norway (grant number 247678). Thanks to the Customer-Driven Project organizers (Prof. Jon Atle Gulla and Prof. John Krogstie) and the participating students at NTNU. Thanks also to Prof. Pekka Abrahamsson and Prof. Laurie Williams for input on the study design. Thanks to Prof. Guttorm Sindre for input on the contents of the paper.

REFERENCES

[1] Icek Ajzen and Martin Fishbein. 1980. *Understanding attitudes and predicting social behavior.* Prentice-Hall.
[2] Paulo Caroli and Taina Caetano. 2015. *Fun Retrospectives - Activities and ideas for making agile retrospectives more engaging.* Leanpub, Layton.
[3] Fred D Davis. 1985. *A technology acceptance model for empirically testing new end-user information systems: Theory and results.* Ph.D. Dissertation. Massachusetts Institute of Technology.
[4] Fred D Davis. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly* (1989), 319–340.
[5] T. Dybå and T. Dingsøyr. 2009. What Do We Know about Agile Software Development? *IEEE Software* 26, 5 (Sept 2009), 6–9.
[6] Tore Dyba, Nils Brede Moe, and Edda M Mikkelsen. 2004. An empirical investigation on factors affecting software developer acceptance and utilization of electronic process guides. In *Software Metrics, 2004. Proceedings. 10th International Symposium on.* IEEE, 220–231.
[7] Martin Höst, Björn Regnell, and Claes Wohlin. 2000. Using students as subjects – a comparative study of students and professionals in lead-time impact assessment. *Empirical Software Engineering* 5, 3 (2000), 201–214.
[8] Michael Howard and Steve Lipner. 2006. *The Security Development Lifecycle.* Microsoft Press.
[9] Martin Gilje Jaatun. 2012. Hunting for Aardvarks: Can Software Security Be Measured? In *Multidisciplinary Research and Practice for Information Systems*, Gerald Quirchmayr, Josef Basl, Ilsun You, Lida Xu, and Edgar Weippl (Eds.). Lecture Notes in Computer Science, Vol. 7465. Springer, 85–92.
[10] Martin Gilje Jaatun, Daniela S. Cruzes, Karin Bernsmed, Inger Anne Tøndel, and Lillian Røstad. 2015. Software Security Maturity in Public Organisations. In *Information Security*, Javier Lopez and Chris J. Mitchell (Eds.). Lecture Notes in Computer Science, Vol. 9290. Springer International Publishing, 120–138.
[11] Long Li. 2008. A critical review of technology acceptance literature. *Department of Accounting, Economics and Information Systems, College of Business, Grambling State University.* (2008).
[12] Gary McGraw. 2006. *Software Security: Building Security In.* Addison-Wesley.
[13] G McGraw, S Migues, and J West. 2015. *BSIMM 6.* Technical Report.
[14] Microsoft. [n. d.]. Microsoft Security Development Lifecycle. ([n. d.]). https://www.microsoft.com/en-us/SDL
[15] Microsoft. 2009. *Security Development Lifecycle for Agile Development, Version 1.0.* Technical Report.
[16] OpenSAMM. [n. d.]. Software Assurance Maturity Model (SAMM): A guide to building security into software development. ([n. d.]). http://www.opensamm.org/.
[17] Adam Shostack. 2014. Elevation of Privilege: Drawing Developers into Threat Modeling. In *2014 USENIX Summit on Gaming, Games, and Gamification in Security Education (3GSE 14).*
[18] Adam Shostack. 2014. *Threat Modeling: Designing for Security.* Wiley.
[19] Forrest Shull and Nancy Mead. 2016. Cyber Threat Modeling: An Evaluation of Three Methods. (2016). https://insights.sei.cmu.edu/sei_blog/2016/11/cyber-threat-modeling-an-evaluation-of-three-methods.html
[20] Mikael Svahnberg, Aybüke Aurum, and Claes Wohlin. 2008. Using students as subjects – an empirical evaluation. In *Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement.* ACM, 288–290.
[21] Viswanath Venkatesh and Fred D Davis. 1996. A model of the antecedents of perceived ease of use: Development and test. *Decision sciences* 27, 3 (1996), 451–481.
[22] Laurie Williams, Andrew Meneely, and Grant Shipley. 2010. Protection poker: The new software security game. *IEEE Security and Privacy* 8, 3 (2010), 14–20.