An Empirical Study on the Comprehensibility of Graphical Security Risk Models Based on Sequence Diagrams

Vetle Volden-Freberg and Gencer Erdogan

SINTEF Digital, Oslo, Norway
{vetle.volden-freberg,gencer.erdogan}@sintef.no

Abstract. We report on an empirical study in which we evaluate the comprehensibility of graphical versus textual risk annotations in threat models based on sequence diagrams. The experiment was carried out on two separate groups where each group solved tasks related to either graphical or textual annotations. We also examined the efficiency of using these two annotations in terms of the average time each group spent per task. Our study reports that threat models with textual risk annotations are equally comprehensible to corresponding threat models with graphical risk annotations. With respect to efficiency, however, we found out that participants solving tasks related to the graphical annotations spent on average 23% less time per task.

Keywords: Security risk models, empirical study, comprehensibility.

1 Introduction

Security risk models based on sequence diagrams are useful to design and select tests focusing on security risks the system under test is exposed to [25, 4]. This testing strategy is referred to as risk-driven security testing [8]. The field of risk-driven testing needs more formality and proper tool support [5]. To address this need, we developed a tool to help security testers design and select security tests based on the available risk picture by making use of risk-annotated sequence diagrams. The tool is freely available as a plugin [2] for Eclipse Papyrus [23].

We specifically developed the tool to support the CORAL approach, which is a model-based approach to risk-driven security testing [4]. The CORAL approach provides a domain specific modeling language that captures security risks in terms of sequence diagrams annotated with graphical icons representing risk constructs. However, as part of the development of the tool, we conducted an empirical study to evaluate the comprehensibility of the graphical icons representing risk constructs in the CORAL language versus corresponding textual representation of the risk constructs in terms of UML stereotypes [22].

The contribution of this paper is the empirical study. We believe the study is useful for the security risk community to better understand the effectiveness of security risk models based on sequence diagrams. The study may also be

The final publication is available at Springer via https://doi.org/10.1007/978-3-030-12143-3_1

useful for others who wish to conduct similar empirical studies, as well as for tool developers who consider to develop similar tools.

The overall goal of our empirical study was to investigate, from the perspective of comprehensibility, whether it is better to use the graphical icons provided by the CORAL language to represent risk constructs or if it is better to use corresponding textual representation in terms of UML stereotypes. Throughout this paper, by graphical annotations, we mean representing risk constructs using graphical icons provided by the CORAL language [4], and by textual annotations, we mean representing risk constructs using UML stereotype annotations [22]. Based on this overall goal, we defined two research questions:

RQ1 Will the use of either graphical or textual annotations to represent risk constructs in threat models based on sequence diagrams affect the objective performance of comprehensibility?

RQ2 Will the use of either graphical or textual annotations to represent risk constructs in threat models based on sequence diagrams affect the participants' efficiency in solving the provided tasks?

In Sect. 2, we present the kind of threat models considered in our empirical study. In Sect. 3, we present an overview of our research method which consists of three main steps: experiment design, experiment execution, and experiment data analysis. Sections 4–6 present our empirical study with respect to the aforementioned steps of our research method. In Sect. 7, we discuss our results in relation to research questions RQ1 and RQ2 as well as threats to validity. In Sect. 8, we discuss related work. Finally, in Sect. 9, we provide our conclusions.

2 Threat Models Considered in the Empirical Study

It is beyond the scope of this paper to explain in detail the CORAL language [4] as well as UML sequence diagrams and stereotypes [22]. However, it is necessary to illustrate the kind of threat models considered in our empirical study.

Figures 1(a) and 1(b) illustrate two semantically identical threat models using graphical and textual risk annotations, respectively. Figure 1(a) is developed using the CORAL language, while in Fig. 1(b) we have replaced the graphical risk annotations with corresponding textual risk annotations using UML stereotypes. The graphical icons representing risk constructs in the CORAL language are inspired by corresponding graphical icons in CORAS, which is a model-driven approach to risk analysis [17].

Both threat models in Figs. 1(a) and 1(b) illustrate a stored cross-site scripting attack [32] on an example web application that stores feedback from users, such as an online forum. The *hacker* first clicks on a button on the web application to add new feedback (*clickAddNewFeedback*), and then updates the feedback with malicious script (*updateFeedbackText(script)*). This causes the unwanted incident *Hacker's script stored in database*, which in turn has an impact on the asset *Integrity of source code* because the script may be executed by the browser when accessed by a user, which modifies the content of the web page.

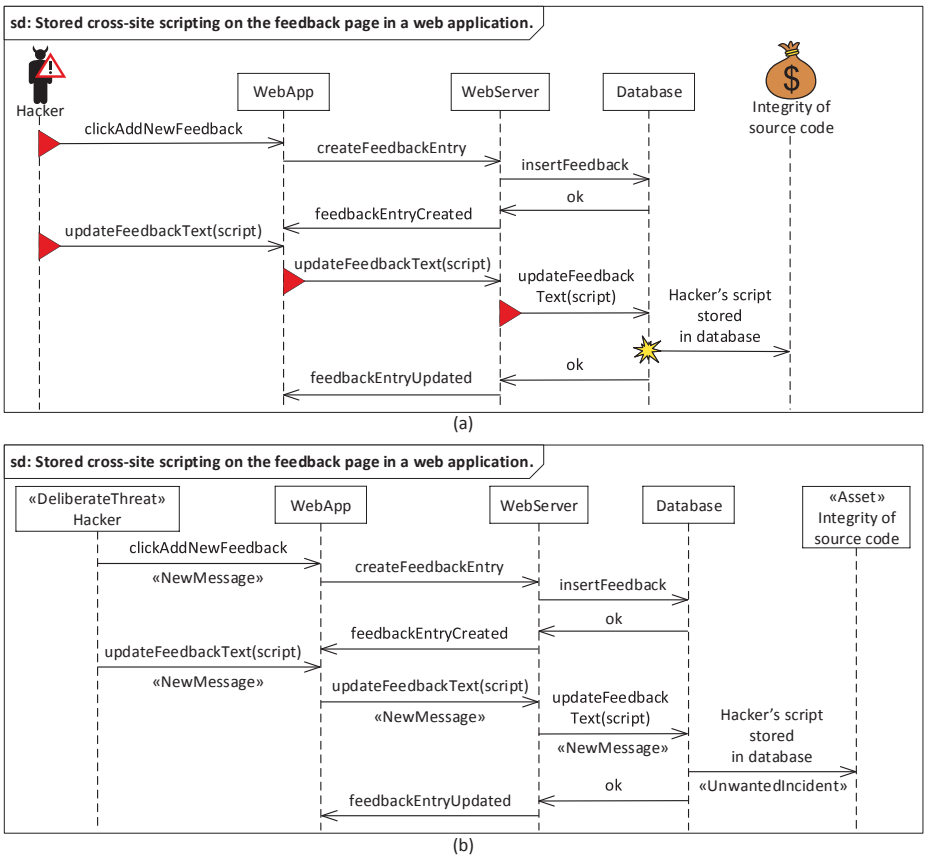

Fig. 1. (a) Threat model with graphical risk annotations based on the CORAL language. (b) Threat model with corresponding textual risk annotations using stereotypes.

From a conceptual point of view, the graphical icon representing a hacker in Fig. 1(a) is referred to as a *deliberate threat* in CORAL [4]. A deliberate threat is a human threat that has malicious intents. During risk assessment, we assess security risks that may harm certain security *assets* we want to protect. In CORAL, an asset is illustrated by a moneybag icon. Messages initiated by a threat with the intention of manipulating system behavior are referred to as *new messages*. A new message is represented by a red triangle which is placed at the transmitting end of the message. An *unwanted incident* is represented by a message with a yellow explosion sign at the transmitting end and conveys that an asset is harmed or its value is reduced. As already mentioned, Fig. 1(b) "mirrors" Fig. 1(a) and represents the above risk constructs as stereotypes. The CORAL language defines additional risk constructs not captured by the threat model in Fig. 1(a), such as *altered messages* and *deleted messages*. These risk

constructs were also included in our empirical study. The reader is referred to [4] for a detailed explanation of the CORAL language.

3 Research Method

Figure 2 shows an overview of our research method which is based on guidelines provided by the widely accepted quality improvement paradigm framework (QIP) [1]. The QIP framework is a generic improvement cycle which can also be used as a framework for conducting empirical studies [30]. We made use of the QIP framework to conduct an empirical study in terms of a controlled experiment [30]. All data related to our empirical study is fully documented and available online including experiment design, execution, and data analysis [29].

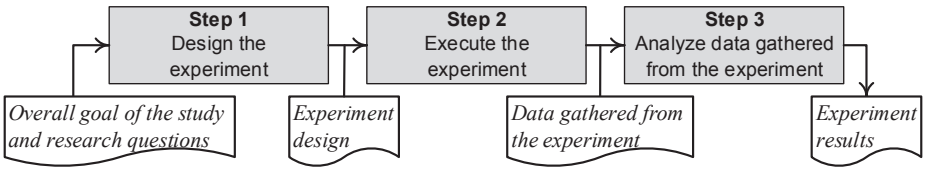

Fig. 2. Research method.

In Step 1, we designed the experiment with respect to the overall goal of the study and research questions defined in Sect. 1. The experiment was designed in terms of formulating the hypothesis, identifying independent and dependent variables, defining the experiment process, and preparing experiment material.

Based on the experiment process defined in Step 1, we executed the experiment in Step 2 as follows. First, we identified the subjects of the experiment and then conducted a demographic survey and based on that divided the participants fairly into groups A and B. Then, we provided Group A and Group B training material for the graphical and textual annotations, respectively. Finally, we conducted a questionnaire focusing on the graphical and textual annotations answered by Group A and Group B, respectively. The demographic survey and the questionnaire were carried out using the online survey tool Eval&Go [6]. The training material was provided via email subsequent to the demographic survey, but prior to the questionnaire.

In Step 3, we analyzed the data gathered from the online questionnaire in terms of visualizing data, using descriptive statistics, and carrying out hypothesis testing of the hypothesis defined in Step 1.

4 Experiment Design

In this section, we first formulate the hypothesis and identify the independent and dependent variables before presenting the experiment material. The process of the experiment is already described in Sect. 3.

4.1 Formulate Hypothesis and Identify Variables of the Experiment

Following the goal and research questions in Sect. 1, we devised hypothesis H_0.

H_0: Threat models with textual annotations are equally comprehensible in comparison to threat models with graphical annotations.

There exists no empirical evidence to support that either annotation is superior to the other. For this reason, we formulate alternative hypotheses H_1 and H_2.

H_1: Threat models with textual annotations are more comprehensible than threat models with graphical annotations.

H_2: Threat models with graphical annotations are more comprehensible than threat models with textual annotations.

In order to assess and compare the two different annotations with respect to the hypothesis, we need to identify independent and dependent variables for our experiment. Independent variables are considered to be the input to an experiment. In effect, the motivation for an experiment is to investigate whether variations in the independent variables have an effect on the dependent variables (output of the experiment) [30].

The independent variable in our experiment is the threat model representation whose notation can hold two different values: graphical or textual risk annotations. The dependent variables are comprehensibility and efficiency.

Comprehensibility refers to the ability of the participant to develop and comprehend models [24]. This is measured by effectiveness, which in our case is the degree to which the participant is able to reach successful task accomplishment, taking into account the task scores in the questionnaire.

Efficiency refers to the ability of the participant to develop and comprehend a model relatively quickly according to the syntax and semantics of the modelling language [24]. Having in mind that there currently is no empirical evidence to suggest what is a relatively quick comprehension of the threat models considered in our experiment, comparison between the two populations (Group A and Group B) with respect to efficiency is necessary.

4.2 Experiment Material

We prepared the experiment material in terms of a letter of consent to be signed by the participants, a demographic survey, training material, and a questionnaire with tasks to solve.

In order to communicate with the participants and divide the participants fairly into two groups, names and corresponding email addresses were recorded and associated with their respective response to the demographic survey. With respect to training material, we prepared one document for each group explaining the experiment. Group A received training material for the graphical annotations, while Group B received training material for the textual annotations. Due to space limitations, we will not go into further details of the training material. However, this is thoroughly documented and available online [29].

As mentioned in Sect. 3, we used the tool Eval&Go to conduct the online demographic survey and questionnaire. In addition to Eval&Go, we considered the tools SurveyGizmo, SurveyMonkey, Zoho Survey, Google Forms, SurveyPlanet, LimeSurvey, and QuestionPro [29]. We selected Eval&Go because it was the only tool fulfilling most of our requirements (two out of three): 1) provide a timer functionality to enforce a time limit per question and 2) not store e-mail/IP addresses, browser information or cookies, as well as prevent the possibility to trace a response to a particular participant. Point 2) was important to take into account the anonymity of the participants. Our third requirement was a time stamp feature to record the individual time *each participant spent per task*. None of the tools provided this feature. However, Eval&Go did record the average time *each group spent per task*.

The demographic survey consists of 22 questions (Q) and were grouped into the following categories to best help us divide the participants fairly into two groups: occupation (Q1-Q4), work experience within IT or engineering (Q5-Q6), academic degree (Q7-Q12), knowledge of UML modeling (Q13), knowledge of sequence diagrams (Q14), work experience within model-driven engineering (Q15-Q16), knowledge and work experience within risk assessment or risk analysis (Q17-Q19), knowledge and work experience within user interface design or usability (Q20-Q22). The questions related to "knowledge" were answered using a Likert scale with the following five values {no knowledge, minor knowledge, some knowledge, good knowledge, expert}.

The tasks in our experiment address comprehensibility and are therefore focused on model-reading [28, 10]. To observe noticable difference in comprehensibility between the two control groups, it is important to have a mixture of easy and difficult tasks [11, 12]. For this reason, we divided the questionnaire in two parts. Part 1 consists of 6 less complicated tasks concerned with identifying different risk constructs in a threat model, for example, *How many altered messages are modeled in the threat model?* Part 2 consists of 7 more complex tasks focusing on model interpretation, for example, *According to the model, describe how the hacker causes the unwanted incident to occur*. Developing tasks with an appropriate level of complexity is not trivial. The tasks were therefore developed in several iterations where for each iteration a third researcher reviewed the tasks and provided feedback to the authors. In total, there were seven iterations until the task set for the questionnaire was finalized. With respect to task scores, a participant can obtain a maximum score of 12 points in Part 1, and a maximum of 15 points in Part 2. This is because in some of the tasks it is possible to obtain more than one point. Wrong or no answer to a question results in zero points. The complete task set for the questionnaire as well as the questions for the demographic survey are available online [29].

To avoid potential situations where a participant overestimates the amount of time required for a given task or that the easier tasks are correctly answered by most of the participants, we enforced a time limit per question. These time limits were also reviewed in the iterative process of developing the tasks. Each task in Part 1 has a time limit of 60 seconds. The last six out of the seven tasks in

Part 2 were presented to the participant as three separate pairs of tasks because each pair addresses one threat model. The first out of the seven tasks in Part 2 has a time limit of 180 seconds, while each pair of tasks in Part 2 has a time limit of 300 seconds. This means that Part 1 has a total time limit of 6 minutes (6 tasks × 60s = 360s), while Part 2 has a total time limit of 18 minutes (180s + 3 pairs of tasks × 300s = 1080s). Thus, the total allocated time for all 13 tasks is 360s + 1080s = 24 minutes.

5 Experiment Execution

The participants were recruited through our network and selected based on two criteria: 1) hold or being in pursuit of a degree within computer science and 2) have knowledge of programming and/or have technical experience within ICT. This type of sampling is referred to as purposive sampling [26]. We recruited in total 16 participants of which 10 were graduates and 6 were undergraduates within the field of computer science.

On June 14th 2017, the invitations for the demographic survey were sent to all participants by email. By June 18th 2017, all participants had submitted their answers. We identified four groups of participants based on their occupation: five students, eight working, two studying and working, and one specified as other. The participants were divided fairly in two groups with respect to their academic degree, years of work experience, and knowledge profiles. Table 1 shows the participants in Groups A and B, where Group A represents the participants solving tasks related to graphical annotations, while Group B represents the participants solving tasks related to textual annotations. With respect to academic degree (AD), both groups have three participants with a bachelor's degree and five participants with a master's degree.

Group A has on average 2 years of work experience (WE), while Group B has on average 5 years of work experience. This difference is because Group B has one participant with 20 years of work experience. However, to keep the groups balanced, we placed five participants with work experience in each group. None of the participants had work experience with model-driven engineering (MDE-WE). One participant had two years of work experience with risk assessment or analysis (R-WE). Finally, four participants had work experience with user interface design or usability (UI-WE), with one, two, four and eight years of experience, respectively.

The columns UML (UML modeling), SD (sequence diagrams), R (risk assessment or analysis), and UI (user interface design or usability) show the participants' assessment of their own knowledge within these domains. The digits in these columns correspond to the steps in the Likert scale defined in Sect. 4. That is, the digit 0 corresponds to "no knowledge", 1 corresponds to "minor knowledge", and so on. As shown in Table 1, the average level of knowledge (with respect to UML, SD, R, and UI) are similar for both groups, except for UML modeling, where Group A has a slightly better score (2.12) compared to Group B (2).

Table 1. Participants of Groups A and B. B=Bachelor's degree, M=Master's degree.

	Participant	AD	WE	UML	SD	MDE-WE	R	R-WE	UI	UI-WE
Group A (graphical)	**P1**	B	0	2	2	0	2	0	2	0
	P2	B	2	1	1	0	1	0	0	0
	P3	B	1	3	3	0	2	0	0	0
	P4	M	1	2	2	0	0	0	2	0
	P5	M	0	2	2	0	1	0	2	0
	P6	M	4	2	1	0	1	0	2	2
	P7	M	8	2	2	0	2	0	4	8
	P8	M	0	3	3	0	3	0	0	0
	Average	**M**	**2**	**2.12**	**2**	**0**	**1.5**	**0**	**1.5**	**1.25**
Group B (textual)	**P9**	B	1	2	2	0	0	0	1	1
	P10	B	20	2	2	0	1	0	2	0
	P11	B	5	2	2	0	1	0	1	0
	P12	M	0	2	2	0	2	0	2	0
	P13	M	5	1	1	0	1	0	2	0
	P14	M	9	2	2	0	3	2	2	4
	P15	M	0	2	2	0	2	0	1	0
	P16	M	0	3	3	0	2	0	1	0
	Average	**M**	**5**	**2**	**2**	**0**	**1.5**	**0.25**	**1.5**	**0.62**

Having divided the participants fairly in two groups, we distributed the questionnaire containing the tasks on June 18th 2017 via email where we included an anonymous link to the survey. All answers were submitted by June 25th 2017. Table 2 shows the complete task scores. The tasks T1-T6 belong to Part 1 of the questionnaire, while tasks T7-T13 belong to Part 2 of the questionnaire.

With respect to time usage, we recorded via the tool Eval&Go the average time *per group* spent for each task in the questionnaire (see Table 3). The column $\bar{x}(t_A)$ shows the average time (seconds) Group A spent for each task, while column $\bar{x}(t_B)$ shows the average time Group B spent for each task. Recall that the last six out of the seven tasks in Part 2 of the questionnaire were presented to the participant as three pairs of tasks. The column Δt shows the difference in average time Group B spent compared to Group A, i.e., $\Delta t = \bar{x}(t_B) - \bar{x}(t_A)$. The column % shows this difference in terms of percentage. Finally, a positive value for Δt and % indicates that Group B spent more time than Group A, while a negative value indicates that Group B spent less time than Group A.

6 Experiment Data Analysis

Figure 3 shows box plots of the total score for Group A and Group B produced by IBM SPSS [9], which is the tool we used for statistical analysis. The box plot on the left hand side in Fig. 3 represents the distribution of Group A, while the box plot on the right hand side in Fig. 3 represents the distribution of Group B. The box plot for Group A reports an outlier of record 4 (i.e., Participant 4), having a total score of 5 (see Table 2). This record has a low score because the

Table 2. Task scores for Group A and Group B. T=Task, P=Participant.

	Group A									**Group B**								
	P1	P2	P3	P4	P5	P6	P7	P8	Avg.	P9	P10	P11	P12	P13	P14	P15	P16	Avg.
T1	1	1	1	0	0	1	1	1	0.75	1	0	1	1	1	1	0	1	0.75
T2	1	1	1	0	1	1	1	1	0.875	1	0	1	1	1	1	0	1	0.75
T3	1	1	1	0	1	1	1	1	0.875	1	1	1	1	1	1	0	1	0.875
T4	3	1	2	0	2	3	0	2	1.625	3	2	0	3	3	3	1	3	2.25
T5	2	0	2	1	2	2	2	2	1.625	2	2	1	2	2	2	1	2	1.75
T6	4	4	4	2	4	4	3	4	3.625	4	4	4	4	3	4	4	3	3.75
T7	0	2	0	0	0	2	1	0	0.625	0	0	0	0	0	0	0	0	0
T8	1	1	0	1	1	1	1	0	0.75	1	1	1	1	1	0	1	1	0.875
T9	1	3	0	0	3	2	1	3	1.625	2	2	0	2	2	0	0	2	1.25
T10	1	1	0	1	1	1	1	1	0.875	1	1	1	1	0	1	1	1	0.875
T11	2	0	0	0	0	0	3	3	1	2	2	0	3	3	0	0	2	1.5
T12	2	3	2	0	0	1	0	2	1.25	3	3	1	3	2	0	0	2	1.75
T13	1	2	2	0	2	2	2	2	1.625	2	2	0	2	2	2	0	2	1.5
Total	20	20	15	5	17	21	17	22	17.125	23	20	11	24	21	15	8	21	17.875

Table 3. The average time per group spent for each task.

Task #	$\bar{x}(t_A)$	$\bar{x}(t_B)$	Δt	**%**
1	22	31	9	40.91%
2	22	24	2	9.09%
3	13	21	8	61.54%
4	49	46	−3	−6.12%
5	36	41	5	13.89%
6	44	51	7	15.91%
7	119	145	26	21.85%
8+9	156	233	77	49.36%
10+11	167	205	38	22.75%
12+13	232	232	0	0.00%
Total	**860**	**1029**	**169**	

participant gave several blank answers. This might be because the participant did not know how to solve the tasks. It can also be because the participant was not interested in participating. For this reason, throughout this section, we analyze both situations; one where the outlier is included, and one where it is excluded. In addition, we analyze the data from three perspectives with respect to the task scores: total score, total score Part 1 only, total score Part 2 only.

In general, the box plots of the three different perspectives (Figs. 3-5) do not give any clear indication whether there is any significant difference between the two groups. It may seem, however, that Group B has a slight improvement over Group A. If we in the total score (Fig. 3) exclude the outlier from Group A, the distributions of both groups seem to be approximately normally distributed. However, if we look at the total score for Part 1 (Fig. 4) and the total score for Part 2 (Fig. 5) individually, the distributions are not as normally distributed. We

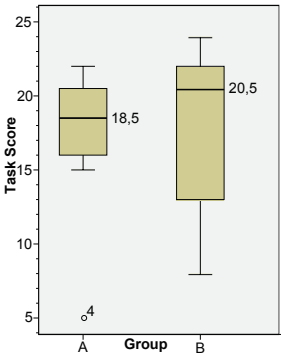

Fig. 3. Total score.

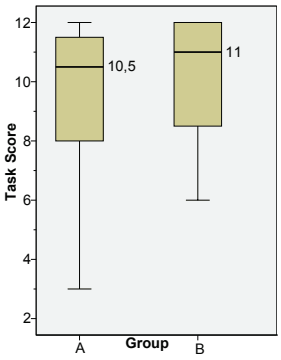

Fig. 4. Total score Part 1.

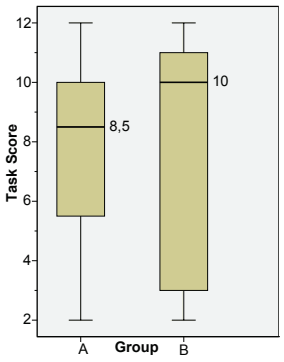

Fig. 5. Total score Part 2.

therefore proceed to apply additional descriptive statistics to investigate further whether there is any significant difference between the two groups.

Table 4 shows additional descriptive statistics of the three different perspectives in terms of mean, variance, standard deviation, standard error, skewness, and kurtosis [7, 18]. The columns A and B show descriptive statistics of the total score for Group A and Group B, respectively, while the column A* shows descriptive statistics of the total score for Group A excluding the outlier.

Table 4. Descriptive statistics of the total task scores.

	Total score			**Total score Part 1**			**Total score Part 2**		
	A	**B**	**A***	**A**	**B**	**A***	**A**	**B**	**A***
Mean (Avg.)	17.13	17.88	18.86	9.38	10.13	10.29	7.75	7.75	8.57
Variance	29.554	34.411	6.476	9.125	4.982	2.905	11.357	18.214	6.952
Std. Deviation	5.436	5.866	2.545	3.021	2.232	1.704	3.370	4.268	2.637
Std. Error	1.922	2.074	0.962	1.068	0.789	0.644	1.191	1.509	0.997
Skewness	-1.862	-0.827	-0.373	-1.515	-1.029	-0.618	-0.638	-0.579	-0.570
Kurtosis	3.956	-0.812	-1.314	2.279	-0.069	-1.396	-0.291	-2.097	0.547

The high mean scores for "Total score Part 1" compared to "Total score Part 2" indicate that the participants had a good understanding of the tasks in Part 1, while the lower mean scores for "Total score Part 2" indicate that the participants struggeled more with the tasks in Part 2, as expected. Moreover, Table 4 shows that Group B is less skewed and more normally distributed compared to Group A. Skewness and kurtosis are measures used to check whether the data follow a normal distribution [18]. A skewness value close to zero indicates a symmetrical distribution, positive and negative values indicate a right-skewed and left-skewed distribution, respectively. Both groups have negative skewness in all perspectives included in Table 4. This confirms our observation from the box plots in Figs. 3-5.

The kurtosis gives an indication of how big the *tails* (distance from the mean) of the distribution are [18]. A normal distribution has a kurtosis value of 0, while positive and negative values indicate larger and smaller tails, respectively. A distribution with a positive kurtosis implies a distribution that is more steep towards the top than a normal distribution. A negative kurtosis implies a distribution that is more flat towards the top than a normal distribution. The values for kurtosis that fall within an acceptable range to be classified as acceptable for a normal distribution is (for a sample size $n = 25$) minimum -1.2 and maximum 2.3 [18]. Note that for our sample sizes (which are $n = 8$) the ranges do not cover this size. For this reason, the kurtosis may be biased, however, it does give an indication whether the distribution is approximately normally distributed or not. The kurtosis values for Group A and Group B differ from each other by some margin. For example, looking at the values for "Total score" in Table 4, we see that the kurtosis for Group B is well within the acceptable range (-0.812). Group A is outside the acceptable range with a difference of $3.956 - 2.3 = 1.656$. Group A is also outside the range when excluding the outlier (-1.314).

In general, the descriptive statistics in Table 4 shows that the two groups are similar, with Group B having a slightly better score than Group A (mean). With the exclusion of the outlier (column A*), however, Group A seems to have a better mean score with a smaller median than Group B. Further, this yields a higher precision of measurement for Group A than Group B, with a lower standard deviation and standard error. Additionally, Group A gains a smaller skewness value, indicating an approximately normal distribution. Thus, we argue that Group B has performed better than Group A. However, the exclusion of Group A's outlier suggests that Group A performed better than Group B. It is important to note, however, that the differences are small and we cannot conclude whether there is a significant difference between the groups only by comparing their descriptive statistics for either of the three perspectives. For this reason, we proceed to hypothesis testing to answer our null hypothesis H_0.

In our experiment we applied two conditions (graphical versus textual annotations) with different participants taking part in each condition. For this reason, an appropriate hypothesis testing method is the independent samples t-test, also called an unpaired t-test [9, 27]. There are two variants of this t-test, one assuming equal variances, and one assuming unequal variances. To determine which variant to use, we carried out Levene's test for equality of variances for each of the three perspectives (including and excluding the outlier in Group A). Moreover, for the t-test, we used a 95% confidence interval and degrees of freedom given by $df = n_1 + n_2 - 2$, which in our case is $df = 8 + 8 - 2 = 14$. Table 5 summarizes the results from the t-tests for testing the null hypothesis H_0 (defined in Sect. 4) for all the perspectives mentioned above. The symbol * in Table 5 denotes the t-tests in which we have excluded the outlier in Group A from the total score. The column "Statistically significant?" provides a yes/no value depending on whether the t-statistics indicate a significant effect between Group A and Group B.

Table 5. Summary of the independent samples t-tests.

	T-statistics	**Statistically significant?**	**Accept H_0?**
Total score	$t = -0.265, p = 0.795$	No	Yes
Total score*	$t = 0.430, p = 0.677$	No	Yes
Total score Part 1	$t = -0.565, p = 0.581$	No	Yes
Total score Part 1*	$t = -0.265, p = 0.795$	No	Yes
Total score Part 2	$t = 0.000, p = 1.000$	No	Yes
Total score Part 2*	$t = 0.454, p = 0.658$	No	Yes

7 Discussion

As shown in Table 5, the independent samples t-tests for all perspectives, including the perspectives in which the outlier in Group A is excluded, report on the acceptance of our null hypothesis. This means that the comprehensibility of threat models with either graphical or textual annotations with respect to the given task set is equally comprehensible. Thus, to answer **RQ1**, the use of either graphical or textual annotations seem not to affect the objective performance of comprehensibility. That is, there is no evidence indicating that graphical annotations are more effective than textual annotations or vice versa, in terms of comprehension, with respect to the threat models considered in our study.

With respect to **RQ2**, we examine the average time each group spent per task and note that Group B spent considerably more time than Group A for the whole task set (see Table 3). Furthermore, eight out of the ten reported differences seen from column Δt in Table 3 are in favour of Group A. On average, Group B spent approximately 23% more time *per task* compared to Group A. Thus, to answer RQ2, this indicates that graphical annotations aid the participant in more efficient task solving, compared to textual annotations. This claim is further substantiated by Moody [20], who argues that visual representations are more efficient than textual because they are processed in parallel by the visual system, while textual representations are processed serially by the auditory system. This is because textual representations are one-dimensional (linear), while graphical are two-dimensional (spatial) [20].

However, it is important to note, since we lack individual time, we cannot ascertain that there were participants who contributed heavily to the average time statistic. This statistic could for example be affected by a participant either having skipped many questions, or spent all/most of the available time for a task. However, we note that there is a difference in time as described above, and that having a more precise measurement of individual time may be of interest in future experiments to further answer RQ2.

In the following, we discuss threats to validity in terms of conclusion validity, internal validity, construct validity, and external validity [30, 31].

Conclusion validity. For our hypothesis test we chose to use the independent samples t-test, which assumes a normal distribution and independent control groups. This choice was motivated by our findings during the data visualisation and use of descriptive statistics. In addition, the two control groups were

completely independent, and each group were only subject to a single treatment. If the data was not normally distributed, we could have performed a non-parametric test such as the Mann-Whitney u-test [9]. Although parametric tests (such as the independent t-test carried out in our study) generally has higher power than non-parametric test, i.e., less data is needed to get significant results [30], the t-tests were in addition carried out from multiple perspectives to mitigate arriving at a false conclusion when rejecting or accepting our null hypothesis. Moreover, we acknowledge that by having a larger sample size, our conclusions would be more robust.

Internal validity. A threat to internal validity is introduced by not having randomized assignment of the treatment for our control groups. This threat is mitigated by dividing the participants fairly in two groups based on competence as explained in Sect. 5. The fair division of groups ensures to some extent that the groups are even in terms of level of knowledge. The fair division could have been further strengthened in the study by, for example, adding an additional step in which the roles of Groups A and B are swapped in terms of solving tasks for the textual and graphical annotations, respectively. However, the measurement of knowledge based on the Likert scale can be imprecise. Imprecision may be due to the Dunning-Kruger effect [3]. This is an effect wherein less competent people tend to overestimate their skills and knowledge, while more competent people tend to underestimate their skills and knowledge. Another threat to internal validity concerns the introductory material since the participants have to go through it on their own. As a consequence, we cannot control the degree to which the participant learns the given material. This uncertainty leads to two different situations in which a participant either spends more or less time learning the material than others. Finally, since we could not control the environment in which the participant answered the questionnaire, there was no way to ensure that the participant did not carry out internet searches to look for clues. In an attempt to mitigate this, the tasks had timers enforcing time restrictions.

Construct validity. A threat to construct validity is introduced by the theoretical constructs comprehensibility and efficiency and the manner in which they are measured in the study. Comprehensibility is measured with respect to task scores, while efficiency is measured with respect to average time. However, these measurement types for comprehensibility and efficiency are often used in similar studies [21, 13, 19]. Furthermore, to prevent bias, all experiment material are the same for both groups with the only difference being the graphical or textual annotations. Finally, as mentioned in Sect. 4, the experiment material was reviewed and improved by a third researcher in seven iterations.

External validity. Our sample of participants does not fully represent the target group of CORAL, which are professionals within security testing and risk assessment, who ultimately are the stakeholders likely to use the CORAL approach. The focus of the study, however, was concerned with the comprehensibility and efficiency when interpreting *predefined* threat models with either graphical or textual annotations. Thus, the study was not concerned with testing nor assessing risks and therefore did not require participants with high exper-

tise in these fields. The sample does, however, consist of developers at different levels, which is also a relevant target group. It can be argued, that developers are most familiar with textual notation used in programming languages. Yet, all participants stated they had experience in using UML in some form. Although the time limitations per task were carefully identified in seven iterations (see Sect. 4.2), the time limitations may have had a potential impact on the results. However, evaluating this would require a separate study.

8 Related Work

Hogganvik et al. [14] empirically investigated the comprehensibility of a domain specific modeling language (DSML) for security risk analysis based on the UML use-case notation [22]. In particular, they investigated the comprehensibility of two versions of their DSML. One version using only stereotypes to capture security risk constructs versus the other version using graphical annotations to capture security risk constructs. This study involved both professionals and students. Their findings, which are similar to ours, report that the participants using graphical risk annotations were able to conclude faster, however, not reaching a higher correctness of interpreting the models.

Meliá et al. [19] compared graphical and textual notations for the maintainability of model-driven engineering domain models in a pilot study. The study was performed with students as participants, and showed that the participants using textual notation performed better with regard to analyzability coverage and modifiability efficiency. This study compares pure textual models against graphical models, and employ metrics different from our study. Furthermore, the graphical models are represented by UML class diagrams, while in our study we address threat models based on sequence diagrams.

Labunets et al. [15, 16] report on an empirical study in which they investigate the comprehensibility of security risk models in terms of tabular versus graphical representations. They conclude that tabular risk models are more effective than graphical ones with respect to extracting certain information about security risks, while graphical risk models are better in terms of solving tasks involving different information cues, different relationships and different judgments. While they evaluate the comprehensibility of tabular risk models versus graphical risk models, we evaluate the comprehensibility of graphical versus textual *risk annotations* on sequence diagrams.

9 Conclusion

We have carried out an empirical study in which we evaluate the comprehensibility of two different annotations representing risk constructs in threat models based on sequence diagrams. The two being either graphical icons provided by the CORAL language [4] or textual UML stereotype annotations [22]. The experiment was carried out on two separate groups A and B, where Group A solved tasks related to the graphical annotations, while Group B solved tasks

related to the textual annotations. We also examined the efficiency of these two annotations in terms of the average time each group spent per task.

With respect to comprehensibility, our study reports that threat models using textual risk annotations to support risk assessment are equally comprehensible to corresponding threat models using graphical risk annotations. With respect to efficiency, our study reports that the use of graphical annotations leads to more efficient task solving in comparison to textual annotations. Participants receiving tasks related to the graphical annotations spent on average 23% less time per task compared to the participants receiving tasks related to the textual annotations. Although Group A was able to conclude faster, they did not reach a higher correctness of interpreting the threat models. Note that our evaluation on efficiency is based on the average time each group spent per task, and not based on the individual time each participant spent per task. Thus, as future work, futher studies should evaluate the efficiency using individual time. However, our findings are in line with and substantiated by similar studies [14].

Acknowledgments. This work has been conducted within the AGRA project (236657) funded by the Research Council of Norway.

References

1. V. R. Basili, G. Caldiera, and H. D. Rombach. *Experience Factory.* John Wiley & Sons, 2002.
2. CORAL plugin for Eclipse Papyrus. `https://bitbucket.org/vetlevo/no.uio.ifi.coral.profile/`. Accessed July 6, 2018.
3. D. Dunning, K. Johnson, J. Ehrlinger, and J. Kruger. Why people fail to recognize their own incompetence. *Current directions in psychological science*, 12(3):83–87, 2003.
4. G. Erdogan. *CORAL: A Model-Based Approach to Risk-Driven Security Testing.* PhD thesis, University of Oslo, 2016.
5. G. Erdogan, Y. Li, R.K. Runde, F. Seehusen, and K. Stølen. Approaches for the combined use of risk analysis and testing: a systematic literature review. *International Journal on Software Tools for Technology Transfer*, 16(5):627–642, 2014.
6. Eval&Go. `http://www.evalandgo.com/`. Accessed July 6, 2018.
7. B.S. Everitt and A. Skrondal. *The Cambridge Dictionary of Statistics.* Cambridge University Press, 2010.
8. M. Felderer and I Schieferdecker. A taxonomy of risk-based testing. *International Journal on Software Tools for Technology Transfer*, 16(5):559–568, 2014.
9. A. Field. *Discovering Statistics Using IBM SPSS Statistics.* SAGE Publications, 2013.
10. I. Hadar, I. Reinhartz-Berger, T. Kuflik, A. Perini, F. Ricca, and A. Susi. Comparing the comprehensibility of requirements models expressed in Use Case and Tropos: Results from a family of experiments. *Information and Software Technology*, 55(10):1823–1843, 2013.
11. G. S. Halford, R. Baker, J. E. McCredden, and J. D. Bain. How many variables can humans process? *Psychological Science*, 16(1):70–76, 2005.

12. G. S. Halford, W. H. Wilson, and S. Phillips. Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. *Behavioral and Brain Sciences*, 21(6):803–831, 1998.
13. I. Hogganvik and K. Stølen. Empirical Investigations of the CORAS Language for Structured Brainstorming. Technical Report A05041, SINTEF Information and Communication Technology, 2005.
14. I. Hogganvik and K. Stølen. On the comprehension of security risk scenarios. In *Proc. 13th International Workshop on Program Comprehension (IWPC'05)*, pages 115–124. IEEE, 2005.
15. K. Labunets, F. Massacci, F. Paci, S. Marczak, and F. M. de Oliveira. Model comprehension for security risk assessment: an empirical comparison of tabular vs. graphical representations. *Empirical Software Engineering*, 22(6):3017–3056, 2017.
16. K. Labunets, F. Massacci, and A. Tedeschi. Graphical vs. Tabular Notations for Risk Models: On the Role of Textual Labels and Complexity. In *Proc. 11th International Symposium on Empirical Software Engineering and Measurement (ESEM'17)*, pages 267–276. IEEE, 2017.
17. M. S. Lund, B. Solhaug, and K. Stølen. *Model-Driven Risk Analysis: The CORAS Approach*. Springer, 2011.
18. B.S. Madsen. *Statistics for Non-Statisticians*. Springer, 2016.
19. S. Meliá, C. Cachero, J. M. Hermida, and E. Aparicio. Comparison of a textual versus a graphical notation for the maintainability of MDE domain models: an empirical pilot study. *Software Quality Journal*, 24(3):709–735, 2016.
20. D.L. Moody. The "Physics" of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering. *Transactions on Software Engineering, IEEE*, 35(6):756–779, 2009.
21. E. G. Nilsson and K. Stølen. The FLUIDE Framework for Specifying Emergency Response User Interfaces Employed to a Search and Rescue Case. Technical Report A27575, SINTEF Information and Communication Technology, 2016.
22. Object Management Group. *Unified Modeling Language (UML), Version 2.5.1*, 2017. OMG Document Number: formal/2017-12-05.
23. Papyrus Modeling Environment. `https://www.eclipse.org/papyrus/`. Accessed July 6, 2018.
24. C. Schalles. *Usability evaluation of modeling languages*. Springer, 2012.
25. I. Schieferdecker, J. Großmann, and M. Schneider. Model-Based Security Testing. In *Proc. 7th Workshop on Model-Based Testing (MBT'12)*, pages 1–12. Electronic Proceedings in Theoretical Computer Science (EPTCS 80), 2012.
26. F. Shull, J. Singer, and D.I.K. Sjøberg. *Guide to Advanced Empirical Software Engineering*. Springer, 2007.
27. K. Singh. *Quantitative Social Research Methods*. SAGE Publications, 2007.
28. M. Staron, L. Kuzniarz, and C. Wohlin. Empirical assessment of using stereotypes to improve comprehension of UML models: A set of experiments. *Journal of Systems and Software*, 79(5):727–742, 2006.
29. V. Volden-Freberg. Development of Tool Support within the Domain of Risk-Driven Security Testing. Master's thesis, University of Oslo, 2017.
30. C. Wohlin, M. Höst, and K. Henningsson. Empirical research methods in software engineering. In *Empirical Methods and Studies in Software Engineering*, pages 7–23. Springer, 2003.
31. C. Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, B. Regnell, and A. Wesslén. *Experimentation in Software Engineering*. Springer, 2012.
32. Cross-site Scripting (XSS). `https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)`. Accessed July 6, 2018.