An Approach to Train and Evaluate the Cybersecurity Skills of Participants in Cyber Ranges based on Cyber-Risk Models

Conference Paper · June 2020

DOI: 10.5220/0009892105090520

Gencer Erdogan
SINTEF
30 PUBLICATIONS **122** CITATIONS

Åsmund Hugo
Norwegian Marine Technology Research Institute
8 PUBLICATIONS **5** CITATIONS

Antonio Álvarez
Atos S.A.
4 PUBLICATIONS **3** CITATIONS

Dario Varano
Università di Pisa
1 PUBLICATION **2** CITATIONS

An Approach to Train and Evaluate the Cybersecurity Skills of Participants in Cyber Ranges based on Cyber-Risk Models

Gencer Erdogan[1], Åsmund Hugo[1], Antonio Álvarez Romero[2],
Dario Varano[3], Niccolò Zazzeri[4] and Anže Žitnik[5]
[1]*Software and Service Innovation, SINTEF Digital, Oslo, Norway*
[2]*Research & Innovation, Atos, Seville, Spain*
[3]*Department of Information Engineering, University of Pisa, Pisa, Italy*
[4]*Trust-IT Services, Pisa, Italy*
[5]*XLAB, Ljubljana, Slovenia*
{gencer.erdogan, aasmund.hugo}@sintef.no, antonio.alvarez@atos.net, dario.varano@ing.unipi.it,
n.zazzeri@trust-itservices.com, anze.zitnik@xlab.si

Keywords: Cyber Range, Cybersecurity, Cyber-risk Models, Training Scenario, Exercise, Evaluation, Real-time, White Team, Green Team, Blue Team, Red Team.

Abstract: There is an urgent need for highly skilled cybersecurity professionals, and at the same time there is an awareness gap and lack of integrated training modules on cybersecurity related aspects on all school levels. In order to address this need and bridge the awareness gap, we propose a method to train and evaluate the cybersecurity skills of participants in cyber ranges based on cyber-risk models. Our method consists of five steps: create cyber-risk model, identify risk treatments, setup training scenario, run training scenario, and evaluate the performance of participants. The target users of our method are the White Team and Green Team who typically design and execute training scenarios in cyber ranges. The output of our method, however, is an evaluation report for the Blue Team and Red Team participants being trained in the cyber range. We have applied our method in three large scale pilots from academia, transport, and energy. Our initial results indicate that the method is easy to use and comprehensible for training scenario developers (White/Green Team), develops cyber-risk models that facilitate real-time evaluation of participants in training scenarios, and produces useful feedback to the participants (Blue/Red Team) in terms of strengths and weaknesses regarding cybersecurity skills.

1 INTRODUCTION

There is an urgent need for highly skilled, multi-disciplined cybersecurity professionals, given the increasingly aggressive cyber-landscape public and private organizations are facing. As pointed out by the European Cyber Security Organization (ECSO), at education level, there is a big awareness gap and lack of integrated training modules on cybersecurity related aspects on all school levels, starting from low awareness and skills of teachers themselves. The same is true for professional training on university level, including lack of cybersecurity modules in higher education training programs for vital service domains etc. In addition, there are only few existing cybersecurity higher education programs in Europe. Moreover, it is reported that at professional level, there is a lack of accessible tools for continuous awareness, training and skills development on cybersecurity aspects (ECSO, 2016).

In order to address the need of continuous cybersecurity awareness, training and skills development, we have developed a method to train and evaluate the cybersecurity skills of participants in cyber ranges based on cyber-risk models. Our method assumes that the technical capabilities to simulate infrastructure on which exercises are executed already exist and that necessary instructions and training for the cyber range has already been given to the participants.

Thus, the contribution of this paper is a method for creating cyber-risk models that facilitate the training and evaluation of cybersecurity skills of participants in cyber-ranges. The method is described using an example in the context of CYBERWISER.eu, which is a web-based cyber

Erdogan, G., Hugo, Å., Romero, A., Varano, D., Zazzeri, N. and Žitnik, A.
An Approach to Train and Evaluate the Cybersecurity Skills of Participants in Cyber Ranges based on Cyber-Risk Models.
In *Proceedings of the 15th International Conference on Software Technologies (ICSOFT 2020)*, pages 509-520
ISBN: 978-989-758-443-5

range platform we have developed as part of the international EU project with the same name (CYBERWISER.eu, 2020a).

Cyber ranges often have different groups of people referred to as "teams" who have different roles. In this paper, we consider the Green, White, Red, and Blue teams (Damodaran & Smith, 2015). The Green Team consists of individuals who operate the range infrastructure and support tool systems. In collaboration with the White Team, the Green Team manages on-demand definition of training scenarios. The White Team represents the instructor/s of the training, whether course based or as an exercise. The White Team collaborates with the Green Team to deploy and configure training scenarios. The White Team also evaluates the participants' progress. The Red Team carries out cyberattacks against the infrastructure simulated on the cyber range as part of a training scenario. The Blue Team detects and responds to the attacks performed by the Red Team and/or automatically by the tools in the cyber range.

To address the abovementioned needs and develop artefacts that appropriately meet these needs, we define the following success criteria.

Success Criterion 1: The method must be easy to use and comprehensible for cyber-range training scenario (exercise) developers.

The main target user group of our method is people who design and develop cyber-range training scenarios/exercises. That is, the Green Team and the White Team. The method must therefore be easy to use and comprehensible for the intended target audience.

Success Criterion 2: The method must provide necessary guidelines to create cyber-risk models that facilitate real-time evaluation of participants.

Cyber range training scenarios are dynamic in nature and the participants being trained need to make decisions and take actions on-the-fly. Thus, to correctly evaluate the cybersecurity skills of the participants, we need to evaluate their decisions and actions in real-time while the exercise is running.

Success Criterion 3: The method must produce useful feedback to the participants in terms of exercise evaluations.

For the participants to learn from their decisions and actions taken in an exercise in the cyber range, they need to receive feedback explaining to what extent they have successfully carried out the exercise. Thus, we need to provide useful feedback to the participants evaluating their achievements.

In Section 2, we describe our research method. In Section 3, we describe the architecture of our cyber range platform, before we explain our method for training and evaluation in Section 4. In Section 5, we describe related work. In Section 6, we discuss the extent to which we have fulfilled our success criteria described above, before concluding in Section 7.

2 RESEARCH METHOD

Figure 1 illustrates the three steps of our research method, which is in line with the design science approach by Wieringa (2014). Although the steps are illustrated sequentially, the method was carried out iteratively where some of the steps were revisited during the process.

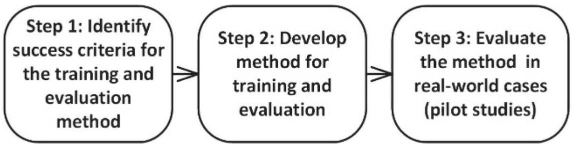

Step 1: Identify success criteria for the training and evaluation method
Step 2: Develop method for training and evaluation
Step 3: Evaluate the method in real-world cases (pilot studies)

Figure 1: Research method.

In Step 1, we identified three success criteria which act as requirements for our method for training and evaluation based on the background and needs as explained in Section 1.

In Step 2, we developed our method for training and evaluation. The method consists of five main steps: create cyber-risk model, identify risk treatments, setup training scenario, run training scenario, and finally evaluate the performance of participant. All steps are supported by tools that collectively comprise our cyber range training platform. The cyber range training platform will be explained in more detail in Section 3.

In Step 3, we evaluated our method for training and evaluation in real-world pilot studies to assess the feasibility of our approach w.r.t. our success criteria.

3 CYBER RANGE ARCHITECTURE

Before we describe the steps of our method for training and evaluation, it is necessary to explain the overall architecture of our cyber range platform in which the training and evaluation method is applied. As illustrated in Figure 2, the platform consists of components that may be grouped into four parts: simulated infrastructure, scenario environment, cyber range, and user interface. The Simulated Infrastructure represents the first layer of the platform. This layer consists of virtual machines and virtual networks that simulate an organization's ICT system. Depending on the objective of a training scenario, the participant being trained will either carry out attacks on the simulated ICT system automatically using the Attack Simulator, or mitigate an ongoing attack using the Countermeasure Simulator. Manual attacks/mitigations are also possible. The participants in a Blue or Red Team may also run scans to check for potential vulnerabilities using the Vulnerability Assessment Tools.

The Monitoring Sensors are software programs implemented in the simulated ICT system to monitor host activity or network activity. The host activity sensors detect potential threats, while the network activity sensors analyse network traffic and detect and prevent network intrusion. These sensors send events to the Anomaly Detection Reasoner and the Economic Risk Evaluator.

The second layer of the cyber range platform is the Scenario Environment which is implemented as an IaaS and contains the components necessary for the training scenarios. At the centre of the Scenario Environment, we find the Economic Risk Evaluator, which uses Economic Risk Models to produce real-time risk assessment in terms of monetary loss based on the observed behaviour of the participants in the training scenario. The real-time feature builds on continuous observation of the dynamic behaviour of the training scenarios with the help of the Monitoring Sensors, Anomaly Detection Reasoner, and the Vulnerability Assessment Tools which all produce input to the Economic Risk Evaluator.

To create a realistic experience for the participants, the Attack Simulator can be configured to automatically launch pre-defined attacks towards a specific target. The Performance Evaluator is notified about the success of an automated attack which indicates whether the participants, acting as defenders, were able to prevent the attack. Similarly, the Countermeasure Simulator can execute mitigation measures that prevent further attacks automatically after a certain event has occurred.

The Performance Evaluator component takes as input the risk assessment produced by the Economic Risk Evaluator, as well as actions taken by the participants, and based on these inputs produces an evaluation report. This evaluation report is forwarded to the Centralized Logging Component where it is archived and made available to the participant.

The third layer of the cyber range platform is the Cyber Range, which includes the components Digital Library, Training Manager, and Simulated Infrastructure Manager, as well as the Scenario Environment and the Simulated Infrastructure. The Training Manager provides user interfaces for easy design and configuration of training scenarios, their creation, deployment, as well as un-deployment and resource removal after a completed training session. The Digital Library is a repository storing all the virtual machine images and additional software, required for building the training scenarios, as well as the designed scenarios themselves. The Simulated Infrastructure Manager acts as an interface to the underlying IaaS to control the virtual machines and networks of the Scenario Environments based on instructions from the Training Manager. Additionally, Simulated Infrastructure Manager provides the participants access to the virtual

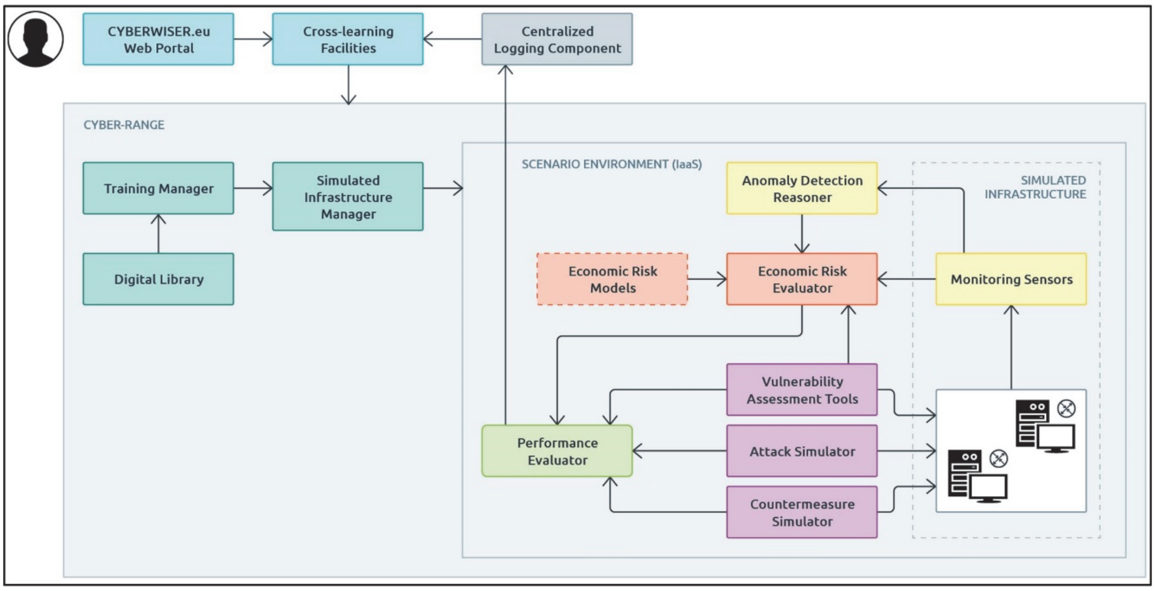


Figure 2: Cyber range architecture.

machines in the Scenario Environment by exposing a VNC interface, simply available through the user's web browser.

The fourth and final layer of the cyber range platform is the User Interface consisting of a Web Portal and the Cross-learning Facilities. The CYBERWISER.eu Web Portal is the single entry-point to the platform and its services for all end-users (white, green, blue, red teams, etc.). The authentication process is provided by the Cross-Learning Facilities component through a Single Sign-On (SSO) service. The Cross-Learning Facilities component provides, among others, training materials in terms of literature, courses, communication tools such a chat service, dashboards for the users and a link to the Cyber-Range Service. Scores, achieved by participants in the Cyber Range training exercises, are transferred to the Cross-Learning Facilities and can be viewed here as well.

4 METHOD FOR TRAINING AND EVALUATION

Figure 3 illustrates our method for training and evaluating cybersecurity skills of participants in cyber ranges based on cyber-risk models. The following sections explain each of the steps.

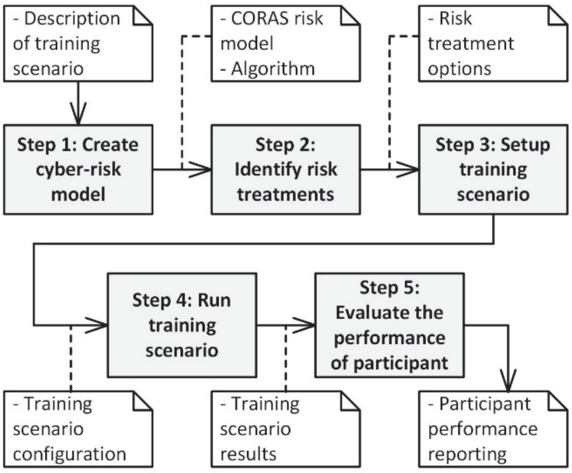

- Description of training scenario
- CORAS risk model
- Algorithm
- Risk treatment options
Step 1: Create cyber-risk model
Step 2: Identify risk treatments
Step 3: Setup training scenario
Step 4: Run training scenario
Step 5: Evaluate the performance of participant
- Training scenario configuration
- Training scenario results
- Participant performance reporting

Figure 3: Method for training and evaluation.

4.1 Step 1: Create Cyber-Risk Model

As indicated in Figure 3, the first step of our method expects as input a description of a training scenario, which is basically a description of an exercise to be carried out on the cyber range platform with the purpose of training cybersecurity skills. This description is expected to be provided by cybersecurity experts. For example, someone who has the role as Chief Security Information Officer and who is interested in training their cybersecurity staff. Examples of roles to be trained are Vulnerability Analysts or Threat Analysts (CIISec, 2019). As part of our cyber range platform, we do also provide guidelines for how to *describe* training scenarios. However, these guidelines are out of the scope of this paper. The reader is referred to (CYBERWISER.eu, 2019b) for a detailed explanation on how to describe training scenarios.

In the following, we consider a training scenario example developed as part of applying the method in a real-world case pilot as part of the CYBERWISER.eu project. The training scenario we consider describes an exercise to train technical security staff in mitigating an SQL injection attack. That is, the exercise concerns defensive training where the technical security staff needs to mitigate an SQL injection attack their ICT infrastructure, which is simulated on the cyber range platform, is exposed to. From a cyber range training perspective, the team trained in defensive exercises are typically referred to as the Blue Team.

Based on the above training scenario description we follow an approach described in earlier work (Erdogan, Gonzalez, Refsdal, & Seehusen, 2017) to first create a graphical cyber-risk model using CORAS (Lund, Solhaug, & Stølen, 2011) in order to capture the SQL injection attack pattern, and then schematically develop a corresponding machine-readable risk assessment algorithm with respect to the graphical risk model. The CORAS risk model and the risk assessment algorithm are the output of Step 1.

Figure 4 illustrates the CORAS risk model created for the SQL injection training scenario example described above. When creating risk models, we make use of existing libraries and catalogues such as CAPEC (CAPEC, 2020), OWASP (OWASP, 2020) and CWE (CWE, 2020) in order to create risk models that are in line with standard attack patterns.

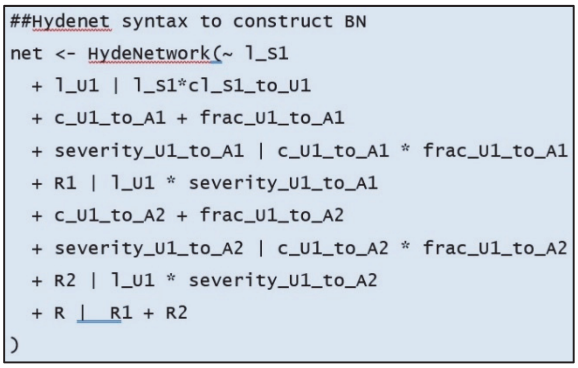
The CORAS risk model in Figure 4 illustrates that a threat *Hacker* initiates the treat scenario *S1: Initiate SQL Injection*. Moreover, the hacker may exploit the vulnerabilities *CWE-89: Improper neutralization of special elements used in an SQL command* and *CWE-390: Detection of SQL-related error conditions without action* which leads to the unwanted incident *U1: SQL injection successful*. Finally, we see that the unwanted incident has an impact on the security assets *A1: Confidentiality* and *A2: Integrity*, which are the assets we want to protect.

In addition to threats, threat scenarios, vulnerabilities, unwanted incidents, and security assets, we use

CORAS risk models to capture the risk assessment values likelihood, conditional likelihood and consequence. In a standard CORAS risk model, these values are written directly in the model. However, in our approach we parameterize these values in the model to later develop the corresponding risk assessment algorithms. Considering our example in Figure 4, the likelihood of *S1* is represented by *l_S1*, the conditional likelihood going from *S1* to *U1* is represented by *cl_S1_to_U1*, the likelihood of *U1* is represented by *l_U1*, the consequence of *U1* on *A1* is represented by *c_U1_A1*, and the consequence of *U1* on *A2* is represented by *c_U1_A2*.

Finally, to capture the dynamic behaviour of the SQL injection attack that the simulated infrastructure is exposed to during the training scenario, we include what we refer to as indicators in the risk models. By indicator we mean a piece of information that is relevant for assessing the risk level. The risk level in our approach is represented as monetary loss. We distinguish between the following four kinds of indicators.

- *Business configuration (IN-32, IN-C81C, and IN-C81I)*: Indicator values are obtained by asking business related questions. The indicator values are thus based on the knowledge of the participant.
- *Test results (IN-37)*: Indicator values are obtained by carrying out tests, such as vulnerability scans or automated attacks. The indicator values are thus based on test results. Test results and business configuration indicators are non-intrusive in the sense that they do not require the implementation of sensors in the simulated infrastructure.
- *Network-layer monitoring*: Indicator values are obtained by monitoring the network layer. This type of indicator is intrusive in the sense that sensors need to be deployed in the network layer of the simulated infrastructure under analysis.
- *Application-layer monitoring (IN-44 and IN-56)*: Indicator values are obtained by monitoring the application layer. This type of indicator is also intrusive; a sensor needs to be installed in the machine under analysis.



```
##Hydenet syntax to construct BN
net <- HydeNetwork(~ l_S1
  + l_U1 | l_S1*cl_S1_to_U1
  + c_U1_to_A1 + frac_U1_to_A1
  + severity_U1_to_A1 | c_U1_to_A1 * frac_U1_to_A1
  + R1 | l_U1 * severity_U1_to_A1
  + c_U1_to_A2 + frac_U1_to_A2
  + severity_U1_to_A2 | c_U1_to_A2 * frac_U1_to_A2
  + R2 | l_U1 * severity_U1_to_A2
  + R | R1 + R2
)
```

Figure 4: CORAS risk model with indicators.

Note that Figure 4 does not illustrate any network monitoring indicator as they were not relevant for this example.

Having created the graphical risk model, next we schematically translate the model into an executable risk assessment algorithm in terms of an **R** script (R-project, 2020). Figure 5 illustrates an excerpt of the **R** script created based on the risk model in Figure 4. The excerpt illustrates that we represent the risk model as a Bayesian Network in order to do calculations using the likelihood and consequence values captured in the risk model. The likelihood and consequence values are in turn calculated with respect to their respective indicators. Due to space restrictions, it is

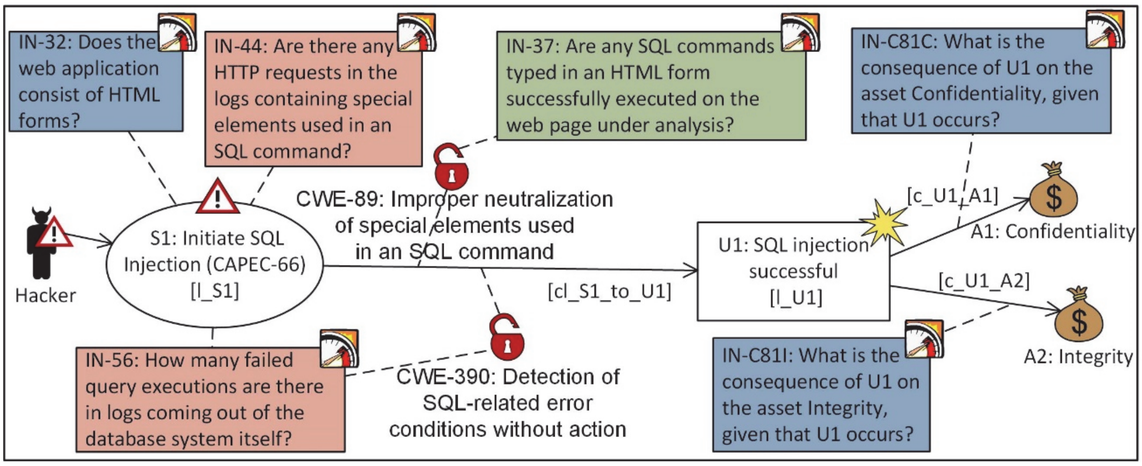

IN-32: Does the web application consist of HTML forms?
IN-44: Are there any HTTP requests in the logs containing special elements used in an SQL command?
IN-37: Are any SQL commands typed in an HTML form successfully executed on the web page under analysis?
IN-C81C: What is the consequence of U1 on the asset Confidentiality, given that U1 occurs?
CWE-89: Improper neutralization of special elements used in an SQL command
Hacker
S1: Initiate SQL Injection (CAPEC-66) [l_S1]
[cl_S1_to_U1]
U1: SQL injection successful [l_U1]
[c_U1_A1]
A1: Confidentiality
[c_U1_A2]
A2: Integrity
IN-56: How many failed query executions are there in logs coming out of the database system itself?
CWE-390: Detection of SQL-related error conditions without action
IN-C81I: What is the consequence of U1 on the asset Integrity, given that U1 occurs?

Figure 5: Excerpt of R script.

not possible to show the complete **R** script. However, the logic for computing likelihood and consequence values are as follows:

- The indicators used to assess likelihood values (IN-32, IN-37, IN-44, and IN-56) are formulated either as Yes/No questions or as quantitative questions. Based on the answer, we increase or decrease the likelihood value of the risk illustrated in Figure 4. For example, consider that the answer to indicators IN-32 and IN-44 are "Yes", and that the answer to indicator IN-56 is "25 in the last 10 minutes", then we would increase the likelihood *l_S1* to Very High (in a scale of {Very Low, Low, Medium, High, Very High}).
- The indicators used to assess consequence values (*IN-C81C* and *IN-C81I*) basically asks the participant what the consequence of an unwanted incident is, given that the unwanted incident materializes.

For the complete **R** script, the reader is referred to a technical report in which the risk model in Figure 4 as well as other risk models tried out in context of real-world pilots are explained in detail (CYBERWISER.eu, 2020b).

4.2 Step 2: Identify Risk Treatments

In Step 2, we base ourselves on the risk model created in Step 1 to identify risk countermeasures. We identify countermeasures using CORAS treatment diagrams which represent strategies and action plans the implementation of which reduces risks to an acceptable level (Lund et al., 2011).

Figure 6 illustrates the same risk model as in Figure 4. However, in Figure 6, we have removed all indicators and identified several risk countermeasures for the vulnerabilities *CWE-89* and *CWE-390*. In total, we identified 10 countermeasures for vulnerability *CWE-89*, and 2 countermeasures for vulnerability *CWE-390*, but due to space restrictions we illustrate 5 of 12 countermeasures in the treatment diagram in Figure 6. The main source from which we identified the countermeasures were the webpages documenting *CWE-89* and *CWE-390* (CWE, 2020).

Having created CORAS treatment diagrams, next we describe each countermeasure in detail using a table template including a unique ID of the countermeasure (for example, *M8* illustrated in Figure 6), the name of the countermeasure (for example, *Validate input*), a detailed description of the countermeasure, and source of the countermeasures (for example, URL to specific countermeasures in CWE). An excerpt of the detailed description of countermeasure *M8*, according to *CWE-89* (CWE, 2020), is: "When performing input validation, consider all potentially relevant properties, including length, type of input, the full range of acceptable values, missing or extra inputs, syntax, consistency across related fields, and conformance to business rules. As an example of business rule logic, 'boat' may be syntactically valid because it only contains alphanumeric characters, but it is not valid if the input is only expected to contain colours such as 'red' or 'blue'."

Thus, the output of Step 2 is a CORAS treatment diagram with countermeasures and an accompanying table, according to the template described above, describing the countermeasures in detail. These outputs are used in the cyber training scenarios to provide the participants a set of countermeasure options.

4.3 Step 3: Setup Training Scenario

In Step 3, we setup the training scenario in the cyber range platform. All the components in the cyber range platform (described in Section 3) play a role in setting

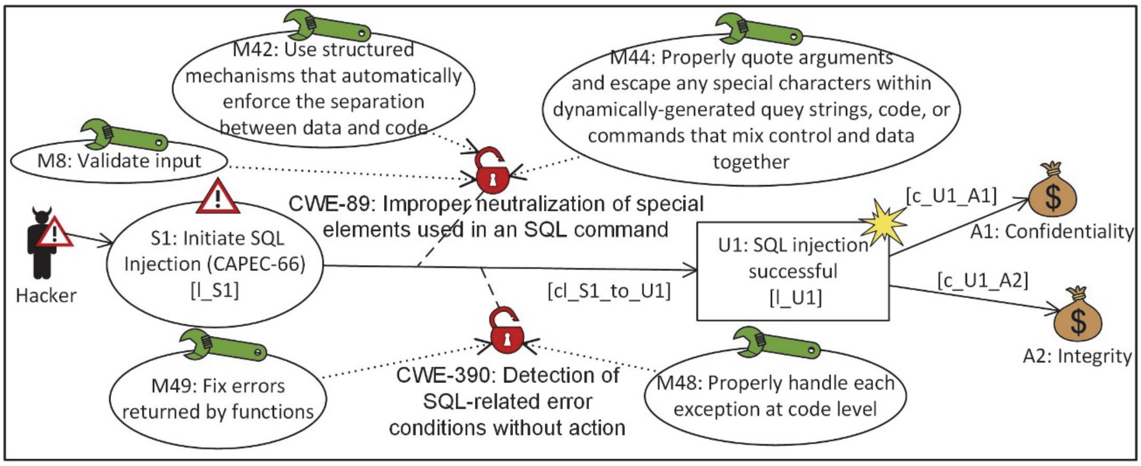

M42: Use structured mechanisms that automatically enforce the separation between data and code
M44: Properly quote arguments and escape any special characters within dynamically-generated quey strings, code, or commands that mix control and data together
M8: Validate input
CWE-89: Improper neutralization of special elements used in an SQL command
Hacker
S1: Initiate SQL Injection (CAPEC-66) [l_S1]
[cl_S1_to_U1]
U1: SQL injection successful [l_U1]
[c_U1_A1]
A1: Confidentiality
[c_U1_A2]
A2: Integrity
M49: Fix errors returned by functions
CWE-390: Detection of SQL-related error conditions without action
M48: Properly handle each exception at code level

Figure 6: CORAS treatment diagram.

up and executing a training scenario. However, our method considers mainly the components that are relevant for training and evaluating based on risk models. When describing the setup and execution of training scenarios, we therefore mainly discuss the components Economic Risk Evaluator, Countermeasure Simulator, and Performance Evaluator, and assume that all other components are setup and work properly.

As illustrated in Figure 7, setting up the training scenario depends on the risk models. As illustrated in the top-left part of Figure 7, we first develop risk models with indicators and then schematically translate a risk model with indicators to an **R** script (that is, Step 1 explained in Section 4.1). Next, as illustrated on bottom-left part of Figure 7, we develop treatment diagrams (risk models with countermeasures) and then describe all identified countermeasures in detail using a table-based template (that is, Step 2 explained in Section 4.2).

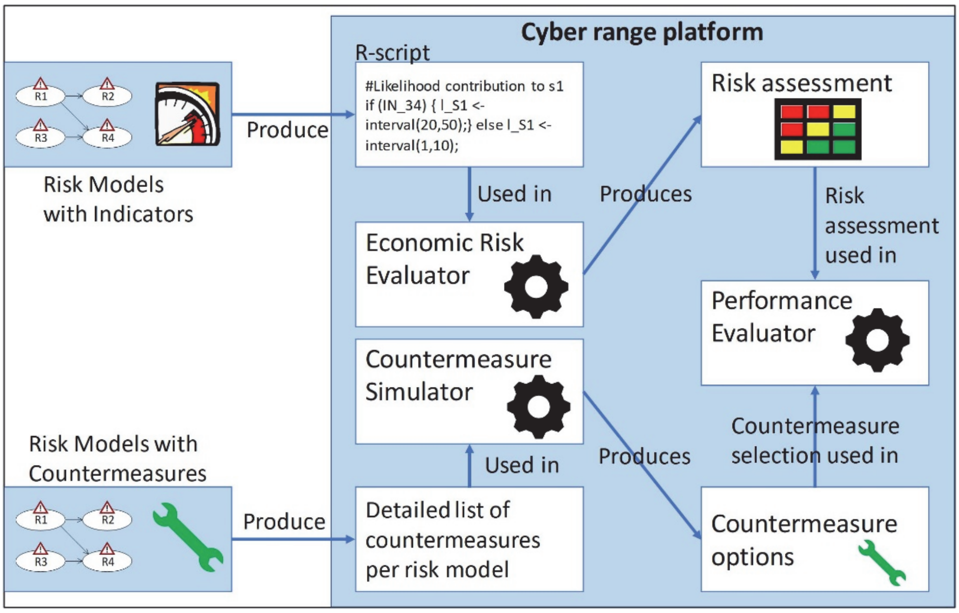
The **R** scripts are implemented in the Economic Risk Evaluator, which has the responsibility of executing the scripts. This includes retrieving values from the components Monitoring Sensors, Anomaly Detection Reasoner, and Vulnerability Assessment Tools and assign the values to their corresponding indicators in the risk model. Recall that we have different types of indicators as described in Section 4.1. The component Vulnerability Assessment Tools provide values to test result indicators, the component Monitoring Sensors and Anomaly Detection Reasoner collectively provide values to network layer monitoring indicators and application layer monitoring indicators.

Each countermeasure identified and described as part of Step 2 is implemented in the Countermeasure Simulator. The Countermeasure Simulator is a component that has a register of a set of possible countermeasures for each risk model considered in a training scenario. Basically, the Countermeasure Simulator is responsible of making a set of relevant countermeasures available to the participants during the training scenario.

4.4 Step 4: Run Training Scenario

In Step 4, we execute the training scenario with respect to the training scenario configuration carried out in Step 3.

The execution of a training scenario depends on the use case of the training scenario. There are four kinds of use cases: (A) Blue Team vs. Red Team where the purpose is for the Blue Team to protect the simulated ICT system from cyber-risk attacks performed by the Red Team, (B) Red Team vs. Blue Team where the purpose is for the Red Team to attack the simulated ICT system protected by the Blue Team, (C) Blue Team vs. Cyber Range Platform where the purpose is for the Blue Team to protect the simulated ICT system from cyber-risk attacks automatically carried out by the platform, and finally (D) Red Team vs. Cyber Range Platform where the purpose is for the Red Team to attack the simulated ICT system protected by the platform.



```
#Likelihood contribution to s1
if (IN_34) { l_S1 <-
interval(20,50);} else l_S1 <-
interval(1,10);
```

Figure 7: Setting up training scenario in the cyber range.

Considering our example described in Section 4.1, the purpose is for the Blue Team to protect the simulated ICT system from SQL injections. In this example, we do not consider a team versus another team. Thus, we are considering use case (C) Blue Team vs. Cyber Range Platform. The automatic SQL injections are initiated by the person having the role as tutor/teacher using the component Attack Simulator in the platform. The person having the teacher/tutor role is typically referred to as a member of the White Team in context of cyber ranges.

Assuming members of the Blue Team and White Team are logged in the cyber range platform, the training scenario will commence as follows. First, the White Team initiates SQL injections using the component Attack Simulator. The Blue Team monitors the status of the simulated ICT system via the graphical user interface of the Economic Risk Evaluator. Then, if the Blue Team detects that the risk level for SQL injection is going up, they need to investigate how and where the attack is carried out (on the simulated ICT system) and select appropriate countermeasures from the Countermeasure Simulator and apply them. Having applied the countermeasures, the Blue Team continuously checks the risk level to see if the level goes down or up. The important task here is to understand what kind of cyber-risk attack the system is exposed to and based on that select appropriate countermeasures. After a predefined time (running time of the exercise), the White Team stops the exercise. All actions taken by the Blue Team is logged. After the exercise has been stopped, the risk assessment and the countermeasures selected by the Blue Team are fed into the Performance Evaluator as illustrated in Figure 7.

4.5 Step 5: Evaluate the Performance of Participants

Based on the training scenario results obtained by carrying out the exercise in Step 4, the Performance Evaluator calculates automatically in Step 5 how well the participants have carried out the training scenario. The Performance Evaluator considers the current status of the exercise, but also its history, that is, how the exercise has evolved.

The evaluation is carried out with respect to predefined assessment algorithms implemented in the Performance Evaluator, which vary depending on the exercise. Each performance evaluation algorithm defined and implemented in the Performance Evaluator expects a series of inputs that must be obtained from the training scenario exercise. The values of those inputs are assigned by analysing and processing the raw logs the exercise generates (output of Step 4).

In the following the different types of information leveraged to produce the performance evaluation reporting for the participants are listed, along with some examples:

- Flags. Flags mark relevant moments in the exercise, mainly related to achievements of the participant. For example, the execution of a Denial of Service attack by a participant who is part of the Red Team or the blocking of the TCP for a suspicious IP address by a participant who is part of the Blue Team. The Attack Simulator and the Countermeasure Simulator are the main components producing this information (flags).
- Elaborated logs coming from monitoring tools deployed in the monitored infrastructure. Such logs produce information about, for example, an attacker performing a dictionary attack to achieve brute-force login into a certain machine.
- Results of vulnerability scans executed against a certain infrastructure element. The vulnerability scans (part of the Vulnerability Assessment Tools) are useful for the Red Team to find weaknesses that can be exploited. It can also be used by a Blue Team to identify vulnerabilities that need to be mitigated. An example of a vulnerability may be the improper neutralization of special characters in an SQL query which may eventually be exploited with an SQL injection attack.
- Cyber risk exposure evolution: the amount of money being exposed and its evolution over time is relevant for the evaluation. Pairs (cyber risk, timestamp) are fed to the Performance Evaluator from the Economic Risk Evaluator to describe the corresponding trajectory. Using configured thresholds, the Performance Evaluator will analyse this trajectory to evaluate the celerity and effectiveness the participant showed to react to the attack (selected countermeasures by the participant).
- Questionnaire: In addition to the above information, the Performance Evaluator takes input coming from a questionnaire filled out by the White Team evaluating the participants. The predefined evaluation algorithm in the Performance Evaluator is configured to give different scorings depending on the answer given by the White Team.

In the case of our SQL injection exercise example, the participant playing the role of defender is presented with different alternatives in terms of countermeasures, thanks to the Countermeasure

Simulator. The countermeasures are presented to the participant in terms of preconfigured scripts. However, some of the countermeasures are more useful than others (expressed as ratings in each countermeasure), and each countermeasure comes with a cost expressed in monetary value.

To make the exercise represent a real-world situation, a limited budget is allocated for the participant to select ("buy") countermeasures and apply them on the underlying cyber risk attack.

When choosing a countermeasure, the corresponding script is executed, and this event is registered with a timestamp and sent to the Performance Evaluator. The Economic Risk Evaluator sends the evolution of the cyber risk exposure which is also used as input of the performance evaluation algorithm. Recall that, depending on the selected countermeasure, the risk level may vary.

Based on the information produced by the participant during a training scenario as pointed out above, the Performance Evaluator will be able to evaluate whether:

- The Blue Team correctly identified the cyber-attack that the simulated ICT system is exposed to.
- The Blue Team selected correct and appropriate countermeasures.
- The selected countermeasures were the most cost-efficient.
- The Blue Team mitigated the cyber-risk attack within the expected time of the training scenario.

In addition to the above, the White Team has the possibility to provide their own assessment based on their expert knowledge by filling out the abovementioned questionnaire. The questionnaire provides the assessment carried out by the White Team in a structured manner to the Performance Evaluator.

As indicated in Figure 3, the main output of the Performance Evaluator is a report which provides a grading and evaluates the performance of the participants. To provide a richer feedback, this grade is broken into chapters which provide the participants a more detailed assessment explaining aspects the participants showed more strength and where the weak points are.

5 RELATED WORK

Cyber ranges have traditionally been developed and used by military institutions for cybersecurity training in the context homeland defence strategy (Damodaran & Smith, 2015; Davis & Magrath, 2013; Ferguson, Tall, & Olsen, 2014). However, over the years, there have been proposed various cyber range solutions to bring cybersecurity training to both public and private organizations (Yamin, Katt, & Gkioulos, 2020).

Secure Eggs (Essentials and Global Guidance for Security) by NRI Secure (NRISecure, 2020), enPiT-Security (SecCap) (EnpitSecurity, 2020), and CYber Defense Exercise with Recurrence (CYDER) are approaches and security training programs focusing on basic cybersecurity hands on and awareness training (Beuran, Chinen, Tan, & Shinoda, 2016).

There are various approaches focusing on cybersecurity skills training within specific domains such as smart grid (Ashok, Krishnaswamy, & Govindarasu, 2016) and cybersecurity assurance (Somarakis, Smyrlis, Fysarakis, & Spanoudakis, 2019). In contrast to domain specific approaches, our approach is generic and may be applied for training cybersecurity skills in any domain. This is also demonstrated by the fact that we have applied our approach in the context of three different large scale real-world pilots: academic pilot, transport pilot, and energy pilot (CYBERWISER.eu, 2020a).

Several approaches focus mainly on the cyber range architecture and improving the efficiency and performance of cyber ranges. Pham, Tang, Chinen, and Beuran (2016) suggest a cyber range framework named CyRIS/CyTrONE focusing on improving the accuracy of the training setup, decreasing the setup time and cost, and making training possible repeatedly and for a large number of participants. The authors also report on an evaluation of the performance of their approach (Beuran et al., 2018). Russo, Costa, and Armando (2018) argue that the design, validation, and deployment of scenarios are costly and error-prone activities that may require specialized personnel for weeks or even months, and that misconfiguration in the resulting scenario can spoil the entire cyber exercise. To address these challenges brought by architectural shortcomings, the authors propose a framework for automating the design, model validation, generation and testing of cyber training scenarios. As part of developing our cyber range platform described in Section 3, we considered requirements related to scalability and efficiency of deploying and running training scenarios. However, as explained in Section 1, the contribution of this paper is our method for creating cyber-risk models that facilitate the training and evaluation of cybersecurity skills of participants in cyber-ranges. The reader is referred to a technical report for further information on considerations related to scalability and efficiency of our cyber range platform (CYBERWISER.eu, 2019a).

The approaches provided by Russo et al. (2018) and Braghin et al. (2019) are similar to our approach in the sense that they use some form of attack models as a foundation to design and execute training scenarios. Russo et al. (2018) introduce a Scenario Definition Language (SDL) based on the OASIS Topology and Orchestration Specification for Cloud Applications (TOSCA). Braghin et al. (2019) provide a domain specific language for scenario construction in which it is possible to capture configuration problems as well as structural vulnerabilities. We use CORAS risk modes which are acyclic directed graphs as described in Section 4. Thus, from a modelling perspective, our approach complements existing approaches. To the best of our knowledge, our approach is unique compared to existing approach in the sense that we develop machine readable risk assessment algorithms based on the cyber-risk models in order to facilitate real-time risk assessment as well as real-time evaluation of the participants' cybersecurity skills as explained in Section 4.

In their systematic literature review, Yamin et al. (2020) report that current cyber range approaches that apply some form of cyber-attack modelling are not validating the models against real world scenarios and use mostly artificial educational scenarios. In this respect, our approach requires that the risk models are developed with respect to training scenario descriptions requested by stakeholders. Based on experience so far in applying our method in real-world academic pilot, transport pilot, and energy pilot, it is reasonable to argue that our approach validates the risk models against real world scenarios, thus supporting its feasibility.

6 EVALUATION

In this section, we discuss the extent to which we have fulfilled our success criteria described in Section 1.

6.1 Fulfilment of Success Criterion 1

The first criterion states: *The method must be easy to use and comprehensible for cyber-range training scenario (exercise) developers.*

The steps of our method (Section 4) are well in line with activities typically carried out in cyber ranges collaboratively by the White Team and the Green Team (Yamin et al., 2020). Step 1 and Step 2 fall under the training scenario design activities typically carried out by the White Team. Step 3 and Step 4 are part of environment configuration and the management of training scenario execution typically carried out by the Green Team. Finally, Step 5 is carried out as part of learning activities including tutoring, scoring and analysis of scoring carried out by the White Team.

As explained in Section 5, there are several approaches that use some form of attack modelling for the purpose of designing and executing training scenarios. In our approach, we use the CORAS risk modelling language. CORAS has been empirically shown to be intuitively simple for stakeholders with different backgrounds (Solhaug & Stølen, 2013). CORAS is also based on international standards like ISO 27005 and ISO 31000, which means that the language supports well known and widely used cybersecurity concepts (see Section 4). In the context of cyber ranges, it is expected that the White Team and Green Team are familiar with concepts such as threat scenario, vulnerability, unwanted incident, etc.

With respect to the technical aspects of our method considering the development of **R** scripts based on the cyber-risk models, this is an activity expected to be carried out by the Green Team as part of environment configuration. However, the White Team on the other hand can support the Green Team by explaining the logic of the expected cyber-risk assessment algorithms. These activities are thus in line with the expected roles of White Team and Green Team members (Yamin et al., 2020).

A threat to validity in terms of the generality of our method is that the method has been applied only on our cyber range architecture described in Section 3. However, according to the taxonomy provided by Yamin et al. (2020), we see that the architecture of our cyber range is in line with general cyber range architectures covering components within scenario development, environment setup and configuration, monitoring, teaming (red, blue, white, green), learning (training material), and management of the cyber range. It is therefore a straightforward task to map our method to existing cyber range architectures, in order to apply our method in different cyber ranges.

At the time of writing, our method has been tried out in the context of the CYBERWISER.eu project, within three large scale pilots (cases) from three different domains: academia, transport, and energy. Although the stakeholders in these pilots come from different domains, they all are cybersecurity experts who took the role as White Team, while the technical team of the project took the role as Green Team. All steps of our method were carried out and tried out in order to train students (in the case of the academic pilot) and security staff (in the case of the transport and energy pilot). More empirical studies of our method are planned (CYBERWISER.eu, 2019b). However, the fact that all steps of our method were successfully carried out collaboratively in different domains with people from various background, supports the feasibility of our method in real world cyber range applications.

Thus, from a methodological point of view, it is reasonable to argue that our method is easy to use and comprehensible by scenario developers.

6.2 Fulfilment of Success Criterion 2

The second criterion states: *The method must provide necessary guidelines to create cyber-risk models that facilitate real-time evaluation of participants.*

The first step of our method provides detailed explanation on how to develop cyber-risk models based on a training scenario description, how to identify indicators to capture the dynamic behaviour of the training scenario, and finally translate the cyber risk models with indicators to machine readable risk assessment algorithms. These risk assessment algorithms are used to assess the risk level of an ongoing cyber-attack in a training scenario in real-time. From a team perspective, these algorithms are used to assess how well the Red Team is carrying out an attack in real-time (the higher the risk level the better).

The second step of our method explains how to identify risk countermeasures by creating CORAS treatment diagrams. These countermeasures are implemented in the cyber range and made available to the Blue Team during a training scenario as risk-countermeasure options. Based on the selected countermeasures in a training scenario, the risk level may go down. This factor is used to assess how well the Blue Team is performing in real-time protecting the simulated infrastructure from exposed cyber-attacks. That is, the lower the risk level (due to selected countermeasures) the better the Blue Team is performing.

To this end, our method provides necessary guidelines to create cyber-risk models that facilitate real-time evaluation of participants.

6.3 Fulfilment of Success Criterion 3

The third criterion states: *The method must produce useful feedback to the participants in terms of exercise evaluations.*

As explained in Section 4.5, our method supports the evaluation of participants carrying out training scenario exercises on cyber ranges. The novel contribution of our approach relies on the automation of the evaluation process, using information collected during the exercise to provide automated evaluation in real-time.

The evaluation process in our approach takes as input the cyber risk exposure of the cyber-attack considered in the training scenario, flags logging the achievements of the participants, logs from monitoring tools, and results from vulnerability scans as explained in Section 4.5. In order to capture evaluation carried out manually by the White Team, complementing the automatic evaluation, our approach provides a questionnaire to be filled out by the White Team based on their expert knowledge and observations.

According to Yamin et al. (2020), current cyber range approaches mainly use scoreboards in which progress of participants is presented based upon the task they completed. Our approach also provides scoring in terms of grading. However, compared to existing approaches, our approach contributes with additional evaluations in the sense that the Performance Evaluator provides a report in which detailed information about strengths and weaknesses of the participant's cybersecurity skills are presented, including indications on how the participant may improve weaknesses. Our initial experience is that the participants find this evaluation very useful in order to plan further customized cybersecurity learning activities.

7 CONCLUSIONS

In general, there is an urgent need for highly skilled, multi-disciplined cybersecurity professionals, and at the same time there is an awareness gap and lack of integrated training modules on cybersecurity related aspects on all school levels. In order to address this need and bridge the awareness gap, we have developed a method to train and evaluate the cybersecurity skills of participants in cyber ranges based on cyber-risk models.

The target users of our method are the White and Green Teams typically considered in cyber ranges. The method uses cyber-risk models to support the design and execution of training scenarios. The *output* of our method is a performance report for the participants being trained, that is, participants in the Red or Blue Teams. The report provides detailed information based on the executed training scenario exercises reporting strengths and weaknesses of the participant's cybersecurity skills, as well as directions for improving the weaknesses.

We have applied our method in three pilot cases from academia, transport, and energy. Our initial results indicate that the method is easy to use and comprehensible for training-scenario developers (White and Green Team), develops cyber-risk models that facilitate real-time evaluation of participants in training scenarios, and produces useful feedback to the participants (Blue and Red Team) in terms of grading and detailed evaluation of strengths and weaknesses regarding cybersecurity skills.

As next steps, we will carry out empirical evaluations focusing on user experience in the abovementioned large-scale pilots and based on our findings continue improving our method.

ACKNOWLEDGEMENTS

This work has been conducted as part of the CYBERWISER.eu project (786668) funded by the European Commission within the Horizon 2020 research and innovation programme.

REFERENCES

Ashok, A., Krishnaswamy, S., & Govindarasu, M. (2016). *PowerCyber: A remotely accessible testbed for Cyber Physical security of the Smart Grid.* Paper presented at the 2016 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT).

Beuran, R., Chinen, K.-i., Tan, Y., & Shinoda, Y. (2016). *Towards effective cybersecurity education and training (IS-RR-2016-003).* Retrieved from https://dspace.jaist.ac.jp/dspace/handle/10119/13769

Beuran, R., Tang, D., Pham, C., Chinen, K.-i., Tan, Y., & Shinoda, Y. (2018). Integrated framework for hands-on cybersecurity training: CyTrONE. *Computers & Security, 78*, 43-59.

Braghin, C., Cimato, S., Damiani, E., Frati, F., Mauri, L., & Riccobene, E. (2019). A Model Driven Approach for Cyber Security Scenarios Deployment. In *Computer Security* (pp. 107-122): Springer.

CAPEC. (2020). Common Attack Pattern Enumeration and Classification. Retrieved from https://capec.mitre.org/index.html

CIISec. (2019). *CIISec Roles Framework, Version 0.3.* Retrieved from https://www.ciisec.org/ CIISEC/Resources/Roles_Framework.aspx

CWE. (2020). Common Weakness Enumeration. Retrieved from https://cwe.mitre.org/

CYBERWISER.eu. (2019a). Deliverable 2.5, Platform Design, Final Version. Retrieved from https://www.cyberwiser.eu/

CYBERWISER.eu. (2019b). Deliverable 5.1, General Requirements and Guidelines. Retrieved from https://www.cyberwiser.eu/

CYBERWISER.eu. (2020a). CYBERWISER.eu - Cyber Range & Capacity Building in Cybersecurity. Retrieved from https://www.cyberwiser.eu/

CYBERWISER.eu. (2020b). Deliverable 2.6, Risk Model Templates, Initial Version. Retrieved from https://www.cyberwiser.eu/

Damodaran, S. K., & Smith, K. (2015). *CRIS Cyber Range Lexicon, Version 1.0.* Retrieved from https://apps.dtic.mil/dtic/tr/fulltext/u2/a627477.pdf

Davis, J., & Magrath, S. (2013). *A survey of cyber ranges and testbeds*. Retrieved from https://apps.dtic.mil/dtic/tr/fulltext/u2/a594524.pdf

ECSO. (2016). European Cybersecurity Strategic Research and Innovation Agenda (SRIA) for a contractual Public-Private-Partnership (cPPP). Retrieved from https://www.ecs-org.eu/documents/ecs-cppp-sria.pdf

EnpitSecurity. (2020). SecCap. Retrieved from https://www.seccap.jp/

Erdogan, G., Gonzalez, A., Refsdal, A., & Seehusen, F. (2017). *A method for developing algorithms for assessing cyber-risk cost.* Paper presented at the 2017 IEEE International Conference on Software Quality, Reliability and Security (QRS).

Ferguson, B., Tall, A., & Olsen, D. (2014). *National cyber range overview.* Paper presented at the 2014 IEEE Military Communications Conference.

Lund, M. S., Solhaug, B., & Stølen, K. (2011). *Model-Driven Risk Analysis - The CORAS Approach*: Springer-Verlag Berlin Heidelberg.

NRISecure. (2020). Secure Eggs (Essentials and Global Guidance for Security). Retrieved from https://www.nri-secure.co.jp/service/learning/secureeggs

OWASP. (2020). Open Web Application Security Project. Retrieved from https://owasp.org/

Pham, C., Tang, D., Chinen, K.-i., & Beuran, R. (2016). *Cyris: A cyber range instantiation system for facilitating security training.* Paper presented at the Seventh Symposium on Information and Communication Technology.

R-project. (2020). The R Project for Statistical Computing. Retrieved from https://www.r-project.org/

Russo, E., Costa, G., & Armando, A. (2018). *Scenario design and validation for next generation cyber ranges.* Paper presented at the 2018 IEEE 17th International Symposium on Network Computing and Applications.

Solhaug, B., & Stølen, K. (2013). *The CORAS Language-Why it is designed the way it is.* Paper presented at the 11th International Conference on Structural Safety and Reliability (ICOSSAR'13).

Somarakis, I., Smyrlis, M., Fysarakis, K., & Spanoudakis, G. (2019). Model-Driven Cyber Range Training: A Cyber Security Assurance Perspective. In *Computer Security* (pp. 172-184): Springer.

Wieringa, R. J. (2014). *Design science methodology for information systems and software engineering*: Springer.

Yamin, M. M., Katt, B., & Gkioulos, V. (2020). Cyber Ranges and Security Testbeds: Scenarios, Functions, Tools and Architecture. *Computers & Security, 88*, 101636.