Computers & Security 113 (2022) 102551

Contents lists available at ScienceDirect

Computers & Security

journal homepage: www.elsevier.com/locate/cose

TC 11 Briefing Papers

Modeling effective cybersecurity training frameworks: A delphi method-based study

Nabin Chowdhury*, Sokratis Katsikas, Vasileios Gkioulos

NTNU, Teknologivegen 22, 2815 Gjøvik, Norway

ARTICLE INFO

Article history:
Received 3 September 2021
Revised 29 October 2021
Accepted 15 November 2021
Available online 18 November 2021

Keywords:
Cyber-security
Training framework
Delphi method
Learning theory
Personalizedp learning

ABSTRACT

Today, cybersecurity training is commonplace in both large companies and Small & Medium Enterprise (SME). Nonetheless, the effectiveness of many of the current training offerings is put into question by reports of increasing successful cyber-attacks. While a number of models for developing Cybersecurity (CS) training frameworks for industrial personnel or general audience have been proposed, these models often lack consideration for humans aspects of learning (cognitive abilities, learning styles, meta-cognition among others) during development. Additionally, the success of a CS training program highly depends on its ability to engage participants. To develop a CS training framework that is able to motivate participants, we must consider individual-specific factors that can affect the result of training, besides establishing optimal training delivery methods and assessment. For this, in this work we propose a CS training framework based on a revised version of the ADDIE model and more recent research personalised learning theory. The Delphi method was used to both develop and validate our decisions during the development of the training framework model. The results of the decision of the Delphi method have later been compared to recommendations in the literature to create the finalised framework. This work presents two major distinctions from other CS training frameworks models described in the literature. First, the developed model is strongly based in learning theory foundations and takes into consideration differences in learning styles, cognitive abilities and metacognition of individuals, to offer tailored solutions optimized for each group of employees and single individual. Second, the use of the Delphi method and the involvement of experts stakeholders from various sides of academia and industry gave a wide insight into current needs and recommendations for CS training, as well as formal validation for the final development.

1. Introduction

The current landscape of threats in the cyberspace often involves attack vectors that exploit lack of readiness and human preparedness to access confidential data or compromise systems. In the private sector, this has been causing significant economic losses for companies by jeopardizing systems' regular functionality or due to attackers asking for substantial ransoms.

Recent examples of these type of occurrences are the 2020 ransomware attacks against the Toll Group, which happened sequentially at a distance of three months (Osbourne, 2020). The attack forced the logistics company to switch many of the services dedicated to their clients offline. Interestingly, the malware used for the attack known as the MailTo ransomware is notorious for not being a stealthy malware, which suggests that improved personnel security awareness may have allowed to detect and prevent the attacks. Another similar incident involved the health insurance company Magellan, which fell victim to a ransomware attack in the first quarter of 2020 (Davids, 2020). In this occurrence, attackers used phishing mails to impersonate one client of the company, and after obtaining access to the system, steal confidential data about other clients.

As it can be noted from these examples, human negligence and unpreparedness are often the contributing factors to the success of cyber attacks. Humans are indeed often described as the weakest link in Cybersecurity (CS) assurance. This characterization has been motivated by researchers because of human tendencies towards negligence, either because of lack of knowledge (Goh, 2021), but also because of psychological attributes and cognitive biases,

* Corresponding author.
E-mail addresses: nabin.chowdhury@ntnu.no (N. Chowdhury), sokratis.katsikas@ntnu.no (S. Katsikas), gkioulos.vasileios@ntnu.no (V. Gkioulos).

https://doi.org/10.1016/j.cose.2021.102551

which can affect an individuals' judgement when it comes to trust management (Hai-Jew, 2019; Wiederhold, 2014).

To counter this issue, research has been focused to develop methods that would allow to move from the current perspective of humans as a problem to becoming a resource and agent against cyber threats. Zimmermann and Renaud (2019) suggests that for such a transition to happen, there needs to be a shift from the current mindset of control & prevention as the basis for current policies and training in attitudes that encourage active learning, communication and collaboration.

Currently, one of the most prevalent methods for improving human capabilities for CS in the private sector consists of instituting internal CS awareness and training programs to educate and train staff against common attack vectors and prepare them in emergency scenarios. Although such forms of training have become commonplace in the private sector for many years, much criticism has been raised on their effectiveness. The continued increase in cyber-attacks against companies in recent years has been one of the motives for the questioning of the effectiveness of current training offerings.

It has been reported that over the last 10 years, the number of successful malware attacks has steadily increased year-over-year(PurpleSec, 2021), with damage caused by ransomware exceeding $7.5 billion in 2019 alone (at Last, 2021). Additional criticisms raised against modern CS training offerings include their poor ability in changing users' risk perception (Malmedal and Røislien, 2016), lack of agreement on most effective training delivery methods, as well as established evaluation criteria and techniques for these programs, which also contributes to their lackluster performance(Chowdhury and Gkioulos, 2021a). Finally, many CS training programs still fail to engage participants and motivate them towards learning. Lack of user engagement has been indicated in the literature as one of the main detractors to the effectiveness of CS training programs (Bada et al., 2019). This lack of engagement has been justified by CS training and awareness campaigns often being perceived as tedious activities (Bada et al., 2019), or due to not considering preferences in content delivery, participants learning styles and other individual-specific factors that can influence training effectiveness (Pashler et al., 2008; Pattinson et al., 2019).

These factors motivated researchers to focus on delivering training that is more engaging, often by adopting more captivating training delivery methods such as game-based and simulation-based training (Beuran et al., 2017; Hendrix et al., 2016; Nagarajan et al., 2012).

In recent years, significant progress has also been made in the area of Personalized Learning Theory (PLT), which refers to providing training that is tailored to a specific individual, based on their learning objectives, learner's profile and overall preferences in learning (Morin, 2020). Additionally, researchers have also been adapting established learning taxonomies to CS education to improve the overall learning and evaluation process (Harris et al., 2015). Unfortunately, the same considerations have yet to be adapted for CS training or incorporated in current CS training programs. Additionally, differences in prioritization of objectives by different groups of stakeholders involved in the CS training development and employment are also often cause of a decrease in the performance of these programs.

To tackle all these issues, in this work, we propose a novel CS training framework model for CS personnel that takes into consideration the aforementioned individual-specific factors of learning. The CS training framework has been developed based on an initial consultation between a panel of experts in CS on different topics relating to CS training framework development. By using the Delphi method, we were able to consult with a panel of experts in CS from different fields of academia and industry and allow for dynamic and active discussion between participants. After reaching an agreement on all raised topics, the results of the Delphi were utilized to develop a CS training framework model that is theoretically founded.

The main novelty of this work comes from developing a CS training framework reliant on the current progress in research in learning theory and its application in digital education and training. Specifically, very recent educational concepts proposed in PLT were incorporated in the design of the model. Another novelty of this work is its use of the Delphi process as an initial validation method for the later developed model. By involving a very heterogeneous panel of experts when it comes to roles covered in both industry and academia, we were able to consider differences in objectives and prioritization between different roles and agents involved in CS and reach a majority agreement on critical aspects of CS training.

The remainder of the work is organized as it follows. In Section 2 we discuss related work found in the literature focused on proposals for CS frameworks. In Section 3 we describe the overall methodology used in this study, by illustrating in detail all the steps of the Delphi process and how the results of the Delphi were utilized to develop the final CS training framework. In Section 4, we describe the findings of the discussion between the panel on selected topics relating to CS training. These include training development methodology, training components and desirable attributes, training evaluation and the human factor in CS training, which is further defined in the related section. Additionally to the findings from the discussions conducted during the Delphi, these sections also describe and analyse the main recommendations for these components of CS training found in the literature, and the subsequent comparison between the Delphi findings and the literature recommendations. All these findings are then summarized and elaborated in Section 4.7 to present the final CS training framework. Finally, the main conclusions, planned future work and possible direction for later research are presented in Section 5.

2. Related work

While a number of CS training framework has been suggested in the literature, to the best of our knowledge there has yet to be a framework that has been developed taking into consideration the individual-specific factors discussed in Section 1. Nonetheless, a selection of CS training frameworks of interest have been found in the literature, which have also been consulted for the development of the CS training framework proposed in this work and are presented in this section.

In our previous work in Chowdhury and Gkioulos (2021a), we conducted an extensive literature review of CS training offerings, which included CS training frameworks, platforms, test-beds, among other types of offerings, with a focus on training offerings for Critical Infrastructure (CI) protection. A total of 68 articles were included in the review. Methods of training delivery, target audiences, analysis of evaluation criteria, together with general discussion regarding advantages and disadvantages of groups of solutions were presented in the work. Based on the findings of the literature review, delivery methods that offered hands-on experience were often preferred over traditional methods. Simulation-based and game-based training in particular were shown to be a popular CS training tool, both for CI-sector specific training and general CS concept training. In the work, it was concluded that agreement on which solution should be considered optimal has yet to be reached and that further research is needed to establish how to optimally integrate desired attributes found across different proposals.

In Patriciu and Furtuna (2009), the authors propose a step-based design and general guidelines to be followed for the devel-

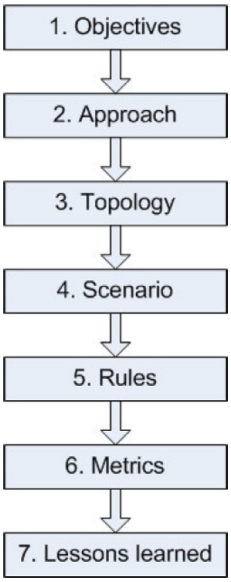

Fig. 1. Design steps for developing CS exercises, proposed by Patriciu and Furtuna (2009).

opment of CS exercises. The model proposed by the authors consists of 7 steps, as shown in Fig. 1.

Aside from design considerations, the authors also suggest possible metrics for evaluating the effectiveness of exercise, by suggesting exemplary Performance Indicators (PIs). The model offers a general initial approach for developing CS exercises, in a methodological and structured manner. Nonetheless, this design lacks considerations for the human factor in training.

Beuran et al. (2017) developed CyTrONE, which is described as an integrated CS training framework, designed and implemented to address shortcomings of CS training which requires manual setup and configuration of training environment, by automating the training content generation and environment setup task (Beuran et al., 2018). CyTrONE uses input from a training coordinator to generate the training content for a particular training session and uploads it to an e-learning system. CyTrONE also creates the cyber range training environment corresponding to that training content. CyTrONE has been developed by combining the following components: (1) a User Interface (UI), a training database, (2) a training description generator that allows to take the organizer input to select the appropriate sources from the training database, (3) a content description processing module which converts the training content description that is generated by the training description generation module to a format that is suitable for e-learning systems and finally (4) a cyber-range instantiation module, named CyRIS (Cyber Range Instantiation System) (Pham et al., 2016). This last component was developed to automatically create a cyber range based on its specification. This includes: (i) training environment setup functions; and (ii) security content generation functions. CyTrONE's high level of personalization, automatic generation of training scenario and cyber range instantiation make it an overall great tool to integrate in any CS training framework that requires high scalability and is to be used by a large number of participants. Since developments of the training content for CyTrONE are still at their initial stages, further work and experimentation is necessary to verify whether the tool can be of use in CS training in enterprise settings.

Brilingaitė et al. (2020) present a framework to aid in the development and assessment of cybersecurity competences during hybrid CS exercises, which involve both CS skilled and non-skilled workers. The framework involves 4 phases: pre-exercise assessment, pre-exercise training, live exercise and post-exercise assessment, as shown in Fig. 2.

The framework proposed by the authors supplements typical CS exercise life cycles to enable competence assessment at an individual level to reach learning objectives. While the authors do give consideration to motivation as a critical aspect of the success of a CS exercise in achieving its initial goal, other factors, such as meta-cognition and learning styles, are not considered. Also, when it comes to assessment, the authors analyze possible evaluation methods only at a high abstraction level, without indicating in detail the advantages of certain methods over others.

Zhang et al. (2021) developed a theoretical framework for conducting a cost-benefit analysis of CS awareness training programs. The authors differentiate three types of CS awareness training programs (constant, complementary and compensatory) in terms of their costs and four types (ineffective, consistent, increasing and diminishing) with respect to their benefits. The authors also investigate the impact of CS awareness training programs with different costs and benefits on a company's optimal degree of security, and found that if a company is to implement such a program with a projected consistent cost and a constant benefit, the optimal degree of security will remain the same, while a program with a compensatory cost will help a company move to a higher level of security since the company can take advantage of such a program to incur a lower cost at a higher security level, whereas the opposite is true for a complementary program. The author's analysis provides an interesting tool to better understand whether a CS training program is valuable for its overall cost, which is often one of the key criteria used by companies to select which training program to implement. That being said, the analysis provided does not consider the impact of CS awareness training programs on a company's total cybersecurity cost because companies with different sizes may vary significantly in their anticipation and failure costs.

Rajamäki et al. (2018) propose a holistic cyber resilience and security framework for developing and delivering a multilateral educational and training scheme based on a proactive approach to cybersecurity. The framework proposed by the authors is built on the principle that "education and training must be interactive, guided, meaningful and directly relevant to the user' operational environment" (Rajamäki et al., 2018). The framework addresses capacity mapping, cyber resilience level measuring, utilizing available and mapping missing resources, adaptive learning technologies and dynamic content delivery.

In Aldawood and Skinner (2018) and later in Aldawood and Skinner (2019a) the authors discuss CS training solutions for assessing and raising awareness of social engineering threats. In the study, a variety of methods for training are identified, including serious games, gamification, virtual labs, tournaments, simulations, and the use of other modern applications. Similarly, current awareness programs that educate against social engineering threats including video streaming, compliance, theme-based trainings, awareness campaigns, and conferences are also included. Serious games and simulations are noted in the work as some of the most effective and latest solutions against social engineering threats, which confirms that their applicability and effectiveness extend outside of regular CS training. Both techniques use real life experiences of social engineering threat scenarios in a single location or for a whole department in which the participants get to know different situations they may face as a threat and the best methods to tackle them.

3. Methodology

To develop a theoretically founded model for CS training frameworks, we started by conducting a literature analysis of theories of learning and training applicable to CS and digital environments.

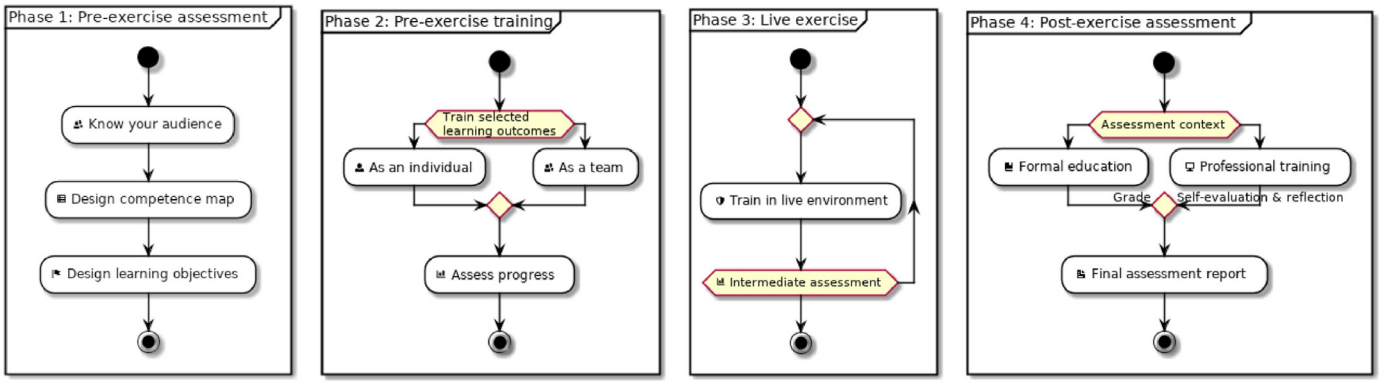

Fig. 2. Phases for CS training based on the CS framework proposed by Brilingaitė et al. (2020).

The purpose of this analysis was to ensure that the later developed model for training considers human attributes that affect the outcome of training programs. These attributes include cognitive abilities, learning styles and various forms of biases, among others. This information was utilized together with the data previously collected in Chowdhury and Gkioulos (2021b), Chowdhury and Gkioulos (2021a) and Chowdhury et al. (2021) focused on CS training offerings in the literature and in the industry to develop the questionnaires utilized during the Delphi method. The method was utilized both for the development and validation of the proposed CS training framework. The Delphi consisted of sending out electronically a description of the central problem and providing the collected background knowledge to a panel of experts and stakeholders in CS training in the industry. The decision of using the Delphi method as a validation technique for our model came from its ability to collect equally weighted feedback from multiple participants and allow for open debate, without requiring practical experimentation. The panel was selected based on a stakeholder analysis for CS training programs, with the goal of finding a sufficient amount of individuals for each of the following categories:

- CS trainers and educators (both individuals and entities that provide CS training services);
- Trainees in the industry (this could be both limited to CS roles or extended to all personnel);
- Researchers working on related topics;
- Personnel in charge of establishing, maintaining and supporting CS training (managerial personnel, human resources, etc).

Invitations for participating in the process were sent to 18 individuals based on their competences, knowledge, previous and current occupation. The final selection came to 10 individuals after acceptance, which is above the suggested minimum requirement of 8 participants for Delphi (Hallowell and Gambatese, 2010). The panel was composed of senior professors in the field of information security as well as researchers in the same area, experienced CS instructors with industry work experience and other CI personnel that covered organizational CS roles and were in charge of managing their respective companies' CS training programs. The remainder of the Delphi method consisted of the activities listed below:

1. Establish a problem statement - The problem statement represents the general question that will be central to the topics discussed during the process. In our case, the problem statement is the following: *How to develop a Cybersecurity (CS) Training Framework that takes into consideration the human factor?*
2. Appointing facilitator - The main author of this work was appointed as the facilitator of the Delphi method. The facilitator had the responsibility of recruiting the panel of experts, developing and sending out all questionnaires for each round of the Delphi method.
3. First round of Delphi - The first round of the Delphi was conducted digitally and consisted of participants answering an interactive questionnaire. The questionnaire has been developed by using Mentimeter and is composed of 3 main topics: (i) CS training development methodology, (ii) CS training components and attributes, and (iii) the human factor in CS training. To facilitate dialogue, each section begun with multiple option questions and ranking questions. At the end of each section, we allowed participants to open discussion and suggestions to the selected topics, through digital open panels.
4. Following round of Delphi - After completion of the first round of Delphi, agreement regarding CS training development methodology and training attributes was reached by participants, while suggested approaches to consider for the human factor in training were more heterogeneous. For this reason, a second round of Delphi was conducted to discuss proposed approaches and come to a majority decision.
5. Conclusion of the Delphi method - Once agreement on each topic proposed and raised by the panel was reached, the Delphi was determined to be concluded. The results were distributed to all participants and used to develop the final model.

Initial input for the topics discussed during the two rounds of Delphi came from our previous work in Chowdhury and Gkioulos (2021b) and Chowdhury and Gkioulos (2021a), additionally to further literature analysis, particularly when it comes to learning theories that account for the human factor. More detail on this is found in Section 4.6. After obtaining and textualizing the final results of the Delphi into a final report, we further compared the final suggestions of the panel with the recommendations collected from the literature. The result of this comparison was then utilized to develop the proposed CS training framework.

4. Results

As briefly mentioned in Section 3, during the Delphi the following aspects of designing and developing a CS training framework were discussed:

Each of the aspects shown in Table 1 was separately discussed during the two Rounds of Delphi, and later compared to the recommendations from the literature. Both findings from the discussion during the Delphi and the comparative analysis with the literature can be found in the following sections. These findings have later been used to develop the model proposed in Section 4.7.

4.1. CS Training development methodology

The initial selection for the development methodology of the proposed CS training framework came to using the ADDIE (Analysis, Design, Development, Implementation, and Evaluation) model,

Table 1
Key aspects of CS training discussed during the two Delphi rounds .

Aspect	Description
CS Training development methodology	Preferred methodology for designing each module of the training and the overarching final product
CS Training: desirable attributes	Attributes often recommended in the literature or by the panel of experts to be incorporated during development of the CS training
CS Training delivery methods	Preferred training methods (game-based training, simulation-based training, for example), based on reported preferences of CS training participants and instructors
Training content	Content of training based on key skills and competencies that are required by CS personnel, based on literature reviewing
Training assessment & evaluation	Methods for evaluating both formatively and summatively both individual training components and the overall training program, additionally to metrics and other methods for evaluating participants' progress in knowledge and skills acquisition
CS training: The human factor	Individual-specific factors that may affect the final outcome of the training. Examples of these include preferences in learning styles, engagement and motivation, among others

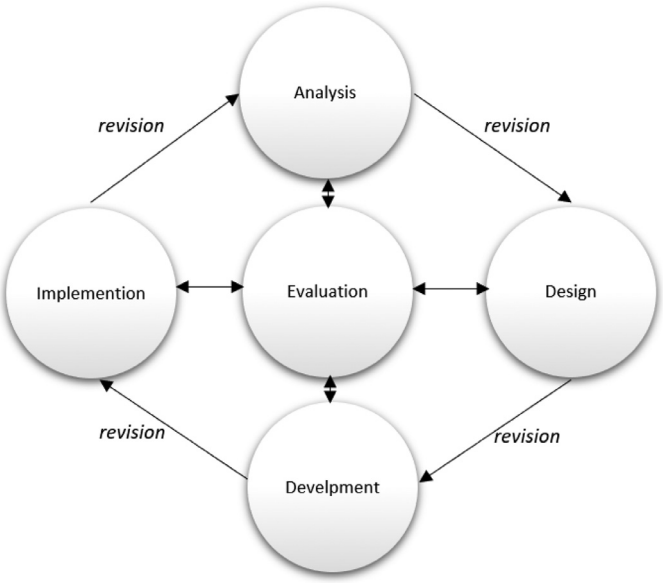

Fig. 3. Revised ADDIE model, based on rapid prototyping.

one of the more renowned models for instructional technology (Molenda, 2003).

While original iterations of the model propose a static, cyclical process for the completion of all phases, starting from the analysis phase to the evaluation phase, during the panel discussion it was concluded that a more modern approach based on rapid prototyping would be preferable. Such an approach uses continual or formative feedback during each phase to assess each output at its time of development, allowing for additional dynamicity and interactiveness (Nixon and Lee, 2001). Two main advantages were noted when using this approach: (i) it allows for easier identification and direct revision of any single component that may require changes, based on feedback and evaluation; (ii) it also decreases the time needed for overall evaluation and revision, due to targeting issues independently. A conclusive evaluation of the final product is still recommended, as the final result of the developed training may differ from the result of each single component.

Definition of all tasks to be completed during each phase of the model was then established. As the model developed is meant to be applicable to different forms of CS training, the tasks were defined to be inclusive and adaptable for any required application. The initial approach was based on the suggested principles of instructional design by Gagne et al. (2005) and the revised activities for the ADDIE model suggested in Allen (2006). These concepts were then adapted, with a particular focus on current research on e-learning applications (Alajmi, 2009).

In Allen (2006) it is also noted that evaluation activities may involve different forms of assessment, which are often distinguished as: (i) **formative evaluation**, comprising process and product evaluations to provide feedback. These are conducted during the analysis, design phases, and the development phase; (ii) **summative evaluation**, consisting of operational test cases and tryouts conducted after the conclusion of the development phase;(iii) **operational evaluation**, consisting of periodic internal and external evaluation of the training and its component during the implementation phase.

Further discussion brought to the final agreement on the activities listed below for each phase:

- **Analysis Phase:** Analyse training needs. Establish goals, possible pre-requirements, target audience and audience preferences (on training delivery, content, etc.), and resource requirements (budget overhead, instructors, hardware and software resources, facilities).
- **Design Phase:** Design training solution aligned with goals and requirements, based on inputs from analysis phase. This includes selection of training delivery method and possible decision on whether new material may need to be developed to satisfy the requirements and goals of training.
- **Development Phase:** Develop an action plan, needed training resources and a pilot test. Validate the components developed. Validation may be done by internal reviewing, test cases, or both.
- **Implementation Phase:** Implement training solution by preparing a training environment and training activities. Engage participants in training.
- **Evaluation Phase:** Evaluate the quality of the training resources, user engagement and satisfaction, and overall training results. Evaluation is to be conducted throughout the life cycle of the training development and also regularly after its installation and use.

Fig. 3 shows the agreed final version of the model.

4.2. CS Training: Desirable attributes & components

As mentioned in Section 4.1, development of the various components of CS training should be dependent on the requirements and goals of training, with trainees preferences and overall resource overhead being additional considerations. That being said, many recommendations are found in the literature when it comes to key components and desirable attributes to consider when developing a CS training program or tool. For this reason, the second part of the Delphi process focused on CS training components and key attributes, as well as evaluation criteria to be used for determining the effectiveness of training. In Chowdhury and Gkioulos (2021a), an analysis and summary of attributes to consider during the development of training frameworks is presented. These include considerations for proposed training delivery and material, content of training, as well as other factors. Similarly, He and Zhang (2019) gives recommendations regarding best practices and key attributes to include when developing CS training for enterprise. These considerations, together with additional suggestions,

Table 2
Key attributes and considerations for the development of CS training activities, according to Chowdhury and Gkioulos (2021a) .

Attributes	Description
Suitability	Training content should be appropriate to the target audience and specific company in terms of content, skills developed and level of training (Adams, 2018).
Real-life Experience	Training should include hands-on activities developed to emulate or simulate real-life scenarios. Such activities should also focus on developing communication and team skills of participants.
Scalability & Adaptability	Training should be developed so that modification, upgrade and extension of content should be possible, based on the skills and level of knowledge of the target audience, as well as new information on technologies and vulnerabilities.
Accessibility	Training activities should be accessible to all staff that may benefit from such activities. Developing remotely accessible training further benefits this goal.
Frequency of Training and Periodical Updates	Training should be conducted and updated periodically. Progress sessions should be planned to ensure that KSAs of personnel are up-to-par to current standards and recommendations (He and Zhang, 2019).
Cost Efficiency	Training activities should take into consideration resource constraints of a company (budget, time and training personnel constraints.)
Consideration for the human factor	CS training should consider participants engagement and motivation (Gross, 2018; Kostadinov, 2018), adapting to different learning styles (Kostadinov, 2018; Nadkarni, 2012) and stimulating metacognition (Pokorny, 2017)

have been summarized in the list of key attributes for CS training shown in Table 2.

Discussion of these attributes with the panel of experts focused on establishing two main conclusions: (i) validity, overall value and prioritization of the mentioned aspects in the development of a CS framework; (ii) additional key attributes to be considered when developing CS training. While overall agreement on all attributes being of relevance when developing a CS training program was reached by participants, certain attributes were weighted higher than others, and discrepancies between different stakeholders' prioritization was noted. **Ease of implementation and use** was mentioned as an additional desirable attribute, and was ranked highest among all attributes when it comes to prioritization. In particular, stakeholders from the industry focused on the importance of training offerings that can easily be implemented, either as complementary to established solutions or as independent ones. Ease of implementation and use would benefit to the satisfaction of other desirable attributes, making scalability and cost efficiency more easily achievable.

When it comes to suitability of training, discussion focused on differences in development between generalized CS training and role-specific CS training. Generalized CS training activities usually refer to training offerings that are meant for all personnel of one firm or multiple firms (Lee et al., 2016). These types of activities are often part of CS awareness campaigns with the objective of providing basic knowledge about CS topics, informing personnel of companies security policies and risk profiles, and overall increment in alertness (Bada et al., 2019; Tirumala et al., 2019).

The experts from the industry suggested that training of personnel should consider companies risk profiles and link them to the training needs of specific roles in the company. Two benefits of doing this type of analysis are "improved tailoring of training to satisfy initial requirements as well as persuading the firm to further invest in training, by indicating the risks that could incur in its absence".

4.3. CS Training delivery methods

In reviewing CS training offerings proposed in the literature in Chowdhury and Gkioulos (2021a), we established 5 major groups of training delivery methods. Table 3 list examples of training for each of the group, together with advantages and disadvantages for each group of training

In the literature, game-based and simulation-based methods of training are often recommended for CS training, or they are suggested as a form of complementary training to traditional delivery methods (Abawajy, 2014b).

The main advantages that these two methods provide include allowing participants to conduct interactive, hands-on activities, develop team skills including communication and organizational skill, as mentioned in Table 3. Additionally, these activities have been demonstrated to be more engaging and stimulating than traditional training methods. As mentioned in He and Zhang (2019) and Bada et al. (2019), user engagement and motivation are two of the most significant factors in the success of CS education and training. Tedious or non-engaging training solutions often fail to change employees security behaviour and attitudes (He and Zhang, 2019). This preference was also confirmed by the panel of experts, with an equal majority of participants selecting these two methods over other suggestions.

When it comes to selecting between game-based training and simulation-based training, a few distinctions in properties and possible applications should be made. Game-based education has as its main strength that of being highly engaging. Gamification and game mechanics have been proved to stimulate collaboration and competition, self-efficacy and self-assessment, while maintaining. The reduced costs of gamified training have also contributed to the increase in popularity of game-based training. On the other hand, simulation-based training suffers from high initial overhead. Nevertheless, simulation-based training is the only training method that allows participants to conduct exercises that are equivalent to possible real-life scenarios.

4.4. Training content

While the specific content of training was not discussed during the Delphi, in Chowdhury and Gkioulos (2021b) we conducted an extensive literature analysis of competencies and skills to develop for CS training.

Based on our previous findings, when it comes to competencies and skills required by CS personnel, 4 main categories can be identified: technical skills, non-technical (soft skills), implementation skills, and managerial skills. Examples of the main competencies required for each of these categories have been listed in Table 4.

Additionally, in Chowdhury et al. (2021), we interviewed several CS workers in Norwegian Critical Infrastructure companies to investigate on the training offerings currently available at their respective companies and the content of these offerings. According to the study, the following topics were the most common focus of current training offerings:

- Network Architecture;
- Information Handling (information disclosure, information sharing and reporting);
- Cyber threats & relevant potential attacks and system vulnerabilities;
- Procedures and preparedness plans for cyber incidents;
- Security Management System (Risk assessment & management, mitigation strategies, control strategies, documentation);
- Human factor aspects (Communication, trust management, teamwork skills, decision making);
- Surveillance;

Table 3
Classification of CS training methods according to Chowdhury and Gkioulos (2021a), with examples found in the literature and associated advantages and disadvantages.

Delivery Method	Examples	Advantages	Disadvantages
Conventional Methods	On-site training; Classroom training and exercises; Presentations & Conferences; On-site training sessions;	Usability; Familiarity of format; Multiple messages can be conveyed at once; Ease of communication between instructor and participants; Real-time resolution of issues;	No guarantee of personnel active participation; Can be perceived as tedious (Leach, 2003); Does not always provide hands-on experience; Provides a static solution for a fluid problem (Valentine, 2006); High cost and resource overhead; Time-consuming;
Online and Software-based	Online courses; Cloud-based training; Web-accessible training material and software; E-mail tests;	Remote and multi-modal accessibility (Abawajy, 2014a); Industry-wide standard use; Cost-effective; Hands-on exercises; (Possible) team skills development;	Users may undermine the value/pay less attention; Not always very scalable and adaptable; High cost and resource overhead, if personalized solution is needed; Does not provide instructor assistance;
Game-based	Serious Games for CS Awareness and Training	Team skills development, Engaging to users; Hands-on exercises, Demonstrated effectiveness (Antonioli et al., 2017; Beuran et al., 2018); Adaptability; (Possible) Remote Usability; (Possible) High scalability;	Older audiences may not be familiar with mechanics; Time-consuming; May not reflect real-life processes. High initial development cost and resource overhead;
Video-based	Educational videos	Accessibility, Usability; Cost-efficient; Time efficient;	Limited content. Lack of interactivity with other trainees or instructors. Lack of hands-on experience. No guarantee of personnel active participation; Requires constant integration and updates for scalability;
Simulation and virtualization-based	Testbeds, Simulation platforms, Simulated Laboratory exercises	Team skill development; Hands-on experience; Replication of real-life incidents; Adaptability; (Possible) Remote Usability; (Possible) High scalability;	Hard to coordinate (Kumar et al., 2015); Requires pre-existing knowledge; Time-consuming; High initial development cost and resource overhead;

- Crisis contingency & management;
- Incident response & management;
- Intrusion detection;
- Managerial skills training;

While this information provides a general overview of what type of content should be trained during CS training sessions, exact subjects should be determined based on the goals of both training participants and the institution offering the training, which should be determined during the initial analysis phase of the ADDIE model.

4.5. Training assessment & evaluation

In our previous work in Chowdhury and Gkioulos (2021a), we found a lack of agreement in preferred evaluation metrics and performance indicators to be used when evaluating CS training output. Additionally, research in the area is limited and no studies on the best uses of different evaluation criteria was found.

This limitation in both literature and in organizations was also noted by the National Institute of Science and Technology (NIST). To address the issue, researchers at NIST presented the following list of suggestions of measures that can be used at an organizational level to indicate whether any implemented CS training is successful :

- **Top-Down Leadership buy-in:** leadership (executives, managers, supervisors) support is noted as one of the key factors to measure effectiveness of CS training (Adams, 2019). Seeing leaders participating or championing training offerings not only indicates positive evaluation of the training, but also motivates employees to participate and be more engaged.
- **Workforce Training Measures:** Specific measures should be used to determine in a continuous cycle the non-technical, technical and managerial capabilities of personnel after training (Adams, 2019).
- **Risks, Vulnerabilities, POA&M Measures, & Cybersecurity Compliance:** risk and vulnerability captured from assessment can determine what security control implementation users need to be trained on. CS compliance can show that a successful training program has been implemented that enables the organization to understand and meet security requirements (Adams, 2019).

Aside from these general recommendations, several evaluation criteria have been utilized in the literature to assess post-training results. Chowdhury and Gkioulos (2021a) provides a summary of metrics and assessment methods commonly reported in the literature for CS training, which are reported in Table 5.

When discussing evaluation criteria with the panel, focus was given on evaluation of the effectiveness of listed criteria, discussion on additional criteria and finally on feedback collection. Feedback collection and comparison of pre and post training evaluation were selected by the panel as the most effective methods of evaluation. The two methods provide significantly different outputs, as the former is a qualitative approach based on training participant input, while the latter is a quantitative approach that analyzes specific performance indicators (PI).

Table 4
Mapping of skills and competencies for Critical Infrastructure Protection (CIP), according to Chowdhury and Gkioulos (2021b) .

Skills & Competencies Mapping Table			
Technical Skills	Soft Skills	Implementation Skills	Management Skills
1. Understanding of digital security concepts; 2. Understanding of evolving threats; 3. Understanding of attack intelligence; 4. Penetration testing skills; 5. Cryptology knowledge; 6. SW & HW security skills; 7. Network security skills; 8. Computer forensics skills; 9. Programming skills; 10. Data analytics skills; 11. Information security skills; 12. Wireless security skills; 13. Ability in using IDS tools;	1. Information sharing and communications; 2. Public speaking and presentation skills; 3. Situational Awareness; 4. Cognitive and behaviour analysis; 5. Ability to work independently; 6. Trust management; 7 Teamwork; 8. Motivation; 9. Time management; 10. Networking; 11. Confidence; 12. Work habits;	1. Threat and vulnerability assessment & management; 2. Event and Incident Response; 3. Continuity of Operations;	1. Risk management; 2. Identity and access management; 3. Asset, change and configuration management; 4. System administration; 5. Workforce management; 6. Cyber-security program management; 7. Supply chain and external dependencies management; 8. Evaluation of policies effectiveness; 9. Project planning;

4.5.1. Performance indicators for training evaluation

Performance indicators or key performance indicators (KPIs) are defined as measurable values that demonstrate the effective achievement of certain key objectives. These indicators are often utilized in businesses and firms to evaluate the performance of individuals, processes and of the organization as a whole.

In the context of CS training, performance indicators are defined in the NIST documentation as computable performance assessment, as derived from a combination of metrics or in other words, compute value based on post-analysis which may utilize one or many primitive values to perform the computation (Tang, 2017).

Many KPIs can be found in the literature relating to CS performance analysis of specific areas, such as process control systems (Tang, 2017), robotic (Zimmerman, 2017), big data (Petrenko and Makoveichuk, 2017) among others. Less research has been conducted to establish preferred KPIs for CS training, due to the variance of objectives for different training and exercises as well as lack of agreement in evaluation methods for training. In Chowdhury and Gkioulos (2021a), a list of exemplary KPIs and metrics of evaluation is given, based on an analysis of evaluation methods adopted for different types of CS training for Critical Infrastructure personnel. Additionally, Boerman (2020) provides a classification of KPIs based on the five NIST perspectives of CS (Identify, Protect, Detect, Respond, Recover). In the work, the authors recommend selecting KPIs based on stakeholders' input and preferences. Further grading on which KPIs may yield the most relevant data to evaluation the training objective should also be used to prioritize certain indicators and avoid over encumbering evaluation.

In the NIST documentation for developing a CS scorecard (Wagner, 2016), it is recommended to start by selecting one key performance indicator (KPI), based on a specific desired outcome. In addition to this first KPI, it is suggested to add complementary indicators that may aid in measuring the various factors that can influence the outcome of the training. Samuel (2019) suggests that KPIs for CS training should measure one of the following attributes: Accuracy, Timeliness, Completeness and Authorization. A number of exemplary KPIs for different types of measurements such as identity & access management, configuration management, security awareness, security incidents, compliance, data leak prevention, vulnerability and patching are given in the literature (Samuel, 2019) and should be considered during the development of specific CS training exercises.

The aforementioned criteria for KPI selection were also confirmed by the panel of experts during the Delphi process, albeit specific definition and selection of KPIs was avoided due to discussion being focused on general parameters for training and not on the development of a specific CS training offering. One of the experts did however suggest utilization of role-specific KPIs, meaning KPIs related to each role of a CS response team, in both team exercises and individual training.

4.5.2. Feedback collection

Feedback is often described as an essential tool and performance indicator for post-training evaluation (Andriotis, 2018; Farooq et al., 2011). Benefits of collecting feedback as an evaluation tool are many, including constant training program improvement based on learners' input, increase participants' motivation and performance (DeFranzo, 2018).

Feedback is differentiated in formative and summative feedback. Formative feedback is collected during the training and is used to enhance or modify training components in real-time (UniTo, 2018), while summative feedback provides an evaluation of how much a student and the class has learned. When tied to specific learning objectives, it can be used as course feedback, providing the instructor with feedback about the effectiveness of the course design. Examples of summative feedback techniques include exams, final projects, and research reports (Miller, 2018).

When gathering feedback, UniTo (2018) suggests that data collection should remain optional, anonymous, and not linked to individual evaluation. The most common methods for post-training feedback collection include questionnaires and surveys, both by using online tools and paper-based data collection. Interviews with participants are also another possible feedback collection method, albeig less used. When collecting feedback, it is critical to establish what type of information need to be collected, as well as how this information can be used to improve the current training offering. According to Andriotis (2018), the following 5 elements should always be included during post-training feedback collection:

- **Effectiveness of training:** Effectiveness is a critical element to measure the performance of a training program as it establishes

learners perception of whether the course helped them attain their learning objectives and how relevant it was for them.

- **Comprehension:** Comprehension refers to the effectiveness of the course delivery and as such is focused about the way the course content was delivered. This element also includes the conciseness and clarity of content.
- **Attractiveness:** Attractiveness of a training program refers mostly to how the material and tools used during training looked and felt to the learners. It is especially relevant for software-based training, such as game-based, simulation-based or online training.
- **Engagement:** One of the most critical aspects in the success of a training program depends on user engagement. As overall training engagement is a multifaceted issue, evaluation of individual training components should be collected from participants to highlight any weak points.
- **Suggestions:** Suggestions for improvement from training participants should also be collected. Andriotis (2018) notices that suggestions are often skipped during feedback surveys and for this reason recommends asking participants to include a minimum required number of suggestions.

When it comes to components of training to evaluate through feedback, Sviridenko (2018) recommends analyzing content, course length, exercises, instructor and tools, platforms and any other type of training media & material. There are also certain limitations, however, to the effectiveness of feedback as an evaluation tool for training. Firstly, gathering and analyzing feedback can be a long, complex process, especially in the case of large number of training participants and if information is collected in a non-automated manner. A way to circumvent this issue is to develop or incorporate automated feedback collection tools to the training programs that can allow to generate a summative log or report of the feedback obtained.

Input from the panel was collected during the first round of Delphi regarding what information and measures should be collected from training participant feedback. Suggestions from the experts included perceived knowledge and skills, motivation and interest towards the type and content of training, level of understanding, self and team assessment and relevancy to their role and work. To circumvent the previously mentioned limitations to the subjective, qualitative assessment provided by feedback, one participant suggested playback possibility as a way to compensate the limitation and allow for a quantitative assessment of improvement.

4.6. Cybersecurity training: The human factor

The final section of the Delphi focused on how individual-specific factors may influence the outcome of CS training. Factors such as cognition & meta-cognition, engagement & motivation, adaptability, human error and learning styles have all been cited by the panel as being of influence in the effectiveness of training and should be considered in the development of training.

To conciliate the technical requirements of specialized CS training and the factors mentioned by the panel as affecting training outcome, research on learning theory, instructional design and learning taxonomies was conducted prior to initiating the Delphi. After preliminary research, an approach based on merging the ADDIE model to two learning taxonomies tailored to training supported by the use of digital instrumentation was proposed. The selected taxonomies included Bloom's digital taxonomy and Solo's taxonomy.

Bloom's digital taxonomy is a modernized variation of the revised Bloom's taxonomy (Churches, 2010). In Fig. 4, the initially proposed taxonomies discussed with the panel of experts are shown.

Several criticisms were raised during the discussion regarding appropriateness of the indicated taxonomies and their possible alignment to CS training development methodologies.

More in detail, the panelists found that aligning the ADDIE model to these taxonomies may not be easily feasible and may require extensive further research and work in adapting the aforementioned learning taxonomies to the requirements of CS training. Panelists also criticized the taxonomies hierarchical structure of learning, citing that while it may be appropriate for knowledge acquisition, they may not be adequate for CS training.

For this reason, during the second round of questionnaires, additional input was collected from the panel on methods and learning theories that may be more suitable to CS training. One of the main conclusions of the discussion between the panelists was that different modes of information communication (utilization of images, text, videos, verbal communication, etc.) as well as different training delivery methods may be required to satisfy and tailor training to specific target groups or individuals. For this reason, an additional suggestion of utilizing modern findings of personalized learning theory (PLT) was taken into consideration.

PLT is generally defined as an education approach that aims to customize learning based on students' needs, interests and abilities (Walkington and Bernacki, 2020). Resarch on PLT is still relatively novel, and many heterogeneous approaches and models have been proposed in the literature. Certain key elements of PLT that are common to most of the proposed models have been established in the literature. In Diana (2019), the following 5 elements are highlighted:

- **Student Agency:** Student Agency (SA) or other times referred to as ownership or control indicate students actively taking responsibility and becoming active participants over their own learning, by becoming more aware of their strengths and weaknesses, advance mastered skills and reinforce skills they lack. Teachers only facilitate the content acquisition while students internalize it and own it too. To support this, it is necessary to provide students with personal learning spaces and activities such as the ones provided by Learning Management Systems (LMS) and other similar tools, online forums and communities. Additionally to giving them more autonomy and self-regulation, this also allow to give and receive feedback from peers, leading to greater achievement levels, greater class participation, better preparation, self-awareness and decreases in behavioral problems Diana (2019).
- **Flexible Learning Environments:** Instruction is often still delivered in traditional learning environments, typically classroom-based, which are known to hinder the learning process. In flexible learning environments, students have more control over how they learn. This is achieved by modifying the traditional learning environment to one that enables additional cooperation between students, as well as more interactivity. Designing such a space with places for solitary work, collaborative work and for debates and mini lessons gives students confidence and it leads to improved academic results, better peer interaction and less bored students (Diana, 2019).
- **Individual Mastery:** With independent modules designed as part of a larger learning goal, students can focus on mastering skills, at preferred schedules, location and pace of learning. For this, teachers have to offer individualized support and guidance and students need self-motivation, grit, perseverance and agency.
- **Personal Learning Paths:** Personalized learning involves adjusting instruction to students learning pace with the goal of tracking long-term learning. This is currently often assisted by LMS which help educators create classes, assess students and add individualized learning paths. An LMS provides the nec-

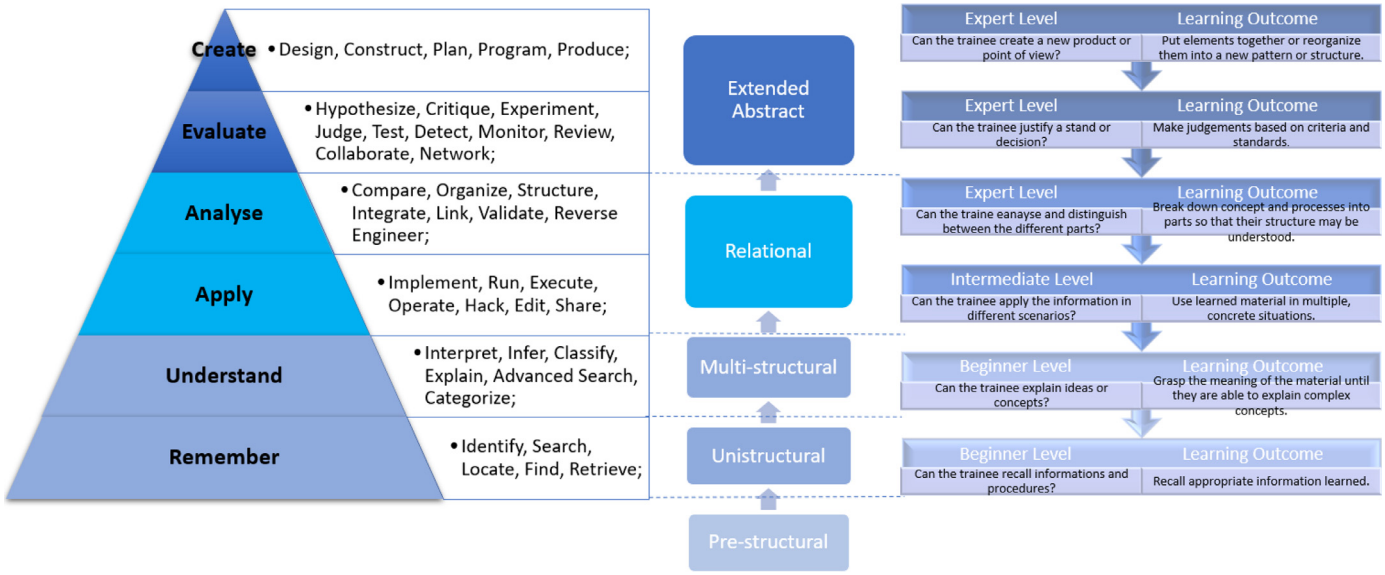

Fig. 4. Revised versions of Bloom's Digital Taxonomies and Solo Biggs' Taxonomy for CS training.

essary tools for teachers to meet students needs by personalizing goals within paths and creating a flexible virtual learning environment. LMSs also give students agency over their learning and allows them to become proactive by having access to self-assessment tools, such as quizzes and surveys with instant feedback, but also a space to express themselves and keep track of their progress (Diana, 2019). Personalized paths adapt to multiple learning styles and focus on how students experience learning offering customizable modules to respond to individual learning needs.

- **Learner Profiles:** Having individual learner profiles give powerful insights on progress, to teachers and students alike. By analyzing students progress teachers can create personalized content, assign individual goals and give customizable feedback while students are able to build on their strengths, overcome their weaknesses and follow their own goals and interests. While the process of creating learner profiles may have been more challenging in previous eras, the introduction of ever more sophisticated LMS has allowed both teachers and learners to better track their progress and develop personalized content based on their profiles.

Models for PLT vary greatly, due to the heterogeneity of students' preferences and needs. Out of the various proposed models in the literature, the following 4 models have been highlighted as the most common in academic settings (Morin, 2020), with possible applicability also for learning in professional environment:

- **Learner Profiles-based models:** This model keeps an up-to-date record that provides a deep understanding of each learner's individual strengths, needs, motivations, progress and goals. The records are associated to each learner's profiles, which are periodically and often updated, to aid both instructor and learner to keep track of the individual's progress or to understand if there is any need for changes in learning methodologies or other requirements.
- **Personalized learning paths-based models:** This model helps each learner customize a learning path that responds or adapts based on progress, motivations, and goals. An example of this would be a learner's schedule based on weekly updates about training progress and interests. Each learner's schedule is unique and might include several learning methods. A personalized learning path allows a learner to work on different skills at different paces.
- **Competency-based progression:** This model continually assesses learners to monitor their progress toward specific goals. This system makes it clear to learners what they need to master. These competencies include specific skills, knowledge and mindsets. Students are given options of how and when to demonstrate their mastery. For example, a student might work with a teacher to weave certain math skills into an internship at a retail store. The student might work on several competencies at the same time. When they master one, they move on to the next. Each student gets the necessary support or services to help master the skills. The emphasis isn't on taking a test and getting a passing or failing grade. Instead, it's about continuous learning and having many chances to show knowledge.
- **Flexible learning environment-based models:** This model simply adapts the environment learners learn in, based on how they learn best.

PLT has been noted in the literature as being particularly advantageous when compared to traditional learning theories due to several of its properties. According to a study by Pane et al. (2015), where usage of PLT in several academic institution was monitored and analyzed, it is reported that compared to peers, students in schools using PLT practices are making greater progress over the course of two school years, and that those students who started out behind are catching up to perform at or above national averages. The study finds that teachers at most schools were using data to understand student progress and make instructional decisions, all schools offered time for individual academic support, and the use of technology for personalization was widespread. However, some strategies, such as competency-based progression, were less common and more challenging to implement. Positive effects on student mathematics and reading performance were shown as result of adoption of PLT, with even lowest-performing students making substantial gains relative to their peers. Adoption of personalized learning practices varies considerably. Personalized learning practices that are direct extensions of current practice are more common, but implementation of some of the more challenging personalized learning strategies is less common. Three elements of PLT have been cited to give the largest achievement effects when implemented in tandem: Learner Grouping, Learning Space Sup-

Table 5
Metrics categories identified from the literature.

Metrics and KPIs cat.	Type	Classification	Measurement Units	Data Source
CS incident records	Quantitative	Effectiveness	Number of data breaches or other incidents that occurred before and after training.	Internal Reports on attacks and incidents.
User Performance	Quantitative	Effectiveness	Outcome of CS exercises and tests; Comparison of pre-training and post-training test results; Evaluation of threats detection, prevention and report rates, from tests and real-life occurrences	Data analytics from exercises; reports from evaluators; analytics about threat detection and reporting times.
User Feedback	Qualitative	Effectiveness & Comprehen-siveness	User evaluation of training program's content, delivery methods, accessibility, usability; Improvement suggestions	Surveys; Questionnaires; Interviews.
Compliance to User Needs and Roles	Quantitative.	Comprehensiveness	Results of maturity models scoring; Internal evaluation (User feedback & user performance evaluation methods);	Standard certification evaluation; Company or National standard/guidelines/ best practices compliance;
Compliance to Companies' Requirements	Quantitative	Comprehensiveness	Results of maturity models scoring; User Performance evaluation methods; Standard certification evaluation;	Maturity Models; Company or National standard/guidelines/ best practices compliance;

ports Model, and Learners Discuss Data (Pane et al., 2015). According to a survey conducted on the students and teachers of the academic institution that participated in the study conducted by Pane et al. (2015), teachers' greater use of practices that support competency-based learning and greater use of technology for personalization in the schools in this study with implementation data was noted.

To successfully integrate any of the PLT models identified in CS training, consideration on overall requirements for training, resources available as well as input from learners should be taken into consideration and be utilized to develop each training component accordingly.

4.7. CS Training framework model

After completing the second round of questionnaire of Delphi, we summarized the results of the discussion and classified the collected feedback into categories associated with various components of CS training and training development. Table 6 summarizes the recommendations and decisions taken by the panel.

A conceptual map of the training framework developed is presented in Fig. 5

As it can be seen from Fig. 5, when developing a CS training framework, 4 main aspects or components are highlighted as requiring prioritization: (1) training development model and learning model selection, (2) training content, (3) training delivery methods and finally (4) assessment and evaluation.

For each of these components, it is recommended to implement the recommendations found in the previous Sections 4.1–4.5, respectively. Additionally, it can be noted that it is also recommended to involve all training participants during the decision and development process on each of these aspects, as well as any of the training components. Continuous feedback is also recommended be collected during the developmental phases, to allow developing personalized profiles or learning paths for participants, following to PLT recommendations. While it is expected that following such process will require an initial high resource overhead as well as a longer set-up period, this should also facilitate and shorten future update and modification requirements, by being preemptively validated by training participants.

To further detail on the steps to be taken during development, we then analyzed possible integrations of the ADDIE model to the considerations raised in Section 4.6. For this, we aligned the objectives defined for each of the phase of the ADDIE model, shown in Section 4.1, to the considerations of PLT.

- **Analysis Phase:** During the analysis phase, when establishing training needs and goals, it is recommended to involve the selected target audience in the process, by both analyzing their preferences in training delivery, but also on whether the goals of training align with their current goals. Aside from problem and goal definition, all other activities to conduct during analysis phase include the previously mentioned desired outcome establishment, pre-requirement definition, selection of possible learning environment and establishing overall duration of training. All these decisions should take in consideration any possible resource constraint that may be present at the organization that is planning on incorporating the training, both in financial and material terms as well as human resources needed. In this phase as well as in other phases, revision should occur based on progressive feedback given by both training designers and participants on each established decision, until a majority agreement is reached on all attributes.
- **Design Phase:** When designing both the overall training program and each of its single modules, it is critical to take into consideration the individual-specific factors mentioned in Section 4.6. This means that learning material should be developed based on the learning style of participants and their preferences (audio, visual or other type of material). Aside from the material, it is important that an overall structure of how the training will be conducted and what content will be utilized during each module is decided by this point, together with more detailed lesson planning. It is important that the design phase is systematic and specific (Instructional, 2021), with systematic meaning logical, orderly method of identification, development and evaluation of a set of planned strate-

Table 6
Recommendations for the development of a CS training framework given by the panel of experts during the Delphi process.

Component	Recommendation	Description
Development methodology	Revised ADDIE model	A revised version of the ADDIE model, where evaluation and revision of any component developed during all phases is conducted has been suggested as the preferred methodology for CS training development. The methodology should be further aligned with PLT models to consider individual-specific factors which can influence the outcome and effectiveness of training
Training design & Desirable attributes	Suitability, Real-life experience, Scalability, Accessibility, Frequency of Training, Cost Efficiency, Consideration for the Human Factor.	The highlighted attributes have been selected as in need of prioritization during the development of CS training. While all of the attributes should be considered in. A more detailed description of each attribute is given in Table 2.
Training content	Technical skills, soft skills, implementation skills, managerial skills	In Chowdhury and Gkioulos (2021b), we summarized the key skills and competencies needed by CS workers, according to the literature. Additionally, in Chowdhury et al. (2021), we summarized the most common topics of CS training in Norwegian Critical Infrastructure companies. Exact training content should still be defined based on goals and requirements established during the analysis phase of the ADDIE model.
Evaluation	KPIs & Feedback collection	Out of all observed evaluation criteria and methods utilized in the literature, a combination of training specific KPIs and feedback collection has been recommended. The two methods are selected as they provide complementary data for evaluation, with the former providing quantitative data and parameters and the latter providing qualitative evaluation from training participants

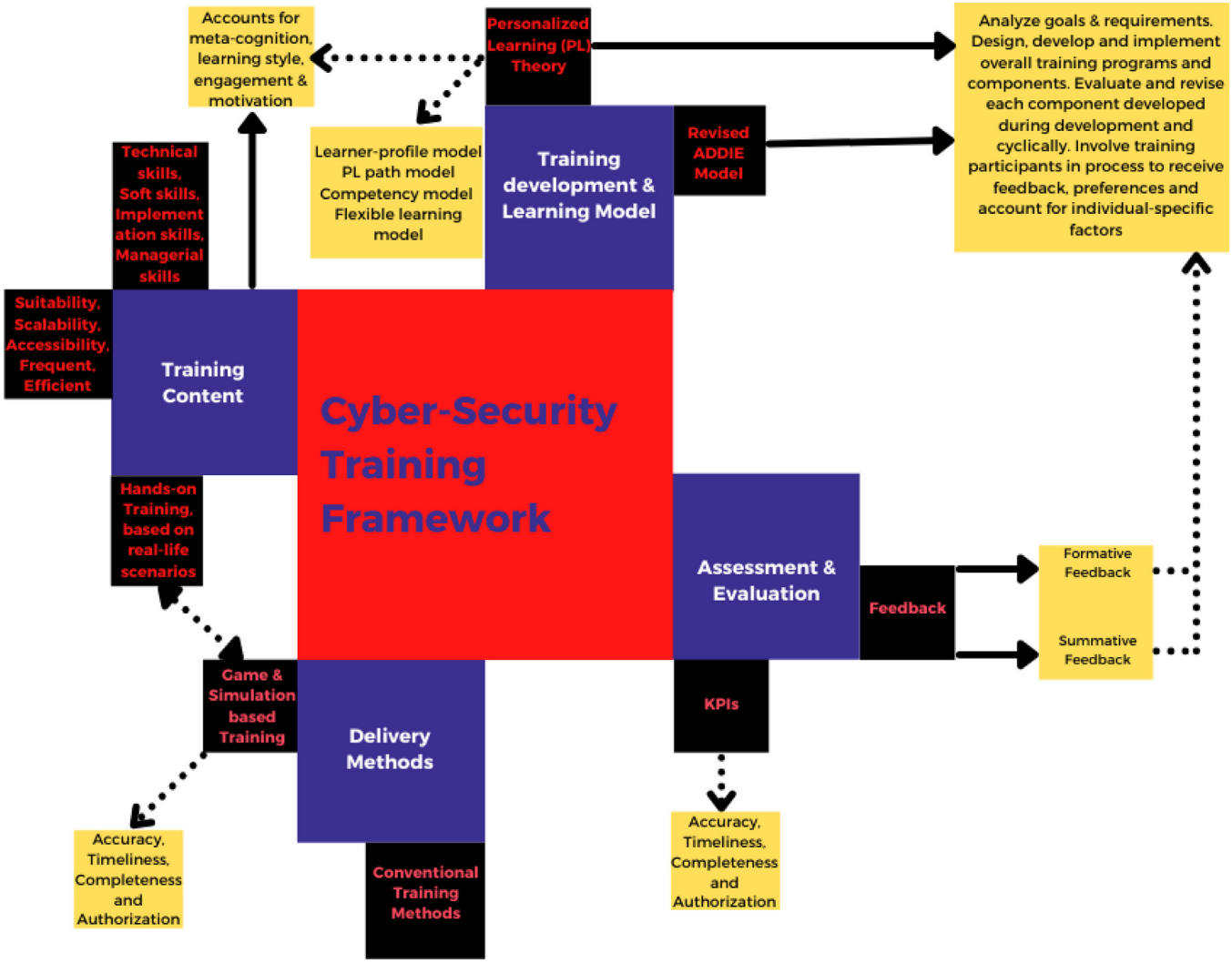

Fig. 5. Conceptual map of the proposed CS training framework.

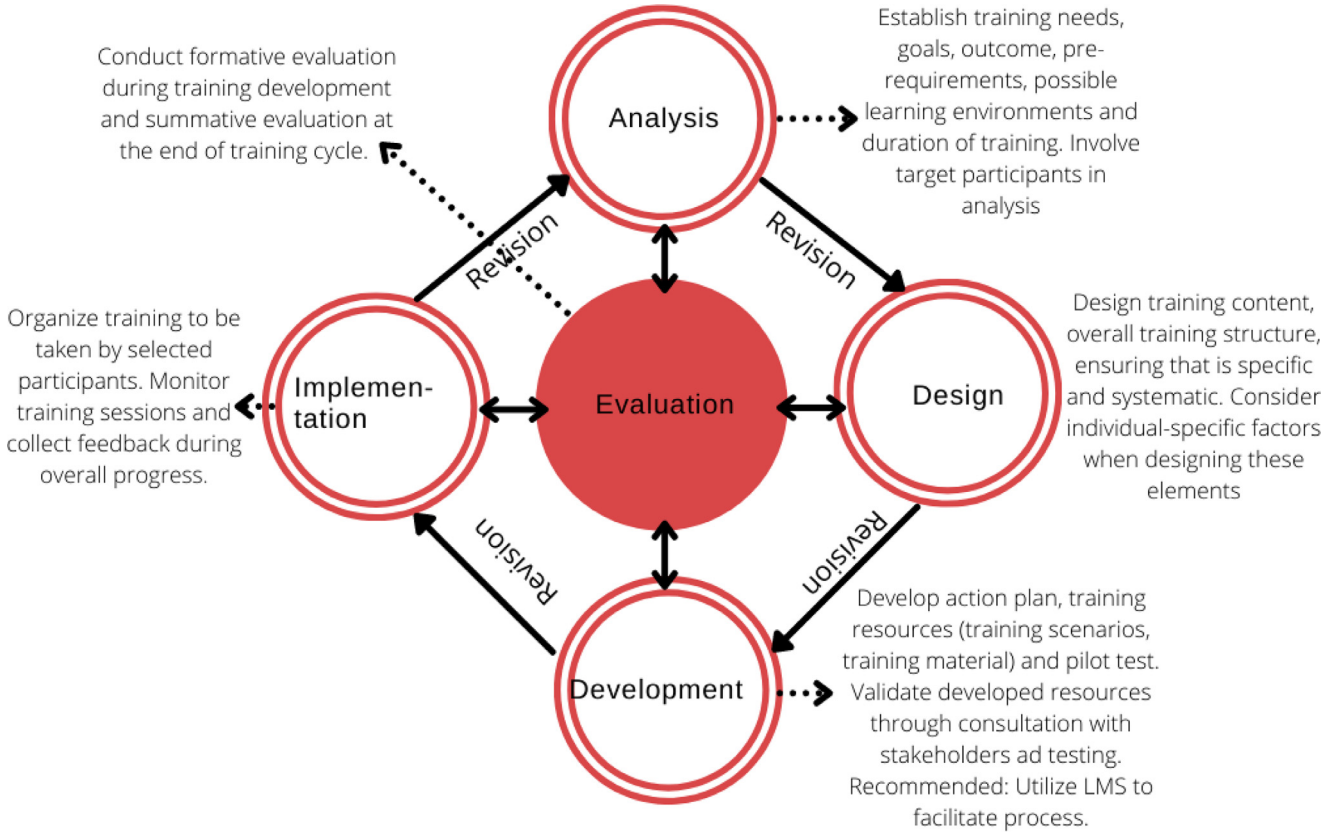

Fig. 6. revised ADDIE model, based on rapid prototyping.

gies targeted for attaining the project's goals, while specific meaning attention to detail for each element of the instructional design plan. The following steps are recommended in Instructional (2021) to ensure systematic and specific design: (1) Document project's instructional, visual and technical design strategy; (2) Apply instructional strategies according to the intended behavioral outcomes by domain (cognitive, affective, psychomotor); (3) Create storyboards; (4) Design the user interface and user experience; (5)Prototype creation; (6) Apply visual design (graphic design);

Many of the activities indicated will be concretized in the following development and implementation phase.

- **Development Phase:** As mentioned in Section 4.1, during this phase, an action plan, training resources and a pilot test will all need to be developed. Another key component to develop during this phase include the training scenarios and other hands-on activities that will be integrated in the training program, which will utilize the tools and learning environment selected in the previous phases. To validate the developed components, consultation with all of the involved stakeholders as well as the pilot test should be used. Validation may be done by internal reviewing, test cases or both. It is recommended to integrate a learning management system (LMS) to the overall training framework during development to facilitate with data collection and reporting during implementation and evaluation phases. Modern LMSs can also facilitate with overall training development, by automating training content generation and facilitating content management.
- **Implementation Phase:** Once the training program and all of its component are completed, they will have to be taken by the participants. Once the participants are engaged in training, it is important to continue monitoring the overall progress, both from an external point of view as well as internally, by collecting formative feedback, to ensure that if any issue arises it can be resolved immediately.
- **Evaluation Phase:** As stated for the previous phases, evaluation should be conducted formatively during design, development and implementation, with summative assessment to be conducted at the conclusion of the first implementation cycle and at any successive iteration. Elements of training to evaluate include quality of the training resources, user engagement and satisfaction, and overall training results. LMS facilitate evaluation by both allowing to collect feedback on all the training components and material, as well as providing logs with information regarding assessment of participants of the training. It is important to involve any instructor that participated to the training sessions also in the evaluation process, and possibly allow training participants to give feedback on the instructor's performance.

Fig. 6 shows the final revised ADDIE model, with a summative description of the activities to be conducted during each phase.

As previously mentioned, the model shown in Fig. 6 presents two main novelties over traditional versions of ADDIE: (1) incorporation of rapid prototyping concepts to allow for continuous revision of each component developed during each phase of the ADDIE. This is later followed by summative evaluation of the final training framework, by means of pilot test assessment; (2) Involvement of training participants, instructors and other relevant stakeholders during the development of the training and its components. Stakeholders' feedback may be collected through individual meetings, group discussions, surveys, or internal advisory groups (Schupmann et al., 2018), and later utilized to develop components after reaching a majority agreement.

Additionally, by collecting training participants' feedback and evaluation logs, it will be possible to develop personalized learning paths and/or learning profiles specific to each individual or group of trainees, based on the key personalized learning theory concepts shown in Section 4.6. These should be then later used for designing future exercises and training material, for overall training progression.

5. Conclusions & future work

The increase of cyber-attack that exploit lack of readiness and human preparedness in recent years has motivated both academic institutions and private institute to increasingly focus on researching and implementing CS training programs, for the purpose of educating and training staff against common attack vectors and prepare them in emergency scenarios.

Nonetheless, many of the offerings are subject of criticism due to their ineffectiveness in engaging training participants, lack of hands-on activities and inability to change user behaviour (Aldawood and Skinner, 2019b; Bada et al., 2019; Chowdhury and Gkioulos, 2021a).

In order to create a more effective CS training framework, in this work we conducted a Delphi method-based study focused on CS training framework modeling. Ten participants were selected as the panel of experts during the Delphi, based on a stakeholder analysis for CS training programs. The stakeholders selected for the Delphi had extensive experience as researchers or professors in academic institutions in the field of CS or covered senior position in the industry in CS roles. Discussion during the Delphi process focused on CS training development methodology, CS training component and desirable attributes, CS training evaluation and finally individual-specific factors that can influence the result of the training. Involving a diverse group of stakeholders allowed to reach an agreement on these topics, based on overall prioritization of different elements of training and by evaluating various techniques and models found in the literature. After reaching an agreement on the selected topics, the information collected was utilized to develop a model CS training framework that followed the recommendations found in the literature and the recommendations given by the experts. Particular focus on considerations regarding learning theory and individual specific-factors of learning such as metacognition, motivation and learning styles was given during discussion with the panel and subsequently during the development of the final training framework. The final decision came to combining a revised version of the ADDIE model, specifically designed for CS training development, with considerations for PLT. According to PLT, by focusing on each participants objectives and preferences, by developing individual learner profiles and personalized learning paths, the overall learning process would be improved and made more effective. Such tailoring would be also made possible by ensuring the developed training solution includes desired attributes discussed in this work, which include suitability, scalability and extandability among others. The final framework proposed in this work offers a general model for developing CS training programs. The main novelties of it are the central considerations for the individual-specific factors of learning previously discussed, by incorporating key and modern PLT concepts, which were not accounted for in previous proposals. This is done by both including training participants as well as other stakeholders during training development, as well as by keeping logs of both feedback and assessment of each trainee, to create individual learning profiles and personalized learning paths as training progresses. Additionally, by utilizing the Delphi process both for the development and validation of the framework, we were able to involve and obtain feedback of a panel of experts stakeholders with heterogeneous CS backgrounds, understand which aspects each group saw as in need of prioritization and finally come to an overall agreement.

Although the CS training framework developed in this work has received initial, albeit informal, theoretical validation, through consultation and discussion conducted during the Delphi process, practical validation is necessary to ensure that the framework is sound. Additionally, the Delphi process itself can present certain limitations: lack of exact methodological outlines and possibility of disregarding the ideas from minority participants being two of the main possible shortcomings. While these limitations were considered and partially resolved by initially consulting with the panel on the exact methodology for conducting the study and by allowing for open discussion after each round of Delphi, additional validation in the form of case studies is suggested for further validation. Experimentation involving both students in information security higher education and industrial CS personnel is currently planned, and would occur by providing different combinations of training approaches, based on participants' preferences, and evaluating the effectiveness of each approach. As research on PLT is still novel, continuous feedback from training participants and monitoring need to be conducted on all different approaches utilized to establish both preferences and better performing solutions.

Acronyms

- SME: Small & Medium Enterprise;
- CI: Critical Infrastructure;
- CS: Cyber-Security;
- KSA: Knowledge, Skills and Ability;
- ISO/IEC: International Organization for Standardization/International Electrotechnical Commission;
- NIST: National Institute of Standards and Technology;
- NICE: National Initiative for Cybersecurity Education;
- ENISA: European Union Agency for Cybersecurity.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Nabin Chowdhury: Conceptualization, Funding acquisition, Formal analysis, Writing – original draft, Writing – review & editing. **Sokratis Katsikas:** Formal analysis, Writing – review & editing. **Vasileios Gkioulos:** Conceptualization, Writing – review & editing.

References

Abawajy, J., 2014. User preference of cyber security awareness delivery methods. Behaviour & Information Technology 33 (3), 237–248.
Abawajy, J., 2014. User preference of cyber security awareness delivery methods. Behaviour & Information Technology 33 (3), 237–248. doi:10.1080/0144929X.2012.708787.
Adams, M., 2019. How to measure the effectiveness of cybersecurity training and awareness programs. Fissea.
Adams, R., 2018. Our approach to employee security training. Pager Duty.
Alajmi, M., 2009. E-learning and addie model. In: E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education. Association for the Advancement of Computing in Education (AACE), pp. 37–42.
Aldawood, H., Skinner, G., 2018. Educating and raising awareness on cyber security social engineering: A literature review. In: 2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE). IEEE, pp. 62–68.
Aldawood, H., Skinner, G., 2019. An academic review of current industrial and commercial cyber security social engineering solutions. In: Proceedings of the 3rd International Conference on Cryptography, Security and Privacy, pp. 110–115.
Aldawood, H., Skinner, G., 2019. Reviewing cyber security social engineering training and awareness programs-pitfalls and ongoing issues. Future Internet 11 (3), 73.
Allen, W.C., 2006. Overview and evolution of the addie training system. Adv Dev Hum Resour 8 (4), 430–441.

Andriotis, N., 2018. 5 Elements to include in any post training evaluation questionnaire. Efront Learning.
Antonioli, D., Ghaeini, H.R., Adepu, S., Ochoa, M., Tippenhauer, N.O., 2017. Gamifying ics security training and research: Design, implementation, and results of s3. In: Proceedings of the 2017 Workshop on Cyber-Physical Systems Security and PrivaCy, pp. 93–102.
Bada, M., Sasse, A.M., Nurse, J.R., 2019. Cyber security awareness campaigns: why do they fail to change behaviour? arXiv preprint arXiv:1901.02672.
Beuran, R., Pham, C., Tang, D., Chinen, K.-i., Tan, Y., Shinoda, Y., 2017. Cytrone: an integrated cybersecurity training framework. JAIST Repositor.
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.
Boerman, D., 2020. Reporting on cybersecurity performance. University of Twente B.S. thesis.
Brilingaitė, A., Bukauskas, L., Juozapavičius, A., 2020. A framework for competence development and assessment in hybrid cybersecurity exercises. Computers & Security 88, 101607.
Chowdhury, N., Espen, N., Kine, R., Gkioulos, V., 2021. A study of cybersecurity training offerings in norwegian critical infrastructure companies. International Journal of Safety and Security Engineering.
Chowdhury, N., Gkioulos, V., 2021. Cyber security training for critical infrastructure protection: a literature review. Computer Science Review 40, 100361.
Chowdhury, N., Gkioulos, V., 2021. Key competencies for critical infrastructure cyber-security: a systematic literature review. Information & Computer Security.
Churches, A., 2010. Bloom's digital taxonomy.
Davids, J., 2020. Magellan health data breach victim tally reaches 365k patients. https://healthitsecurity.com/news/magellan-health-data-breach-victim-tally-reaches-365k-patients.
DeFranzo, S., 2018. 5 reasons why feedback is important. Snap Surveys.
Diana, 2019. 5 key elements of personalized learning. Neo Blog.
Farooq, M., Khan, M.A., et al., 2011. Impact of training and feedback on employee performance. Far east journal of psychology and business 5 (1), 23–33.
Gagne, R.M., Wager, W.W., Golas, K.C., Keller, J.M., Russell, J.D., 2005. Principles of instructional design. Wiley Online Library.
Goh, P., 2021. Humans as the weakest link in maintaining cybersecurity: building cyber resilience in humans. In: Introduction To Cyber Forensic Psychology: Understanding The Mind Of The Cyber Deviant Perpetrators, pp. 287–305.
Gross, A., 2018. Effective security training requires change in employee behavior. Health IT Answers.
Hai-Jew, S., 2019. The electronic hive mind and cybersecurity: mass-scale human cognitive limits to explain the weakest link in cybersecurity. In: Global cyber security labor shortage and international business risk. IGI Global, pp. 206–262.
Hallowell, M.R., Gambatese, J.A., 2010. Qualitative research: application of the delphi method to cem research. J Constr Eng Manag 136 (1), 99–107.
Harris, M.A., et al., 2015. Using bloom's and webb's taxonomies to integrate emerging cybersecurity topics into a computic curriculum. Journal of Information Systems Education 26 (3), 219–234.
He, W., Zhang, Z., 2019. Enterprise cybersecurity training and awareness programs: recommendations for success. Journal of Organizational Computing and Electronic Commerce 29 (4), 249–257.
Hendrix, M., Al-Sherbaz, A., Victoria, B., 2016. Game based cyber security training: are serious games suitable for cyber security training? International Journal of Serious Games 3 (1).
Instructional, D., 2021. Addie model. Instructional Design.
Kostadinov, D., 2018. The components of a successful security awareness program. InforSec Institute.
Kumar, A., Chaudhary, M., Kumar, N., 2015. Social engineering threats and awareness: a survey. European Journal of Advances in Engineering and Technology 2 (11), 15–19.
at Last, S., 2021. 22 shocking ransomware statistics for cybersecurity in 2021. https://safeatlast.co/blog/ransomware-statistics/#gref.
Leach, J., 2003. Improving user security behaviour. Computers & Security 22 (8), 685–692.
Lee, J.-W., Song, J.-G., Lee, C.-K., 2016. Study on nuclear facility cyber security awareness and training programs.
Malmedal, B., Røislien, H. E., 2016. Cybersecurity risk perception.
Miller, A., 2018. Formative and summative feedback. Teaching at Tufts.
Molenda, M., 2003. In search of the elusive addie model. Performance improvement 42 (5), 34–37.
Morin, A., 2020. Personalized learning: what you need to know. Teachhub.
Nadkarni, S., 2012. Security awareness training made easy. Computer Weekly.
Nagarajan, A., Allbeck, J.M., Sood, A., Janssen, T.L., 2012. Exploring game design for cybersecurity training. In: 2012 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER). IEEE, pp. 256–262.
Nixon, E.K., Lee, D., 2001. Rapid prototyping in the instructional design process. Performance Improvement Quarterly 14 (3), 95–116.
Osbourne, C., 2020. Logistics giant toll group hit by ransomware for the second time in three months. https://www.zdnet.com/article/transport-logistics-firm-toll-group-hit-by-ransomware-for-the-second-time-in-three-months/.
Pane, J.F., Steiner, E.D., Baird, M.D., Hamilton, L.S., 2015. Continued progress: promising evidence on personalized learning. Rand Corporation.
Pashler, H., McDaniel, M., Rohrer, D., Bjork, R., 2008. Learning styles: concepts and evidence. Psychological science in the public interest 9 (3), 105–119.
Patriciu, V.-V., Furtuna, A.C., 2009. Guide for designing cyber security exercises. In: Proceedings of the 8th WSEAS International Conference on E-Activities and Information Security and Privacy. World Scientific and Engineering Academy and Society (WSEAS), pp. 172–177.
Pattinson, M., Butavicius, M., Lillie, M., Ciccarello, B., Parsons, K., Calic, D., McCormac, A., 2019. Matching training to individual learning styles improves information security awareness. Information & Computer Security.
Petrenko, S.A., Makoveichuk, K.A., 2017. Big data technologies for cybersecurity. In: CEUR workshop, pp. 107–111.
Pham, C., Tang, D., Chinen, K.-i., Beuran, R., 2016. Cyris: A cyber range instantiation system for facilitating security training. In: Proceedings of the Seventh Symposium on Information and Communication Technology, pp. 251–258.
Pokorny, B., 2017. Cybersecurity training. Big Data Analytics in Cybersecurity 115.
PurpleSec, 2021. 2021 cyber security statistics the ultimate list of stats, data & trends. https://purplesec.us/resources/cyber-security-statistics/.
Rajamäki, J., Nevmerzhitskaya, J., Virág, C., 2018. Cybersecurity education and training in hospitals: Proactive resilience educational framework (prosilience ef). In: 2018 IEEE Global Engineering Education Conference (EDUCON). IEEE, pp. 2042–2046.
Samuel, J., 2019. Cyber security key performance indicators. Infosec Write-ups.
Schupmann, W., Fudge, K., Reynolds, K., Whyman, D., 2018. Engaging stakeholders in learning agenda development. Washington, DC: Evidence-Based Policymaking Collaborative.
Sviridenko, A., 2018. How to effectively collect feedback for your online course. eLearning Industry.
Tang, C., 2017. Key performance indicators for process control system cybersecurity performance analysis. US Department of Commerce, National Institute of Standards and Technology.
Tirumala, S., Valluri, M.R., Babu, G., 2019. A survey on cybersecurity awareness concerns, practices and conceptual measures. In: 2019 International Conference on Computer Communication and Informatics (ICCCI). IEEE, pp. 1–6.
UniTo, 2018. Asking for feedback from students. University of Toronto.
Valentine, J.A., 2006. Enhancing the employee security awareness model. Computer fraud & security 2006 (6), 17–19.
Wagner, J., 2016. Developing a cybersecurity scorecard. Farm Service Agency.
Walkington, C., Bernacki, M. L., 2020. Appraising research on personalized learning: Definitions, theoretical alignment, advancements, and future directions.
Wiederhold, B. K., 2014. The role of psychology in enhancing cybersecurity.
Zhang, Z.J., He, W., Li, W., Abdous, M., 2021. Cybersecurity awareness training programs: a cost–benefit analysis framework. Industrial Management & Data Systems.
Zimmerman, T.A., 2017. Metrics and key performance indicators for robotic cybersecurity performance analysis. US Department of Commerce, National Institute of Standards and Technology.
Zimmermann, V., Renaud, K., 2019. Moving from a human-as-problem to a human-as-solution cybersecurity mindset. Int J Hum Comput Stud 131, 169–187.

Nabin Chowdhury was born on the 2nd December 1992, in BrahmanBaria, Bangladesh. He moved in Italy at the age of 2, and received his full education, including his higher education. He completed his bachelor and master's degree in Computer Engineering at the University of Bologna, located in Bologna, Italy. During his studies, he participated in two exchange programs; the first one was completed at Reykjavik university while the second one at the Norwegian University of Science and Technology (NTNU) in Gjøvik, Norway. During this second exchange, he focused his attention on the field of Information Security, by completing his thesis work focused on penetration testing in IoT devices in Smart Homes. After completing his master's degree, he commenced working as a computer engineer for the consulting company Akka Technologies, at their location in Geneva, Switzerland. He was then relocated in Nice, France for a contractual mission as a software engineer at Amadeus. While completing the mission, he continued working with his previous supervisors and collaborators at NTNU in various project and for the completion of a scientific work, all focused in information security areas. His passion for the field grew larger as time passed, which ultimately made him decide to leave his position as a software engineer to return at NTNU to complete a doctoral program, which he is currently completing. The objective of the doctoral work he is currently completing is to develop an effective cyber-security training framework for critical infrastructure protection.