Raising Awareness of Cloud Security through a Serious Game

Tiange Zhao

Vollständiger Abdruck der von der Fakultät für Informatik der Universität der Bundeswehr München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

angenommenen Dissertation.

Gutachter/Gutachterin:

1: Prof. Dr. Ulrike Lechner
2: Prof. Dr. Maria Pinto-Albuquerque

Die Dissertation wurde am 12. Dez. 2024 bei der Universität der Bundeswehr München eingereicht und durch die Fakultät für Informatik am 25. Apr. 2025 angenommen. Die mündliche Prüfung fand am 14. Mai 2025 statt.

Acknowledgements

This work was done in Siemens in partnership with the Universität der Bunderswehr München (UniBW) and with Instituto Universitário de Lisboa (ISCTE-IUL). This collaboration was crucial to the success of the present work. From an academia perspective, the novel scientific results were presented in several publications with two best-paper awards and two journal papers. From an organizational perspective, the designed useful artifact was applied in training programs in industry. The present work counted additionally with the support of two working students: Ece Ata and Didem Ongu.

The premises for the present work have been given by the following constellations:

- CSC infrastructure developed by my colleague Dr. Tiago Gasiba;
- My position in the company as cyber-security consultant and trainer;
- The recognition and long-term commitment by the company to perform research on innovative training methodologies;
- The funding for project CONTAIN by the Federal Ministry of Education and Research under the project number 13N16585.

First and foremost, I would like to express my sincere gratitude to all the game participants from Siemens and other organizations. Their active participation was instrumental in making this study possible.

I am deeply grateful to all the anonymous reviewers of our papers, whose genuine feedback and constructive comments have significantly enhanced our research.

My heartfelt thanks go to my working students, Didem Ongu and Ace Ata, for their valuable assistance throughout this journey.

I extend special appreciation to my colleagues and friends at Siemens: Fabiola, Iosif, Prabha, Santi, Tiago and everyone in Security Lifecycle Research Group. Particular thanks to my boss Dr. Holger Dreger, whose unwavering support throughout my PhD journey has been invaluable.

To my dear UniBW colleagues, I cherish the wonderful memories we created during summer schools and conferences both in Germany and abroad. I am

especially touched by Judith and Maxi, who crafted the adorable PhD hat that now holds pride of place in my living room as my most treasured possession.

I am grateful to Jorge for his wisdom and guidance. His presence during my Oberseminar and Defense provided me with much-needed confidence.

I owe an immense debt of gratitude to my professors, Ulrike and Maria. Through countless discussions and mentoring sessions, you have been my guiding stars, illuminating the path forward. After each meeting, the direction of my research became clearer, building my confidence and substantially improving the quality of my work.

Finally, I want to express my deepest gratitude to my beloved husband, Mr. Gao You Lin, who was the first person I wanted to share this achievement with after my defense. His love and support made everything possible. I am also thankful to my mother, Ms. Yang Jin Ying; father, Mr. Zhao Guang Yi; my 96-year-old grandmother, Ms. Jin Guang Mei; and my parents-in-law, Ms. Lin Chun Ping and Mr. Gao Cong Jia. Their love has been my source of strength throughout this journey. A special mention goes to our loyal companion, family dog Kirin, whose company during morning and evening walks often sparked moments of inspiration.

I dedicate this work to the newest member of our family, my son. You have been the perfect companion throughout this journey, resting peacefully while offering encouraging kicks from time to time. Though you may not realize it yet, we accomplished the defense together. Since then, we have been an incredible team. We eagerly await your arrival.

Abstract

Cloud deployment is widely used in the industry due to its irreplaceable convenience and flexibility. However, it is vital to understand the security risks associated with cloud deployment and the shared responsibility model that determines the responsibilities in cloud deployment. In this work, we present Cloud of Assets and Threats (CATS), a serious game designed for industry practitioners to raise awareness about cloud security and the mitigation strategies in the shared responsibility model.

The research follows the design science research paradigm. The game idea of CATS is to build a cloud defense strategy by assigning different defensive cards to various roles. An evaluator algorithm determines the success rate of a defense strategy. CATS is designed and implemented in three design iterations and evaluated through three game trial runs and twelve game events with more than 150 participants from the industry. The first design iteration focuses on the design and validity of game logic. In the second design iteration, a digital platform is implemented. The third design iteration refines the game elements, notably the evaluator algorithm. The evaluation process illustrates that CATS fosters the knowledge of cybersecurity-relevant aspects of cloud deployment and the shared responsibility model. A road map towards implementing CATS in an organization is designed and can be used to implement the results of this thesis in practice.

The research was conducted in the Security Lifecycle research group in the Technology department of Siemens AG from 2021 to 2024, in collaboration with the Universität der Bundeswehr München and the ISCTE - Instituto Universitário de Lisboa.

Abstrakt

Cloud Deployment ist für die Industrie aufgrund von Nützlichkeit und Flexibilität praktisch unersetzlich. Es ist jedoch wichtig, auch die mit Cloud assoziierten Sicherheitsrisiken zu kennen und wie das Shared-Responsibility Modell Verantwortlichkeiten für Cloud Deployment regelt. In dieser Arbeit stellen wir das Serious Game Cloud of Assets and Threats (CATS) vor. CATS wurde für die Praxis in der Industrie entwickelt. Das Serious Game CATS soll das Bewusstsein für Cloud-Sicherheit und die Risikomitigierungsstrategien im Shared-Responsibilty Modell schärfen.

Die Forschung folgt dem Design Science Paragdima. Die Spielidee von CATS besteht darin, eine Cloud-Verteidigungsstrategie zu entwickeln, indem verschiedenen Rollen unterschiedliche Verteidigungskarten zugewiesen werden. Ein Auswertealgorithmus bestimmt die Erfolgswahrscheinlichkeit einer Verteidigungsstrategie. CATS wird in drei Design-Iterationen entworfen und implementiert und durch drei Spiel-Probeläufe und zwölf Spiel-Events mit mehr als 150 Teilnehmerinnen und Teilnehmern aus der Industrie evaluiert. Die erste Design-Iteration konzentriert sich auf das Design und die Gültigkeit der Spiellogik. In der zweiten Design-Iteration wird eine digitale Plattform implementiert. Die dritte Design-Iteration verfeinert die Spielelemente, insbesondere den Auswertealgorithmus. Der Evaluierungsprozess zeigt, dass CATS das Wissen um die für IT-Sicherheit relevanten Aspekte des Cloud-Deployments und des Shared-Responsibility Modells fördert. Es wird eine Roadmap für die Implementierung von CATS in einer Organisation vorgeschlagen. Diese Roadmap kann zur Nutzung der Ergebnisse dieser Arbeit in der Praxis verwendet werden.

Für diese Arbeit wurde von 2021 bis 2024 in der Security Lifecycle Forschungsgruppe der Technologieabteilung der Siemens AG und in Zusammenarbeit mit der Universität der Bundeswehr München und dem ISCTE - Instituto Universitário de Lisboa geforscht.

Contents

List of Figures

List of Tables

Acronyms

AC	Attack Complexity
AD	Active Directory
ADR	Action Design Research
AICPA	American Institute of Certified Public Accountants
AWS	Amazon Web Service
API	Application Programming Interface
AS	Attack Scenario
AV	Attack Vector
BMBF	The Federal Ministry of Education and Research
BSI	Federal Office for Information Security
C5	Cloud Computing Compliance Criteria Catalogue
CATS	Cloud of Assets and Threats
CCM	Cloud Control Matrix
CIP	Critical Infrastructure Protection
CIS	Center for Internet Security
COBIT	Control Objectives for Information Technologies
COPYCAT	CONTAIN SupPlY Chain ATtack
COVID-19	Coronavirus disease 2019
CSA	Cloud Security Alliance
CSC	CyberSecurity Challenges
CSP	Cloud Service Provider
CVE	Common Vulnerabilities and Exposures
CVSS	Common Vulnerability Scoring System
DK	Design Knowledge
DoS	Denial of Service
DSR	Design Science Research
DSS	Data Security Standard
EC2	Amazon Elastic Compute Cloud
FedRAMP	Federal Risk and Authorization Management Program
IaaS	Infrastructure as a Service
IaC	Infrastructure as Code
IDS	Intrusion Detection System
IEC	International Electrotechnical Commission

IS	Information System
ISA	Information Security Awareness
ISACA	Information Systems Audit and Control Association
ISDT	Information System Design Theory
ISO	International Organization for Standardization
IT	Information Technology
KAB	Knowledge Attitude and Behavior
LM-GM	Learning Mechanics–Game Mechanics
MFA	Multi-factor Authentication
NERC	North American Electric Reliability Corporation
NIST	National Institute for Standards and Technology
NSA	National Security Agency
NPC	Non-player Character
PaaS	Platform as a Service
PCI	Security Standards Council
PII	Personally Identifiable Information
PR	Privileges Required
SaaS	Software as a Service
SID	Security Identifier
SP	Specal Publication
SSI	Semi-Structured Interview
SSR	Single Success Rate
S3	Simple Storage Service
TR	Trial Run
TSC	Trust Services Criteria
UI	User Interaction

1

Introduction

In cloud computing, the cloud service provider and the cloud service customer share the responsibility of securing the cloud assets. The responsibility separation is addressed by the topic of the shared responsibility model (Sisodia and Khan, 2022), and raising awareness of cloud security roles and responsibilities is of great importance. Companies move their products and solutions from on-premises to cloud deployment. The cloud service providers publish their shared responsibility model as a basis for contracting with cloud service customers. The context of the shared responsibility model in cloud security is defined in various standards and white papers. Despite the differences between cloud service providers and the service provisioning model, it is always the responsibility of the cloud service customer to protect their information and data, and the cloud service provider takes care of the physical layer. Both are necessary to have a secure cloud service. The shared responsibility model describes the separation of tasks of both parties based on the service provisioning model. Sometimes, the cloud service customers take more responsibility, and sometimes, the cloud service provider.

Standards and white papers aim to regulate the industry field and provide the basis for a shared understanding of roles and responsibilities in securing cloud assets among all stakeholders involved. Several stakeholders, including cloud asset managers, cloud asset owners, and cloud service providers, are involved in the design and development life cycle of cloud services (Sisodia and Khan, 2022). Each stakeholder must understand their role and responsibility to secure a cloud asset. Everyone has a part to play. Cloud service providers can only do so much, and it is the responsibility of the data owners to secure the

parts that the service providers cannot reach, such as establishing a protection concept for corresponding data policy. The constraints of one party have led to the further development of the shared responsibility model (Walsham, 2024). Note that a phenomenon described in (Calder, 2022) shows that "shared responsibility" without an agreed understanding of who does what will lead to "no responsibility." It is a reminder to take the topic of shared responsibility seriously.

On the one hand, the convenience and flexibility of cloud services boost the efficiency of the software and system development cycle in the industry. On the other hand, cloud deployment can expose industry systems to novel cybersecurity threats (Cybersecurity Advisory, 2020). The common cyber security vulnerabilities in on-premises systems are critical as cloud deployment increases system exposure, and cloud-specific threats, e.g., breaches of cloud storage objects, need to be addressed.

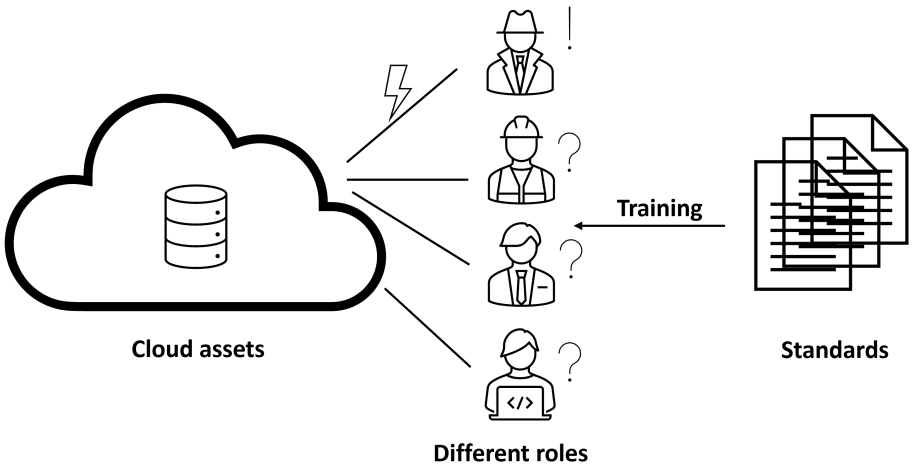


Figure 1.1: The complexity of cloud security

Figure 1.1 depicts the complexity of standards and the mapping between the roles and responsibilities in cloud security from a high level. Within an organization, some practitioners take the role of development and configuration, while others make business decisions. The separation of different roles and responsibilities needs to be communicated. Security training is essential to convey concepts and strategies to decision-makers, developers, system architects, cloud service customers, and providers.

Various studies emphasize the need for innovative security awareness training methods (Harford, 2022). During my career as a cyber security consultant and trainer in the industry, experience has shown me that developers and managers need to raise awareness about cloud security and, more importantly, realize who should do what to protect cloud assets. In cyber security training, serious games serve as a means to enhance awareness and facilitate experiential

learning within a secure and playful environment. This research contributes to enriching and improving the training methods in cloud security through a serious game. We design a serious game to improve cloud security by increasing cloud security awareness with a special focus on shared roles and responsibilities in securing cloud assets.

Since I started my career as a cyber security consultant in the industry, our research group has been successful in designing a serious game, which is now known as CyberSecurity Challenges (CSC) (Gasiba, 2021), and applying it in training in industry to help practitioners raise awareness on cyber security and exercising their secure coding skills. My special research interests are cloud and cloud deployment since cloud provisioning has become a mature business with increasing connectivity and remarkable efficiency. The market for cloud services is growing rapidly. Applying cloud service reduces the maintenance workload of the operator in the industry. I decided to focus on cloud security training for practitioners in the industry. Cloud security is a complex topic that includes technical skills to protect cloud assets and collaboration with decision-makers to work out a comprehensive protection strategy with different roles and responsibilities working hand-in-hand. Inspired by the success of CSC, I started designing a serious game dedicated to raising awareness about cloud security and addressing its complicated nature.

The practitioners should understand the shared responsibility model and what concrete tasks should be completed to improve cloud security status despite their technical backgrounds. Therefore I decide to design an interactive game, Cloud of Assets and Threats (CATS), for players from different levels of technical knowledge. Following the footsteps of CSC, CATS, as a digital tabletop game, can be deployed online and easily integrated into training events, providing a valuable hands-on experience for participants. CATS is designed and implemented under the design science paradigm proposed by Hevner et al. (Hevner, 2007). The design and implementation went through three design cycles for evaluation. The evaluation was conducted in game events among industrial practitioners.

1.1 Cloud and cloud security incidents

Cloud-based applications are common in the industry, as the cloud service provider enables developers to build and deploy their applications efficiently. The National Institute for Standards and Technology (NIST) summarizes five characteristics of cloud computing (Mell and Grance, 2011):

- On-demand self-service;
- Broad network access;
- Resource pooling;
- Rapid elasticity;

- Measured service.

These characteristics contribute to flexibility and convenience, improving development efficiency and business success. However, cloud assets are prone to various cyber security threats (Nafea and Almaiah, 2021). Due to the broad network exposure and architecture that involves cloud service providers and customers, the attack surface increases compared to on-premise and other service provisioning models. Also, there are security challenges specific to the cloud. The Cloud Security Alliance (CSA) provides a ranking table of the top 11 threats in cloud computing (Cloud Security Alliance, 2019): data breaches; misconfiguration and inadequate change control; lack of cloud security architecture and strategy; insufficient identity, credential access and key management; account hijacking; insider threat; insecure interfaces and APIs; weak control plan; metastructure and applistructure failures; limited cloud usage visibility; abuse and nefarious use of cloud services.

Cloud security is a joint effort involving the cloud service customer and the cloud service provider. The shared responsibility model describes, on a high level, the responsibilities of cloud service customers and cloud service providers, depending on different service provisioning models. The details need to be communicated to the cloud service customers and negotiated in the contract between the cloud service customer and the cloud service provider. According to the shared responsibility model of all the international cloud service providers (Sisodia and Khan, 2022), the cloud service customers always need to configure the cloud assets securely and take full responsibility for the customer's data and information security (Cloud security guidance, 2023).

Cloud service providers endeavor to improve cloud security by offering numerous services. Despite these various security-relevant services, security incidents are reported to target top-rated companies, critical infrastructure, and governments. Raising awareness about cloud security and establishing a common understanding of who should do what to protect cloud assets are necessary. Due to the complexity of cloud deployments, a better understanding of each stakeholder's responsibilities regarding the security of company assets is essential.

1.2 Data and facts

The global cloud computing market was valued at USD 483.98 billion in 2022 and is expected to expand at a compound annual growth rate (CAGR) of 14.1% from 2023 to 2030 (Statista, 2022). Nearly half (45%) of all security incidents target cloud-based services. Moreover, 80% of business organizations experienced at least one cloud security breach incident last year (Raza, 2023).

Amazon Web Service (AWS) is one of the biggest cloud service providers. It offers over 200 cloud services, 22 of which are security-relevant services (Amazon Web Services, 2024a).

For instance, the AWS S3 (Simple Storage Service) bucket is an object storage that AWS provides to host data for web applications, industry applications, etc., to retrieve data. In Australia, a misconfigured Amazon S3 bucket has accidentally compromised 48,270 personally identifiable information (PII) from Australian employees working in government agencies, banks, and a utility company. The leaked PIIs include full names, passwords, IDs, phone numbers, email addresses, and credit card numbers. Salary and expense details were also exposed (Trendmicro, 2017). Another example of S3 bucket leakage is from the US Army and the National Security Agency (NSA) (UpGuard Team, 2017). Despite this, over 50,000 patient records were stored on two publicly accessible AWS S3 Buckets for Utah-based COVID-19 testing service. The cause of the breach was also a damaging security misconfiguration (Sebayan, 2021). Such security breaches can be avoided if the practitioners are equipped with basic awareness of the cloud security issues (Amazon Web Services, 2024b). Microsoft has also reported a similar data leakage. The Microsoft AI research division accidentally leaked 38TB of private data via unsecured Azure storage (Gatlan, 2023).

1.3 Background and motivation

Numerous industry security standards propose requirements and best practices in cloud security. The controls in Cloud Control Matrix (Cloud Security Alliance, 2021) (CCM) from CSA are mapped against industry-accepted security standards, regulations, and control frameworks including but not limited to: ISO 27001/27002/27017/27018 (ISO27001, 2017; ISO27002, 2013; ISO27017, 2015; ISO27018, 2019), NIST SP 800-53 (NIST, 2020), AICPA TSC (AICPA TSC, 2017), German BSI C5 (Bundesamt für Sicherheit in der Informationstechnik, 2020), PCI DSS(PCI DSS, 2022), ISACA COBIT(ISACA, 2019), NERC CIP(North American Electric Reliability Corporation, 2020), FedRAMP (General Services Administration of United States, 2019), CIS (Center for Internet Security, 2020) and many others.

There are three types of cloud computing service models in cloud deployment: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). In all three service models, it is the responsibility of the cloud service customers to configure the service securely. The shared responsibility model describes the responsibility of cloud service providers and customers for a cloud service based on different service models.

Cloud security is a complex yet important topic that industry practitioners need to understand. The study of Gleeson (Gleeson and Walden, 2014) has shown the complexity of cloud security standards. In the work of Iosif et al. (Iosif et al., 2022), we examined the quality of infrastructure as code (IaC) in open code repositories regarding security. We found almost 300,000 security violations from over 8000 code repositories. Our study concludes that developers miss basic concepts of cloud security and that improving awareness

of certain issues related to cloud deployment in an industrial environment is necessary. Those fundamental concepts are generally conveyed to the developers through training. Traditional training is typically lecture-based in a face-to-face or virtual format. Cyber security serious games are designed to enrich or provide an alternative to traditional training. Yet, none of the existing games focus on the uniqueness of cloud security issues and solutions. Therefore, we decide to design a serious game, Cloud of Assets and Threats (CATS), to help raise awareness of cloud security in the industry.

CATS can be deployed as a standalone game in a training event or as a part of the CyberSecurity Challenge (CSC), developed by Dr. Tiago Gasiba (Gasiba, 2021). It is integrated into the company's training curriculum. During our research, CSC is used as a platform to deliver the CATS game. In most game events, CSC and the CATS were delivered together for in-house training, which levitates the burden of organizing and focusing on the research necessary to design and complete the game elements. Game logic and core elements of CATS have a unique design and implementation.

In this thesis, we present the design of the game and evaluate the impact of the game on industrial practitioners through open discussion, semi-structured interviews, and surveys.

1.4 Collaborations and publications

In the research of CATS, I have collaborated and published several papers with Prof. Dr. Ulrike Lechner from the Universität der Bundeswehr München and Prof. Dr. Maria Pinto-Albuquerque from the ISCTE Instituto Universitário de Lisboa. I also published with Dr. Tiago Gasiba, a colleague at Siemens AG and the creator of Cybersecurity Challenges. The present work counted additionally with the support of two working students: Ece Ata and Didem Ongu.

The following articles (Zhao, Gasiba, et al., 2024; Zhao, Lechner, et al., 2024; T. Zhao, Gasiba, et al., 2021a,b, 2024; T. Zhao, Lechner, Pinto-Albuquerque, and Ata, 2022; T. Zhao, Lechner, Pinto-Albuquerque, Ata, and Gasiba, 2023; T. Zhao, Lechner, Pinto-Albuquerque, and Ongu, 2023; T. Zhao, Lechner, Pinto-Albuquerque, Ongu, and Gasiba, 2024), sorted by publication year, have been published in different journals and conferences and represent the preliminary version of the content in this thesis:

- Tiange Zhao, Tiago Gasiba, Ulrike Lechner, Maria Pinto-Albuquerque. "Exploring a Board Game to Improve Cloud Security Training in Industry". In: Second International Computer Programming Education Conference (ICPEC) (May. 2021). Ed. by Henriques, Pedro Rangel and Portela, Filipe and Queirós, Ricardo and Simões, Alberto. Open Access Series in Informatics (OASIcs) Schloss Dagstuhl – Leibniz-Zentrum für Informatik: Dagstuhl, Germany, p. 11:1-11:8.
  DOI: 10.4230/OASIcs.ICPEC.2021.11.

- Tiange Zhao, Tiago Gasiba, Ulrike Lechner, Maria Pinto-Albuquerque. "Raising Awareness about Cloud Security in Industry through a Board Game." In: Information, Special Issue Future Trends in Computer Programming Education , Vol. 12, No. 11. Ed. by Ricardo Queirós, Mário Pinto, Carlos Filipe Portela, Alberto Simões and Pedro Rangel Henriques. Information 2021, 12, 482.
  DOI: https://doi.org/10.3390/info12110482.

- Tiange Zhao, Ulrike Lechner, Maria Pinto-Albuquerque and Ece Ata. "Cloud of Assets and Threats: A Playful Method to Raise Awareness for Cloud Security in Industry". In: Third International Computer Programming Education Conference (ICPEC) (Jun. 2022). Ed. by Simões, Alberto and Silva, João Carlos. Open Access Series in Informatics (OASIcs) Schloss Dagstuhl – Leibniz-Zentrum für Informatik: Dagstuhl, Germany, p. 6:1-6:13.
  DOI: 10.4230/OASIcs.ICPEC.2022.6.

- Tiange Zhao, Ulrike Lechner, Maria Pinto-Albuquerque, Ece Ata, and Tiago Gasiba. "CATS: A Serious Game in Industry Towards Stronger Cloud Security." In: Ubiquitous Security (UbiSec) (Dec. 2022). Ed. by Guojun Wang, Kim-Kwang Raymond Choo, Jie Wu, and Ernesto Damiani. Springer Nature Singapore: Singapore, p. 64-82. Communications in Computer and Information Science (CCIS, volume 1768).
  DOI: 10.1007/978-981-99-0272-9_5.

- Tiange Zhao, Ulrike Lechner, Maria Pinto-Albuquerque, Didem Ongu. "An ontology-based model for evaluating cloud attack scenarios in CATS – a serious game in cloud security." In: IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC) (Jun. 2023). IEEE, p.1-9.
  DOI: 10.1109/ICE/ITMC58018.2023.10332371

- Tiange Zhao, Ulrike Lechner, Maria Pinto-Albuquerque, and Tiago Gasiba. "Thriving in the era of Hybrid Work: Raising Cybersecurity Awareness Using Serious Games in Industry Trainings." In Journal of Systems and Software. Volume 210. p. 111946.
  DOI: 10.1016/j.jss.2023.111946

- Tiange Zhao, Ulrike Lechner, Maria Pinto-Albuquerque Tiago Gasiba and Didem Ongu. "COPYCAT: Applying Serious Games in Industry for Defending Supply Chain Attack." In Innovations for Community Services (I4CS) (Jun. 2024). Springer Nature Switzerland: Cham p. 321-336
  DOI: 10.1007/978-3-031-60433-1_18

- Tiange Zhao, Ulrike Lechner, Maria Pinto-Albuquerque Tiago Gasiba and Didem Ongu. "A Deep Dive Into CATS Evaluator Algorithm: Quantification of the Probability in Serious Game Cloud Security Defense Scenarios." 2024 36th International Conference on Software Engineering Education and Training (CSEE&T) (Jul. 2024). IEEE p. 227-231
  DOI: 10.1109/CSEET62301.2024.10663050

- Tiange Zhao, Ulrike Lechner, Maria Pinto-Albuquerque, Tiago Gasiba. "Thriving in the era of hybrid work: Raising cyber security awareness using serious games in industry trainings" 2024 36th International Conference on Software Engineering Education and Training (CSEE&T) (Jul. 2024). IEEE p. 227-231
  DOI: 10.1109/CSEET62301.2024.10663040

Individual contributions from the articles to the individual chapters and sections are also declared in their corresponding introductions. In this research, I collaborated with two students. I supervised one master's thesis of Ece Ata in 2022 at the Technical University of Munich, with the title "A Serious Game for Cloud Security." Her work contributed to the design of CATS's digital platform. Didem Ongu also participated in the present work as a working student under my supervision. She helped to implement the refinement of the evaluator algorithm, manage the deployment of the game platform, and process the collected data.

1.5 Structure of the thesis

This thesis consists of eight chapters. Each chapter is developed based on a topic relevant to our study.

In chapter 1, we introduce the topic's background and motivation. We aim to illustrate the complexity of cloud security, the theoretical models, and the urgency to raise awareness about cloud security issues. On the one hand, the shared responsibility model provided by the cloud service provider is dedicated to showing the responsibilities of the cloud service provider and the customer. On the other hand, the shared responsibility model needs refinement, and the stakeholders from the cloud service customer aspect also need further instructions on who should do what. We name a handful of cloud security incidents that have occurred due to the inherent characteristics of cloud computing. Such incidents could have been avoided if the cloud service were configured properly with strengthened cloud security awareness. Besides, we list our publications during our study in chapter 1.

In chapter 2, we describe the methodology we follow to conduct the research. Our study is guided by the design science research paradigm, where the core of design science research is the cycle of Design & Implementation and Justify & Evaluation. In the chapter, we provide a differentiation of Action-Design

Research, Action Research, and Design Research and detail the elements of our research design. We complete three design cycles: the initial design, the online cycle, and the refinement cycle. In each design cycle, we collect feedback as input for the improvement in the next iteration.

In chapter 3, we present the state of the art of our work. Our research involves several topics. The previous work contributes to our study differently. The third chapter starts with the reference scenario, including the typical scenario of how cloud assets are deployed and used and the common threats that prevail for cloud assets. Then, we present the topic of cyber security awareness, which is essential in our study. We follow the three dimensions of IT security awareness and extend them to cloud security: perception, protection, and behavior. Then, we list the precious works in cloud security, serious games, security ontology, Common Vulnerability Scoring System (CVSS) calculation, and requirements. Those works provide important inspiration for our study,

We describe the three design cycles in chapter 4. In each design cycle, we present the problem formulation, design and implementation, evaluation, and, at last, the reflection and learning of the current design cycle. It illustrates how our design evolves during the iterations and demonstrates the improvement we accomplish in concrete problem setting.

In chapter 5, we provide a deep dive into CATS and a detailed description of the game towards the end of our study. It is a result of the previous design cycles. We introduce the background, game modes, game process, interface, and winning condition.

In chapter 6, we present the road map to raising cyber security awareness with CATS in the industry. In this chapter, we focus on the setting of our study. More specifically, we describe the environment that triggers and motivates the study in the era of hybrid work. We develop our method of collecting feedback and conducting evaluations in the unique setting of our study. We present CATS as a method to improve interactivity and keep participants focused despite the challenging aspect of online training.

In chapter 7, we reflect on the existing body of knowledge from our design and share how we contribute to the theory by designing and implementing CATS. The chapter starts with a reflection on design science research principles and further reflects on research embedded into practice. Previous chapters mention that CATS is designed under the Design Science Research paradigm. In this chapter, we share our research experience and contribute to the existing body of knowledge.

In chapter 8, we conclude our work. We state the course of action, our contribution, and the final words. This study is conducted under the scientific guidance of design science research and accomplishes three design iterations. In the meantime, the design artifact is useful. It also guides and inspires similar studies on raising cyber security awareness using serious games. In this chapter, we retrospectively look at and recapitulate the important points

we discover in our journey of designing and applying a serious game to raise awareness of cloud security in the industry.

2

Methodology

Our research is guided by the design science paradigm proposed by Hevner et al. (Hevner, March, and Park, 2004). The method literature on design science describes the design of a useful artifact as a creative search process for a useful solution. Design Science Research (DSR) provides theoretical support to our work searching and designing a useful artifact in the industry.

Section 2.1 introduces the design science research methodology. Our work went through three major design cycles, and section 2.2 gives an overview of the design cycles. Finally, section 2.3 summarizes the chapter.

2.1 Design Science Research

Our work is guided by the Design Science Research paradigm (Hevner, 2007; Hevner, March, and Park, 2004) proposed by Hevner et al. In the work of Hevner et al., they describe the core of Design Science Research as the cycle of Design & Implement and Justify & Evaluate. We apply the engineering method in our research. We designed and implemented the serious game artifact and organized game events for justification and evaluation.

According to Hevner et al. (Hevner, March, and Park, 2004), the result of Design Science Research in information systems is, by definition, a purposeful IT artifact created to address an important organizational problem. It must be described effectively, enabling its implementation and application in an appropriate domain. Research in information systems aims to acquire knowledge and understanding that enable developing and implementing technology-based solutions to unsolved and important business problems. Good Design Science Research must provide clear contributions in the areas of the design artifact, design construction knowledge, and/or design evaluation knowledge. Design

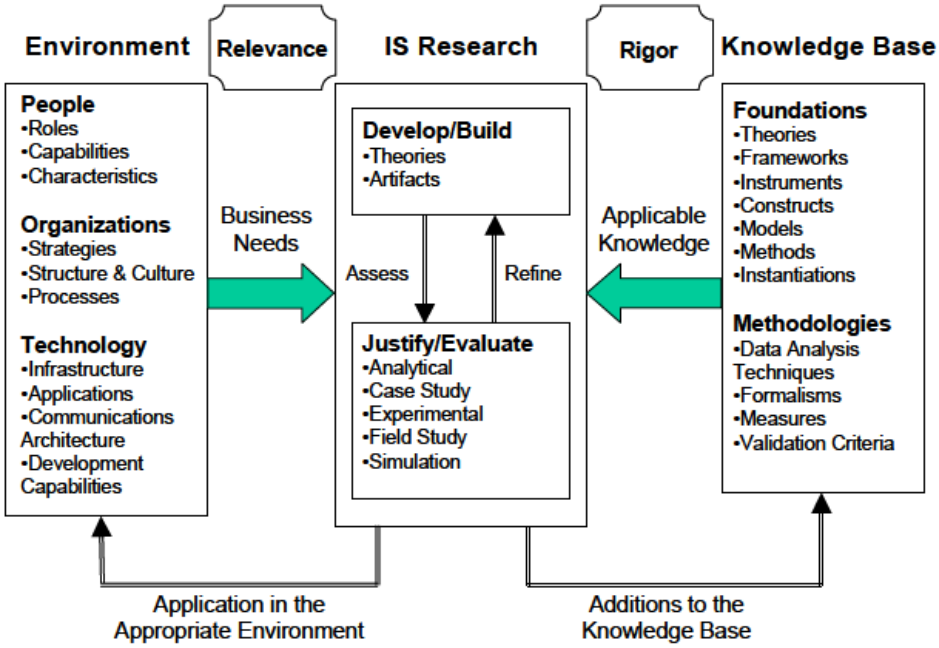


Figure 2.1: Information Systems Research Framework (Hevner, March, and Park, 2004)

Science Research relies on rigorous methods in constructing and evaluating the design artifact, as illustrated in figure 2.1. Technology-oriented audiences need sufficient detail to enable the described artifact to be constructed (implemented) and used within an appropriate organizational context (Hevner, March, and Park, 2004).

We need to differentiate between Action-Design Research (ADR), Action Research (AR), and Design Science Research (DSR). As Baskerville et al. described in their work (Baskerville and Wood-Harper, 1998), AR is an interventionist approach to acquiring scientific knowledge that has found foundations in the post-positivist tradition. ADR originates from Action Research, which focuses on the design of an artifact and the relevant theories. ADR is about working on problems of organizations and doing research with practitioners together while solving a practical problem. Sein et al. elaborate that ADR emphasizes the relevance of the learning cycle of DSR (Hevner, March, and Park, 2004) by providing concrete guidance on building, intervening, and evaluating in a concerted research effort. Our work has the nature of ADR as it is based in an organization. We focus, however, on the design and evaluation of the artifact and do not consider the organizational context. Therefore, we present our work as a DSR project. We take inspiration from ADR to have the learning and reflection phase after the design and evaluation of the game have been completed. We also work closely with practitioners in the industry within

an organization, which is also one of the characteristics of the ADR method.

Vom Brocke et al. mention in their work (Vom Brocke and Maedche, 2019) that each DSR project is a complex matter with many aspects that must be considered. DSR is an iterative process, starting with identifying a problem in the problem space and evaluating alternative solutions in the solution space. Through multiple iterations, a DSR project seeks to contribute means-end relationships between problem and solution spaces.

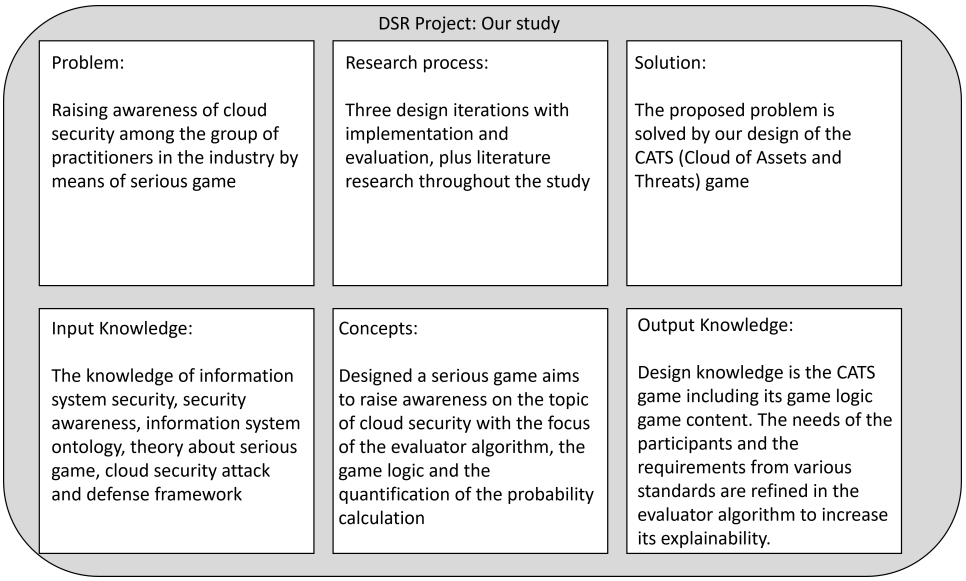


Figure 2.2: Mapping of the six core dimensions of DSR

All DSR activities must be aligned to such intended contributions so modifications in such positioning lead to subsequent changes in how to scope and plan a DSR project. Vom Brocke et al. have identified six core dimensions of a DSP project that can lay the foundation for a high-level, one-page framework that describes DSR projects. We map the six dimensions to the scope in our work as illustrated in figure 2.2.

- Problem description
  Problems should be formulated using problem statements and characterized by positioning the problem in a problem space. Research has identified the context, described by the domain, the stakeholder, time and place, and goodness criteria, the last of which tells when a problem should be considered solved, as necessary to capture the problem appropriately (Vom Brocke and Maedche, 2019).

  In our work, we would like to solve the problem of raising awareness of cloud security among the group of practitioners in the industry through a serious game. The criteria for goodness are reflected in the feedback we

collected in the game events and afterwards: 1) Was the game logic valid? 2) Does the participant find the game enjoyable? 3) does the participant find the game helpful? 4) How is awareness improved according to the three dimensions of perception, protection, and behavior?

- Input knowledge
  Input knowledge refers to the prior knowledge that will be used in the DSR project. Our project includes the knowledge of information system security, security awareness, information system ontology, theory about serious games, cloud security attacks, and defense frameworks.

- Research process
  The research process refers to the essential activities planned and conducted. In our work, we follow the design iterations, and along with the three design iterations, with implementation and evaluation, we conduct literature research to enrich our knowledge of the research topic.

- Key concepts
  The key concepts refer to the most important concepts used in the DSR project, such as the words used to describe the research. In our study, we designed a serious game used in the industry that aims to raise awareness among the practitioners in training on cloud security. Knowledge such as the cloud service provisioning model, shared responsibility model, information security standards focusing on cloud security, IT security ontology, Common Vulnerability Scoring System, IT security awareness, and serious game design are essential to reach the goal. The key concepts include the game logic, the quantification of the probability calculation, and the evaluator algorithm. A detailed description of those key concepts can be found in chapter 3.

- Solution description
  The solution description refers to the solution to the problem being investigated by a DSR project. In our work, the proposed problem is solved by our design of the CATS game. This tabletop game can be played online. Players can join as a team or as single players. The game aims to help players learn about typical defenses and attacks and cloud security roles and responsibilities. The primary target group of the players is industry practitioners.

- Output knowledge
  Output knowledge is produced in the DSR project. It can be design knowledge (DK). If design entities are generated in the DSR project, the design entity's evaluation description constitutes DK. Output knowledge includes the theory-ingrained artifact. In our work, the design

knowledge we gain is the CATS game, including its game logic, game content, and the creative search process for a good and useful design. We identify the needs of the participants and the requirements based on various standards. The evaluations contribute to the understanding of the usefulness of the artifact.

Gleasure (Gleasure, 2013) and Hevner et al. (Hevner, 2007; Hevner, March, and Park, 2004) describe that when the prescriptive aspect of a research problem is less mature than its descriptive or normative dimensions, the information system (IS) research problem is 'wicked.' Such problems are not suitable for traditional science approaches and instead require the situated theorizing afforded in the context of active design. Design Science Research can handle the cognitive and social limitations we encounter in practice and the industry. There is no single best solution to such a 'wicked' problem, but a creative search process can shape a useful solution to such a problem. Therefore, we conclude that the DSR approach suits our problem.

2.2 Overview of design cycles

This work follows the design science paradigm proposed by Hevner et al. (Hevner, 2007). Their work describes the core of Design Science Research as the cycle of Design & Implement and Justify & Evaluate.

Table 2.1: Three design interactions following the design science paradigm

Iteration	Design & implement	Justify & evaluate
1	Initial design: - Defensive / Offensive - Game logic - Game features	**Participants:** 3 x trial runs with 11 participants **Method:** Observers in trial runs Open discussion in the end (recorded)
2	Online cycle: - Defensive - Online platform - Attack scenarios	**Participants:** 10 game events with 123 industrial practitioners **Method:** Survey Open discussion in the end Semi-structured interview
3	Refinement cycle - Ontology extension - Introducing CVSS - Evaluator algorithm	**Participants:** 2 game events with 24 industrial practitioners **Method:** Survey Open discussion in the end

In this study, we finished three design iterations, as shown in table 2.1. In the first design iteration, *Initial design*, the game prototype is proposed and validated by three trial runs. Based on the feedback we collected in *Initial design*,

we decided to develop a digital platform for the game in the second design iteration, *Online cycle*, and the digital platform is evaluated by ten game events organized with 123 industrial practitioners. In the third design iteration, we focus on refining the evaluator algorithm and applying systematical mappings with CVSS and CVEs that occurred in real-world cloud security incidents. This design iteration is the *Refinement cycle*. The refined evaluator algorithm is tested in 2 game events with 24 industrial practitioners. In the tests conducted during the third design iteration, we aim to refine the evaluator algorithm and enhance the realism of attack scenarios, thereby providing players with a better understanding of the consequences of their defense strategies. The related tests and the results validated the efficacy of the refined evaluator algorithm in enhancing the gameplay experience and fostering better cloud security awareness.

2.3 Chapter summary

This chapter describes the methodology that guides us in our work. Design Science Research is an appropriate methodology for designing a serious game to raise cloud security awareness in the industry due to the factor introduced in section 2.1. In section 2.2, we describe the three design iterations in light of DSR.

3

State of the art

This chapter presents state of the art of our study. Our work is inspired by the cyber security awareness program conducted in the company, which shows the importance of such a program. Through training, practitioners in the industry and the organization can benefit from such activities.

The first section 3.1 describes the requirements of such a successful security awareness program within the scope of our study setting. The second section 3.2 presents how cloud assets are deployed and operated in a typical industry background, and this is the scenario that our serious game CATS addresses. The section 3.3 summarizes relevant cloud security know-how, cloud security best practices, and standards that are reflected in the game element. The section 3.4 describes the relevant literature on awareness programs and how awareness can be measured in training programs. This literature provides valuable insights for the evaluation process of our game. The section 3.5 shows the landscape of existing serious games to improve cyber security in general, and those game designs have inspired us to design CATS. The sections 3.6 and 3.7 explain in detail the CVSS metrics and the security ontology which are used in the last design iteration of our game and provide a foundation for the core evaluator algorithm that simulates and calculates the success rate of the defense strategy against the given attack scenario. Section 3.8 summarizes the chapter.

3.1 Requirements for successful security awareness programs

This first section of the state-of-the-art chapter is dedicated to a case study on the implementation of success in selecting content for a cyber security awareness program. This case study identifies success factors that shaped the design and implementation of CyberSecurity Challenges and guides our

approach to the design and implementation of CATS. An early version of the case study is published in the thesis of Dr. Tiago Gasiba (Gasiba, 2021). An extended version that adds information on the context of hybrid work is published in Journal of Systems and Software (T. Zhao, Gasiba, et al., 2024), emphasizing the generalizability of the case study's findings. This section gives a brief outline of the case study and its results.

The case study started with an initiative to update the company's secure coding guidelines to reflect the latest advances in technical know-how and industry best practices. The motivation was to refile a collaborative effort to emphasize the requirements and the role standards play in improving the organization's cyber security level. The main outcome of this project was a set of secure programming guidelines, an understanding of the importance of industrial standards, and the resources an organization is willing to spend on improving cyber security.

The CyberSecurity Challenges as a security awareness program is a follow-up activity in the case study. It is integrated into the company's software development lifecycle. The rationale behind this is 1) to lower the burden of compliance towards secure coding guidelines and cloud security standards by automating as much as possible and 2) to recognize that automation tools are not sufficient to detect and eliminate weaknesses in cloud architecture design and cloud asset configuration; the practitioners in the industry and the operator of cloud assets need to be trained in securing cloud assets and in particular in the newly defined security-related cloud services and cloud defense mechanism. The analyses emphasized the role of the individual and IT security awareness and empowered the software developers to be excellent in secure cloud assets deployment and operation.

The project to create the CyberSecurity Challenges (Gasiba, 2021) as a serious game to raise awareness for secure coding resulted from the considerations to facilitate the uptake of the guidelines and how to empower industrial software developers for more excellence in secure coding. This attempt is extended to develop CATS further to enable the participants to learn interactively about the roles and responsibilities, the cloud security issues, and their effective mitigation.

The project of establishing the secure coding policy and the CyberSecurity Challenges as serious games for awareness was a success story in the company. While the activities to implement the serious game as part of the company training activity curriculum unfolded, the topic of cloud security became more prominent — also through the need to advance cloud security as part of hybrid work both in the company and in critical infrastructure for which the company operates digital infrastructure and services. Shared responsibilities for a secure cloud were determined to be the topic to be addressed to increase security.

As part of the company, we, as the researchers and the department, decided to follow the example of secure coding guidelines to identify cloud security

rules and raise awareness for secure cloud usage through a serious game. Information from Cloud Security Alliance (CSA) and MITRE ATT&CK cloud matrix (MITRE ATT&CK, 2020e) were employed to identify cloud vulnerabilities. The experience with CSC expanded into a finding of willingness to expend funding to support industry practitioners in participating in cyber security training. This work also follows CSC in emphasizing the defensive side of cyber security, differentiating it from other cyber security serious games or interactive training methods.

Following the analysis result from the case study, we identified cloud security and shared responsibility model as important aspect of cyber security. Cloud Security Alliance (CSA) has listed the 11 most common cloud security issues in "Top Threats to Cloud Computing: The Egregious 11" (Cloud Security Alliance, 2019). The listed cloud security issues are data breaches; misconfiguration and inadequate change control; lack of cloud security architecture and strategy; insufficient identity, credential access and key management; account hijacking; insider threat; insecure interfaces and APIs; weak control plane; metastructure and applistructure failures; limited cloud usage visibility; abuse and nefarious use of cloud services. To address those potential issues due to the nature of cloud deployment and the human factor, the company puts much effort into cloud security training by providing an example of implementation recommendations to ensure the cloud assets are configured securely, and the standards are met.

The secure coding guidelines were turned into elements of a serious game. This game was designed to raise awareness of secure cloud programming and deployment and empower developers to apply secure coding to open architectures and business models. The case study helps us understand the organization's needs and identify important factors such as the cloud control matrix, the shared responsibility model, and the defense-in-depth concept that should be reflected in our serious game design.

3.2 Reference scenario

The reference scenario includes how cloud assets are deployed and operated by the practitioners in the industry and the shared responsibility model in our setting. A contract must be signed between the company (cloud service customer) and CSP to use the cloud services CSP provides. Additionally, a couple of typical threat vectors will be introduced. In the contract, a detailed service level agreement needs to be specified. Then, the practitioners can access the CSP platform with the agreed authentication method and configure the cloud assets accordingly to guarantee a secure deployment and provide the service to second-level customers.

The shared responsibility model describes from a high level what the CSP should provide to customers and the responsibilities of both parties, as illustrated in figure 3.1.

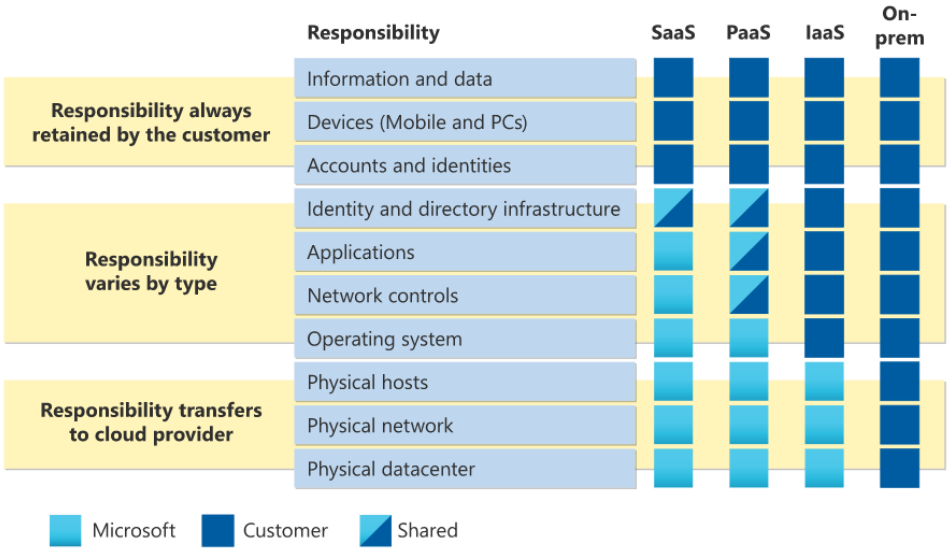


Figure 3.1: The shared responsibility model from Microsoft Azure (Azure, 2023)

The responsibility shared between the CSP and the customer differs slightly in different service models. However, the customer is always responsible for securing the information, data, devices and configuring the accounts and identities. This means the cloud service customer should always use what the CSP provides and configure them securely to meet the requirements of the customer's organization. In our game, we focus on the customer side of the shared responsibility model and further divide the customer's responsibility into business responsibility and technical responsibility.

In 2022, CSA published a list of the top eleven threats to cloud computing where traditional cloud security issues become less concerning, but CSP security issues become more concerning (Cloud Security Alliance Top Threats Working Group, 2022). In this section, we will list the first five threats of the ranking list and introduce the cloud threats landscape. Some generic threats, such as the fifth point of the list below, could threaten the cloud environment's security.

- Insufficient identity, credential, access and key management (Cloud Security Alliance Top Threats Working Group, 2022)
  This threat ranks first in the pandemic 11. Even though CSP offers numerous security services, it is still confusing for practitioners to establish a tight access control that fulfills the least privilege principle.

- Insecure interfaces and APIs (Cloud Security Alliance Top Threats Working Group, 2022)

The security and availability of general cloud services depend on the security of the APIs provided by the CSP. CSPs and customers protect those APIs and interfaces against accidental and malicious attempts to circumvent the security policy.

- Misconfiguration and inadequate change control (Cloud Security Alliance Top Threats Working Group, 2022)
  The customer takes full responsibility for setting up the cloud services securely. Misconfiguration of cloud resources is a leading cause of data breaches and could allow deletion or modification of resources and service interruption. It is introduced after the CSA Top Threat 2019 (Cloud Security Alliance, 2019).

- Lack of cloud security architecture and strategy (Cloud Security Alliance Top Threats Working Group, 2022)
  This threat refers to the challenges of implementing appropriate security architecture when migrating IT infrastructure to the public cloud. Customers are fully responsible for designing and implementing a secure cloud-based architecture.

- Insecure software development (Cloud Security Alliance Top Threats Working Group, 2022)
  This threat is not a cloud-specific threat. If the application has security issues, migrating it to the cloud will not eliminate them but rather expose them to a broader attack surface. It is introduced after the CSA Top Threat 2019 (Cloud Security Alliance, 2019).

3.3 Cloud security

There are many standards for information security in the industry, and cloud security is a critical subset of cyber security. Best known are ISO/IEC27001 (ISO27001, 2017) and MITRE ATT&CK (MITRE ATT&CK, 2020e). The standard ISO/IEC27001 (ISO27001, 2017) describes how to provide customers with certified products and services. The standards specify how cloud assets should be protected, e.g., by monitoring and data encryption, and mandate secure deployment and maintenance. MITRE ATT&CK (MITRE ATT&CK, 2020e) categorizes attack action and defense mechanisms in a cloud matrix based on industry standards and real-world observation. Our work helps to transfer the requirements of industry standards to industry practitioners by assisting them in understanding cloud security concepts through a serious game.

In the industry, standards define necessary protections for cloud assets. The best known among them are the ISO 27017 (ISO27017, 2015) and ISO 27001 (ISO27002, 2013), which require the practitioners to participate in training to learn about security technologies and raise awareness on cyber security issues. The CSA CCM (Cloud Security Alliance Cloud Controls

Matrix) (Cloud Security Alliance, 2021) compares 44 cloud security standards and shows an overview of the coverage of cloud security controls. MITRE ATT&CK cloud matrix (MITRE ATT&CK, 2020e) categorizes cloud attack actions and defense mechanisms based on real-world observations. It provides an adequate framework to derive the important game elements in CATS.

3.4 Awareness

Kruger et al. propose a prototype for assessing information security awareness. In their prototype, information security awareness is measured in three dimensions: what does a person know (knowledge); how do they feel about the topic (attitude); and what do they do (behavior) (Kruger and Kearney, 2006), known as the KAB model.

The work of Hänsch et al. on IT security awareness (Hänsch and Benenson, 2014) and its refinement by Gasiba (Gasiba, 2021) are the theoretical foundation for our research. In their work, the model returns to three dimensions of IT security awareness: perception, protection, and behavior. The KAB model shares significant similarities with the work of Hänsch et al. Perception is a combination of knowledge and attitude; both models have the same behavior definition. Hänsch et al.'s model emphasizes the idea of "protection," which overlaps with the idea of "knowledge" to some extent in the model of Kruger et al. According to Hänsch et al., perception is related to knowledge about IT security, protection is related to knowledge of how to protect IT assets, and behavior is related to the intention to protect IT assets actively. Gasiba refined three dimensions of secure coding: perception, which involves knowing about software security vulnerabilities; protection, which involves knowing how to protect against these vulnerabilities; and behavior, which involves the intention to write secure code.

In our work, the concepts presented in these three dimensions are used to evaluate our artifact in the industry and to understand how serious games affect the cyber security awareness level of the participants. Our evaluations focus mainly on the level of awareness in the three dimensions mentioned above. We find a certain degree of overlap between those three dimensions and the decomposition of quality factors proposed by Petri et al. In the work of Petri et al. on MEEGA+ (Petri, Wangenheim, and Borgatto, 2016), a model they proposed to evaluate the quality of educational games proposes different factors to evaluate the player experience and perceived learning.

In (Gasiba, 2021), Gasiba also investigates compliance with security policies as a means of behavior analysis. Neutralization theory captures the arguments employees use to rationalize security policy non-compliance. Siponen et al. (Siponen and Vance, 2010) propose in their work that the "Defense of Necessity," the need to get work done, triggers the intent not to comply with security policies. Moody et al. (Moody, Siponen, and Pahnila, 2018) in the Unified Model of Security Policy Compliance studies compliance with IT-security

policies. This model results from a meta-analysis that marks a milestone as it synthesizes different studies on security policy compliance. Social factors and facilitating conditions are important for the intention to comply with security policies. Various studies on the unified model find that deterrence or fear does not contribute to compliance with IT security policies (Vance, Siponen, and Pahnila, 2012). Note that these studies and models generally study white-collar work and are not tailored to the particular context of software engineers.

Khando et al. (Khando et al., 2021) did a systematic literature review on awareness programs. They discovered that various methods and factors enhance employees' information security awareness (ISA) in organizations. Theoretical models and gamification are widely used in private and public organizations. In contrast, the constructivist approach and violation detection are some methods used only in private organizations.

It is mentioned in the white paper from Secure Code Warrior (Warrior, 2021) that developers do not need to become security experts. Still, they must be empowered to be their organization's first line of defense. Graziotin et al. (Graziotin et al., 2018; D. Graziotin, Wang, and Abrahamsson, 2015) study human factors in code quality and found consequences of happiness and unhappiness that are beneficial or detrimental to developers' mental well-being, the software development process, and the produced artifacts. The contrast mentioned above, together with the fact that developers are perceived to be generally under pressure to deliver software with as little business risk as possible. However, the time and resources to help write secure code from the start are limited. Therefore, good practical knowledge and strong cyber security awareness are advantageous to raise the level of security (T. Zhao, Gasiba, et al., 2024). Studies are not concerned with cloud security or shared responsibility models.

3.5 Serious games on cyber security

Various serious games are designed and implemented to raise awareness of cyber security. Hendrix et al. (Hendrix, Al-Sherbaz, and Bloom, 2016) did a literature review and a product search on game-based cyber security training. They selected 28 papers in the literature review, while only 11 studies conducted an evaluation which can be scrutinized. These studies report a positive outcome, indicating that the game studies contributed to training or raising cyber security awareness. In the product search, 15 training games were identified. The most popular target audiences were children, teenagers, and students.

Larson did a literature review (Larson, 2020) on serious games and gamification in the corporate training environment. The author identified 50 journal articles, 21 conference proceedings, six white papers and dissertations, seven professional reports, and six books to be included in this review on serious games and gamification and found significant empirical evidence of successful applications of game-based learning in the business realm. Various corporate entities have successfully implemented gamification and serious games as a

component of their overall training strategy. The results of numerous studies have evidenced a significant decrease in lost time, increase in engagement, and revenue impacts up to $30 million.

Coenraad et al. (Coenraad et al., 2020) investigated cyber security digital games on the Apple App Store, the Google Play Store, Steam, and the web. The systematic review of the 181 digital games found that few currently available games provide deep content engagement with cyber security concepts and fulfill the potential of game-based learning to broaden participation in cyber security. This suggests a significant game of developing such a digital game in greater depth to reflect elements from real-world cyber security attacks.

Shostack maintains a list of security tabletop games on his website (Shostack, 2021). One early example of a video game with cyber security simulation is CyberCIEGE (Thompson and Irvine, 2011), where players purchase and configure workstations, servers, operating systems, applications, and network devices. They make trade-offs as they struggle to balance budget, productivity, and security. CyberCIEGE is mainly targeted at students in schools and institutes, and it successfully avoids the fear of failing by allowing students to explore simulated scenarios.

Another example of a cyber security serious game in the company is the game Riskio of Hart et al. (Hart et al., 2020). Riskio is dedicated to people without technical backgrounds and successfully increases cyber security awareness for people working in organizations. Riskio is a tabletop game focusing on defensive and offensive skills in IT security in general. It provides important insights into the impact of such serious games. Comparing CSC to Riskio, the target group of CSC is people with a technical background, and the purpose is to empower the developers to write secure code.

Another Week at the Office (AWATO) (Ferro et al., 2022) by Ferro et al. is another serious game developed based on a systematic literature review. The game focuses on the human factor by raising awareness of phishing attacks. The evaluation of the game shows that it is an effective tool for improving users' awareness of cyber security best practices. Their work further proves that serious games could be a useful approach to solving awareness issues.

Hofmeier (Hofmeier, 2024) proposed a serious game called Operation Digital Butterfly following the Design Science Research paradigm and focuses on insider threats. In an organization, insider threats can be highly harmful and difficult to detect. Operation Digital Butterfly helps to raise awareness to such problem and is useful to detect different roles and attack scenarios.

The work of Švábensk ý, et al. (Švábenskỳ, Cermak, and Laštovička, 2018) shows another possibility of applying serious games in university teaching. They found that creating serious games contributes to fostering adversary thinking. Their study lasted over three semesters, and the game's purpose was to teach undergraduate students about network attack and defense by creating educational games in a cyber range environment. The students

report they had a unique opportunity to understand the topic deeply. The game created is played by their college mates, who rated the quality and educational value of the games overwhelmingly positively. Their work shows exciting results in the academic environment. The work of Mendes (Mendes, 2021) aims to build an automatic tool to generate programming problems that can be used in teaching in an academic environment.

One of the most obvious serious game approaches would be an online platform with programming tasks for programming education. One example is Codewars (Codewars, 2023), which could improve the players' development skills by training with coding tasks that continuously challenge and push coding practice. This approach is helpful in self-paced training but is not applicable in our industry training setting. Additionally, in capture-the-flag activity, players play the role of an attacker in team red and the role of a defender in team blue and play against each other (HITB CyberWeek, 2023). The setting of our environment makes our study focus more on the defensive aspect of serious games. Dewes et al. (Dewes, Gasiba, and Schreck, 2022) provides a mapping of famous serious games and their mapping to different job profiles in the industry since different skill sets are required for the various roles in cyber security.

The work of Arnab et al. (Arnab et al., 2015) proposes the Learning Mechanics–Game Mechanics (LM-GM) model, which supports serious game analysis and design by allowing reflection on the various pedagogical and game elements in a serious game.

While many games exist to raise cyber security awareness, the ones mentioned above can be considered to represent the variety of games and applications. These games provide evidence of the success of using serious games in cyber security education. In our work, we report on our experience in designing and applying serious games in training within the industry under different workplace conditions with work-from-home and hybrid work (T. Zhao, Gasiba, et al., 2024). Our study addresses a relevant yet specific topic of cloud security. The target group is industry practitioners and software developers. Those factors distinguish our work from the existing serious games in cyber security.

3.6 Security ontology

Our security ontology was developed based on the security relationship model described in the National Institute of Standards and Technology (NIST) Special Publication 800-12 (Guttman and Roback, 1995). Fenz et al. proposed a specific security ontology in their work (Fenz and Ekelhart, 2009), modeling the security relationship with *Security Attribute, Asset, Vulnerabilities, Threat and Security Scale*. In this work, we take the security ontology proposed by Fenz et al. as a starting point and further adjust it to our context: designing a serious game dedicated to raising awareness of cloud security for industrial practitioners. We provide a brief description of each concept.

- Security Attribute (*Original*) $\Rightarrow$ defines which security attributes (e.g.,

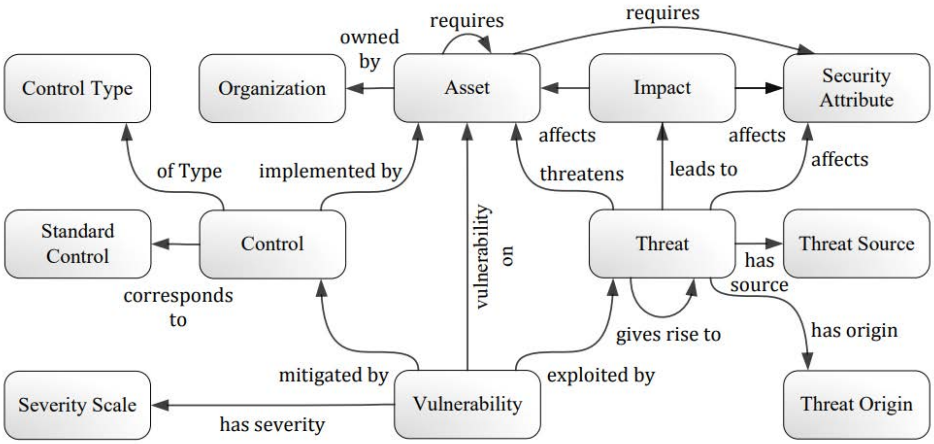


Figure 3.2: The ontology overview from work of Fenz et al. (Fenz and Ekelhart, 2009)

accountability, availability, confidentiality, integrity, reliability, or safety) can be affected by a certain threat. Our work considers only confidentiality, availability, and integrity, as used in impact metrics in the CVSS base group.

- Asset (*Original*) ⇒ can be a tangible or intangible asset in the original ontology. Our work mainly considers intangible cloud assets, e.g., cloud applications, data, user accounts, etc.

- Vulnerability (*Original*) ⇒ is the absence of a proper safeguard that a threat can exploit. Vulnerabilities on *Asset*s exist and could be mitigated by *Control*s. In our work, we map the attacks derived from MITRE ATT&CK cloud matrix (MITRE ATT&CK, 2020e) to a vulnerability and assign a CVSS score.

- Threat (*Original*) ⇒ The threat taxonomy comprises natural (e.g., earthquake, monsoon, or lightning), accidental (e.g., hardware failure or liquid leakage), and intentional (e.g., theft or alteration of software) threats at the highest level, followed by a detailed classification. The threat threatens the assets, and the threat exploits the vulnerability.

- Control (*Original*) ⇒ has to be implemented to mitigate an identified vulnerability and to protect the respective assets by preventive, corrective, deterrent, recovery, or detective measures. In our work, we map *Control*s to the defenses derived from MITRE ATT&CK cloud matrix (MITRE ATT&CK, 2020e) and these can be used to reduce the harm of a given *Vulnerability*.

- Severity Scale (*Original*) ⇒ of each vulnerability concept in the original security ontology is rated by a three-point scale (high, medium, and low) to enable a machine to interpret the significance of the vulnerability. Since we included the CVSS score in our work, we used the five-point scale according to CVSS vulnerability metrics (National Vulnerability Database, 2023b) (None, Low, Medium, High, Critical). The vulnerability with a base score of 0.0 gets the severity scale *None*. The vulnerability with a base score of 0.1-3.9 gets the severity scale *Low*. The vulnerability with a base score of 4.0-6.9 gets the severity scale *Medium*. The vulnerability with a base score of 7.0-8.9 gets the severity scale *High*. The vulnerability with a base score of 9.0-10.0 gets the severity scale *Critical*.

We adapted the security ontology proposed by Fenz et al. (Fenz and Ekelhart, 2009). The ontology as it was proposed by Fenz et al is shown in figure 3.2. The adaptation details are described in chapter 4.3.

3.7 CVSS Metrics

Common vulnerability scoring system (CVSS) metrics are a uniform way to communicate the characteristics of a cyber security vulnerability. We apply CVSS 3.1 since it is the mainstream version of CVSS for now. CVSS metrics describe how dangerous a vulnerability can be from the perspective of exposure, exploitability, and impact. It is tremendously helpful in our work since it is a widely agreed standard to describe different characteristics of a certain vulnerability, so introducing the CVSS metric into the evaluator algorithm could improve the explainability of the derived results. The richness of the vulnerability database allows us to map the attack actions in the attack scenarios to a vulnerability, and the well-defined mathematical calculation helps our evaluator algorithm quantify the probability calculation. Our evaluator algorithm and the CVSS mapping are described in chapter 4.3. In this chapter, we focus on the introduction of original CVSS metrics. The metric values are explained as the following (National Vulnerability Database, 2023b):

- Attack Vector (AV): Network (N), Adjacent (A), Local (L), Physical (P)
  This metric reflects the context by which vulnerability exploitation is possible. This metric value (and consequently the Base Score) will be larger, and the more remote (logically and physically) an attacker can be in exploiting the vulnerable component. The assumption is that the number of potential attackers for a vulnerability that could be exploited from across a network is larger than the number of potential attackers that could exploit a vulnerability requiring physical access to a device and, therefore, warrants a greater Base Score.

- Attack Complexity (AC): Low (L), High (H)
  This metric describes the conditions beyond the attacker's control

to exploit the vulnerability. The Base Score is greatest for the least complex attacks.

- Privileges Required (PR): None (N), Low (L), High (H)
  This metric describes the privileges an attacker must possess before successfully exploiting the vulnerability. The Base Score is greatest if no privileges are required.

- User Interaction (UI): None (N), Required (R)
  This metric determines whether the vulnerability can be exploited solely at the attacker's will or whether a separate user (or user-initiated process) must participate somehow. The Base Score is greatest when no user interaction is required.

- Scope (S): Unchanged (U), Changed (C)
  The Scope metric captures whether a vulnerability in one vulnerable component impacts resources in components beyond its security scope. The Base Score is greatest when a scope change occurs.

- Confidentiality (C): High (H), Low (L), None (N)
  This metric measures the impact on the confidentiality of the information resources managed by a software component due to a successfully exploited vulnerability. The Base Score is greatest when the loss to the impacted component is highest.

- Integrity (I): High (H), Low (L), None (N)
  This metric measures the impact on the integrity of a successfully exploited vulnerability. Integrity refers to the trustworthiness and veracity of information. The Base Score is greatest when the consequence to the impacted component is highest. When the integrity of information is deemed trustworthy, the data has not been tampered with or altered unauthorizedly. It is highest when the consequence to the impacted component's integrity is greatest.

- Availability (A): High (H), Low (L), None (N)
  This metric measures the impact on the availability of the impacted component resulting from a successfully exploited vulnerability. The Base Score is greatest when the consequence to the impacted component is highest.

3.8 Chapter summary

This chapter introduced the state of the art topics relevant to this thesis. The literature covers reference scenarios, awareness studies on information security, standards and best practices in cloud security, serious games in cyber

security, IT security ontology, CVSS, and requirements for successful security awareness programs.

The reference scenario provides a basis for the current thesis. Our work is dedicated to raising awareness of cloud security. The existing literature on serious games helps comprehend the landscape of serious game applications in cyber security and the industry. Security ontology and the CVSS calculation method provide a method for designing the concept of our serious game. Finally, we examined the requirements for a successful security awareness program and utilized the results in the design of our game.

4

Design of CATS

In this chapter, we describe the three design cycles. In each design cycle, we present the problem formulation, design and implementation, evaluation, and, at last, the reflection and learning of the current design cycle. It illustrates how our design evolves during the iterations and demonstrates the improvement we accomplished in concrete problem setting. In all the design cycles, we conducted game events or trial runs as a method for evaluation. In the first design cycle, the trial runs are standalone events conducted among groups of cyber security experts. In the second and third design cycles, we can deploy CATS in two ways as the game design matures. One way is a standalone game event, normally hosted after a workshop where concepts about cloud security are introduced to the participants. The second way is as a joint exercise with the CyberSecurity Challenge, which is a full-day event, and different categories of challenges are included. In this type of event, CATS is deployed as a category of challenges. In the second and third design cycles, CATS reuses the CyberSecurity Challenge Terraform script to deploy the infrastructure. This allows us to focus on serious game design and other elements.

Note that the earlier version of the first design iteration is published in the journal Information (T. Zhao, Gasiba, et al., 2021b). The earlier version of the second design iteration is published in ICPEC (T. Zhao, Lechner, Pinto-Albuquerque, and Ata, 2022). The earlier version of the third design iteration is published in ICE (T. Zhao, Lechner, Pinto-Albuquerque, and Ongu, 2023).

4.1 First design cycle - initial design

The design of CSC started in 2017 to address software developers' lack of awareness of secure coding. One of the conclusions from a previous case study

(T. Zhao, Gasiba, et al., 2024) is that software developers need guidance to understand the purpose and reasoning behind the need to follow secure coding guidelines. In general, the current security training based on classroom or online lectures is not adequate for the practitioners in the industry. This observation can be extended to the aspect of cloud security. Therefore, we decided to design a serious game in which practitioners in different roles in cloud asset protection can participate, learn about how to build a sufficient cloud asset defending strategy and learn about different roles and responsibilities.

The approach of applying serious games in on-job training in the industry has proven very successful during the design of the CyberSecurity Challenges, as shown in (Gasiba, 2021). As such, the CATS game was originally developed as an onsite event. However, in 2020, the game had to undergo significant changes to adapt to new working conditions caused by the COVID-19 pandemic. After the adaptation, the industry used the game as a standalone event. The resulting design of the game, which can then be used in a hybrid environment (onsite, remote, or mixed), is successful in the industry. Among others, one factor contributing to the game's success is its adoption into the standard training curriculum in the company where the game was originally developed. This successful design of CSC, combined with the rich set of lessons learned, led to the creation in 2021 of CATS in Siemens to raise awareness of the different roles needed in the secure deployment of cloud assets.

4.1.1 Problem formation

Cloud computing is a relevant and important architecture to provide IT services. The promised value of cloud service providers (CSP) is to provide high levels of service and security and save business costs. Significant players have dedicated themselves to providing cloud-based solutions to the customer. MindSphere (Petrik and Herzwurm, 2019) is an example of the industrial Internet of Things (iIoT) as a service. It powers IoT solutions from the edge to the cloud. A cloud solution can also be found in Siemens' healthcare industry. Teamplay (Veen, 2020) is a cloud-based network that brings together healthcare professionals in a team effort—the number and volume of applications and systems deployed and maintained in the cloud increases.

Cloud-based systems are exposed to security threats as listed in (Cloud Security Alliance, 2019), such as nefarious use of cloud services and lack of cloud security architecture and strategy. The current standards (Cloud Security Alliance, 2020; Di Giulio et al., 2017) and guidelines (Cloud Security Alliance, 2021, 2017) on cloud security describe the roles and responsibilities in cloud computing. These standards and guidelines are currently communicated to developers and managers using security training. It becomes imperative that a company helps its developers and managers understand the importance of cloud security and, more precisely, how it relates to daily work. Despite the many measures taken, cloud security awareness needs to be

improved. We propose a serious game to address this challenge. Our research interest is to design a serious game to facilitate the training of developers and managers about cloud security, especially the roles and responsibilities and the collaboration between cloud service providers and customers. This work presents a preliminary result of a tabletop game prototype designed to introduce fundamental concepts in cloud security.

Due to the cloud-based systems' nature, they face various security threats (Cloud Security Alliance, 2019). These threats include the nefarious use of cloud services and the lack of cloud security architecture and strategy. Poor management of cloud cyber security can lead to severe consequences. In 2017 (UpGuard Team, 2017), the United States Department of Defense disclosed login credentials to their cloud environment, which led to the disclosure of secret government intelligence data hosted in their cloud-deployed infrastructure. Also, misconfigurations of S3 buckets in Amazon's cloud environment have allowed several high-profile pieces of information to be leaked, resulting in severe monetary consequences due to the data breaches (Cimpanu, 2017). In (Scheffler, 2019), Scheffler provides details on the Man-in-the-cloud attack. In this type of attack, the goal of the malicious party is to gain control of the victims' cloud account by capturing credentials such as those present in OAuth tokens. One of the ways that the author proposes to address this problem is through regular security training to raise awareness of cloud cyber security.

The origin of cloud vulnerabilities is often two-fold. On the one hand, there are technical vulnerabilities, e.g., the poor configuration of cloud environments (either manual or automated). On the other hand, there is a lack of cloud security awareness among managers, cloud asset operators, and customers of cloud services. This work addresses cloud security awareness issues, considering particular roles and responsibilities in cloud service provisioning.

There are numerous standards that regulate cloud security, such as (Cloud Security Alliance, 2020; Di Giulio et al., 2017) and security guidelines (Cloud Security Alliance, 2021, 2017), that describe the roles and responsibilities in cloud computing. Security training is the primary method to communicate these standards and guidelines to developers and managers. Enterprises must help their developers and managers understand the importance of cloud security and, more precisely, how cloud security standards relate to daily work. Otherwise, the continuity of business is put in danger.

To address this challenge, we propose a serious game in this work — raising awareness of roles and responsibilities related to cloud security. In this research, we are interested in designing a serious game that facilitates the training of developers and managers about cloud security, especially the roles and responsibilities and the collaboration between cloud service providers and customers. This work presents a tabletop game prototype designed to introduce fundamental concepts in cloud security and the first results of the validation of this game.

4.1.2 Design and implementation

In the first brainstorming sessions, we identified the topic "Cloud Security" and the game framework: it should be a board game that can be played face-to-face or online in a virtual session to cope with the restrictions imposed by the COVID-19 situation. It was also determined that the insider perspective of non-compliance with security policies should be considered.

By its nature, several fundamental facts about cloud security could be transformed into a board game: 1) Cloud security can be described as a constant fight between attackers and defenders. 2) Both defenders and attackers are facing resource constraints. 3) Defenders might play different roles and thus take different responsibilities determined by their role in cloud security. 4) Attackers take attack actions to compromise cloud assets. Those elements above can be built into a board game prototype geared to help trainees better understand the basic concepts of cloud security.

The following core design elements for the serious game are identified in the design phase. The game prototype aims to address these design elements:

- Feature 1: Cloud Security Kill Chain.
- Feature 2: 100 percent security does not exist.
- Feature 3: Defense-in-depth helps.

Feature 1 refers to the attackers never doing a single-step attack. Instead, they plan the attack step-by-step to form a kill chain. Feature 2 refers to the restrictions that we have in reality. It is too expensive and unrealistic to have a perfect protection concept that eliminates the possibility of all types of attacks. The purpose is to make the effort of the attack bigger than the value of the protected assets. Feature 3 means that the game mechanism encourages the players to defend the cloud assets with more than one defense action, and this behavior will increase the defense success rate.

The classic game prototype needs a game master to organize and host the game. Before the game begins, the game master explains the game flow and rules to the participants and handles the questions raised by the participants during the game.

During the game, the defender team develops an attack plan, and the attacker team develops an attack plan. Each team uses a game board to place different cards to model attack and defense plans. Attackers and defenders can only place a limited number of cards. This constraint reflects the reality that neither the attacker nor the defender has unlimited resources, and both of them need to prioritize accordingly. This drafting of attack and defense plans is done in teamwork. Teams use breakout rooms virtually to discuss and develop plans in an online game. The concept can be adapted to an

Table 4.1: No. of cards for defender team and attacker team

Defender Team	Total no. of defense cards	24
	No. of defense cards belong to Asset Owner	8
	No. of defense cards to Asset Owner on Defense Plan	2
	No. of defense cards belong to Asset Manager	18
	No. of defense cards to Asset Manager on Defense Plan	4
Attacker Team	Total no. of attack cards	16
	No. of attack cards to choose from for Initial Access	5
	No. of attack cards for Initial Access on Attack Plan	2
	No. of attack cards to choose from for Launch Attack	8
	No. of attack cards for Launch Attack on Attack Plan	3
	No. of attack cards to choose from for Make Impact	3
	No. of attack cards for Make Impact	1

in-person game. Then, different teams should sit apart and work on their defense or attack plan separately.

In total, there are 40 cards, of which 24 are available to the defender team and 16 to the attacker team, as table 4.1 shows. Each card for the defender team states one countermeasure to secure cloud assets, for example, "Information Encryption" or "Network Segmentation" (see figure 4.1). On each card, the attacker team states one attack action to cloud assets, for example, "Monitoring Escaping" or "Network Service Discovery" (see figure 4.2).

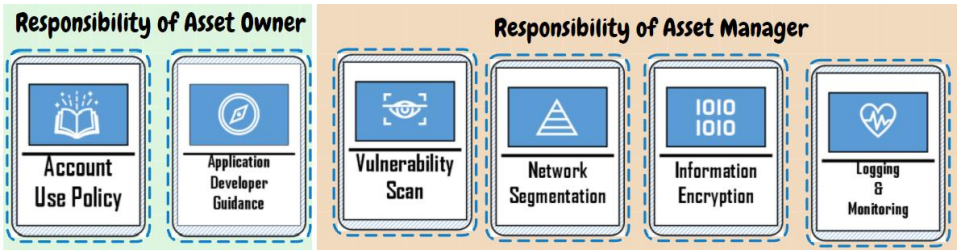


Figure 4.1: Trial run: a screenshot of the game board on the defender side

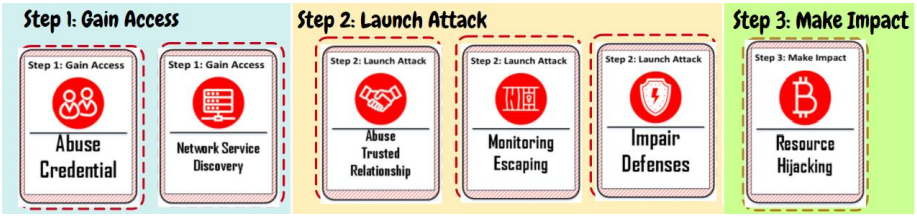


Figure 4.2: Trial run: a screenshot of the game board on the attacker side

Figure 4.3 presents the phases of the game as a flow chart.

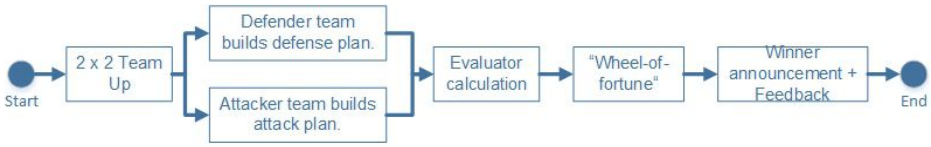


Figure 4.3: Game Process in Flowchart

When the game starts, the game master randomly assigns players to the two teams: *defender* and *attacker*. During the game, the task of the *defender* team is to build a defense plan from scratch by selecting defense cards and assigning them to one of the roles: cloud asset owner and cloud asset manager. They should discuss and decide with their teammates to assign two cards to the Asset Owner and four cards to the Asset Manager. The defender team needs to determine which of the 24 cards are the responsibility of the Cloud Asset Owner and which ones should be taken care of by the Cloud Asset Manager based on their understanding and knowledge. If a card is assigned to the wrong role, it will be sorted out from the defense plan and not contribute to a successful defense.

Table 4.1 illustrates the number of cards in each category. Two cards could be assigned either to the Asset Owner or Asset Manager. That is why the sum of the second and fourth row of table 4.1 is two more than the total number of defense cards. The *attacker* should build a three-step attack plan: Gain Access, Launch Attack and Make Impact. In total, 16 attack cards categorized into the three steps are made available to them, and they should discuss and decide with their teammates to assign 2, 3, and 1 card(s) to each step. Both teams have 20 minutes to build their defense plan or attack plan.

We derived the attack cards, defense cards, and the mapping relation between them from MITRE ATT&CK (MITRE ATT&CK, 2020e) primarily and the CSA cloud control matrix (Cloud Security Alliance, 2021) for additional information. The cloud matrix demonstrates the typical attack and defense actions in a cloud environment based on real-world incidents. In case the players are unfamiliar with cloud security defenses and attacks, cheat sheets with the key information are made available to them for assistance throughout the game process.

We list all the defense cards in the first design iteration. They are organized based on the attack steps in which they can be applied. The responsibility mapping is summarized in table 4.2.

- Account Management: Manage the creation, modification, use, and permissions associated with privileged accounts.
- Account Use Policies: Configure features related to account use, like login attempt lockouts, specific login times, etc.

Table 4.2: An overview of the defense cards are their mapping to either business responsibility or technical responsibility

	Business Responsibility	Technical Responsibility
Account Management		x
Account Use Policies		x
Active Directory Configuration		x
Application Developer Guidance	x	
Application Isolation and Sandboxing		x
Asset management	x	
Audit	x	
Backup Concept	x	
Code Signing		x
Critical Data Protection	x	
Disable or Remove Feature or Program		x
Encrypt Sensitive Information	x	x
Filter Network Traffic		x
Logging and monitoring		x
Multi-factor Authentication		x
Network Intrusion Prevention		x
Network Segmentation		x
OS Hardening		x
Password Policies	x	x
Restrict Permissions		x
Software Configuration		x
Update Software		x
User Training	x	
Vulnerability Scanning		x

- Active Directory Configuration: Configure directory service such as Active Directory to prevent the use of certain techniques; use SID (Security Identifier) Filtering, etc.
- Application Developer Guidance: This mitigation describes any guidance or training given to developers of applications to avoid introducing security weaknesses that an adversary may be able to take advantage of.

- Application Isolation and Sandboxing: Restrict code execution to a virtual environment on or in transit to an endpoint system.
- Asset Management: Identify and register cloud assets in a respective management system and keep on track if any change occurs.
- Audit: Perform audits or scans of systems, permissions, insecure software, insecure configurations, etc., to identify potential weaknesses.
- Backup Concept: A backup concept is a document that describes the strategy for protecting data from loss.
- Code Signing: Enforce binary and application integrity with digital signature verification to prevent untrusted code from executing.
- Critical Data Protection: Protect hosts and process highly critical data with dedicated cloud resources.
- Disable or Remove Feature or Program: Remove or deny access to unnecessary and potentially vulnerable software to prevent abuse by adversaries.
- Information Encryption: Protect sensitive information with strong encryption.
- Filter Network Traffic: Use network appliances to filter ingress or egress traffic and perform protocol-based filtering. Configure software on endpoints to filter network traffic.
- Filter Network Traffic: Use network appliances to filter ingress or egress traffic and perform protocol-based filtering. Configure software on endpoints to filter network traffic.
- Logging and Monitoring: Engage logging and monitoring measurements to supervise the overall health and performance of the cloud system.
- Multi-factor Authentication: Use two or more pieces of evidence to authenticate to a system.
- Network Intrusion Prevention: Use intrusion detection signatures to block traffic at network boundaries.
- Network Segmentation: Instead of having a flat network architecture, architect network sections to isolate critical systems, functions, or resources. Configuration changes related to the operating system or a common feature of the operating system that results in system hardening against techniques are made.

- Password Policies: Set and enforce secure password policies for accounts.
- Restrict Permissions: Restrict access by setting directory and file permissions that are not specific to users or privileged accounts.
- Software Configuration: Implement configuration changes to software (other than the operating system) to mitigate security risks associated with the software's operation.
- Update Software: Perform regular software updates to mitigate exploitation risk.
- User Training: Train users to be aware of access or manipulation attempts by an adversary to reduce the risk of successful spearphishing, social engineering, and other techniques that involve user interaction.
- Vulnerability Scanning: Vulnerability scanning is used to find potentially exploitable software vulnerabilities to remediate them.

We provide a list of all the attack cards in the first design iteration: In the first attack step, initial access, the attacker aims to gain the first access to the system by various means. Two out of the following attacks can be chosen.

- Abuse Credential: Attacker obtains and abuses credentials of existing accounts.
- Brute Force: The attacker uses brute force techniques to access accounts when passwords are unknown or password hashes are obtained.
- Cloud Infrastructure Discovery: The attacker attempts to discover available resources within an infrastructure-as-a-service (IaaS) environment.
- Exploit Public-Facing Application: The attacker takes advantage of a weakness in an Internet-facing computer or program using software, data, or commands to cause unintended or unanticipated behavior.
- Network Service Discovery: The attacker attempts to get a listing of services running on remote hosts, including those vulnerable to remote software exploitation.

In the second attack step, launch attack, the attacker aims to launch the attack on the system and damage the system in different ways. Three out of the following attacks can be chosen.

- Abuse Trusted Relationship: Attacker breaches through 3rd party providers who have access to intended victims.

- Account Manipulation: The attacker manipulates accounts to maintain access to victim systems.
- Cloud Storage Breach: The attacker accesses data objects from improperly secured cloud storage.
- Exploit Unused Region: To evade detection, the attacker creates cloud instances in unused geographic service regions.
- Impair Defenses: The attacker maliciously modifies components of a victim environment to hinder or disable defensive mechanisms.
- Implant Container Image: The attacker implants cloud container images with malicious code to establish persistence.
- Infrastructure Manipulation: To evade defenses, the attacker attempts to modify a cloud account's compute service infrastructure.
- Monitoring Escaping: Attackers exfiltrate data by transferring the data, including backups of cloud environments, to another cloud account they control on the same service to avoid typical file transfers/downloads and network-based exfiltration detection.

The third step, "make impact," describes the ultimate goal of the attack. One out of the following attacks can be chosen.

- Defacement: Attacker modifies visual content available internally or externally to an enterprise network.
- Denial of Service: The attacker performs Endpoint Denial of Service (DoS) attacks to degrade or block the availability of services to users.
- Resource Hijacking: The attacker leverages the resources of co-opted systems to validate cryptocurrency transactions and earn virtual currency.

The defender and attacker teams submit their defense and attack plans to an evaluator. The evaluator runs an algorithm to simulate the attack and defense steps and compute the probability of the defense plan withstanding the attack plan.

The game master shows the evaluator's results to participants and explains them when necessary. For instance, how an attack action is blocked by a single or multiple defense action card(s) and which attack action is left undefended if there is any. The evaluator finally outputs a percentage of the probability for the Defender Team to survive the attack.

In the next step, a "Wheel-of-Fortune" spins. This step involves a virtual spinning wheel with different slices marked as "Attacker wins" or "Defender

wins." The size of each area is determined by the probability calculated by the evaluator in the previous step. For instance, if the evaluator calculates that the defense plan has 80% of chance withstanding the attack plan, the marked "Defender wins" area will take up 80% of the wheel. The areas are distributed evenly on the surface of the wheel. At the end of the game, the game master will spin the "Wheel-of-Fortune," and the winner will be announced.

An evaluator runs an algorithm to analyze the defense plan against the attack plan and calculates the probability of the input attack plan tearing down the defense plan. Along with the attack steps, it shows the reasoning also step-by-step. The ultimate output of the evaluator is the calculated probability in percentage numbers. The evaluator has some adjustable parameters, such as the number of hints and the single success rate.

- Number of Hints

The hint is only available for the defender team to help them assign one card to the correct role before the next game step. When the team assigns a valid defense card to the correct role, as the hint suggests, this guarantees that the card will enter the evaluation and contribute to the overall success of the defense plan. Based on the game master's observation of the game and the players, they can decide whether to give the defender team a maximum of two hints. No hints are available for the attacker team to improve their attack plan.

- Single Success Rate

The single success rate (SSR) describes the likelihood of a single defense card successfully blocking a mapped attack card. It is the same for all defense-attack mapping pairs. It reflects the important fact that there is no 100% security (Feature 2 in section 4.1.2). For instance, if we consider Multi-factor Authentication (MFA) as effective mitigation against account manipulation and the SSR is set to 80%, it means MFA has an 80% chance of successfully stopping an account manipulation attack. MFA reduces the chance of account manipulation but does not remove the danger completely. The attacker could also intercept the authentication traffic or steal the additional credentials. As the SSR is set higher, the Defender Team would be more likely to win if they select the effective defense action card. In the trial run, we set the SSR to 80%.

- Evaluator Algorithm

The evaluator algorithm is described briefly here. The evaluator takes the defense plan and attack plan as input. It first sorts out all the defense action cards in the defense plan assigned to an incorrect role and then discards them. Secondly, it analyzes the attack plan step by step. For each attack action card, it counts the number of mapped defense action cards in the defense plan. The higher the number, the more likely this attack action card will be blocked. The

attack action cards in the same step are considered in parallel. This means that as long as one attack action in an attack step is successful, this attack step is successful. The three attack steps are considered sequentially, which means the attack plan is successful if and only if all the attack steps are successful.

After design and implementation, the evaluator algorithm was presented to an expert from industry and academia. Two one-hour feedback and improvement sessions took place in May and June of 2021. The feedback provided by the expert was taken into consideration when designing the evaluator algorithm.

- Different Ways of Deployment

The game could also be played in person. However, theoretically, it would work similarly. The game will be presented to all the players first, and then two teams will be assigned. Each team will sit together, away from the other, and work on the defense or attack plan. When time runs up, the game master will collect the defense and attack plan and simulate the attack against the defense in front of all the players. Note that as the game was designed in 2021, the world was heavily impacted by the COVID-19 pandemic, and measures were taken to keep social distance and restrict traveling to stop the virus from spreading. Due to the traveling restrictions, we had no trial runs in person.

Initially, the game is designed to be played by two teams and, optimally, more than one player in each team, which is a multiple-player mode. With the help of an evaluator, it is possible to replace the attacking team with a non-player character (NPC). In that case, the game could also be played in a single-player mode, independent of the availability of other players, allowing the game organization more flexibility. In single-player mode, the attacker's action is simulated by a computer program based on an analysis of the defense attack matrix and the statistics from the previous trial runs.

4.1.3 Evaluation

For the evaluation of the game, we have organized three trial runs in an industry setting in 2021. In this section, the first trial run (TR 1) will be described in detail as an example, and we will compare the second and third trial run (TR 2 and 3). The results of all the trial runs will be presented and discussed.

The first three trial runs in industry are presented in this section. Details are summarized in Table 4.3. Each trial run has three to four players participating. They come from different backgrounds; some were security experts in the industry, some were students from the university. In TR 1, we also invited two observers to provide objective opinions and feedback. The players were divided into two teams: defender and attacker. All the trial runs were hosted online and took 30 minutes for each game. After the game, open discussions were held for 15 minutes with the participants to gather feedback from their experience and the game's design. The first trial run also included

Table 4.3: Details of each trial run: TR 1, 2 and 3 - Industry = ind.; University = uni.

	TR 1	TR 2	TR 3
Participants in total	3	4	4
Participants (ind.)	2	2	2
Participants (uni.)	1	2	2
Observer	2	0	0
Players in defender team	2 (ind. + uni.)	2 (ind. + ind.)	2 (ind. + ind.)
Players in attacker team	1 (ind.)	2 (uni. + uni.)	2 (uni. + uni.)
Date	4th Feb. 2021	23rd Apr. 2021	23rd Apr. 2021
Duration	30 mins	30 mins	30 mins
Online / Onsite	Online	Online	Online

two observers. These two observers are cyber security experts involved in developing serious games. They observed the game without interfering. One of them joined the defender breakout room; the other joined the attacker breakout room. They did not participate directly in the gameplay but were included to gather additional feedback on the game design.

As an example, we present the details of TR 1. Three players participated in the game. One of them is a university student with an elementary level of cyber security knowledge as required by the game. Two of them are security experts with years of experience in the field.

Participants were assigned to two teams. Each team built its defense or attack plan on an online whiteboard application prepared with the cards and game board in advance. Figure 4.1 and figure 4.2 are the final defense and attacker plan screenshots in our game environment.

The defender team assigned the card "Account Use Policy" to the asset owner. So, a meaningful account use policy is established and enforced. They also assigned the card "Application Developer Guidance" to the asset owner to reduce the number of security weaknesses in the early development phase. The defenders agreed to select "Vulnerability Scan" to track potentially vulnerable software; use "Network Segmentation" to separate critical systems in subnets and avoid lateral movement; apply "Information Encryption" to put protection on confidential data and enable "Logging & Monitoring" to keep an eye on the cloud system performance to detect anomaly behavior. These cards are assigned to the Asset Manager. The defender team made no mistake in assigning the roles, so all the defense cards chosen could contribute to the defense's success. On the attacker side, several decisions were made: in the first step, Gain Access, the attacker team chose the card "Abuse Credential" to extract credentials and opened up their attack plan. The selected attack action card, "Network Service Discovery," helped list all those vulnerable services that had not yet been shut down on remote hosts. In the second

step, Launch Attack, they played "Abuse Trusted Relationship" to find their way of compromising through a third-party provider; "Monitoring Escaping" techniques were selected to bypass detection, and "Impair Defenses" helped them to manipulate components of a victim environment maliciously. In the last step, Make Impact; the attacker team did a "Resource Hijacking," which meant they would like to take advantage of the resources of co-opted systems in the cloud and earn virtual currency by means such as bitcoin mining.

At the game's end, attack and defense plans were transmitted to the evaluator. The evaluator simulated the defenses and attacks step-by-step and displayed the intermediate result during the calculation. Finally, the overall probability of the defense plan against the attack plan was computed at 96%. This meant the defender had 96% of the chance to stop the attacker. This probability was then configured to the Wheel-of-Fortune. To everyone's surprise, even though the attacker team had only a 4% chance of winning the game, the Wheel-of-Fortune finally stopped at the slice of "Attacker wins." However, it reflects Feature 2 in section 4.1.2 in a way that no defense is 100% secure.

The chosen cards in each trial run are summarized in Table 4.4. Surprisingly, the attacker team won the game in all the trial runs.

After each trial run game, we gathered the participants and observers to share their opinions about the game in an open discussion. In general, the majority of the players agreed that the game was helpful to understanding cloud security concepts, and it was engaging playing the game. There was also constructive feedback on improvement. Below, we reveal some of the reviews we collected.

The student who participated in the TR 1 mentioned, *"I think it was pretty cool. It has some cyber security notions that I still don't really know, so I tried to see both the cheat sheet and the task. Team environment helped."* The students had limited background knowledge compared to the industrial software engineers working in the field. Yet, from this review, we understand that despite the background people can enjoy the game with the assistance of a prepared cheat sheet and benefit from teaming up with experienced ones.

One of the security experts mentioned, *"The number of rectangles you can assign to the roles represents the resource - you cannot do everything."* In the designing phase of the game prototype, we learned that there are never enough resources to implement every defense mechanism in the real world. The defenders should make decisions based on priority and consider the countermeasures individually. That implies the game design manages to reflect real-world defensive thinking to a certain extent.

Some participants in TR 2 noted, *"We can improve our strategy by repeatedly playing the game"*. By repeating and debriefing the game, the participants can familiarize themselves with the concept of cloud security and, therefore, improve their strategy.

There was also feedback such as *"Why the attack 'Impair Defenses' is covered*

by the defense 'Logging & Monitoring' is unclear to me...". This feedback shows that limited knowledge impedes the interpretation of the result. As described in section 4.1.2, the evaluator checks the card mapping of attack and defense pairs. For such feedback, we learned there is a lack of rationale for the participants, and further refinement needs to be done to the evaluator's algorithm.

4.1.4 Reflection and learning

Table 4.4 provides details of which cards are chosen in TR 1, 2, and 3 by the corresponding attacker and defender teams.

In these trial runs that we have conducted, some interesting patterns can be observed from the gathered results. Some cards are frequently chosen, such as Abuse Credential and Abuse Trusted Relationship for the attacker team.

In step 1 of the attack plan, abuse credentials almost always provide an attack surface for further actions. This might be why the card "Abuse Credentials" was always chosen. The reason for the teams choosing those cards might also be related to the participants' previous cyber security awareness training.

Abuse trusted relationship was also chosen in all trial runs for step 2. This could be because, in cloud security, we rely more on the products and services of a trusted third party, and it worries us if the trusted relationship gets abused. Playing this card can also indicate that the participants consider the insider threat important for cloud systems. Surprisingly, all the attacker

Table 4.4: Chosen cards in previous trial runs

			TR 1	TR 2	TR 3
Attack Plan	Step 1	Exploit Public-Facing Application		x	
		Abuse Credential	x	x	x
		Cloud Infrastructure Discovery			x
		Network Service Discovery	x		
	Step 2	Abuse Trusted Relationship	x	x	x
		Impair Defenses	x		x
		Infrastructure Manipulation		x	
		Monitoring Escaping	x	x	x
	Step 3	Defacement		x	
		Resource Hijacking	x		x
Defense Plan	Asset Owner	Application Developer Guidance	x	x	x
		Audit		x	x
		Account Use Policy	x	x	
	Asset Manager	Network Segmentation	x	x	x
		Application Isolation and Sandboxing			x
		Vulnerability Scan	x		x
		Logging & Monitoring	x		
		MFA		x	x
		IDS		x	
		Information Encryption	x		

teams selected the card "Monitoring Escaping" as their second step. This could be related to the participants' thinking that cloud-deployed systems are more heavily monitored than non-cloud-deployed systems. The defender implies that the attack would disable the monitoring mechanism to avoid being caught. Further investigation is needed to understand this point.

Also, on the third step of the attacker plan, Resource Hijacking was selected two times compared to Defacement, only once. We think this might be related to cloud resources being a "popular" victim of crypto-mining and distributed denial of service. However, further investigation is needed to understand the attackers' motivation.

For the defender team, Application Developer Guidance and Network Segmentation were chosen all the time. This indicates that the players believed teaching application developers to write code securely is essential, employing developer guidelines. According to our experience in the industry, this result was expected, as the participants also have an industry background. Furthermore, the importance of network segmentation was also prominent in our results. This result aligns with previous internal training in the company where the game took place, which raises awareness about network segmentation to enhance cyber security.

An unexpected result was that only the defender team selected the Information Encryption card in the first trial run. Previous incidents have shown that information placed in cloud environments can be leaked. Encrypting the information can drastically reduce the usability of the data by malicious parties. According to our experience, data in the cloud should be stored encrypted. Therefore, the results that we have collected are surprising since only the first team chose to play this card.

Our preliminary results indicate that attackers choose the following attack path: abuse credentials, abuse trusted relationships, and escape monitoring, while defenders have consistently chosen the following defense plan: application developer guidance and network segmentation.

Organizing the trial run aims to test the game logic, collect players' direct feedback, and gather new ideas for the next design iteration. The participants agreed that the game was engaging and could reflect the difficulties in implementing helpful defenses for cloud assets in real life from the feedback we collected. We want to integrate constructive feedback as a part of future work. From the designer's perspective, the trial runs showed that the game logic was reasonable. Additionally, the prepared material, including the game board, cards, and cheat sheets, fulfilled their design purposes and provided help to the players during the game. We argue that there are positive indicators that the game is adequate for the industry.

Our results give us a positive indicator of the validity and adequacy of the game for the industry. However, further work needs to be performed to collect more players' additional results and solidify our conclusions. Never-

theless, our results align with previous research in the field, particularly in the industry, and the number of participants aligns with previous empirical work. Limitations on the number of players and variations in the participants' background and experience are inherent to this industrial setting, preliminary study, and (wicked) problem.

Our results and conclusions are obtained based on participants from the industry, and no other user groups have been included up to now. Nevertheless, since the goal of the design of our game is to address industry practitioners, we do not see this as an issue.

4.2 Second design cycle - digital platform

In the second design cycle, the game design matures. In our internal discussion, the game's name is determined: Cloud of Assets and Threats (CATS). In this chapter, we describe the process and the refinement of CATS in the second design cycle. In the section 4.2.1, we introduce the problem formation and our motivation for the implemented adjustment. In the section 4.2.2 we introduce the design and implementation in this design cycle in detail. In the section 4.2.3, we share the evaluation approach and the corresponding result we collected in the evaluation phase of the second design cycle. In the section 4.2.4, we discuss the lessons learned in the current design cycle. In this design cycle, we kept the game logic. The attack scenarios are defined, enabling us to remove the role of the attackers since the attack scenarios are pre-defined. We also designed and implemented the digital platform of the game. "Wheel-of-fortune" is removed since it contributes little to the learning objective of raising awareness of cloud security issues and mitigation.

4.2.1 Problem formation

The first design cycle concluded that the game's purpose should focus on the defensive side. The players should be trained to become successful defenders. Additionally, it would be practical to implement the game logic in a digital platform to simplify the deployment and improve the scalability of the game.

In this design cycle, the following topics are addressed:

- design of a digital platform on which CATS can be played.
- a definition of six attack scenarios
- an innovative approach used to evaluate CATS.

4.2.2 Design and implementation

In this section, the design and implementation details are presented. This section consists of three parts: the attack scenarios and defense action, submission of the players, and calculation of a single success rate.

A digital platform for the game is developed as shown in Figure 4.4. It consists of a front end and a back end. The front end application is built on

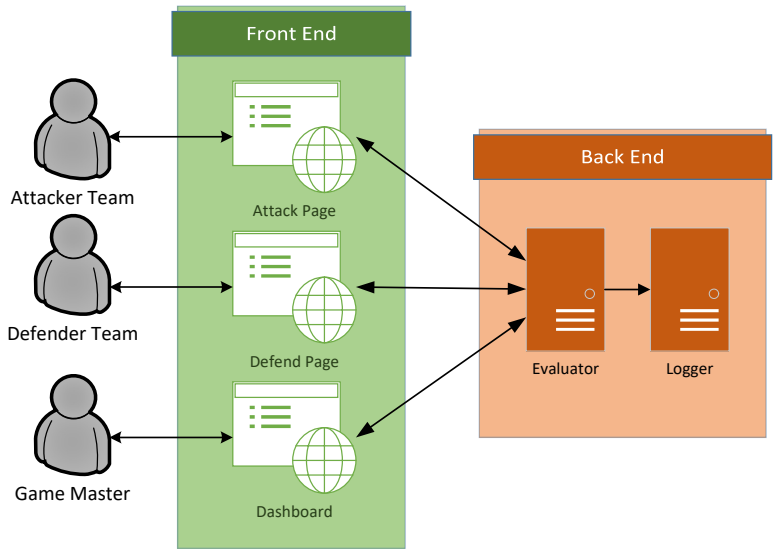


Figure 4.4: Digital Platform

Konva library (Konva, 2019). It provides a game board with cards that can be dragged around for both teams. The page is implemented with a magnetic effect so the cards can fit into the correct area. The attacker team can only access the Attack Page, and the defender team can only access the Defend Page. The Attack and Defend pages send a request to the evaluator in the back-end. Both submissions are recorded anonymously in the logger. When time is up, the game master will share the dashboard, where the output of the evaluator is displayed, as well as a visualization of the chosen defense and attack actions.

4.2.2.1 Attack scenario and defense action

The attack scenarios (AS) are derived based on real-world cyber security attacks that have occurred and were reported in practice to reflect facts in cloud security. In 2022, Koay et al. proposed SDGen to generate real-world cyber security datasets (Koay et al., 2022), which is a promising proof-of-concept. Our research uses the MITRE ATT&CK cloud matrix to acquire such information (MITRE ATT&CK, 2020e). We simplified the attack kill-chain into a three-step pattern: initial access, launch attack, and make impact. We introduce the concept of a tutorial to explain the game. The effective defense actions are based on the mitigation listed for each technique (MITRE ATT&CK, 2017d) in the cloud matrix. In table 4.5, we summarize each attack scenario's difficulty level and goal. We derived six attack scenarios in total. AS1 and AS2 are the tutorial attack scenarios, showing the impact of effective cards and correct roles. AS3 is the first attack scenario where players

must build a full defense plan without pre-selected cards. In AS4 and AS5, the difficulty is increased by raising the threshold. In AS6, a seldom-used attack, "Exploit Unused Region," increases the coverage of different attack actions and reduces repetitions from previous scenarios.

Table 4.5: The difficulty level and goal of each attack scenario

Attack Scenario	Difficulty Level	Goal
AS1	Tutorial	Show the player the impact of choosing effective defense cards.
AS2	Tutorial	Show the player the impact of assigning effective cards to the correct roles.
AS3	Elementary	Let the player build the first full defense plan without pre-selected cards.
AS4	Advanced	Increase the difficulty by raising the threshold.
AS5	Advanced	Increase the difficulty by raising threshold.
AS6	Expert	Increase defense coverage by using seldom-used cards.

4.2.2.2 Submission of the players

By hitting the "Submit" button on the game interface, the player triggers the back end to calculate the defense success rate. The submission that is sent to the back end is consolidated in a JSON format (ECMA-404, 2022). The submission data includes the chosen defense cards and their corresponding assignment to the responsibilities and roles. Each submission is captured in the back end as game dynamics data for analysis. We present a brief statistic on the captured submission data in section 4.2.3.

4.2.2.3 Calculation of Single Success Rate (SSR)

The SSR describes the quality of the submitted defense plan against the given attack scenario. The result is a percentage value that is limited between 0% to 99%. The success rate never reaches 100%, reflecting that, in reality, a perfectly secure system does not exist. Those characteristics of the SSR remain the same as those from the first design cycle, which an evaluator calculated. There are two reasons for a low success rate: 1) the defense card chosen does not mitigate the attack actions used in the attack scenario, and 2) the defense card is assigned to an incorrect responsibility, and thus, the defense cannot be performed. In this design cycle, the calculation of SSR is implemented in the web application's back end, allowing a seamless extraction of the result, which can be presented in the front end.

The game platform is implemented as a single-page web application. The front end is implemented with Konva (Konva, 2019), a JavaScript library providing the gadget necessary for the game interface, for instance, a canvas, floating images of defense cards, and a magnetic effect when the player is dragging and dropping the cards in the supposed area. We implemented

the evaluator with Python3 (Python3, 2022) in the back end. It calculates the success rate based on the presented attack scenario and the submission, then sends results and hints to the front end. The application is packed into a docker image and deployed in AWS EC2 virtual machine (Amazon Web Services, 2022). Before each game event, we prepare a new virtual machine in AWS with automated scripts, and after the game event, we collect the data and dispose of the used AWS resources.

4.2.3 Evaluation

Our evaluation in this design iteration consists of two phases. Phase 1 refers to the evaluation collected directly after game events. Phase 2 refers to the evaluation collected two weeks to one month after the game event. This section presents the design evaluation obtained during the ten game events in the industry in Phase 1 and the result obtained from the Semi Structured Interview (SSI) in Phase 2. In the first part, we show the result from game dynamics data on the correlation of the player behavior to our expectations. In the second part, we present the results collected from the questionnaire and SSI. In the third part, we share the feedback in open discussion.

4.2.3.1 Game dynamics data evaluation

The players can choose from the 24 defense cards provided during the game. Each card can be helpful or useless in defending different attack actions in the given scenario, depending on whether it is assigned to the correct role and if it defends any of the attack actions as provided in the attack scenario. We count the number of attack actions in our attack scenarios, to which the defense card is a proper mitigation. In that way, we can get a ranking of theoretically most helpful cards, as table 4.6 shows in the third column. The card "Account Management" is in the first place in the ranking in theory, which indicates it helps mitigate most of the attack actions in our scenarios. The exact mapping of defenses and attacks are presented in table 4.14 and 4.15. These tables show that "Account Management" can defend 10 out of the 16 attack actions.

In the game dynamics data, we counted the number of each card that appeared in all the valid submissions and got another "Ranking in Game" list in the fourth column of table 4.6. In the most optimal condition, assuming the players completely know which defense cards are helpful against which attack action in all scenarios, we should get the same ranking list in the third and fourth columns in table 4.6. Measuring the similarity and correlation helps to gain insight into how well the game fosters the learning process of the players. There are various ways to compare the similarity and correlation of the two ranking lists. In this work, we use Spearman's ρ (Spearman, 1904) as a way to measure the correlation of two ranking lists.

Table 4.6: The defense cards ranking in theory, game, and survey

No.	Defense Card	Ranking in Theory	Ranking in Game	Ranking in SSI
1	**Account Management**	1	7	2
2	**Network Segmentation**	2	3	1
3	**Restrict Permission**	3	11	2
4	**Logging & Monitoring**	4	1	6
5	**Asset Management**	5	12	2
6	**Filter Network Traffic**	5	8	6
7	**Password Policy**	7	2	6
8	**Audit**	7	6	11
9	**MFA**	7	13	2
10	**Critical Data Protection**	10	10	11
11	**Update Software**	10	18	17
12	**Information Encryption**	10	16	20
13	**Backup Concept**	13	14	6
14	**Application Isolation and Sandboxing**	13	9	6
15	**Vulnerability Scan**	13	15	11
16	**OS Hardening**	13	16	17
17	**IDS**	13	5	11
18	**Remove Unnecessary Feature**	13	21	22
19	**Application Developer Guidance**	19	19	22
20	**User Training**	19	20	11
21	**Account Use Policy**	19	4	11
22	**Software Configuration**	22	22	20
23	**Code Signing**	23	24	17
24	**ACP Process**	24	23	22
Spearman's ρ			0.66	0.75

4.2.3.2 Questionnaire and SSI evaluation

We distributed a questionnaire to the participants in Phase 1 directly after each game event. Based on the ten game events we organized, we have obtained 24 valid answers from 94 participants. Table 4.7 gives an overview of the questions asked and the distribution of the answers. We listed nine statements in the questionnaire in table 4.7. The respondents were asked to answer with a 5-Likert scale whether they "Strongly Disagree (- -)," "Disagree (-)," "Neutral (N)," "Agree (+)," or "Strongly Agree (++)" to the statement.

We randomly selected game participants two weeks to one month after each game event and invited them to join an SSI in Phase 2.

We present the results of questions Ph2Q2 to Ph2Q8 in figure 4.5. The blue bar shows the players' performance in the game, and the orange bar shows the percentage of the correct answer in the survey in terms of assigning the defense actions to a correct role. We see that the players perform nicely in the game and survey for some defenses such as Intrusion Detection System (IDS) and Network Segmentation. However, in some other defenses, the

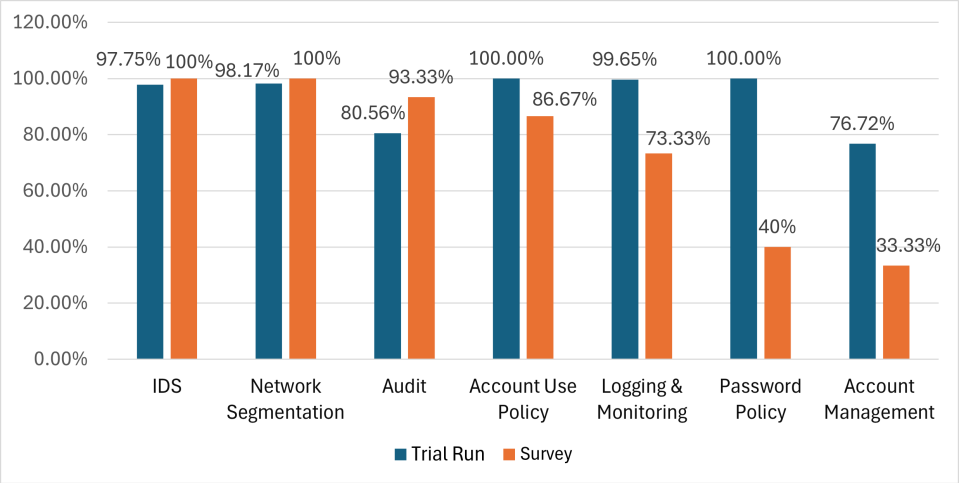


Figure 4.5: The percentage of finding correct roles for cards on the survey

Table 4.7: Questionnaire after each game event - Phase 1

No.	Questions	- -	-	N	+	++
Ph1Q1	Playing this cloud security game helps me to understand roles and responsibilities.	0%	0%	16%	63%	21%
Ph1Q2	Playing this cloud security game helps me to understand cloud attacks and defenses.	0%	0%	4%	79%	17%
Ph1Q3	I benefit from the collaboration with teammates in this cloud security game.	0%	8%	38%	29%	25%
Ph1Q4	I benefit from the discussion with teammates in the cloud security game.	0%	8%	29%	42%	21%
Ph1Q5	I feel my cloud security know-how has improved by playing this cloud security game.	0%	13%	8%	75%	4%
Ph1Q6	I would recommend this cloud security game to other colleagues.	0%	8%	4%	58%	30%
Ph1Q7	Our strategy for cloud security will improve by repeatedly playing this cloud security game.	0%	12%	29%	42%	17%
Ph1Q8	I think it is hard to calculate the actual probability of a successful defense.	0%	0%	25%	42%	23%
Ph1Q9	I think it is hard to consider all relevant factors for a successful defense.	0%	0%	21%	42%	27%

players made more mistakes in the survey at least two weeks after the game event. In general, we could see that the participants performed better during the game event and made more mistakes in the survey. The correctness rate in the survey also varies for different cards.

In Ph2Q9, we asked them to identify the helpful cards. We ranked their answer and summarized the result into the last column in table 4.6. As shown in the table 4.6, the correlation to the ranking in theory is reflected by Spearman's ρ, and the SSI value is higher than the game value. The rest of the questions are open-ended, and the result will be summarized and presented in section 4.2.4.

Table 4.8: Questions in SSI - Phase 2

No.	Question	- -	-	N	+	++	N.A
Ph2Q1	Please rate how much do you still remember from the cloud security game	0%	0%	20%	67%	13%	-
Ph2Q13	The game helped me in understanding the weakness in cloud security	13%	0%	13%	67%	7%	-
Ph2Q14	I think my cloud asset is secure.	0%	6%	7%	27%	20%	40%
Ph2Q15	I think I still need more training in cloud security.	13%	7%	27%	20%	33%	-

In table 4.6, we see that the card "Account Management" helps defend against most of the attack actions in theory. The importance of account management is sufficiently discussed in the work of Tang et al. in (Tang et al., 2022) regarding active malicious accounts detection. However, it is not the most selected card in the game, being at the seventh position in the ranking list. In the game, the most selected card is "Logging & Monitoring". This indicates that participants believe relying on logging and monitoring will improve cloud security, whereas account management contributes more to defending cloud assets. We use Spearman's ρ as a way to measure the correlation between two ranking lists.

Table 4.9: Degree of Correlation according to Spearman's ρ

Range	**Degree of Correlation**
$0<\|\rho\|<0.3$	Weak
$0.3<\|\rho\|<0.7$	Moderate
$\|\rho\|>0.7$	Strong

We refer to the table 4.9 to interpret the calculated value as proposed in (Casinillo and Tavera, 2021). Spearman's ρ ranges between "-1" and "1". The value "-1" suggests that the two compared ranking lists are negatively correlated. That is the case when one list is the reserve of the other. "0" suggests there is no correlation between the two lists. "1" suggests that the two compared ranking lists are perfectly correlated. That is the case when two lists are identical. In our case, Spearman's ρ of our expectation and the player's behavior in the game reached 0.66, which suggests a moderate correlation as shown in table 4.9. This is a positive indicator that players' performance in the game seconds our expectations. The players understand the game logic and grasp the fundamental concept of cloud security. In the last column of table 4.6, we calculated the expectation and SSI correlation. The value reached 0.75, which shows a strong correlation according to table 4.9. We take it as a positive sign that the players' understanding of cloud security defense actions has improved. One possible explanation could be, during the game, the player is learning about the defenses

and attacks, thus players might make mistakes. By correcting the mistakes, players deepen their knowledge about defenses and attacks and remember them. In the survey, when they are asked again, their answer shows more similarity to the optimal case. Since the SSI was conducted two weeks to one month after the game event, we can interpret the results as a possible indicator of the retention of knowledge and also the impact of the game on the players.

4.2.3.3 Evaluation from open discussion and open-ended questions in SSI

We list the questions in table 4.10. The table 4.8 depicts the result of questions Ph2Q1, Ph2Q13 to Ph2Q15. Note the name of the game was not decided by the time the first questionnaire was distributed, so the game was referred as "Cloud Security Game" instead of CATS. We asked the players for their

Table 4.10: Overview of questions in SSI - Phase 2

Theoretical Construct	**ID**	Questions
Perception	**Ph2Q1**	Please rate how much do you still remember from the cloud security game.
Protection	**Ph2Q2∼8**	Does the defense XXX belong to Business Responsibility or Technical Responsibility?
Protection	**Ph2Q9**	Please identify the cards that were helpful in your defense strategy (By the time of the game event).
Perception	**Ph2Q10**	What is the most important thing you learn from the Cloud Security Game?
Behavior	**Ph2Q11**	What have you changed in your daily work after the game?
General	**Ph2Q12**	Do you want to add any feedback or suggestion about the Cloud Security Game?
Perception	**Ph2Q13**	The game helped me in understanding the weakness in cloud security.
Behavior	**Ph2Q14**	I think my cloud asset is secure.
Behavior	**Ph2Q15**	I think I still need more training in cloud security.

opinions in the open discussion after each game event. In table 4.11, we present a selection of the feedback and answers to the open-ended questions. The second column represents whether the feedback is collected in Phase 1 (Ph1D) discussion or the answers to open-ended questions in Phase 2 (Ph2Q10, Ph2Q11, and Ph2Q12). In general, the comments we received were quite positive. We will discuss the feedback in more depth in the next section.

Table 4.7 summarizes the answers to the questionnaire in Phase 1. Most respondents agree that CATS helps them understand roles and responsibilities in cloud security, and know-how is improved by playing CATS. We imply that the game improves the player's perception of cloud security, and the player is

Table 4.11: Selection of representative feedback collected in Phase 1 and 2

No.	Questions	Feedback / Answer
FB1	**Ph1D**	"Thank you so much! It is possible to learn new technical vocabulary (with the game)."
FB2	**Ph1D**	"It is great to have hands-on experiences in building a cloud defense strategy! I enjoyed the game."
FB3	**Ph1D**	"Provide some explanations for both responsibilities (Asset Owner/Manager) as well as for the cards."
FB4	**Ph1D**	"Less abstraction and more context would be helpful. E.g., an architecture overview about the system under attack would be helpful."
FB5	**Ph1D**	"More time for the game."
FB6	**Ph2Q10**	"Cloud deployment is not one person responsibility but shared responsiblity."
FB7	**Ph2Q10**	"I improve my awareness"
FB8	**Ph2Q10**	"The game is too abstract to learn anything."
FB9	**Ph2Q11**	"I did't change anything."
FB10	**Ph2Q12**	"Add animations to the hints."

more aware of how to protect cloud assets. Most of them would recommend CATS to other colleagues. We interpret those answers as a positive sign that the players enjoyed CATS and could benefit from it. In question Ph1Q6, 30% of the respondents strongly agree, and 58% of them agree that they would recommend the game to other colleagues, which hints at a good design of the game. In the questionnaire, we did not get any "strongly disagree" answers to all the questions, which shows the participants received the game well.

Table 4.11 shows some feedback and answers collected in open discussion in Phase 1 and open-ended questions in Phase 2. Most are excited about using the game as a learning method (FB1). They are embracing the interactive exercises (FB2) and want to spend more time with the game (FB5). For some of them, the game is too abstract (FB8), and more concrete examples (FB4) and explanations (FB3) are needed. The players learned that cloud security is a shared responsibility (FB6) and feel their awareness of it is improved by playing the game (FB7). Some give constructive feedback on how to improve the game interface (FB10). In question Ph2Q11, we received many answers that the game did not trigger any change in their daily work (FB9) despite the increased awareness.

We want to conduct future research on the reason behind that and to improve the game further. According to the feedback and answers we collected, it is safe to conclude that the game is suitable for raising awareness of cloud security, especially the defenses versus the attacks and the roles and responsibilities. The feedback is constructed as improvements in the next design iteration.

4.2.4 Reflection and learning

In observing ten game events, the participants mostly identified the card "Account Management" as helpful in many attack scenarios. They learn about the impact of this card in the game, and in the SSI in Phase 2, they rank "Account Management" as the second most helpful card, as shown in table 4.6. This indicates that they use the game to correct their wrong understanding. Additionally, the participants seem to enjoy the game.

In Phase 2, we asked the respondents to assign certain defense cards to the correct role. The results are illustrated in Figure 4.5, which reflects what the players still remember after the game. For some cards such as "IDS" (Intrusion Detection Systems), "Network Segmentation", and "Audit", the correct rate increases in SSI of Phase 2. For cards such as "Account Use Policy", "Logging & Monitoring", "Password Policy" and "Account Management", the correct rate decreases. Surprisingly, although the participants understand the importance of "Account Management," only 33% of the participants assigned it to the correct role in the SSI of Phase 2. The card "Password Policy" has a 100% of correct rate on the role assignment during the game, however, the correct rate drops to only 40% in the SSI. There might be multiple factors that could lead to such results, e.g., daily work and chores. It might be an indicator that the game should be played more often to solidify the lessons learned, which seconds with the results of Ph1Q7 in table 4.7, almost 60 % of the participants agree or strongly agree that their defense strategy for cloud security will be improved by repeatedly playing CATS.

In this design cycle, we concretized the design elements and implementation details of CATS, and we invited 94 industrial practitioners to join the game and collect feedback from them. As the statistics and feedback show, the participants enjoyed the interactive game, and their understanding of basic concepts in cloud security was improved. We validated our design and ideas with ten game events. We provided a preliminary analysis of collected game dynamics data and proposed a measurable way to evaluate our game participants' level of understanding of basic concepts of cloud security. We used questionnaires and semi-structured interviews to collect feedback; the result is presented and discussed. Our work shows CATS has the potential to be applied as a useful artifact to raise awareness of cloud security in the industry. The level of complexity and difficulty is suitable for our target audience. However, there are still gaps between the game elements and reality, and the mapping of the defense and attacks reflects only the general aspect. The individual impact of each defense-attack mapping pair is not considered in the current design cycle. We address those opening points in the next design cycle.

4.3 Third design cycle - algorithm refinement

In this section, the third design cycle is presented. The feedback from the second design iteration is that the game is well-received, and we do not receive

feedback from the game logic or the design of the evaluator. However, we opted for a re-design of the evaluator to accommodate the explainability of the result just in case we need to explain the derivation of the calculation in the future. In this way, the game could reflect more details from reality and have a better grounding in models. We refined the details of the existing game design. In this chapter, we describe the process and the improvement of CATS in the third design cycle. In the section 4.3.1, we introduce the problem formation and our motivation for the implemented refinement. In the section 4.3.2 we introduce the design and implement in this design cycle of algorithm refinement in detail. In the section 4.3.3, we share the evaluation approach and the corresponding result we collected in the third design cycle evaluation phase. In the section 4.3.4, we discuss the lessons learned in the last design cycle. The attack scenarios and game logic remain the same in this design cycle. We refined the game element according to the information security ontology and the extension we made to support our game design. We also adapted common vulnerability scoring system calculation (CVSS) to guide the probability calculation in our core evaluator algorithm.

4.3.1 Problem formation

The second design cycle concluded that the game logic is valid and the difficulty is adequate for the players. However, in the simulation of the attack scenarios, we can integrate more elements from real-world cloud defense and attack activities and refine the evaluator algorithm further by introducing CVSS and IT security ontology. In this design cycle, the following topics are addressed:

- an extension of the IT security ontology proposed by Fenz et al. (Fenz and Ekelhart, 2009)
- a description of the evaluator algorithm based on straightforward conditional probability calculation
- an evaluation focusing on comparing the current design cycle and the previous design cycle.

4.3.2 Design and implementation

CATS is a tabletop board game played on a digital platform. In this design, the game characteristics remain the same. Players can play in teams or as single players. The game consists of six attack scenarios derived from real-world cloud security incidents. It takes approximately 60 minutes to finish one round of all six scenarios.

In one attack scenario, the attack actions of the attacker are abstracted and simulated into a three-step kill chain with different numbers of attack actions: step 1 - initial access (two attack actions), step 2 - launch attack (three attack actions), and step 3 - make an impact (one attack action). To defend a certain

attack, the players should select six defense action cards from 23 available cards and assign the selected cards to the correct responsibilities. Note that the card "ACP Process" is merged with the card "Asset Management". Starting from this design iteration, there is no card called "ACP Process".

In CATS, there are two responsibilities: business responsibility and technical responsibility. The former symbolizes the business decision that needs to be made to implement a certain defense, e.g., determine the account use policy. The later symbolizes the technical implementation, e.g., configure the segmentation of the network. If a selected card is assigned to the wrong responsibility, the card will be discarded in the evaluation process and will not contribute to the success of the defense plan. The player can assign a maximum of two cards to the business responsibility and four to the technical responsibility to strategize the defense plan.

The defense plan and the attack scenario are the inputs to the evaluator algorithm. The output is the probability of the defense's ability to withstand the attack. For each attack scenario, we assign a threshold. Suppose the algorithm's output is higher than the threshold with completely correctly placed defense cards, and all attack actions are mitigated by at least one defense card. In that case, the player solves the scenario and can move on to the next scenario. If the output probability of the evaluator algorithm is lower than the threshold or if one of the other conditions is not met, the evaluator will propose some hints, and the player can adjust the defense plan accordingly. There is no limit to the number of trials since we encourage the players to try different combinations and learn about cloud security while trying.

Note that in the game events, we never observed any attempts of brute forcing, and some players even continued trying to reach perfection after passing the threshold.

The game logic and workflow remain similar to the previous design cycle in the current design cycle. The majority of the improvement is done in the back end evaluator algorithm. We mapped the attack actions to existing vulnerabilities from the real world. The mapping of defense and attack is still derived from MITRE ATT&CK cloud matrix (MITRE ATT&CK, 2020e); however, the impact of the defense against the attack is considered individually, breaking into each CVSS metric, which enables the probability calculation to apply the CVSS scoring calculation.

4.3.2.1 Game workflow

The evaluator algorithm operates in the back end of the digital platform. It evaluates the defense plan submitted by the player in the front end and displays the calculated results to the player. The workflow of the algorithm is captured in figure 4.6. The algorithm first checks whether defense cards submitted in the defense plan are assigned to the correct responsibility. The wrongly assigned cards would be discarded from the defense plan. In the next step, the defense

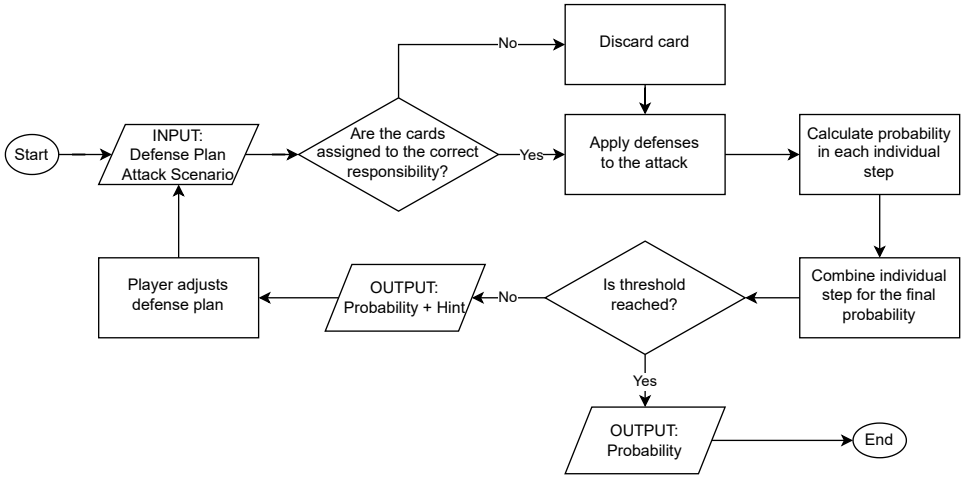


Figure 4.6: Flowchart of the evaluator algorithm

cards are applied to the attack cards based on the mapping of defenses versus attacks (see table 4.14 and 4.15) and the impact of each defense card (see table 4.12). The attack scenarios are enabled by the vulnerabilities existing in the cloud assets. We mapped each attack action to an existing CVE, and when a defense action mitigates the vulnerability, the attack is weakened, and the mapped CVE gets a lower CVSS score. As soon as all the defenses are applied, it is possible to calculate the probability in each step. Then, the intermediate results are combined to derive the final probability that a defense plan will withstand the given attack plan. If the probability reaches the pre-defined threshold, each attack card is mitigated by at least one defense card, and there is no misplaced defense card in terms of responsibilities, the final results are displayed to the player, and the scenario is solved. If the probability is lower than the threshold, an undefined attack card, or a defense card is assigned to the wrong responsibility, hints would be displayed to the player to adjust the defense plan. The player can submit again and again after adjustment.

4.3.2.2 Extension of security ontology

In this design iteration, we extended the security ontology proposed by Fenz et al. (Fenz and Ekelhart, 2009) to support the refinement of the evaluator algorithm.

In our work, we extended the security ontology and improved the core evaluator algorithm with CVSS and MITRE ATT&CK as shown in figure 4.7. We extend the green boxes to support the design of the CATS game.

- Defense Plan (*Adapted*) ⇒ In our game, the players are required to combine six *Control (Defense)* cards to build a defense plan. It is a strategy to block the given attack scenario. A full defense plan consists

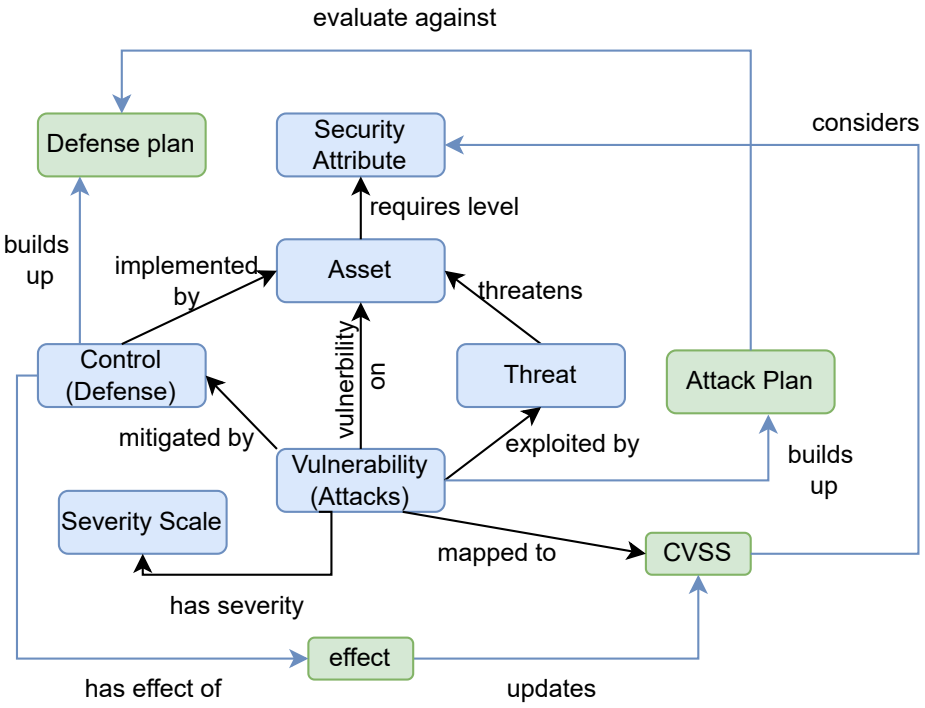


Figure 4.7: The extension of ontology overview from work of Fenz et al. (Fenz and Ekelhart, 2009)

of two cards assigned to business responsibility and four to technical responsibility.

- Attack Plan (*Adapted*) ⇒ In our work, the players are shown six different attack scenarios (attack plan), which describe the step-by-step approach and symbolize the kill chain. The attack plan consists of three steps: initial access (2 *Vulnerabilities (Attacks)*), launch attack (3 *Vulnerabilities (Attacks)*), and make an impact (1 *Vulnerability (Attack)*).
- CVSS (*Adapted*) ⇒ In our game, we use the CVSS base score to support the probability calculation in the evaluator algorithm. We map each different *Vulnerability (Attack)* to a CVSS score. Some are directly available as examples linked to existing incidents on the cloud matrix (MITRE ATT&CK, 2020e). When direct mapping is unavailable, we assign a CVSS base score with all the vectors.
- Effect (*Adapted*) ⇒ In our game, it is possible to mitigate the vulnerability by applying individual *Control (Defense)*. The *Effect* describes how the defense mitigates the vulnerability. For example, we assume there is a vulnerability Network Service Discovery

(MITRE ATT&CK, 2023) mapped to the Common Vulnerability Exposure (CVE) CVE-2020-1206 (National Vulnerability Database, 2023a). The vector in base metric of CVE-2020-1206 is: Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N with a score of 7.5 (High). We apply the *Control (Defense)* against it, whose effects are AV-1 and S-1. "AV-1" means to reduce the attack vector (AV) by one category, so the AV is updated from Network (N) to Adjacent (A). "S-1" means to reduce the Scope (S) by one category. Since the scope of the original vulnerability is Unchanged (U), which is already the lowest category and cannot be further reduced, "S-1" does not impact the given vulnerability. By applying the control, the vulnerability's vector is updated to CVSS:3.1/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N with a score of 6.5 (Medium).

By involving the mentioned elements above, we manage to 1) consider the different impact of each defense action individually 2) enhancing the real-world characteristics of the simulated CATS game without significantly increasing the difficulty.

4.3.2.3 Impact of the defense actions

As mentioned in the previous part, there is a many-to-many mapping between the defense and attack actions. When a defense action mitigates the vulnerability, the defense action impacts its CVSS vector. The mapping of the defense actions and their impact on CVSS vectors are illustrated in table 4.12. In CATS, the base metric group of CVSS vectors is used. We apply CVSS 3.1, since it is the main stream version of CVSS for now. The adaptation of CVSS 4.0 can be studied in future work. The metric values are explained as the following (National Vulnerability Database, 2023b):

- Attack Vector (AV): Network (N), Adjacent (A), Local (L), Physical (P)
  This metric reflects the context by which vulnerability exploitation is possible. This metric value (and consequently the Base Score) will be larger, and the more remote (logically and physically) an attacker can be to exploit the vulnerable component. The assumption is that the number of potential attackers for a vulnerability that could be exploited from across a network is larger than the number of potential attackers that could exploit a vulnerability requiring physical access to a device and, therefore, warrants a greater Base Score.

- Attack Complexity (AC): Low (L), High (H)
  This metric describes the conditions beyond the attacker's control to exploit the vulnerability. The Base Score is greatest for the least complex attacks.

- Privileges Required (PR): None (N), Low (L), High (H)
  This metric describes the privileges an attacker must possess before successfully exploiting the vulnerability. The Base Score is greatest if no privileges are required.
- User Interaction (UI): None (N), Required (R)
  This metric determines whether the vulnerability can be exploited solely at the attacker's will or whether a separate user (or user-initiated process) must participate somehow. The Base Score is greatest when no user interaction is required.
- Scope (S): Unchanged (U), Changed (C)
  The Scope metric captures whether a vulnerability in one vulnerable component impacts resources in components beyond its security scope. The Base Score is greatest when a scope change occurs.
- Confidentiality (C): High (H), Low (L), None (N)
  This metric measures the impact on the confidentiality of the information resources managed by a software component due to a successfully exploited vulnerability. The Base Score is greatest when the loss to the impacted component is highest.
- Integrity (I): High (H), Low (L), None (N)
  This metric measures the impact on the integrity of a successfully exploited vulnerability. Integrity refers to the trustworthiness and veracity of information. The base score is the greatest when the impact component's consequence is highest. When the integrity of information is deemed trustworthy, the data has not been tampered with or altered unauthorizedly. It is highest when the consequence to the impacted component's integrity is greatest.
- Availability (A): High (H), Low (L), None (N)
  This metric measures the impact on the availability of the impacted component resulting from a successfully exploited vulnerability. The Base Score is greatest when the consequence to the impacted component is highest.

In the table 4.12, there are two different signs for each vector: either "-1" or "x0". The sign "-1" means reducing the mapped vector by one value in the direction of reduction on the CVSS Base Score. The sign "x0" means to set the mapped vector directly to the lowest possible. The defense measures are so strong that they greatly minimize or eliminate the potential impact on a certain vector of the system or component.

For example, we assume there is an attack action Network Service Discovery (MITRE ATT&CK, 2017c) mapped to the Common Vulnerability Exposure (CVE) CVE-2020-1206 (MITRE ATT&CK, 2020b), as

Table 4.12: Part 2: Mapping of defense actions and their impact on CVSS vectors

No.	Defense Action	AV	AC	PR	UI	S	C	I	A
1	Application Developer Guidance		-1						
2	Password Policy			-1	-1				
3	Audit		-1						
4	User Training		-1						
5	Asset Management		-1					-1	
6	Critical Data Protection						-1	-1	-1
7	Backup Concept							-1	-1
8	Account Use Policy		-1						
9	Network Segmentation	-1				-1			
10	Application Isolation and Sandboxing	-1				-1			
11	Update Software		-1						
12	Vulnerability Scan		-1						
13	Account Management		-1	-1	-1				
14	Logging & Monitoring				-1				-1
15	MFA				-1				
16	OS Hardening		-1						
17	IDS		-1						-1
18	Code Signing							x0	
19	Restrict Permission		-1	-1					
20	Software Configuration		-1						-1
21	Information Encryption						x0		
22	Filter Network Traffic								-1
23	Remove Unnecessary Feature	-1	-1						

suggested in table 4.13. The vector in base metric of cve-2020-1206 is: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N with a score of 7.5 (High). We apply the defense action Network Segmentation (No. 9 in table 4.12) against it, whose impacts are AV-1 and S-1 according to table 4.12. The sign "-1" in the column of AV means to reduce the attack vector (AV) by one value, so the AV is updated from Network (N) to Adjacent (A). The sign "-1" in the S column means reducing the Scope (S) by one value. Since the Scope of the original CVE is Unchanged (U), which is already the lowest value, it cannot be further reduced. The second impact does not substantially impact the given attack action. By applying Network Segmentation, the vulnerability's vector is updated to CVSS:3.1/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N with a score of 6.5 (Medium).

4.3.2.4 Mapping of attacks and common vulnerabilities and exposures (CVE)

As mentioned in previous sections, in CATS, we designed six attack scenarios. Each attack scenario simulates a step-by-step attack consisting of six attack actions. The attack actions are possible due to the vulnerabilities that might

exist in cloud assets or simply due to a misconfiguration. We endeavor to map each attack action to an existing vulnerability in the first place. When there are no suitable ones to take, we derive the CVSS Vector of our own for attack action No. 3, 7, 12, and 13 in table 4.13. For No. 3 "Abuse Credentials," it is a severe attack. Once the attacker can impersonate any user in the system, the attack immediately gains all privileges. Therefore, we rated it according to the worst-case scenario. The same holds for No. 7 "Account Manipulation". For No. 12 "Exploit Unused Region", the attacker would need privilege to migrate to other region, therefore it requires "Low" privilege as we have "PR:L". The impact on confidentiality, integrity and availability is limited, therefore we rated "C:L/I:L/A:L". For No. 13, "Monitoring Escaping," all situations are the same as No. 12, except that privilege is not required in the worst-case scenario. We take the CVE vectors as input for the next step of calculation. Table 4.13 maps the attack actions to the CVEs. The definition

Table 4.13: Mapping of attacks and CVEs

No.	Attack Action	CVE Mapping	CVSS Base Score	CVSS Vector (CVSS 3.1)
1	Brute Force (MITRE ATT&CK, 2017b)	(ICS-CERT, 2020)	9.8 CRITICAL	AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
2	Exploit Public-Facing Application (MITRE ATT&CK, 2018a)	(MITRE ATT&CK, 2019a)	9.8 CRITICAL	AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
3	Abuse Credential(MITRE ATT&CK, 2017e)	-	10.0 CRITICAL	AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H
4	Cloud Infrastructure Discovery (MITRE ATT&CK, 2020a)	(MITRE ATT&CK, 2021)	9.8 CRITICAL	AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
5	Network Service Discovery (MITRE ATT&CK, 2017c)	(MITRE ATT&CK, 2020b)	7.5 HIGH	AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N
6	Abuse Trusted Relationship (MITRE ATT&CK, 2018b)	(Apple Inc., 2020)	5.5 MEDIUM	AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N
7	Account Manipulation (MITRE ATT&CK, 2017a)	-	10.0 CRITICAL	AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H
8	Implant Container Image (MITRE ATT&CK, 2019e)	(MITRE ATT&CK, 2019b)	8.6 HIGH	AV:L/AC:L/PR:N/UI:R/S:C/C:H/I:H/A:H
9	Cloud Storage Breach (MITRE ATT&CK, 2019c)	(Oracle, 2021)	10.0 CRITICAL	AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H
10	Impair Defenses (MITRE ATT&CK, 2020d)	(Microsoft Corporation, 2020)	7.8 HIGH	AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H
11	Infrastructure Manipulation (MITRE ATT&CK, 2019f)	(Jenkins Project, 2019)	4.3 MEDIUM	AV:N/AC:L/PR:N/UI:R/S:U/C:N/I:L/A:N
12	Exploit Unused Region (MITRE ATT&CK, 2019i)	-	7.4 HIGH	AV:N/AC:L/PR:L/UI:N/S:C/C:L/I:L/A:L
13	Monitoring Escaping (MITRE ATT&CK, 2019h)	-	8.3 HIGH	AV:N/AC:L/PR:N/UI:N/S:C/C:L/I:L/A:L
14	Defacement (MITRE ATT&CK, 2019d)	(National Vulnerability Database, 2022)	6.1 MEDIUM	AV:N/AC:L/PR:N/UI:N/S:C/C:L/I:L/A:L
15	Denial of Service (MITRE ATT&CK, 2020c)	(Siemens AG, 2021)	9.8 CRITICAL	AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
16	Resource Hijacking (MITRE ATT&CK, 2019g)	(Atlasian, 2022)	9.8 CRITICAL	AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

of each attack action can be found in the references in the table from the MITRE ATT&CK cloud matrix. A summary can be found in section 4.1.2. For example, the first attack action, Brute Force, is an attack technique that the adversaries may use to access accounts when passwords are unknown or password hashes are obtained. Without knowing the password for an account or set of accounts, an adversary may systematically guess the password using a repetitive or iterative mechanism (MITRE ATT&CK, 2017b). We mapped this attack action to CVE-2020-14494. This vulnerability is due to a flaw

in an authentication mechanism that does not provide sufficient complexity to protect against brute force attacks, which may allow unauthorized users to access the system after no more than a fixed maximum number of attempts (ICS-CERT, 2020). CVE-2020-14494 is a critical vulnerability, and the CVSS vector is CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H with a score of 9.8 (Critical) out of 10.0. We applied the CVSS version 3.1 (National Vulnerability Database, 2023b) for all the mapped vulnerabilities.

For attack actions 3, 7, 12, and 13, no vulnerability can be mapped to the attack action because those actions are due to misconfiguration. However, to ensure the completeness of the mapping, we manually determined the CVSS vectors.

4.3.2.5 Mapping of attack and defense actions

Multiple defense actions can defend one attack action, and one defense action can mitigate multiple attack actions. This many-to-many mapping is illustrated in MITRE ATT&CK cloud matrix (MITRE ATT&CK, 2020e). In CATS, we derive the mapping between the attacks and defenses we use in the simulation as shown in table 4.14 and table 4.15. In the header of the table, we show 16 attack actions. In the first column, there are 23 defense actions. The "x"s in table 4.14 and table 4.15 show which attack action can be mitigated by the defense action in the first column. For instance, the defense "Application Developer Guidance" refers to the defense that the developers should be trained to develop applications securely. This defense helps prevent the attack "Abuse credential" by avoiding the misuse of debugging and development access to the application. However, the impact is limited, and it only increases the complexity of the attack.

4.3.2.6 Algorithm walk-through of an example

In this section, we walk through the algorithm with a concrete example. In this example, the attack applies "Abuse Credential", "Cloud Infrastructure Discovery" in step 1 to gain access. In step 2, the attacker uses the attack action (AA) "Abuse Trusted Relationship", "Monitoring Escaping" and "Impair Defense" to launch the attack. In step 3, the attacker conducts "Resource Hijacking" to make an impact. This is shown in Figure 4.8.

Within one step, attack actions are considered in parallel, which means the attack success rate for this step will only be reduced to 0%, if all attacks within this step are mitigated to 0%. If there is one undefended attack with 100% of attack success rate, the attack success rate would be 100%. That case symbolizes that the attack finds an open vulnerability and applies an exploit, bypassing this step without any trouble. We assume the normalized percentage of an attack m in step n is p_{nm}, then the attack success rate of step n p_n is:

Table 4.14: Mapping of attacks and CVEs (Part 1)

	Brute Force	**Exploit Public-Facing Application**	**Abuse Credential**	**Cloud Infrastructure Discovery**	**Network Service Discovery**	**Abuse Trusted Relationship**	**Account Manipulation**	**Implant Container Image**	**Cloud Storage Breach**	**Impair Defenses**	**Infrastructure Manipulation**	**Exploit Unused Region**	**Monitoring Escaping**	**Defacement**	**Denial of Service**	**Resource Hijacking**
Application Developer Guidance			x													
Password Policy	x		x										x			
Audit			x					x	x		x					
User Training			x													
Asset Management				x				x		x	x	x				
Critical Data Protection									x				x			
Backup Concept														x	x	
Account Use Policy	x															
Network Segmentation		x			x	x	x									
Application Isolation and Sandboxing		x														
Update Software		x	x													
Vulnerability Scan		x														
Account Management	x		x	x		x	x	x	x	x	x		x			
Logging & Monitoring										x						x
MFA	x						x		x							
OS Hardening			x				x									
IDS					x											

Table 4.15: Mapping of attacks and CVEs (Part 2)

	Brute Force	**Exploit Public-Facing Application**	**Abuse Credential**	**Cloud Infrastructure Discovery**	**Network Service Discovery**	**Abuse Trusted Relationship**	**Account Manipulation**	**Implant Container Image**	**Cloud Storage Breach**	**Impair Defenses**	**Infrastructure Manipulation**	**Exploit Unused Region**	**Monitoring Escaping**	**Defacement**	**Denial of Service**	**Resource Hijacking**
Code Signing								x								
Restrict Permission			x						x	x						
Software Configuration												x				
Information Encryption			x						x							
Filter Network Traffic									x				x		x	
Remove Unnecessary Feature					x											

$$p_n = 1 - \prod_{1...m}(1 - p_{nm}).$$

The defense success rate of step n is:

$$p_{n\_defense} = \prod_{1...m}(1 - p_{nm}).$$

Then, we consider the three steps in serial order. If an attacker attacks the cloud asset in a three-step approach, he or she must succeed in all the steps to finally compromise the cloud asset. Therefore, if we assume the final attack success rate is p, then:

$$p = \prod_{1...3}(p_n).$$

The final defense success rate p_{final} can be derived:

$$p_{final} = 1 - p.$$

Following this logic, we can derive the values in the table 4.17 on the left. The defense actions are shown in the left part in figure 4.8, and the mapped CVEs before and after mitigation are summarized in table 4.16. We assume those cards are assigned to the correct role; therefore, the impact remains

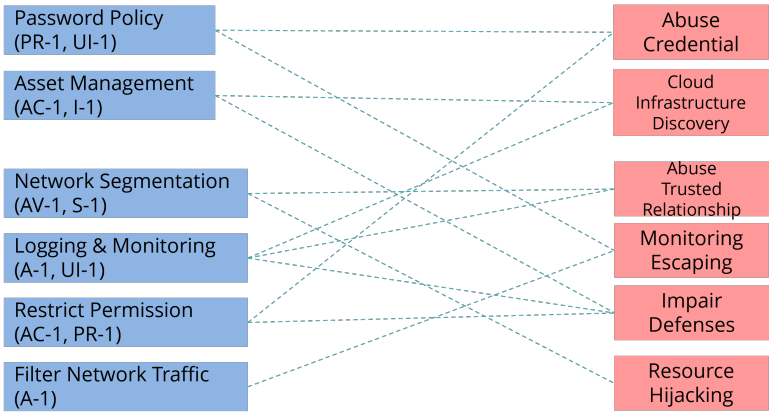


Figure 4.8: Mapped defense and attack action in example

valid. After applying the defense actions, as shown in table 4.16, the CVSS base score of each attack action drops. The updated values are captured in the table 4.17 on the right. After applying the calculation above again based on the new values, we derive that the defending probability has increased from 3% (before mitigation) to 34% (after mitigation). This is still not good enough to solve the scenario. The back end generates hints that would further improve the defense's success rate and propose hints for the player to improve the quality of the defense plan and finally reach the threshold.

Table 4.16: The mapped CVE vectors before and after mitigation

	Before mitigation	After mitigation
Abuse Credential	10.0 AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H	7.6 AV:N/AC:H/PR:H/UI:R/S:C/C:H/I:H/A:H
Cloud Infrastrucure Discovery	9.8 AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H	6.4 AV:N/AC:H/PR:N/UI:R/S:U/C:H/I:L/A:L
Abuse Trusted Relationship	5.5 AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N	4.1 AV:P/AC:L/PR:L/UI:R/S:U/C:H/I:N/A:N
Monitoring Escaping	8.3 AV:N/AC:L/PR:N/UI:N/S:C/C:L/I:L/A:L	5.4 AV:N/AC:L/PR:L/UI:R/S:C/C:L/I:L/A:N
Impair Defense	7.8 AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H	5.1 AV:L/AC:H/PR:H/UI:R/S:U/C:H/I:L/A:L
Resource Hijacking	9.8 AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H	8.3 AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:L

4.3.3 Evaluation

In this section, we present the method we use to evaluate the implemented algorithm. Then, we will show the collected results and discuss them.

4.3.3.1 Evaluation Method

We evaluate the new algorithm by applying it in game events and comparing the statistics of the new algorithm and the previous version of CATS. At

Table 4.17: Algorithm walk-through: before mitigation on the left, after mitigation on the right.

	Success rate for the attacker (single AA)			Intermediate probability
Step 1	100%	98%	-	100%
Step 2	55%	83%	78%	98%
Step 3	98%	-	-	98%
Defending probability				3%

	Success rate for the attacker (single AA)			Intermediate probability
	76%	64%	-	91%
	41%	54%	51%	86%
	83%	-	-	83%
Defending probability				34%

the end of each game event, we also invited the participants to an open discussion to exchange ideas and collect feedback in the form of SSI. Table 4.18 summarizes the organized game events. The evaluation of the improved algorithm is based on two game events (No. 11 and 12 in table 4.18) with 24 players in total. As mentioned in the previous section 4.3.2, CATS can be played as single players or in teams. The table 4.18 captures the basic information about each game event in our evaluation.

Table 4.18: Overview of game events organized in 2022-2023

Game Event	Date	Player	Team	Algorithm	Valid Submissions
1	2022-01-21	17	4	Version 1	177
2	2022-03-15	14	-	Version 1	477
3	2022-03-22	14	-	Version 1	493
4	2022-03-29	13	-	Version 1	312
5	2022-04-14	13	4	Version 1	178
6	2022-04-26	11	-	Version 1	100
7	2022-05-03	8	-	Version 1	171
8	2022-06-02	4	2	Version 1	169
9	2022-09-29	14	4	Version 1	298
10	2022-11-16	15	4	Version 1	147
Total number of players		123			
Total number of submissions		2522			
11	2023-03-22	14	-	Version 2	538
12	2023-04-19	10	-	Version 2	337
Total number of players		24			
Total number of submissions		875			

4.3.3.2 Evaluation Result

In table 4.19, the comparison of the algorithm before and after the improvement. Comparing the players' performance before and after the algorithm refinement, we observed that with the new algorithm, the players took fewer submissions per scenario to solve the scenario. The players' average time in each scenario drops slightly in the new algorithm. On average, the players tend to improve more in increasing the defense success rate in each submission in the new algorithm. However, players seem to make more mistakes in the role assignment per scenario in the game events with the new algorithm.

Table 4.19: Game event statistics: comparing the improved algorithm to the original algorithm

	Ver. 1 algorithm	**Ver. 2 algorithm**
Average no. of submissions per scenario	7.85	7.30
Average time to solve each scenario	5 min. 45 sec.	5 min. 18 sec.
Average improvement in each submission	5.21	6.55
Mistakes in role assignment per scenario	2.11	3.04

Table 4.20: Game event feedback: quote from the participants

ID	**Feedback**	**Quote**
1	Positive	Very nice and well-developed game.
2	Positive	Enjoyed. More interesting (than training without serious games).
3	Positive	It is great that the exercises are wide spread, so that also non-developer (application owner + tester) could be of help and had fun.
4	Positive	Very interesting activity! Thank you so much! It is possible to learn new technical vocabulary.
5	Positive	It is great to have hands-on experiences in building a cloud defense strategy! I enjoyed the game.
6	Negative	More time for the game.
7	Negative	For each measure selected, a description of its influence should be given after submission (for the player) to profit from playing.

4.3.4 Reflection and learning

In general, as we can see from table 4.20, the participants enjoy the game and find it helpful in improving their know-how in cloud security: *"It is great to have hands-on experiences in building a cloud defense strategy!"*. CATS provides a possibility to simulate the attacks and quantify the validity of the mapped defenses with the help of CVSS. Therefore, we fulfilled our intention of introducing more elements from reality to refine the explainability of the evaluator result and to make the whole game logic fit for future extension of further game scenarios. The players take less time to solve the scenarios, which implies the gap between their cognition and the simulation is reduced. The players are

motivated to solve the challenges and agree with the game's logic. Comparing the original primitive algorithm and the new algorithm, the players spend approximately equal time solving each scenario and making approximately equal submissions. This implies that by improving the algorithm, the changes did not introduce additional difficulty in the game. The difficulty level for the game participants remains the same before and after the algorithm refinement.

By involving the mentioned elements above, we manage to 1) consider the different impact of each defense action individually 2) enhancing the real-world characteristics of the simulated CATS game without significantly increasing the difficulty.

The design cycle is conducted under the guidance of the design science paradigm, which is an approach that combines scientific research with practical design to create innovative solutions to real-world problems. As for the evaluation, we organized two game events with industry practitioners. The collected result shows that the players enjoyed the game and found it very helpful in raising awareness about the challenges in cloud security and exercising the skills for designing a successful defense strategy.

The players react similarly to the improved algorithm compared to the primitive one. This implies that in the refinement of the core algorithm for simulation, the difficulty level for the game remains similar while more details, in reality, are reflected in the game.

4.4 Chapter summary

In this chapter, we introduce the three design cycles in depth. In each design cycle, we introduced problem formation, design and implementation, evaluation and reflection, and learning. The first design cycle is called the initial design. In the initial design, CATS is intended to be a tabletop card game that can be played online with both a defender and attacker teams. We organized three trial runs in an industry setting to verify the game's feasibility and collect feedback. Based on the feedback we collected on the initial design, we adapted CATS to a digital platform and entered the second design cycle: digital platform. In this cycle, we conducted ten game events with 123 industrial practitioners. In this cycle, we use a simulator to simulate the attacker in 6 different attack scenarios and design an evaluator algorithm to calculate the step-by-step success rate based on the defense and attack. We conducted a survey, open discussion, and semi-structured interviews for the evaluation. We entered the next design cycle by refining the algorithm based on the feedback. In this design cycle, we mapped each attack to an existing vulnerability and quantified the conditional probability calculation of the success rate according to the CVSS calculation. For evaluation, we conducted two game events with 24 industrial practitioners and a survey and open discussion. The feedback suggests that the game is improved without adding too much difficulty.

The players enjoyed the game and achieved the learning objectives in

the evaluation. With the data collected from all three design iterations, we conclude that CATS is playable and useful and has become an established part of the internal training curriculum. Our research shows how to solve the relevant problem, and therefore, research activities have been successful and can be stopped. Eventually, the game will need further development, i.e., when user interface expectations change or new cloud security regulations and best practices emerge.

5

CATS: the game

In this chapter, we provide a detailed description of CATS towards the end of our study. CATS results from previous design cycles and has been validated by multiple game events. We introduce the background, game modes, process, interface, and winning conditions.

Note that the earlier version of the game interface was first published in ICPEC (T. Zhao, Gasiba, et al., 2021a) and then refined in the Journal of Information (T. Zhao, Gasiba, et al., 2021b).

5.1 Game background

In the first brainstorming sessions, the topic "Cloud Security" was identified, and the format of the game should be a board game that can either be played face-to-face or as a virtual session to accommodate the restrictions imposed by the COVID crisis. It was also determined that the insider perspective of non-compliance with security policies should be considered.

Cloud security provides several elements that could be built into a board game: 1) Cloud security is a constant fight between defender and attacker. 2) For both the defender and attacker, some resource constraints exist. 3) Defenders might have different responsibilities determined by their cloud security role. 4) Attackers could take cloud-specific attack actions to take down cloud assets. Using the elements above together, we can build a board game prototype geared to help trainees better understand cloud security.

In the design phase, the following core features are identified. The game prototype aims to address the following features:

Feature 1: Cloud Security Kill Chain As Assante et al. mentioned in (M. and Lee, 2015), Cyber attackers do not target systems in a single

incident and breach. Attackers in cloud security incidents plan the attack step by step. In the game, the attack element should consist of at least several phases instead of a "single shot."

Feature 2: 100 percent security does not exist. Due to resource constraints, we never aim to achieve 100 percent security in products and solutions. A correct countermeasure does not remove the threat completely. For example, a strong password policy is an effective countermeasure to abusing credentials. A strong password policy as a countermeasure does not eradicate the threat but lowers the risk. Therefore, the defending element should not guarantee 100 percent security in the game.

Feature 3: Defense-in-depth helps. Using a defense-in-depth strategy, as Kuipers et al. advise (Kuipers and Fabro, 2006), in general, improves protection against cyber threats. Due to the uncertainty of the attacker's move, a defense-in-depth strategy covers more possible attacks and should be encouraged in the game. The game prototype should address defense-in-depth strategy, too.

5.2 The game mode

CATS has four game modes through three design iterations: multiple-player mode, single-player mode, user mode, and team mode.

- Multiple player mode

The multiple-player mode comes from the first design iteration. In this game mode, the players can join the game as either defenders or attackers. This mode focuses on both defensive and offensive aspects. During the game, both the defender and the attacker are given a certain amount of time to build an attack or defense plan. When time runs up, the game master coordinates with both teams and evaluates the defense plan against the attack plan. The evaluator in the back end outputs the probability, and the result determines the size of the area in a "wheel-of-fortune": "Attacker wins!" vs. "Defender wins!".

The idea of "wheel-of-fortune" adds randomness and playfulness to the game. The participants appreciated it well. However, it does not contribute to the learning objective; therefore, this game mode is not further developed in the second and third design iteration.

- Single player mode

In the single-player mode, the player can only join as a defender. It is the same as the multiple-player mode except that the attack plan is fixed and pre-defined.

This game mode is also designed as a proof-of-concept to verify whether we can discard the offensive aspect of the game and focus completely on the defensive aspect. The success of the single-player mode led to the design of the six attack scenarios in the next two game modes.

- User mode

In user mode, we focus on the defensive aspect, and players can join as single players. We designed six attack scenarios for the player to choose from. The first and second scenarios are tutorial scenarios. In the first scenario, the player only needs to select one more defense action card to complete the defense plan. In the second scenario, the player learns about the roles and responsibilities. There are two cards placed in each other's responsibility. The player is supposed to swap them and submit the swapped defense plan to the back end. In the other four attack scenarios, the player builds the defense plan.

User mode is evaluated in the second and third design iterations. In the typical use case, after a full-day training session, including cloud security, the game is introduced to the participants, who join it individually.

- Team mode

In team mode, the participants join as teams. The rest is similar to user mode. Team mode uses the game mechanism of CyberSecurity Challenges (CSC) (Gasiba, 2021). CATS is deployed as a category of challenges in a full-day CSC event. If the team successfully solved the challenge, a flag that looked like a random string would appear. By copying this string to the CSC dashboard, the team wins 1500 points per scenario. Team mode is evaluated in the second and third design iteration.

5.3 Game process

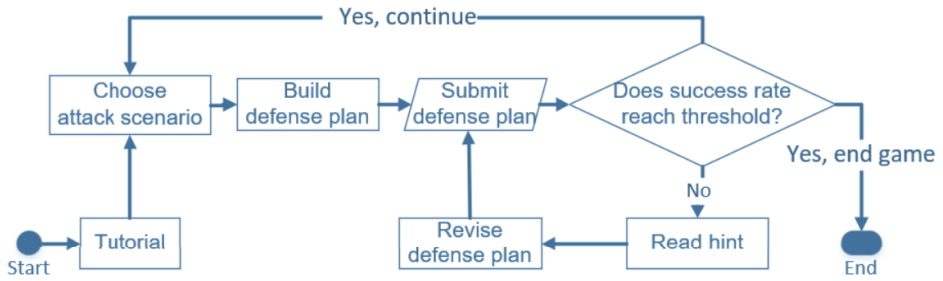


Figure 5.1: The game process from player perspective

The flowchart in figure 5.1 shows the game process. As the game starts, the players follow a tutorial to learn about the rules and game elements. The players are free to choose from the available attack scenarios. Details about the attack scenario are provided in the next section. During the game event, we offer the players two attack scenarios for the tutorials and four attack scenarios to be solved. We will introduce the details of attack scenarios in the next sections. The game aims for players to build a defense plan by assigning

defense cards to the correct responsibility. When the defense plan is ready, players submit it to the back end by clicking a "submit" button. The back end then evaluates the chances of the cloud deployment being attacked based on the scenario, the players' selected cards, and their positions. The evaluator calculates a success rate: the probability that the submitted defense plan withstands the given attack scenario. The game is pre-configured with a threshold visible to the player in the game interface. If the calculated success rate is bigger or equal to the given threshold, the player successfully solves the scenario and can move on to the next one. Hints are automatically generated and sent back to the player if the success rate does not reach the threshold. These hints justify the player as to why the card selection did or did not work. At this stage, the player can further adjust the defense plan based on the received hints. The player can change the defense plan and submit the new plan to the back end until the game scenario is solved.

5.4 Game interface

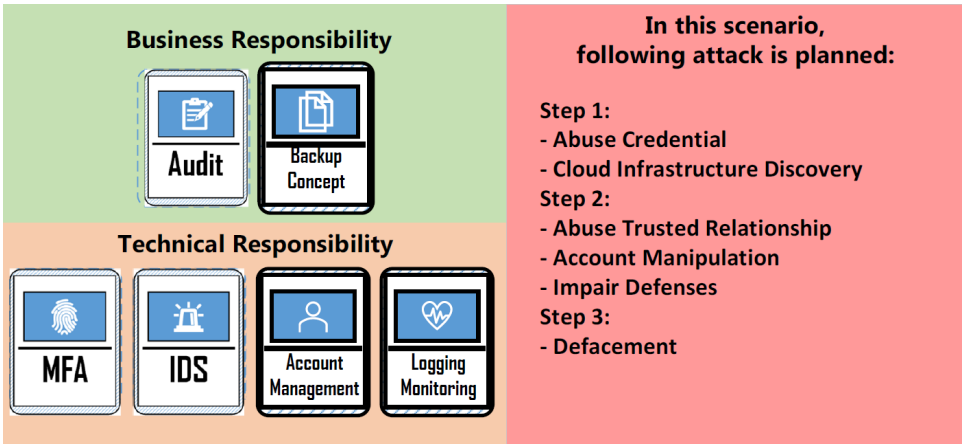


Figure 5.2: Illustrative example of the game design elements

Figure 5.2 depicts an example of the game interface. Depicted on the left are the six chosen cards assigned to either the "Business Responsibility" or "Technical Responsibility" area. "Business Responsibility" refers to the high-level defense actions in which important business-related decisions should be made, typically by the asset owners. "Technical Responsibilities" refers to the concrete technical defense actions typically implemented by the asset manager. On the right side, the attack scenario is listed with three steps: step 1, initial access; step 2, launch attack; and step 3, make impact. In the first step, the attack "Abuse Credential" can be mitigated by the defense card "Audit"; the attack "Cloud Infrastructure Discovery" can be defended by "Account Management." In the second step, attacks "Abuse Trusted Relationship" and "Account Manipulation" can also be secured with the defense

"Account Management." The defense card "Logging & Monitoring" can detect "Impair Defense." In the last step, the defense card "Backup Concept" can alleviate the attack "Defacement." There are, in total, 24 defense cards for the players to choose from. In this example, all the attack actions are defended by at least one defense card, which results in a defense success rate of 98% according to the evaluator. If the threshold is set to 95%

5.5 Wining condition

To win the game, three conditions need to be fulfilled at the same time.

- The calculated probability of the defense to withstand the given attack must be higher than the pre-defined threshold.
- All the attack actions included in the scenario must be mitigated by at least one defense card.
- All the selected defense cards must be assigned the correct responsibility.

If any of the conditions listed above are not fulfilled, a hint will be displayed in the hint area of the page, and the player can adjust the defense plan accordingly until all three conditions are fulfilled.

5.6 Chapter summary

This chapter is a brief paragraph on CATS, which results from three design cycles, as presented in the previous chapter. The game reflects three features: 1) cloud security kill chain, 2) 100 percent security does not exist, and 3) defense-in-depth helps. CATS supports different game modes. It can be deployed as a standalone game, as part of a training program, or as a category of challenges in a CSC event. The game process remains unchanged during the three design iterations: the players follow the tutorial and are free to choose from different attack scenarios. They are encouraged to try different combinations and revise their defense based on the hint provided by the back end. The game interface experienced minor changes to highlight the most important messages to the players. The winning condition is threefold: 1) the calculated success rate of the defender to withstand the attack must be higher than the pre-defined threshold, 2) All attack actions should be mitigated by at least one defense card, and 3) All selected defense cards must be assigned to the correct responsibility.

This chapter summarized the current status of CATS after all the design iterations.

6

Road map to implementing cyber security awareness programs on cloud security with CATS in the industry

This chapter presents the road map to implementing a cyber security awareness program on cloud security with CATS in the industry. Our goal is to show what needs to be done to adapt CATS to another organization, which includes adapting the standards and requirements mapping and integrating them into the training portfolio accordingly. This chapter focuses on the steps beyond the serious game design itself.

In this chapter, we describe the road map to the successful serious game CATS that helps to raise awareness about cyber security in the industry. Researchers, cyber security trainers, or cyber security experts design and implement an awareness program using CATS. CATS works as an instrument to engage trainees and keep participants focused in online, onsite, and hybrid training.

We selected three topics that are relevant to the road map. The first topic is the case study we conducted to identify the necessary preconditions to implementing a serious game into an organization's training program. The first topic enables the execution of the road map. The second topic is the concrete integration possibilities into the existing training framework of the organization. The second topic refines the road map. The third topic

describes the integration into a hybrid work environment, which is –after the COVID-19 pandemic– a relevant topic for organizations.

Note that the earlier version of the considerations towards implementing CATS in an organization has been published in the Journal of Systems and Software (T. Zhao, Gasiba, et al., 2024).

6.1 The conditions to make successful awareness programs in industry

This section presents the necessary preconditions to initiate our study, summarized in a case study (Gasiba, 2021) on a project to increase the levels of security in software engineering products and solutions.

Note that the case study marks the beginning of the research of CyberSecurity Challenges by Dr. Tiago Gasiba (Gasiba, 2021). Our journal paper (T. Zhao, Gasiba, et al., 2024) refined the case study and chapter is, again, a refinement of our previous work. This section highlights the two versions of the case study, presents the efforts to work with the organizations in which our game is implemented, and presents a road map of how to implement the serious game CATS in other organizations seeking to raise their cyber security awareness.

The original case study was conducted in 2019 at an international company that provides software and cyber-physical systems for critical infrastructure. The study started with an initiative to update the company's secure coding guidelines to reflect the latest advances in technical know-how and industry best practices. We followed the basic steps to identify what is relevant for cloud security.

The main outcome of this project was a set of secure programming guidelines. All stakeholders approved the new secure coding guidelines (Gasiba et al., 2020), baselined them and deployed them as an internal policy. The case study identifies the project's success factors. It has shown the need to develop and update the existing training content and extend the interpretation of secure coding guidelines to cloud security.

6.1.1 Motivation and trigger of activities

The case study is written from the perspective of a security expert. It focuses on a small department in a multi-national organization that delivers software to other parts of the organization, which then provides software and cyber-physical systems to critical infrastructure.

Software used in critical infrastructures must be secure and implemented following modern product and service methods. Software companies delivering software for critical infrastructures must meet specific IT security standards that mandate a secure software development lifecycle. When software is to be part of a critical infrastructure, both security and safety are highly relevant. Throughout the industry, there is a strong push to introduce measures that produce high-quality, reliable, and secure software and secure the deployed

cloud asset.

The media and security professionals have been calling attention to software vulnerabilities and the fact that software developers keep making the same programming mistakes (Connory, 2023; Darling, 2023; Poston, 2023; Schneier, B., 2023; Vaughan-Nichols, 2023). Acknowledging this factor and the security requirements imposed by standards, the company started an improvement program initiative to address the current status quo of software development and improve its current guidelines, processes, and documentation.

Since the department had a secure software development lifecycle, higher management decided to address the baseline documentation of secure software development first. This baseline needed to be updated.

6.1.2 Company profile

The company in this case study provides services and solutions for critical infrastructures to various end customers around the world. The company operates mainly in the energy sector, according to the classification given by the German Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik – BSI (Bundesamt für Sicherheit in der Informationstechnik, 2023)). In 2020, the company had more than 40.000 employees in various departments, with about 3% working for Finances, 9% for Human Resources, Legal, Supply Chain, and Solutions, about 50% for Services, 30% for Operations, and 8% for other activities. The software produced by the company may be deployed to critical infrastructure in a product or service after several years of product or service development and coding. It may run in critical infrastructure for decades.

This case study focuses on a relatively small department inside the company working on and being responsible for software analysis and simulation of instrumentation for critical infrastructures. The department is not an operator and does not interact with end customers. However, it provides software incorporated into end products for critical infrastructure, which is then utilized and integrated into other products by other divisions within the company. The other divisions then offer these products directly to end customers or in a contractual agreement according to which the company operates the infrastructure directly.

The company involved in this case study wishes to remain anonymous. Hence, it is referred to as the company. A researcher, a higher management representative, a lead developer, and an external consultant conducted the case study.

6.1.2.1 IT security in the company

Since the company works with and delivers products for critical infrastructures, it is very self-conscious of the need to be excellent in cyber security. The company is also strongly motivated to comply with cyber security standards, which results in a commitment to the end customer and a low risk appetite.

This requirement applies to the whole business and business sectors of the company. The important cyber security standards that apply to this industry and are relevant to this case study are IEC 62443 (International Society of Automation, 2023c), in particular IEC 62443-4-1 (International Society of Automation, 2023b) and IEC 62443-4-2 (International Society of Automation, 2023a), and ISO 27001 (ISO27001, 2017). The standards also define necessary protections for cloud assets. The best known among them are ISO 27017 (ISO27017, 2015) and ISO 27001 (ISO27002, 2013), which require practitioners to participate in training to learn about security technologies and raise awareness on cyber security. The CSA CCM (Cloud Security Alliance Cloud Controls Matrix) (Cloud Security Alliance, 2021) compares 44 cloud security standards and shows an overview of the coverage of cloud security controls. MITRE ATT&CK cloud matrix (MITRE ATT&CK, 2020e) categorizes cloud attack actions and defense mechanisms based on real-world observations. The parts of ISO 27001 that are particularly interesting for this case study are part of Annex 14 of ISO 27001.

The software developer workforce is heterogeneous. There are employees with many years of software engineering experience and young software developers fresh from universities. Both young and experienced developers might be reluctant to follow current computer science or cyber security advances and best practices in secure coding. A heterogeneity of technical backgrounds and secure coding awareness can also be found in the group of lead developers. Balancing the different points of view of the established developers and the younger generation proved challenging.

6.1.2.2 Risk analysis

The software developed in the department is integrated into final products used in critical infrastructures. As such, the same safety, security requirements, and goals are shared across different divisions. As cloud computing is an important service provisioning model, protecting deployed cloud assets must be carefully designed according to security law. The cloud accounts of each department are closely monitored for the health status of the deployed resources. Software development security requirements are also driven by the IT security standards the company wishes to comply with and internal guidelines on software excellence.

Developing and delivering software integrated into the company's end products exposes these components to attacks that can result in severe business consequences. From coding, it can take several years for the software to be deployed to a critical infrastructure. In the critical infrastructures it may run for years, even decades. A lot can happen over the lifecycle: patches, re-configurations, and an evolving threat landscape. The risk levels are constantly monitored, and appropriate measures are taken to address weak links so that the company fulfills its accepted risk appetite established

by higher management.

This case study addresses one action the higher management took to lower the company's risk levels. These risk levels accepted by higher management consider the possible consequences of a breach in security. To address these negative consequences, the company invests heavily in secure software development to ensure that the software it produces meets high-quality standards.

6.1.3 Updating the secure coding guidelines

The project to update the secure coding guidelines can be described by one goal and three requirements:

- **Update secure coding guidelines**: update the existing secure coding guidelines with more modern and updated guidelines that address the current state-of-the-art and industry best practices, with the following requirements,
- **Business Alignment**: in order not to cause unnecessary stress to the software developers, the number of secure coding guidelines should be kept at a minimum; the minimal number of secure coding guidelines are derived by alignment with business goals and priority of the guidelines,
- **Mutual Agreement**: the secure coding guidelines should have a mutual agreement of all the lead developers and address their experience, expertise, and concerns,
- **Management Endorsement**: the goal was that the updated document be endorsed by management to turn it into a mandatory policy for a broader number of software developers.

The previously existing secure coding guideline internal document was a starting point for selecting the sources. The external consultant proposed using additional secure coding guidelines from external parties as a reference — instead of writing all coding guidelines from scratch.

These standards and textbooks were used to determine an initial set of coding guidelines. Additional sources, including information on previous software incidents and lessons learned from previous projects, helped the lead developers select this initial set. Additionally, to secure coding guidelines, the lead developers also wanted to address core and clean code guidelines related to code style and software architecture. It was decided that these documents would also be evaluated and considered from a secure coding perspective. Workshops were conducted to identify relevant coding guidelines.

The guidelines were placed in the Excel tracking document and several iterations of classification, classification, prioritization, and verification were carried out. In verification, it was determined how to ensure compliance with secure coding guidelines. The options considered for verification were manual

verification, automated verification, code review, and check-in trigger. The project's final activity was to present the guidelines to management, review the document, seek approval, and implement them as per company policy.

The original secure coding guideline document consisted of 12 secure coding guidelines. After all the decisions taken during the joint workshops, the total number of secure coding guidelines went up to 31, and nine were kept from the original secure coding guidelines document. In this process of defining the guidelines, a presentation format and additional information on how to implement and verify the secure coding guideline were developed and used for a uniform and easy-to-read presentation of the guidelines.

Establishing the coding guidelines included document reviews, several workshops, and a presentation to management. The resulting secure coding guidelines document receives backing and support from higher management, addresses the specific business case, and distills the experience from the lead developers and the external security expert.

The main outcome is a document containing the updated secure coding guidelines in the form of a secure coding policy.

6.1.4 Reflection on the identified preconditions from the case study to CATS

We reflected on the preconditions identified in the case study: updating secure coding guidelines, business alignment, mutual agreement, and management endorsement.

CATS focuses on the topic of cloud security; therefore, the organizational-level interpretation of the cloud security guidelines should be identified and updated accordingly to address the feasible industry best practices.

Business alignment is necessary to guarantee the successful implementation of CATS. This applies for business and technical roles and responsibilities, as cloud security is a joint effort of multiple stakeholders.

Mutual agreement means that stakeholders should see the cloud security guidelines in the same way and be clear about their roles and responsibilities. An agreement needs to be established among practitioners.

Management endorsement is the key to raising awareness of cloud security topics and enforcing cloud security guidelines. At the management level, they should know the guidelines and commit to enforcing them.

In the design and implementation of the CATS game as an element of the organization´s training program, we ensured that the identification and selection of the standards and game content identified relevant and agreed-upon content, that the initiative had management endorsement, that the topic of cloud security and shared responsibility is aligned with the company's strategy. We involve stakeholders who were already involved in the design and implementation of the CSC in the design. This allows us to follow the successful example of the CSC and be respectful of the resources of the organization.

6.1.5 Summary of the preconditions as a part of the road map

We analyze the following necessary work to implement CATS in our organization: the motivation to understand the relevant standards and the established role and responsibility distribution. The relevant topics are enforced in the organization, which becomes the cornerstone of the road map: updating secure coding guidelines, business alignment, mutual agreement, and management endorsement. The road map estimates the amount of work and necessary resources needed to implement CATS as a training for cloud security in an organization.

6.2 Integration into training program

Flexibility is important in the design of training programs, as measures must be adopted to the needs, resources, interests, and expectations of the organization as well as the participants. In our study we used CATS in different settings.

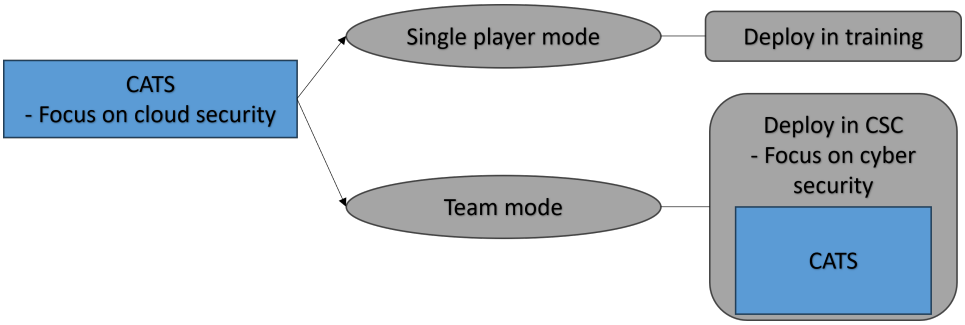


Figure 6.1: Different ways of integration into training program

Figure 6.1 demonstrates different variants of CATS deployment. Integrating CATS into an existing training program saves organizational overloads compared to creating a new training program. However, having CATS as a standalone game and training could raise awareness and increase visibility — more than in an integration of CATS with other training measures.

There are various ways to integrate CATS into a training program. As Figure 6.1 shows, CATS offers different types of deployment that can be used in training programs. CATS, as an online card game, focuses on cloud security as a topic. It can be deployed as a standalone game in training or as part of a CSC. It can be played virtually, hybrid, or in a face-to-face setting. The deployment possibilities make the game available to the practitioners in the industry with relatively little effort and offer various options.

CATS game is integrated into the training curriculum as a balanced instantiation of the game event in a hybrid training environment. CSC focuses more on secure technical know-how for coding. Developers find training interesting and engaging. CATS has a broader target group than CSC because it focuses more on strategic planning in cloud security defenses, being more relevant

for managers than secure coding challenges. When developers and managers work as a team, they find CATS engaging and helpful; both can contribute to the team's winning.

From our previous game events, the game design shows positive results for both games. We learned the following lessons:

- Participants welcome hands-on exercises.
- Serious games can keep trainees engaged during a training session.
- It is important to automate the deployment and recycle the game infrastructure.
- It is important to align activities with the business context in terms of business alignment, mutual agreement, and management endorsement.

CATS was developed after CSC, and we could use lessons learned in developing CSC and adapt the design to an online environment with positive preliminary results. The results of individual games are discussed in separate publications; this work focuses on applying those serious games in the industry under hybrid work.

The takeaway for building the road map is that there are various ways of integrating CATS into the organization and that it needs to be determined which deployment and presentation of the serious game is the most suitable in the given situation.

6.3 Adaptation to new way of work: hybrid working environment

This section introduces the context of a hybrid working environment. As a change in the training program, serious game contributes to a more interactive and engaging training format. During our studies, we observed the impact of COVID-19 pandemics on the workplace: work-in-the-office, work-from-home, work-from-anywhere, and different hybrid settings became valid options. Organizations differ in expectations in terms of workplace and, moreover, these expectations change and adapt to organizations´ strategies and goals.

We evaluated hybrid work from different perspectives. From the trainer's perspective, the goals of our evaluation are: 1) The effort was made to adapt the on-premises training to online training. 2) the pros and cons of training in hybrid mode from the trainers' perspective.

Such activities ensure mutual understanding, resource efficiency for the whole program, and alignment with the organization's goals and employees' preferences. They are also a resource-efficient way to obtain significant feedback to improve the games and content. Therefore, the evaluation contributes to the road map element of the communication plan.

We apply a systematic, qualitative content analysis process following the guidelines by Corbin and Strauss (Corbin and Strauss, 2014) based on the

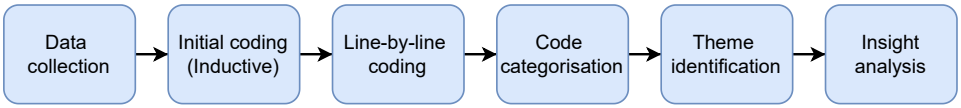


Figure 6.2: Qualitative data analysis process

methodology and process steps proposed in (Crosley, 2020; Frampton, 2020). Figure 6.2 describes the steps of our data analysis. The process starts with data collection through the semi-structured interview (SSI).

We conducted an online SSI with seven trainers in our organization. All interviewees were expert trainers before the pandemic, and they experienced the digital transformation of training in hybrid work. In the interview, we asked them two questions:

- What is better in online training compared to onsite training?
- What is worse in online training compared to onsite training?

During the interview, we transcribed the answers provided by the interviewee or let the interviewee write down their answer to the questions above as an online survey. We removed sensible information from the answer we collected and applied a Python library (Mueller, 2021) to generate a word cloud as shown in Figure 6.4.

Table 6.1 summarizes the general information on the SSIs conducted. We invited ten trainers who hold training events on secure coding in various programming languages, operational technology (OT) security, security activities and processes, and threat & risk analysis. Seven trainers accepted the invitation and provided their answers to the above questions.

Table 6.1: Semi-structured interview general information

Semi-structured interview general information	
# of invited trainers	10
# of collected result	7
# of interviewed female trainers	2 (28%)
# of interviewed male trainers	5 (72%)
# of interviewed trainers (>10 years of training experience)	3 (42%)
# of interviewed trainers (5-10 years of training experience)	3(42%)
# of interviewed trainers (<5 years of training experience)	1(16%)
Average time for the semi-structured interview	39 mins. 46 sec.

Then, we applied an initial inductive coding process to the free text we gathered during the interview, as suggested in the process in Figure 6.2. All the coding processes mentioned below are manual because the sample size is small, and manual coding provides the best precision for a small sample

size. The purpose of initial coding is to identify the essence of the text and code it accordingly. The inductive approach develops the initial code set descriptively in this step. The next step is line-by-line coding, in which the text is reviewed and coded line-by-line. In this step, details will be added to each line. For example, when the interviewee mentions the new tools used in online training in the initial coding step, it is coded as "tool/technology." This step of line-by-line coding was coded more specifically as "meeting software" or "virtual whiteboard"; this allows us to dig deeper into the data we collected in the interviews. After that, the code categories are created to organize the data and guide the analysis in the next step. Based on the result of code categorization, the themes are identified and articulated. In the last step, insight analysis, we derive the insights based on the steps before and produce a narrative that answers the goal of our evaluation.

In the insight analysis, we assign emotions or sentiments to each piece of data in a 5-level Likert scale (strongly positive - 5, positive - 4, neutral - 3, negative - 2, and strongly negative - 1) (Godsay, 2015) and combine the sentiments to conclude. Figure 6.3 shows two examples of how original texts are categorized and rated according to sentiment.

Original text in interview	Category Tags	Sentiment
"Participation of participants located gobally is easier...but (they) cannot chat with each other in coffee breaks."	Flexibility Networking	5/5 1/5
"...Preparation of online training is easier..."	Preparation	5/5

Figure 6.3: Example showing category assignment and sentiments

We collect the interview responses and go through an analysis process as illustrated in Figure 6.2. We grouped the categories into seven themes, counted how many times a certain theme occurred in the SSI, and assigned a sentiment scale each time it occurred. It should be noted that a theme could occur multiple times in the interview with one training. We derived insights on each theme on the sentiment scale and raw data. Table 6.2 provides a summary of the results.

The most frequently mentioned theme is "Tool / Technology" from the trainer's perspective. Adapting conventional training in a hybrid environment involves many tools, such as virtual whiteboard, meeting software, etc. The new tools make the material easier to read than in onsite training, where a poor-quality projector could leave the text barely readable. The trainers are generally positive about the improvement in "Tool / Technology," except that the training in a hybrid environment depends heavily on network quality and

Table 6.2: Theme identification and insights from the trainer's perspective

Themes	# of occurring in semi-structured interviews	Sentiment on average	Insights
Tool / Technology	19	4.4	Trainers are concerned with the new tooling in hybrid work and find tools helpful in training as an improvement. There are some complaints about dependencies on network quality and the availability of webcams.
Participation	16	2.5	For training in hybrid environment, it is challenging to keep the participants engaged and to remain the same level of interaction as on-premises training. Trainers no longer get direct feedback from trainees in hybrid environment.
Flexibility	16	4.4	Training in hybrid environment comes with great flexibility, allowing global participation despite the geographical and time differences. Business trips are cut regardless, which reduces carbon footprint.
Training organization	10	2.6	Work needs to be done for training the organization, e.g., re-designing the group work, introducing more breaks, etc.
Networking	8	2.3	Trainers are worried that the networking opportunities are reduced significantly when there is no coffee breaks and informal idea-exchanging channels are missing.
Preparation	5	5.0	The preparation for training in hybrid environments is easier and more efficient with higher re-usability.
Managing COVID pandemic	1	5.0	Training in hybrid environment helps managing COVID pandemic.

adequate webcam usage during the online session.

The next frequently-mentioned theme is "Participation," which the trainers feel slightly negative about. In the hybrid environment, keeping the participants engaged and stimulating interaction requires more effort. In addition, the trainers missed direct feedback from the participants, making it difficult to judge whether the messages were delivered to the trainees.

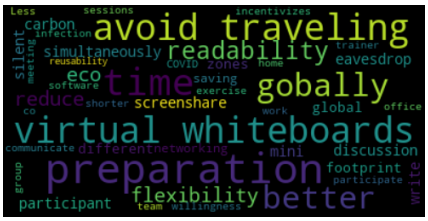
The theme "Flexibility" is mentioned 16 times in the SSI, reaching 4.4 on the sentiment level. Training in a hybrid environment allows greater geographical flexibility in different time zones. Reducing business trips contributes to a lower carbon footprint. However, some people miss the business trips that are perceived as "spice" to daily work.

"Training organization" is mentioned ten times in the SSI with a rating of 2.6 on the sentiment level. Additional work must be done to organize the training in a hybrid environment, e.g., re-designing the group work. Some trainers are neutral about the new "Training organization," and some complain about the additional effort required for adaptation.

The theme "Networking" is mentioned eight times in the SSI, with the lowest rating of merely 2.3 in the sentiment level. Trainers observe that in

a hybrid environment, networking remains a weak element. Due to missing coffee breaks, participants lost the opportunity to exchange casual information.

The SSI mentions the theme "Preparation" only five times. However, the sentiment level is overwhelmingly positive, with 5.0 points. Trainers agree that it is easier to prepare for training in a hybrid environment. One trainer interviewee mentions the theme "Managing the COVID pandemic " only once. However, we consider this theme equally important because online training avoids gathering and helps prevent the pandemic from spreading.



(a) Wordcloud: what is better in an online training

(b) Wordcloud: what is better in an onsite training

Figure 6.4: Wordcloud: comparing online and onsite

We generated a word cloud based on the collected text for both questions. The resulting word clouds for what is better in online training and for what is better in onsite training are shown in Figure 6.4.

The trainers find the new tools helpful in organizing hybrid training, and the preparation efforts and costs are reduced. The hybrid training mode is also more flexible, overcoming the geographical limitations. However, the trainers' task of keeping the level of concentration and engagement of the participants is more challenging compared to the classroom training. In addition, casual talks and networking become difficult in a hybrid working environment.

The takeaway to building the road map is that the integration of serious games must contribute to the organization's migration. In our cases, migration refers to adapting to the hybrid working environment. It is necessary to perform a continuous evaluation to reflect the needs of the trainers, trainees, and the organization.

6.4 Road map to implementing cyber security awareness program on cloud security with CATS in the industry

This section concludes with the road map to adapting CATS into an organization. If further practitioners would like to introduce CATS into their organization in the future, the road map provides instructions for a successful integration.

In total, we identified six elements: 1) The Pre-conditions; 2) the Goal & objective; 3) the Dependency and alignment with the given setting; 4) the

Timeline; 5) the Resource; and 6) the Communication plan. In the following part of this section, each element will be described, and an example of how this element is implemented in our study will be given.

Road map is a concept we borrow from agile DevOps, for the concept we follow the technical blog from Radigan (Radigan, 2024). A product road map is a plan of action for how a product or solution will evolve. According to Radigan, product teams build a road map by considering market trajectories, company goals, customer feedback and insights, and engineering constraints. The company aims to design a useful artifact to enhance the existing training framework, focusing on cloud security. Once these factors are reasonably well understood, they are expressed in a road map as initiatives and timelines. In our study, a road map consists of the following elements: pre-conditions, goal & objective, timeline, dependency, and alignment with the given setting, resources, and communication plan.

- The Precondition:

Road map element: The pre-conditions must be fulfilled to execute the road map. In the case study, we identified those conditions: update secure coding guidelines, business alignment, mutual agreement, and management endorsement.

Road map element: As described in section 6.1, we reflected on those identified preconditions: organizational-level interpretation of cloud security guidelines, business alignment on roles and responsibilities, mutual agreement among the practitioners and the management endorsement to raise awareness on cloud security topic and enforce the cloud security guidelines.

- Goal & objective:

Road map element: Our goal is to implement a serious game, CATS, in an organization in the format of a serious game that addresses cloud security. The focus of the serious game should be to raise awareness about various cloud security issues and their mitigation, as well as the shared responsibility model within the organization. In the original case study, the goal was to update the secure coding guidelines.

Our effort: To reach this goal, we investigated cloud security standards and controls to secure cloud assets. We gained an overview of the game elements by reviewing numerous cloud security standards and shared responsibility models from different cloud service providers. Furthermore, we studied the organization's interpretation of the standards, ranking the commonest cloud security threats and different roles and responsibilities in defending cyber security attack targets at cloud assets. Last but not least, across the three design iterations, we invited security experts in the organization to review the game logic and give feedback to improve CATS.

- Dependency and alignment with the given setting:

Road map element: Driven by the organizational need to improve cloud security awareness level, the CATS game was developed in such a way that it fits into the training. Internally, serious games should fit the training curriculum and the content needs to be aligned with the training material to avoid confusion. It must fit into the internal setting with training certificates. A process needs to be established to identify relevant industry standards and frameworks and narrow them down to the issues of the organization. To make CATS suitable for the organization, we adapted the evaluator algorithm to reflect the key facts of how the organization is affected by cloud security issues and how such issues should be mitigated by different roles and responsibilities from within. The game logic and element must reflect important facts about cloud security requirements and standards, cloud attack and mitigation mapping derived from real-world examples, shared responsibility model, cyber security kill chain, information security ontology, and the probability calculation should have a solid theoretical foundation.

We must understand which format is more suitable for deploying a CATS game event. The possible options are: 1) to have CATS as an independent game played by the practitioners regularly. This is not feasible because of the organization's overhead. 2) Deploy CATS as additional game events after a full-day training to exercise the know-how conveyed during training. This is feasible, and we have organized game events as such. 3) to deploy CATS as a category of challenges in a Cybersecurity Challenge event. We have also organized game events.

Our effort: We did a fundamental case study to understand the organization's requirement to improve cyber security and, more specifically, cloud security. To increase the exposure of CATS to industrial practitioners, we integrated CATS with training. We offer different game modes for use cases, as shown in figure 6.1. The digital platform has been developed to make it easier for CATS to be present in the training setting. This point corresponds to the original case study's Business Alignment and Management Endorsement.

- Timeline:

Road map element: Timeline refers to the time frame within which the goals are expected to be achieved, and the road map should evenly focus on short-term tactics and long-term strategic goals (Radigan, 2024). As the development cycle of the CyberSecurity Challenge suggests, the quality of the designed artifact improves as it goes through multiple design cycles. Design cycles can be mapped to milestones. The first milestone is to verify the game logic with a concrete game prototype with a cyber security expert in the industry. The second milestone is to improve the prototype and evaluate the

CATS in a larger group. The ultimate goal in the long term is to improve cloud security awareness levels among the target group.

Our effort: We planned our three design iterations according to the identified timeline as a road map element. In each design iteration, we published our results at conferences. The publication went through a scientific reviewing process. In the evaluation phase, feedback is collected. Milestones are met by sticking strictly to the timeline.

- Resource:

Road map element: The resource refers to the effort required to execute the road map and the resource necessary to plan and conduct game events.

The resources required to conduct the CATS game event are mainly the time spent by the players and trainers. The flexibility in CATS deployment allows us to organize a game event as short as one hour or include it in a full-day game event or take more time. The game can be adapted to both the organizations and the trainees resources and expectations.

Resources are necessary to organize game events. Learning from the successful story of the CyberSecurity Challenges, it is better to reuse the available resources as much as possible rather than re-invent the wheel and make a double effort. The deployment process of CATS reuses the deployment script of the CyberSecurity Challenge. The disposable deployment process minimizes the maintenance effort for the trainer and minimizes the possibility of having infrastructure issues for the participants, which again would cost time and resources. The time spent learning the game logic and interface is saved by reusing the available material.

All these points above contribute to the necessity of respecting the organization's resources.

Our effort: We identified the available resources for us to use in the organization: 1) the training schedule allows us to verify the game logic and collect feedback,k 2) working in the current research group allows us to stay in consistent contact with cyber security expert 3) the deployment method of an existing serious game can be reused to deploy CATS 4) the funding of CONTAIN by the Federal Ministry of Education and Research under the project numbers 13N16581 and 13N16585 helped us to continue our research 5) my experience in training also helps me to develop the game elements and deliver the training with CATS to the suitable target group. This point corresponds to the Management Endorsement from the original case study.

- Communication plan:

Road map element: According to Radigan, once a road map is built, it must be shared with the entire product, leadership, and delivery teams so that everyone understands the vision and direction. Our communication

plan includes the identification of stakeholders and how we should approach them in the event of necessary changes. The most important stakeholders in training activities in industry are the trainers and trainees, additionally, the support of higher management is needed to launch the training activities. Last but not least, since the security standards and best practices are interpreted differently in different organizations, the opinion of security experts is valuable in adjusting the designed artifact.

Our effort: We have identified the stakeholders and the separation of roles and responsibilities and designed the cards and elements of the game in CATS as such. Since the game is self-sufficient and independent of trainers' knowledge, any qualified trainer could organize CATS game events. Documentation is available for the designed game element and is open for review. Internally, CATS is presented in the training curriculum. The results are also published externally. This point corresponds to the Mutual Agreement from the original case study.

To implement CATS into our organization, we first checked the pre-conditions and verified that those conditions were fulfilled. Then, we made the following effort: 1) We investigated cloud security standards and controls to secure cloud assets and established the goal and objective of implementing CATS to raise awareness about cloud security. 2) We did a fundamental case study to understand the organization's requirements for improving cyber security and cloud security. This effort guarantees that CATS is aligned with the given setting. 3) We planned our design activities in three design iterations to meet the timeline and derived our short-term tactics and strategic, long-term goals accordingly. 4) We gathered the available resources for the development, acquired funding from several projects, and utilized both the tangible and intangible resources that substantiated the study. 5) We identified the stakeholders and reinforced the separation of roles and responsibilities in our designed artifact to commit to the communication plan. These five effort elements have contributed to the successful implementation of CATS in our organization, and the ideas can be generalized. More specifically, if someone would like to adapt CATS into their organization, a similar effort should be made to follow the road map elements described above.

6.5 The road map

The figure 6.5 depicts the road map to implementing CATS in our organization. The green boxes are available, and the blue boxes were developed for this study. At the beginning of our study, we set the goal and objectives for implementing CATS in an organization to address cloud security. Then, we collect the available material and work step-by-step towards the goal.

The road map starts with checking the pre-condition that consists of organizational-level interpretation of cloud security guidelines, business alignment on roles and responsibilities, mutual agreement among the practitioners,

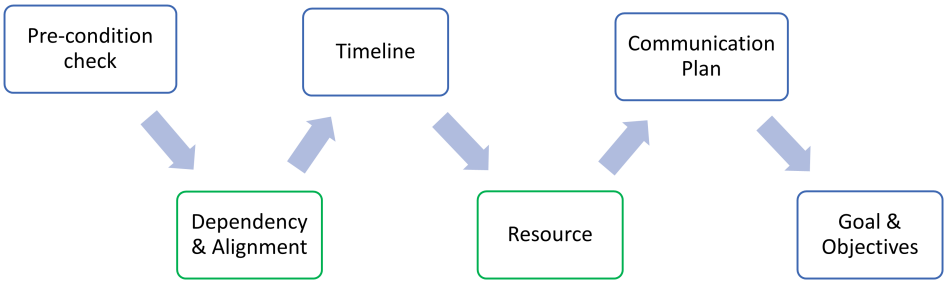


Figure 6.5: Road map to implementing CATS in a training program

and the management endorsement to raise awareness on cloud security topics and enforce the cloud security guidelines. Then, the dependencies are identified and aligned with the given setting. CATS is integrated into the training framework to increase the visibility and exposure of CATS. In the next step, a timeline is formulated with the three design iterations. Performing the timeline requires resources. In our study, the resource is mainly the time of trainers and, in particular, the trainees. It takes time for the trainers to become familiar with the game and for the participants to participate in the training. We are respectful of the organization's constraints that provide us with resources. The last part of the road map is the communication plan. CATS is communicated both internally and externally by different means. In the end, we work towards our final station, goal and the objective, that has been set since the beginning of the study.

6.6 Chapter summary

Our research began with a case study to increase the levels of security in software engineering. The case study captures an initiative to update the company's secure coding guidelines to reflect the latest advances in technical know-how and industry best practices. In an international company, cyber security plays an important role, and the value of security training is well recognized. In this environment, CyberSecurity Challenges was designed and applied in the company's training curriculum to enhance security training. CATS was then developed and learned from the successful experience of CSC since the year 2021. Jan. 2022 (first game event of CATS, see table 4.18), CATS can be applied as a standalone game in training or CSC as a specific category of challenges. The roadmap provides guidance on essentials in the design and implementation of a training program in an organization. It details the precondition check, the dependency and alignment topics, the timeline, the resources, the communication plan to meet the goals and objectives.

7

Reflection on the research design

In this chapter, we reflect on the existing body of knowledge from our design and share how we contribute to the theory by designing and implementing CATS. The chapter starts by reflecting on Design Science Research principles and further reflects on research embedded into practice. Previous chapters mention that CATS is designed under the Design Science Research paradigm. In this chapter, we would like to share our experience conducting such research and contribute to the existing body of knowledge.

7.1 Reflection on Design Science Research principles

Hevner et al. (Hevner, March, and Park, 2004) derived seven design guidelines of fundamental principles of design-science research. "Good design science" quality of the project contributes to the quality of the results. In this section, we will discuss all of those guidelines and show how the guidelines have guided our study and our reflection on the guidelines.

7.1.1 Guideline 1: Design-science research must produce a viable artifact as a construct, a model, a method, or an instantiation.

According to Hevner et al. (Hevner, March, and Park, 2004), the result of design-science research in IS is, by definition, a purposeful IT artifact created to address an important organizational problem. It must be described effectively, enabling its implementation and application in an appropriate domain. In section 6.1, we have shown the environment of our study. CATS

is designed to solve the important organizational problem of enhancing cloud security, educate practitioners on the roles and responsibilities of cloud security, and raise awareness of cloud security issues and solutions. The viable artifact is the CATS game. With three design iterations, it is validated by the game trial runs and game events.

7.1.2 Guideline 2: The objective of design-science research is to develop technology-based solutions to important and relevant business problems

According to Hevner et al. (Hevner, March, and Park, 2004), the objective of research in information systems is to acquire knowledge and understanding that enable the development and implementation of technology-based solutions to unsolved and important business problems. The present work is relevant to business problems as it enriches the traditional training material and raises the practitioner's awareness of cloud security. It is designed based on a concrete problem, and the result of the current work is validated in the same environment. To be more concrete, in the game, participants could learn about the necessity of various defense mechanisms and how they contribute to a better defense strategy. With the acquired knowledge, cloud security incidents due to inadequate awareness, such as misconfiguration, can be avoided (Sebayan, 2021; Trendmicro, 2017; UpGuard Team, 2017).

7.1.3 Guideline 3: A design artifact's utility, quality, and efficacy must be rigorously demonstrated via well-executed evaluation methods.

According to Hevner er al. (Hevner, March, and Park, 2004), evaluation is a crucial component of the research process. Our design artifact went through three design iterations. In all the iterations, we have collected feedback in various ways. We observed the gameplay process in three trial runs with 11 participants in the first design iteration. In the end, we evaluated the validity of the game logic based on the open discussion and our observations during the game.

We conducted ten game events with 123 industrial practitioners in the second design iteration. The feedback was collected during the game as we observed the game process, and the back end tracked the game dynamics performance of the players. Additionally, we surveyed the game players and invited them to open discussion in the end and semi-structured interview. The collected data is analyzed scientifically, and we conclude that the game helps raise awareness of cloud security. The questionnaire we designed is based on the three dimensions of IT security awareness: perception, protection, and behavior.

We organized two game events with 24 industrial practitioners in the third design iteration. We collected feedback through a survey and open discussion. Additionally, we compared the results of the game dynamics data capture

in game runtime with those of the previous iteration.

In all design iterations, well-executed evaluation methods rigorously demonstrate a design artifact's utility, quality, and efficacy.

All the evaluations in the design iteration mentioned above went through the review process in our publications. The strict reviewing process and final acceptance highlight the rigor of our evaluation. The case study, as mentioned in chapter 6.1, provides us with an understanding of our study's utility, quality, and efficacy in its context. The case study underlines the relevance of our research activities.

7.1.4 Guideline 4: Effective design-science research must provide clear and verifiable contributions of the design artifact, design foundations, and/or design methodologies.

According to Hevner et al. (Hevner, March, and Park, 2004), effective design-science research must contribute to the design artifact, design construction knowledge and/or design evaluation knowledge. The ultimate assessment for any research is the new and interesting contributions. We extended the existing ontology and adapted the evaluator algorithm in the present study. The work contributes to the existing knowledge on cyber security through the designed artifact, the CATS game. Within the organization, it helps raise awareness of cloud security among practitioners who participate in training or game events. Externally, the work instantiates design science theory and can inspire studies in a similar area.

7.1.5 Guideline 5: Design-science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact.

According to Hevner er al. (Hevner, March, and Park, 2004), rigor addresses the way in which research is conducted. Design-science research requires the application of rigorous methods in both the construction and evaluation of the designed artifact. Our work is based on the well-established existing body of knowledge on cloud security (Amazon Web Services, 2023; Mell and Grance, 2011; MITRE ATT&CK, 2020e) and IT security ontology (Fenz and Ekelhart, 2009) and extended them to cloud security awareness in industry. The awareness theory (Hänsch and Benenson, 2014) of Hänsch et al. is a theoretical guideline. For the construction and design of our artifact, we followed the design science paradigm proposed by Hevner et al. The design of CATS is guided by the instruction of Dörner et al. on serious game (Dörner et al., 2016). Moreover, the evaluation methods follow well-established methods of survey methodology, semi-structured interviews, and qualitative data analysis.

7.1.6 Guideline 6: The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment.

According to Hevner er al. (Hevner, March, and Park, 2004), design science is inherently iterative. Searching for the best or optimal design is often intractable for realistic information systems problems. The present work follows the case study. In the original case study, we discovered the organization's need to raise cyber security awareness. In the extended version of the case study, we generalize the topic of cyber security to cloud security and develop a serious game that could be played in cyber security training to raise awareness about cloud security and the shared responsibility model. The case study lays a solid foundation for the present work.

Additionally, inspired by the success of CSC and with the support of the management level, we are granted freedom in our design. Learning the successful experience and reusing the deployment infrastructure of CSC partially has also accelerated the present research so that we can focus on the nature of the problem itself, e,.g. what defense actions are effective under which circumstances, how different roles and responsibilities collaborate in CATS and how defense standards and real-world cloud security elements should be reflected in the game. The game logic is validated in the first design iteration; in the second design iteration, we focus on the defense perspective and develop the digital platform. In the third design iteration, we quickly reached saturation of knowledge. We reached our intended end of understanding how to raise awareness of cloud security through serious games in the industry.

7.1.7 Guideline 7: Design-science research must be presented effectively in both to technology-oriented as well as management-oriented audiences

According to Hevner er al. (Hevner, March, and Park, 2004), technology-oriented audiences need sufficient detail to enable the described artifact to be constructed (implemented) and used within an appropriate organizational context. The present work is communicated to both technology-oriented and management-oriented audiences. It resulted in seven conference papers and two journal papers. Those publications include the following: second International Computer Programming Education Conference (ICPEC) (May 2021), third International Computer Programming Education Conference (ICPEC) (Jun. 2022), Ubiquitous Security (UbiSec) (Dec. 2022), IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC) (Jun. 2023), Innovations for Community Services (I4CS) (Jun. 2024), International Conference on Software Engineering Education and Training (CSEE&T) (Jul. 2024), and two journal papers in Information and Journal of Systems and Software. It also won the best paper award in ubiquitous Security (UbiSec) (Dec. 2022). Additionally, the funding for project CONTAIN by the Federal

Ministry of Education and Research under projects 13N16581 and 13N16585 is acknowledged for this project. The topic is also presented at Universität Passau. One result of the present work is the success of one master's thesis from the Technical University of Munich.

The idea of CATS is also communicated within the organization. It is integrated into standard security training and presented to the participants as enhancement material for traditional training. It has been deployed 12 times, and towards the end of 2023, it will also be presented as part of CSC to different business units in Germany and China to train practitioners locally.

7.2 Reflection on the result

According to Hevner et al. (Hevner, March, and Park, 2004), the design science paradigm guides our research approach. The method literature on design science defines the design of a useful artifact as a creative search process for a useful solution. Gleasure has pointed out that when the prescriptive aspect of a research problem is less mature than its descriptive or normative dimensions, the information system (IS) research problem is 'wicked' (Gleasure, 2013). The abovementioned theory provides valuable theoretical support to our work since Design Science Research can handle the changing and varying requirements we encounter in practice and industry. Baskerville (Baskerville and Pries-Heje, 2010) and Hevner (Hevner, March, and Park, 2004) note that a 'wicked' problem has several possible solutions. We present our work as one of the possible solutions. Our designed artifact has been developed under the Design Science Research paradigm and has undergone three design iterations. The usefulness of the artifact depends on the social and cognitive abilities and social context.

The evaluation phase in all the design iterations illustrates that our artifact is useful in the organizational context: the participants find the game helpful in raising awareness about cloud security issues, building cloud defense strategies, and learning about the different roles and responsibilities within the organization. Furthermore, it is integrated into the standard training material as an enrichment. We argue that our research design fulfills both requisites as a useful artifact following the Design Science Research paradigm.

7.3 Chapter summary

In this chapter, we reflect on the existing body of knowledge from our design and share how we contribute to the theory by designing and implementing CATS. The chapter starts by reflecting on Design Science Research principles and further reflects on research embedded into practice. We reflected on seven guidelines in DSR. As mentioned in the previous chapter, CATS is designed under the Design Science Research paradigm. In this chapter, we shared our experience conducting such research and contribution to the existing body of knowledge. The study process of CATS strictly follows the methodology of the design science approach. As a result of the successful design process, the

result we achieved from the evaluation phase strongly indicates that CATS is a well-designed and useful artifact. By achieving such results, we contribute to the body of knowledge on how to raise cyber security in the industry setting.

8

Conclusion

In this last chapter, we conclude our work. This chapter states the course of action, our contribution, and the final words. This study was conducted under the scientific guidance of Design Science Research and accomplished three design iterations. In the meantime, the design artifact is useful and valid in the given setting. It also guides and inspires similar studies on raising cyber security awareness using serious games. In this chapter, we look retrospectively and re-capture all the important points we discover in our journey of designing and applying serious games to raise awareness of cloud security in the industry.

In the industry, cloud deployment has gained popularity due to its flexibility and the abundance of cloud service modules provided by cloud service providers. Yet, without adequate training to raise awareness about the typical pitfalls and mitigation, cloud services are prone to configuration errors and pose great threats to cloud assets' security. Confidentiality of critical customer data can be compromised and exposed to the public. The integrity of the configuration or the cloud accounts can be impacted, leading to cloud services malfunctioning. Availability is less considered in cloud deployment due to the elasticity nature of cloud service. However, an attacker could take advantage of the computation resources in the cloud account and generate additional costs charged to the victim. Security standards regulate how cloud services should be secured and define best practices accordingly. The cloud service provider and the customer have different roles and responsibilities, as defined by the shared responsibility model. The cloud service provider shall provide the customer with the necessary infrastructure and services to secure the resources in the cloud, and the customer shall follow the documentation and configure those services

securely. There are further definitions of roles and responsibilities among cloud service customers, which should be specified and worked on jointly. This know-how is conveyed to the practitioners in the industry through training.

Studies have shown that there are limitations to the traditional training method, and cloud security awareness needs to be further improved to reduce the number of cloud security incidents. Therefore, we are searching for a training enhancement method to improve the group's overall level of cloud security awareness. With the guidance of the Design Science Research approach, we started designing and implementing our serious game, CATS. As I gained experience in the case study, I was able to identify the important factors for such a serious game in the same organizational setting. Those factors build up a solid road map that leads us to the successful design and implementation of CATS.

In this chapter, the course of action will be presented in section 8.1, where we summarize our action within the framework of this study from an organizational perspective, academic perspective, and methodology perspective. In Section 8.2, our contribution will be recapitulated. In the Section 8.3, we conclude our work.

8.1 Course of action

Our research is dedicated to solving a practical problem raised in an industry setting. In the research process, the effort combined academic work and evaluation in the industry. The design of the useful artifact is guided by the Design Science Research paradigm proposed by Hevner et al. (Hevner, 2007; Hevner, March, and Park, 2004). We went through three design iterations with different evaluation methods.

From the organization's perspective, the organization's management level supported the research. The research project is approved through internal research and funded by the CONTAIN BMBF project, collaborating with multiple business partners. The Security Lifecycle research group in Munich, Germany, accomplished the research project. The research group belongs to the Cyber-Security and Trust task force within Siemens' Technology department. CATS game is presented within the company during different cyber security training sessions, in which the participants are practitioners from the industry. Access to our target group allows us to collect data and feedback directly from the target group for a meaningful evaluation. Being in a cyber security research group also enables me to contact specialists daily, who contribute valuable opinions and feedback for improving CATS.

From the academic perspective, the research was aligned with the research group Information Systems of Universität der Bundeswehr München, under Prof. Dr. Ulrike Lechner, and also by the Department for Sciences and Information Technologies of the ISCTE - Instituto Universitário de Lisboa, under Prof. Dr. Maria Pinto-Albuquerque. Our close collaboration with the academic community inspired countless sparkling moments in the idea exchanges

and discussions. The publications achieved within this study provide constant access to the research community, and the scientific review process contributes to the substantial academic foundation of our work. We collected formal and informal feedback not only during the review process but also during the presentation of our work at international conferences, which increased the exposure of our work and enabled me to exchange ideas with other experts in the same field.

From the methodology perspective, our work is conducted under the guidance of the Design Science Research paradigm. The core is the cycle of design & implementation and justify & evaluation. It took three design iterations to reach the current version of the CATS game.

In the first design iteration, we developed the game logic and a prototype. The game is played with one attacker team and one defender team. Each team is provided with defense and attack cards to build a strategy. Towards the end of the game session, the game master will input the defense and attack strategy into the evaluation to calculate a success rate, and a wheel-of-fortune will be introduced to add to the playfulness of the game. Three game events with 11 participants from the industry and academic world participated and provided valuable feedback. One conference paper and one journal paper are published in this iteration. The game logic is validated, and in the next design iteration, we focus on developing a digital platform. A simulator should replace the attacker team with fixed attack scenarios to help the game participants concentrate on the defending perspective.

In the second design iteration, we have developed a digital platform and six attack scenarios to replace the attacker team and focus on the defensive perspective. The participants can interact with the platform in different gaming modes that suit the deployment context. In this design iteration, we conducted ten game events with 123 players from the industry to evaluate the game. Two papers were published in this design iteration, and one won the Best Paper award. In the evaluation phase, we confirmed that the CATS game left a positive impression on the players, and the results show an increase in cloud security awareness. CATS game assisted in reaching learning objectives during training. In the next design iteration, we fixed the game logic and the digital platform and focused on refining the evaluator algorithm to reflect more facts from the real world.

In the third design iteration, we adapted an existing security ontology to support the game logic, and we mapped the attack actions to vulnerabilities discovered in real-world hacking activities in cloud security. The defense mechanisms are also mapped to impact, reducing the vulnerability's risk rating. The mapping supports a finer granular calculation of the conditions of the winning probability for each attack scenario. We conducted two game events with 24 players from the industry to verify the adjustment we made. The result shows that the positive influence on the participants preserves, and more elements from reality are reflected without significantly raising the

difficulty and complexity of the game itself. We published one journal paper and three conference papers in this design iteration.

8.2 Contributions

In this section, we recapitulate the contribution of our research: the design of CATS, the road map that leads to CATS, and an adaptive serious game framework that allows the generalization of cyber security issues.

- Design of CATS

Under the Design Science Research paradigm, we designed CATS, a useful, serious game to raise awareness of cloud security among industry practitioners. The design of CATS went through three iterations, each with a different focus on improvement. CATS game supports various game modes. The player can join the game as a single player or join the group. The flexibility allows us to support divergent use cases in training.

The game contains a tutorial at the beginning to show the players the game interface and logic. The game interface has a defense area, an attack area, and a hint area. The game process does not require an expert game master. We designed six different attack scenarios in the game to simulate the attacker's activity. The players could design a defense strategy by assigning cards to different roles. After hitting the submit button, the player would get instant feedback on how well the defense strategy would work and which attack action is still not addressed by the current defense strategy and can work further on improving the defense strategy. Cloud security awareness is raised by following the hint system, thinking, and searching.

We claim CATS's design is novel, and its functionality is proven in the previous game events we organized. We used these game events as a solid evaluation method.

- Road map leads to CATS

In the process toward the useful artifact, we have identified a road map to raising cyber security awareness in industry training with serious games. An early case study done with a focus on secure coding guidelines is updated and published with a focus on using serious games as a viable approach. We understand the game elements CATS needs to refine to support the organizational setting better. Following this road map, we have successfully developed CATS. The road map is a concept we borrow from agile DevOps; for the concept, we follow the technical blog from Radigan (Radigan, 2024). A product road map is a plan of action for how a product or solution will evolve. We considered market trajectories, company goals, customer feedback and insights, and engineering constraints to build a road map. The company aims to design a useful artifact to enhance the existing training framework, focusing on cloud

security. The customer feedback refers to the feedback we received in the design iteration. We did various tests to gain insights into how well our game performs in the sense of achieving the objective. The constraints for us are the dependencies on the existing training framework, available resources, and the given setting in which we conduct our study. Once these factors are reasonably well understood, they are expressed in a road map as initiatives and timelines.

- An adaptive serious game framework

The game framework of CATS is derived from information security ontology, and CVSS calculations support the calculation. It has proven to be a viable solution for cloud games and is highly flexible. In the later phase of our research, we also adapted the CATS game to address the supply chain threat use case. Another serious game, COPYCAT, is born by updating the defense and attack cards, defining the mapping to existing vulnerabilities, and simplifying the attack scenarios. We did three game events for COPYCAT and received positive participant feedback. The development of COPYCAT is an ongoing research project, and many game events have yet to be planned.

The example above shows how easy it is to adapt the CATS serious game framework to another use case. The smooth adaptation can be attributed to a self-sustained ontology and valid game logic. Hence, CATS presents an adaptive serious game framework that could be reused in future research projects on information security awareness.

8.3 Final words

In this work, we presented the CATS serious game designed under the Design Science Research approach. CATS game finds its application in industry training sessions. Serious games are games with more than entertainment purposes (Azadegan, Riedel, and Baalsrud Hauge, 2012). Our work aims to raise awareness about cloud security issues and enhance the understanding of different roles and responsibilities among cloud service customers in building defense strategies against possible cloud attack kill chains. Furthermore, the game can be flexibly adjusted to address further topics in information security.

We started with a simple whiteboard exercise where the "defender" and "attacker" played against each other. Then, more elements were integrated to reflect reality and support the exercise's educational purposes. In total, 147 players participated in the 12-game events. Solid data analysis of the game event participants proved the usefulness of the designed artifact.

The design and implementation of CATS come to an end with this thesis. Ongoing research focuses on the adaption, application, and further improvement of COPYCATS. This thesis captures all the aspects of our study on CATS, the serious game for raising awareness of cloud security in the industry. Through this thesis, I aim to positively contribute to enhancing cloud security in the industry. By letting the practitioner raise awareness about potential

issues and emphasizing the importance of collaboration, this research seeks to improve cyber security overall. Therefore, ultimately, this work also benefits civil society as a whole.

References

AICPA TSC (2017). *2017 Trust Services Criteria (With Revised Points of Focus – 2022)*. https://www.aicpa-cima.com/resources/download/2017-trust-services-criteria-with-revised-points-of-focus-2022. (Visited on 12/21/2023).

Amazon Web Services (May 2022). *Amazon EC2 Secure and resizable compute capacity for virtually any workload*. https://aws.amazon.com/ec2.

– (2024a). *AWS Cloud Products*. https://docs.aws.amazon.com/. (Visited on 10/30/2024).

– (2024b). *Creating, configuring, and working with Amazon S3 buckets*. https://docs.aws.amazon.com/AmazonS3/latest/userguide/creating-buckets-s3.html. (Visited on 04/05/2024).

– (2023). *Shared Responsibility Model*. https://aws.amazon.com/compliance/shared-responsibility-model/?nc1=h_ls. (Visited on 09/22/2023).

Apple Inc. (2020). *CVE-2020-9979 Detail*. https://nvd.nist.gov/vuln/detail/CVE-2020-9979. (Visited on 05/04/2023).

Arnab, S., T. Lim, M. Carvalho, F. Bellotti, S. De Freitas, S. Louchart, N. Suttie, R. Berta, and A. De Gloria (2015). "Mapping learning and game mechanics for serious games analysis". In: *British Journal of Educational Technology* 46 (2), pp. 391–411. DOI: https://doi.org/10.1111/bjet.12113. eprint: https://bera-journals.onlinelibrary.wiley.com/doi/pdf/10.1111/bjet.12113. URL: https://bera-journals.onlinelibrary.wiley.com/doi/abs/10.1111/bjet.12113.

Atlasian (2022). *cve-2022-26134 Detail*. https://nvd.nist.gov/vuln/detail/cve-2022-26134. (Visited on 05/04/2023).

Azadegan, A., J. Riedel, and J. Baalsrud Hauge (2012). "Serious Games Adoption in Corporate Training". In: *Serious Games Development and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 74–85. DOI: 10.1007/978-3-642-33687-4_6.

Azure (2023). *Shared Responsibility Model*. https://learn.microsoft.com/en-us/azure/security/fundamentals/shared-responsibility. (Visited on 02/13/2024).

Baskerville, R. and J. Pries-Heje (2010). "Explanatory Design Theory". In: *Business & Information Systems Engineering* 2, pp. 271–282. URL: https://aisel.aisnet.org/bise/vol2/iss5/2.

Baskerville, R. and A. Wood-Harper (1998). "Diversity in information systems action research methods". In: *European Journal of Information Systems* 7.2,

pp. 90–107. ISSN: 1476-9344. DOI: 10.1057/palgrave.ejis.3000298. URL: https://doi.org/10.1057/palgrave.ejis.3000298.

Bundesamt für Sicherheit in der Informationstechnik (2023). *BSI IT-Grundschutz-Kompendium*. https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Standards-und-Zertifizierung/IT-Grundschutz/IT-Grundschutz-Kompendium/it-grundschutz-kompendium_node.html. Reguvis Fachmedien GmbH, Köln, Germany. (Visited on 11/04/2024).

– (2020). *Cloud computing C5 criteria catalogue*. https://www.bsi.bund.de/EN/Themen/Unternehmen-und-Organisationen/Informationen-und-Empfehlungen/Empfehlungen-nach-Angriffszielen/Cloud-Computing/Kriterienkatalog-C5/kriterienkatalog-c5_node.html. (Visited on 12/21/2023).

Calder, A. (May 2022). *Where security is concerned, shared accountability equals no security!* https://twitter.com/AlanPCalder/status/1522286058645430272. (Visited on 09/25/2023).

Casinillo, L. and G. Tavera (Jan. 2021). "On the Dark Side of Learning Calculus: Evidence From Agribusiness Students". In: *IJIET (International Journal of Indonesian Education and Teaching)* 5, pp. 52–60. DOI: 10.24071/ijiet.v5i1.2825.

Center for Internet Security (2020). *CIS (Center for Internet Security) Controls*. Standards. East Greenbush, NY: Center for Internet Security. (Visited on 04/11/2024).

Cimpanu, C. (2017). *Misconfigured Amazon S3 Buckets Expose Users, Companies to Stealthy MitM Attacks*. https://www.bleepingcomputer.com/news/security/misconfigured-amazon-s3-buckets-expose-users-companies-to-stealthy-mitm-attacks/. (Visited on 11/04/2024).

Cloud Security Alliance (2021). *Cloud Controls Matrix v4*. https://cloudsecurityalliance.org/artifacts/cloud-controls-matrix-v4/. (Visited on 06/07/2022).

– (2020). *Requirements for Bodies Providing STAR Certification*. https://cloudsecurityalliance.org/artifacts/requirements-for-bodies-providing-star-certification/. (Visited on 03/12/2020).

– (2017). *Security Guidance for Critical Areas of Focus in Cloud Computing v4.0*. https://cloudsecurityalliance.org/artifacts/security-guidance-v4/. (Visited on 07/26/2017).

– (2019). *Top Threats to Cloud Computing: The Egregious 11*. https://cloudsecurityalliance.org/press-releases/2019/08/09/csa-releases-new-research-top-threats-to-cloud-computing-egregious-eleven. (Visited on 04/11/2024).

Cloud Security Alliance Top Threats Working Group (2022). *Top Threats to Cloud Computing Pandemic Eleven*. https://cloudsecurityalliance.org/artifacts/top-threats-to-cloud-computing-pandemic-eleven/. (Visited on 12/21/2023).

Cloud security guidance (2023). *Cloud security shared responsibility model*. https://www.ncsc.gov.uk/collection/cloud/understanding-cloud-s

ervices/cloud-security-shared-responsibility-model/. (Visited on 04/07/2023).

Codewars (2023). *Achieve mastery through challenge - Improve your development skills by training with your peers on code kata that continuously challenge and push your coding practice.* https://www.codewars.com/. (Visited on 04/07/2023).

Coenraad, M., A. Pellicone, D. Ketelhut, M. Cukier, J. Plane, and D. Weintrop (2020). "Experiencing Cybersecurity One Game at a Time: A Systematic Review of Cybersecurity Digital Games". In: *Simulation & Gaming* 51.5, pp. 586–611. DOI: 10.1177/1046878120933312. eprint: https://doi.org/10.1177/1046878120933312. URL: https://doi.org/10.1177/1046878120933312.

Connory, M. (2023). *Software companies keep making these same cyber security mistakes.* https://isuggi.com/software-companies-keep-making-these-same-cyber-security-mistakes. (Visited on 01/13/2023).

Corbin, J. and A. Strauss (2014). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory.* URL: https://books.google.de/books?id=hZ6kBQAAQBAJ. SAGE Publications. ISBN: 9781483315683.

Crosley, J. (2020). *Qualitative Data Coding 101.* https://gradcoach.com/qualitative-data-coding-101/. (Visited on 04/23/2023).

Cybersecurity Advisory (2020). *Iran-Based Threat Actor Exploits VPN Vulnerabilities.* https://www.cisa.gov/news-events/cybersecurity-advisories/aa20-259a/. (Visited on 12/20/2023).

Darling, E. (2023). *Why SQL Developers Keep Making The Same Mistakes.* https://www.brentozar.com/archive/2018/07/why-sql-developers-keep-making-the-same-mistakes. (Visited on 01/13/2023).

Dewes, T., T. Gasiba, and T. Schreck (2022). "Understanding the Usage of IT-Security Games in the Industry and Its Mapping to Job Profiles". In: *Third International Computer Programming Education Conference (ICPEC 2022).* Vol. 102. Open Access Series in Informatics (OASIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 3:1–3:12. ISBN: 978-3-95977-229-7. DOI: 10.4230/OASIcs.ICPEC.2022.3. URL: https://drops.dagstuhl.de/entities/document/10.4230/OASIcs.ICPEC.2022.3.

Di Giulio, C., R. Sprabery, C. Kamhoua, K. Kwiat, R. Campbell, and M. Bashir (2017). "Cloud standards in comparison: Are new security frameworks improving cloud security?" In: *2017 IEEE 10th International Conference on Cloud Computing (CLOUD).* IEEE, pp. 50–57.

Dörner, R., S. Göbel, W. Effelsberg, and J. Wiemeyer (2016). *Serious Games: Foundations, Concepts and Practice.* Springer.

ECMA-404 (May 2022). *JSON format.* https://www.json.org/json-en.html. (Visited on 04/11/2024).

Fenz, S. and A. Ekelhart (2009). "Formalizing information security knowledge". In: *Proceedings of the 4th International Symposium on Information, Computer, and Communications Security.* ASIACCS '09. Sydney, Australia: Association for Computing Machinery, pp. 183–194. ISBN: 9781605583945. DOI: 10.1145/1533057.1533084. URL: https://doi.org/10.1145/1533057.1533084.

Ferro, S., A. Marrella, T. Catarci, F. Sapio, A. Parenti, and M. De Santis (2022). "AWATO: A Serious Game to Improve Cybersecurity Awareness". In: *HCI in Games*. Cham: Springer International Publishing, pp. 508–529. ISBN: 978-3-031-05637-6. DOI: https://doi.org/10.1007/978-3-031-05637-6_33.

Frampton, S. (2020). *Coding Qualitative Data: A Beginner's How-To + Examples*. https://chattermill.com/blog/coding-qualitative-data#3-steps-for-coding-qualitative-data-from-the-top-down. (Visited on 04/23/2023).

Gasiba, T. (2021). "Raising Awareness on Secure Coding in the Industry through CyberSecurity Challenges". PhD thesis. Universität der Bundeswehr München.

Gasiba, T., U. Lechner, J. Cuellar, and A. Zouitni (2020). "Ranking Secure Coding Guidelines for Software Developer Awareness Training in the Industry". In: *First International Computer Programming Education Conference (ICPEC 2020)*. Vol. 81. Open Access Series in Informatics (OASIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 11:1–11:11. ISBN: 978-3-95977-153-5. DOI: 10.4230/OASIcs.ICPEC.2020.11. URL: https://drops-dev.dagstuhl.de/entities/document/10.4230/OASIcs.ICPEC.2020.11.

Gatlan, S. (2023). *Microsoft leaks 38TB of private data via unsecured Azure storage*. https://www.bleepingcomputer.com/news/microsoft/microsoft-leaks-38tb-of-private-data-via-unsecured-azure-storage/. (Visited on 09/18/2023).

General Services Administration of United States (2019). *FedRAMP (Federal Risk and Authorization Management Program)*. https://www.fedramp.gov. Program. Washington, D.C. (Visited on 04/11/2024).

Gleasure, R. (2013). "What Is a 'Wicked Problem' for IS Research?" In: *SIG Prag Workshop on IT Artefact Design & Workpractice Improvement, 5 June, 2013, Tilburg, The Netherlands*, pp. 1–12.

Gleeson, N. and I. Walden (Jan. 2014). "'It's a Jungle Out There'?: Cloud Computing, Standards and the Law". In: *SSRN Electronic Journal*. DOI: 10.2139/ssrn.2441182.

Godsay, M. (Sept. 2015). "Article: The Process of Sentiment Analysis: A Study". In: *International Journal of Computer Applications* 126.7. Published by Foundation of Computer Science (FCS), NY, USA, pp. 26–30.

Graziotin, D, F. Fagerholm, X. Wang, and P. Abrahamsson (2018). "What happens when software developers are (un)happy". In: *Journal of Systems and Software* 140, pp. 32–47. ISSN: 0164-1212. DOI: https://doi.org/10.1016/j.jss.2018.02.041. URL: https://www.sciencedirect.com/science/article/pii/S0164121218300323.

Graziotin, D., X. Wang, and P. Abrahamsson (2015). "Do feelings matter? On the correlation of affects and the self-assessed productivity in software engineering". In: *Journal of Software: Evolution and Process* 27.7, pp. 467–487. DOI: https://doi.org/10.1002/smr.1673. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/smr.1673.

Guttman, B. and E. Roback (Oct. 1995). *An Introduction to Computer Security: the NIST Handbook*. https://www.nist.gov/publications/introduction-computer-security-nist-handbook. DOI: https://nvlpubs.nist.go

v/nistpubs/SpecialPublications/NIST.SP.800-12r1.pdf. (Visited on 04/11/2024).

Hänsch, N. and Z. Benenson (2014). "Specifying IT security awareness". In: *25th International Workshop on Database and Expert Systems Applications*. IEEE, pp. 326–330. DOI: 10.1109/DEXA.2014.71.

Harford, I. (2022). *How effective is security awareness training? Not enough*. https://www.techtarget.com/searchsecurity/feature/How-effective-is-security-awareness-training-Not-enough. (Visited on 12/20/2023).

Hart, S., A. Margheri, F. Paci, and V. Sassone (2020). "Riskio: A Serious Game for Cyber Security Awareness and Education". In: *Computers & Security* 95, p. 101827. ISSN: 0167-4048. DOI: 10.1016/j.cose.2020.101827.

Hendrix, M., A. Al-Sherbaz, and V. Bloom (Mar. 2016). "Game Based Cyber Security Training: are Serious Games suitable for cyber security training?" In: *International Journal of Serious Games*. 3.1. DOI: DOI:https://doi.org/10.17083/ijsg.v3i1.107.

Hevner, A. (2007). "A Three Cycle View of Design Science Research". In: *Scandinavian Journal of Information Systems* 19, pp. 1–6. URL: http://aisel.aisnet.org/sjis/vol19/iss2/4.

Hevner, A., S. March, and J. Park (2004). "Design Science in Information Systems Research". In: *MIS Quarterly* 28.1, pp. 75–105. DOI: https://doi.org/10.2307/25148625.

HITB CyberWeek (2023). *THIRD EDITION OF ADVERSARIES vs DEFENDERS CTF COMPETITION - NOV 18, 19 Welcoming Red Teams and Blue Teams Upcoming village and CTF at HITB CyberWeek*. https://redteamvillage.org/HITB-CyberWeek-2020-Red-vs-Blue-CTF/. (Visited on 04/07/2023).

Hofmeier, Manfred (2024). "Operation Digital Butterfly : Ein Serious-Game-basierter Ansatz zur Identifikation und Analyse von Intentionalen Bedrohungen durch Innentäter und Innentäterinnen (Malicious Insider Threats)". PhD thesis. Universität der Bundeswehr München.

ICS-CERT (2020). *CVE-2020-14494 Detail*. https://nvd.nist.gov/vuln/detail/CVE-2020-14494. (Visited on 05/04/2023).

International Society of Automation (2023a). *ANSI/ISA-62443-4-2-2018, Security for industrial automation and control systems, Part 4-2: Technical security requirements for IACS components*. https://www.isa.org/products/ansi-isa-62443-4-2-2018-security-for-industrial-au. (Visited on 01/30/2023).

– (2023b). *ANSI/ISA-62443-4-1-2018, Security for industrial automation and control systems - Part 4-1: Secure product development lifecycle requirements*. https://www.isa.org/products/ansi-isa-62443-4-1-2018-security-for-industrial-au. (Visited on 01/30/2023).

– (2023c). *ISA/IEC 62443 Series of Standards*. https://www.isa.org/standards-and-publications/isa-standards/isa-iec-62443-series-of-standards. (Visited on 01/30/2023).

Iosif, A., T. Gasiba, T. Zhao, U. Lechner, and M. Pinto-Albuquerque (2022). "A Large-Scale Study on the Security Vulnerabilities of Cloud Deployments". In: *Ubiquitous Security (UbiSec 2021)*. Singapore: Springer Singapore, pp. 171–188. ISBN: 978-981-19-0468-4. DOI: 10.1007/978-981-19-0468-4_13.

ISACA (2019). *COBIT (Control Objectives for Information and Related Technologies)*. Framework. Rolling Meadows, IL: ISACA. URL: https://www.isaca.org/resources/cobit.

ISO27001 (2017). *ISO/IEC 27001 Information Security Management*. https://www.iso.org/isoiec-27001-information-security.html. (Visited on 11/05/2021).

ISO27002 (2013). *ISO/IEC 27002:2013Information technology — Security techniques — Code of practice for information security controls*. https://www.iso.org/standard/54533.html. (Visited on 11/05/2021).

ISO27017 (2015). *ISO/IEC 27017:2015 Information technology — Security techniques — Code of practice for information security controls based on ISO/IEC 27002 for cloud services*. https://www.iso.org/standard/43757.html. (Visited on 11/05/2021).

ISO27018 (2019). *ISO/IEC 27018:2019Information technology — Security techniques — Code of practice for protection of personally identifiable information (PII) in public clouds acting as PII processors*. https://www.iso.org/standard/76559.html. (Visited on 11/05/2021).

Jenkins Project (2019). *CVE-2019-10456 Detail*. https://nvd.nist.gov/vuln/detail/CVE-2019-10456. (Visited on 05/04/2023).

Khando, K., S. Gao, S. Islam, and A. Salman (2021). "Enhancing employees information security awareness in private and public organisations: A systematic literature review". In: *Computers & Security* 106, p. 102267. ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2021.102267. URL: https://www.sciencedirect.com/science/article/pii/S0167404821000912.

Koay, A.., M. Xie, R.. Ko, C. Sterner, T. Choi, and N. Dong (2022). "SDGen: A Scalable, Reproducible and Flexible Approach to Generate Real World Cyber Security Datasets". In: *Ubiquitous Security (UbiSec 2021)*. Singapore: Springer Singapore, pp. 102–115. ISBN: 978-981-19-0468-4.

Konva (2019). *JavaScript 2D Canvas Library*. https://https://konvajs.org/. (Visited on 08/10/2023).

Kruger, H. and W. Kearney (2006). "A prototype for assessing information security awareness". In: *Computers & Security* 25.4, pp. 289–296. ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2006.02.008. URL: https://www.sciencedirect.com/science/article/pii/S0167404806000563.

Kuipers, D. and M. Fabro (2006). *Control systems cyber security: Defense in depth strategies*. Tech. rep. Idaho National Laboratory (INL).

Larson, K. (2020). "Serious Games and Gamification in the Corporate Training Environment: a Literature Review". In: *TechTrends* 64.2, pp. 319–328. ISSN: 1559-7075. DOI: 10.1007/s11528-019-00446-7. URL: https://doi.org/10.1007/s11528-019-00446-7.

M., Assante. and R. Lee (2015). "The industrial control system cyber kill chain". In: *SANS Institute InfoSec Reading Room* 1. (Visited on 04/11/2024).

Mell, P. and T. Grance (Sept. 2011). *SP 800-145 The NIST Definition of Cloud Computing*. https://csrc.nist.gov/publications/detail/sp/800-145/final. (Visited on 06/07/2022).

Mendes, R. (2021). "Moopec: A Tool for Creating Programming Problems". In: *Second International Computer Programming Education Conference (ICPEC 2021)*. Vol. 91. Open Access Series in Informatics (OASIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 9:1–9:7. ISBN: 978-3-95977-194-8. DOI: 10.4230/OASIcs.ICPEC.2021.9. URL: https://drops.dagstuhl.de/entities/document/10.4230/OASIcs.ICPEC.2021.9.

Microsoft Corporation (2020). *CVE-2020-0835 Detail*. https://nvd.nist.gov/vuln/detail/CVE-2020-0835. (Visited on 05/04/2023).

MITRE ATT&CK (2017a). *Account Manipulation*. https://attack.mitre.org/versions/v13/techniques/T1098/. (Visited on 05/04/2023).

– (2017b). *Brute Force*. https://attack.mitre.org/versions/v13/techniques/T1110/. (Visited on 05/04/2023).

– (2020a). *Cloud Infrastructure Discovery*. https://attack.mitre.org/versions/v13/techniques/T1580/. (Visited on 05/04/2023).

– (2019a). *CVE-2019-19781 Detail*. https://nvd.nist.gov/vuln/detail/CVE-2019-19781. (Visited on 05/04/2023).

– (2019b). *CVE-2019-5736 Detail*. https://nvd.nist.gov/vuln/detail/CVE-2019-5736. (Visited on 05/04/2023).

– (2020b). *CVE-2020-1206 Detail*. https://nvd.nist.gov/vuln/detail/cve-2020-1206. (Visited on 05/04/2023).

– (2021). *CVE-2021-44833 Detail*. https://nvd.nist.gov/vuln/detail/CVE-2021-44833. (Visited on 05/04/2023).

– (2019c). *Data from Cloud Storage*. https://attack.mitre.org/versions/v13/techniques/T1530/. (Visited on 05/04/2023).

– (2019d). *defacement*. https://attack.mitre.org/versions/v13/techniques/T1491/. (Visited on 05/04/2023).

– (2020c). *Denial of Service*. https://attack.mitre.org/versions/v13/techniques/T0814/. (Visited on 05/04/2023).

– (2018a). *Exploit Public-Facing Application*. https://attack.mitre.org/versions/v13/techniques/T1190/. (Visited on 05/04/2023).

– (2020d). *Impair Defenses*. https://attack.mitre.org/versions/v13/techniques/T1562/. (Visited on 05/04/2023).

– (2019e). *Implant Internal Image*. https://attack.mitre.org/versions/v13/techniques/T1525/. (Visited on 05/04/2023).

– (2020e). *MITRE ATT&CK cloud matrix*. https://attack.mitre.org/versions/v8/matrices/enterprise/cloud/. (Visited on 02/16/2021).

– (2019f). *Modify Cloud Compute Infrastructure*. https://attack.mitre.org/versions/v13/techniques/T1578/. (Visited on 05/04/2023).

– (2017c). *Network Service Discovery*. https://attack.mitre.org/versions/v13/techniques/T1046/. (Visited on 05/04/2023).

– (2023). *Network Service Discovery*. https://attack.mitre.org/techniques/T1046/. (Visited on 01/15/2023).

MITRE ATT&CK (2019g). *Resource Hijacking*. https://attack.mitre.org/versions/v13/techniques/T1496/. (Visited on 05/04/2023).
– (May 2017d). *Techniques*. https://attack.mitre.org/techniques/.
– (2019h). *Transfer Data to Cloud Account*. https://attack.mitre.org/versions/v13/techniques/T1537/. (Visited on 05/04/2023).
– (2018b). *Trusted Relationship*. https://attack.mitre.org/versions/v13/techniques/T1199/. (Visited on 05/04/2023).
– (2019i). *Unused/Unsupported Cloud Regions*. https://attack.mitre.org/versions/v13/techniques/T1535/. (Visited on 05/04/2023).
– (2017e). *Valid Accounts*. https://attack.mitre.org/versions/v13/techniques/T1078/. (Visited on 05/04/2023).
Moody, G., M. Siponen, and S. Pahnila (Mar. 2018). "Toward a Unified Model of Information Security Policy Compliance". In: *MIS Quarterly* 42.1, pp. 285–311. DOI: 10.25300/MISQ/2018/13853. URL: https://misq.org/misq/downloads/download/article/1302/.
Mueller, A. (2021). *Word-cloud*. https://pypi.org/project/wordcloud/. (Visited on 12/12/2022).
Nafea, R. and M. Almaiah (2021). "Cyber security threats in cloud: Literature review". In: *2021 International Conference on Information Technology (ICIT)*. IEEE, pp. 779–786. DOI: 10.1109/ICIT52682.2021.9491638.
National Vulnerability Database (2023a). *CVE-2020-1206*. https://nvd.nist.gov/vuln/detail/cve-2020-1206. (Visited on 01/15/2023).
– (2022). *cve-2022-1234 Detail*. https://nvd.nist.gov/vuln/detail/cve-2022-1234. (Visited on 05/04/2023).
– (2023b). *Vulnerability Metrics*. https://nvd.nist.gov/vuln-metrics/cvss. (Visited on 01/05/2023).
NIST (2020). *NIST SP 800-53 Rev. 5 Security and Privacy Controls for Information Systems and Organizations*. https://csrc.nist.gov/pubs/sp/800/53/r5/upd1/final. (Visited on 12/21/2023).
North American Electric Reliability Corporation (2020). *Critical Infrastructure Protection Reliability Standards*. Standards. Atlanta, GA: NERC-CIP. URL: https://www.nerc.com/pa/Stand/Pages/CIPStandards.aspx.
Oracle (2021). *CVE-2021-2317 Detail*. https://nvd.nist.gov/vuln/detail/CVE-2021-2317. (Visited on 05/04/2023).
PCI DSS (2022). *PCI Security Standards Council (PCI SSC)*. https://www.pcisecuritystandards.org/. (Visited on 12/21/2023).
Petri, G., C G. von Wangenheim, and F. Borgatto (2016). "MEEGA+: an evolution of a model for the evaluation of educational games". In: *INCoD/GQS* 3, pp. 1–40.
Petrik, D. and G. Herzwurm (2019). "iIoT ecosystem development through boundary resources: a Siemens MindSphere case study". In: *Proceedings of the 2nd ACM SIGSOFT International Workshop on Software-Intensive Business: Start-ups, Platforms, and Ecosystems*, pp. 1–6.
Poston, H. (2023). *The Need For Secure Coding*. https://securityboulevard.com/2019/11/the-need-for-secure-coding/. (Visited on 01/13/2023).

Python3 (May 2022). *Python is a programming language that lets you work quickly and integrate systems more effectively*. https://www.python.org/. (Visited on 04/11/2023).

Radigan, D. (2024). *Agile roadmaps: build, share, use, evolve*. https://www.atlassian.com/agile/product-management/roadmaps. (Visited on 04/05/2024).

Raza, M. (2023). *The Shared Responsibility Model for Security in The Cloud (IaaS, PaaS & SaaS)*. https://www.splunk.com/en_us/blog/learn/shared-responsibility-model.html. (Visited on 10/19/2023).

Scheffler, M. (2019). *Datensicherheit in der Cloud: Best Practices Gegen Man-in-the-Cloud-Attacken*. https://ap-verlag.de/datensicherheit-in-der-cloud-best-practices-gegen-man-in-the-cloud-attacken/50038/. (Visited on 08/10/2023).

Schneier, B. (2023). *Software Developers and Security*. https://www.schneier.com/blog/archives/2019/07/software_develo.html. (Visited on 01/13/2023).

Sebayan, D. (Mar. 2021). *Another S3 Bucket Leads to Breach of 50k Patient Records*. https://securityboulevard.com/2021/03/another-s3-bucket-leads-to-breach-of-50k-patient-records/. (Visited on 09/25/2023).

Shostack, A. (2021). *Tabletop Security Games & Cards*. https://https://shostack.org/games.html. (Visited on 02/16/2021).

Siemens AG (2021). *CVE-2021-33719 Detail*. https://nvd.nist.gov/vuln/detail/CVE-2021-33719. (Visited on 05/04/2023).

Siponen, M. and A. Vance (2010). "Neutralization: New insights into the problem of employee information systems security policy violations". In: *MIS Quarterly* 34.3, pp. 487–502. DOI: https://doi.org/10.2307/25750688.

Sisodia, J and M. Khan (2022). "Understanding the Shared Responsibilities Model in Cloud Services". In: *ISACA Journal* 3. URL: https://www.isaca.org/resources/isaca-journal/issues/2022/volume-3/understanding-the-shared-responsibilities-model-in-cloud-services.

Spearman, D. (Jan. 1904). "The Proof and Measurement of Association between Two Things". In: *The American Journal of Psychology* 15, pp. 72–101. DOI: https://doi.org/10.2307/1412159.

Statista (2022). *Cloud Computing Market Size, Share & Trends Analysis Report By Service (SaaS, IaaS), By End-use (BFSI, Manufacturing), By Deployment (Private, Public), By Enterprise Size (Large, SMEs), And Segment Forecasts, 2023 - 2030*. https://www.grandviewresearch.com/industry-analysis/cloud-computing-industry. (Visited on 04/28/2023).

Švábenskỳ V. Vykopal, J., M. Cermak, and M. Laštovička (2018). "Enhancing cybersecurity skills by creating serious games". In: *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*, pp. 194–199. DOI: https://doi.org/10.48550/arXiv.1804.03567.

Tang, Y., D. Zhang, W. Liang, K. Li, and N. Sukhija (2022). "Active Malicious Accounts Detection with Multimodal Fusion Machine Learning Algorithm". In: *Ubiquitous Security (UbiSec 2021)*. Singapore: Springer Singapore, pp. 38–52. ISBN: 978-981-19-0468-4.

Thompson, M. and C. Irvine (2011). "Active Learning with the CyberCIEGE Video Game". In: *Proceedings of the 4th Conference on Cyber Security Experimentation and Test*. CSET'11. San Francisco, CA: USENIX Association, p. 10.

Trendmicro (Nov. 2017). *A Misconfigured Amazon S3 Exposed Almost 50 Thousand PII in Australia*. Available Online. URL: https://www.trendmicro.com/vinfo/us/security/news/virtualization-and-cloud/a-misconfigured-amazon-s3-exposed-almost-50-thousand-pii-in-australia (visited on 10/30/2024).

UpGuard Team (2017). *Black Box, Red Disk: How Top Secret NSA and Army Data Leaked Online*. https://www.upguard.com/breaches/cloud-leak-inscom. (Visited on 08/10/2023).

Vance, A., M. Siponen, and S. Pahnila (2012). "Motivating IS security compliance: Insights from Habit and Protection Motivation Theory". In: *Information & Management* 49.3, pp. 190–198. ISSN: 0378-7206. DOI: https://doi.org/10.1016/j.im.2012.04.002. URL: https://www.sciencedirect.com/science/article/pii/S0378720612000328.

Vaughan-Nichols, S. (2023). *No Love Lost Between Security Specialists and Developers*. https://www.zdnet.com/article/no-love-lost-between-security-specialists-and-developers/. (Visited on 01/13/2023).

Veen, Annelinde (2020). "Teamplay—Streamline Clinical Operations to Unlock Productivity Gains". In: *Siemens Healthineers, Erlangen, Germany* 14.

Vom Brocke, J. and A. Maedche (2019). "The DSR grid: six core dimensions for effectively planning and communicating design science research projects". In: *Electronic Markets* 29, pp. 379–385.

Walsham, M. (2024). "Cloud security: sharing is caring". In: *Computer Fraud & Security* 2024.3. DOI: 10.12968/S1361-3723(24)70012-8.

Warrior, Secure Code (Jan. 2021). *Whitepaper: Empowering developers to write secure code*. Tech. rep. Secure Code Warrior. URL: https://discover.securecodewarrior.com/Empowering-Developers.html.

Zhao, T, T. Gasiba, U. Lechner, and M. Pinto-Albuquerque (Aug. 2024). "Thriving in the era of hybrid work: Raising cybersecurity awareness using serious games in industry trainings (Journal First)". In: *36th International Conference on Software Engineering Education and Training (CSEE&T)*. Vol. 1, p. 293.

Zhao, T, U. Lechner, M. Pinto-Albuquerque, D. Ongu, and T. Gasiba (Aug. 2024). "A Deep Dive Into CATS Evaluator Algorithm: Quantification Of The Probability in Serious Game Cloud Security Defense Scenarios". In: *36th International Conference on Software Engineering Education and Training (CSEE&T)*. Vol. 1, pp. 227–231.

Zhao, T., T. Gasiba, U. Lechner, and M. Pinto-Albuquerque (2021a). "Exploring a Board Game to Improve Cloud Security Training in Industry". In: *Second International Computer Programming Education Conference (ICPEC 2021)*. Vol. 91. Open Access Series in Informatics (OASIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 11:1–11:8. ISBN: 978-3-95977-194-8. DOI: 10.4230/OASIcs.ICPEC.2021.11.

– (2021b). "Raising Awareness about Cloud Security in Industry through a Board Game". In: *Information, Special issue Future Trends in Computer Programming Education* 12.11. ISSN: 2078-2489. DOI: 10.3390/info12110482.
– (Apr. 2024). "Thriving in the era of hybrid work: Raising cybersecurity awareness using serious games in industry trainings". In: *Journal of Systems and Software* 210, p. 111946. DOI: 10.1016/j.jss.2023.111946.

Zhao, T., U. Lechner, M. Pinto-Albuquerque, and E. Ata (2022). "Cloud of Assets and Threats: A Playful Method to Raise Awareness for Cloud Security in Industry". In: *Third International Computer Programming Education Conference (ICPEC 2022)*. Vol. 102. Open Access Series in Informatics (OASIcs). Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 6:1–6:13. ISBN: 978-3-95977-229-7. DOI: 10.4230/OASIcs.ICPEC.2022.6.

Zhao, T., U. Lechner, M. Pinto-Albuquerque, E. Ata, and T. Gasiba (2023). "CATS: A Serious Game in Industry Towards Stronger Cloud Security". In: *Ubiquitous Security*. Vol. 1768. Singapore: Springer Nature Singapore, pp. 64–82. ISBN: 978-981-99-0272-9.

Zhao, T., U. Lechner, M. Pinto-Albuquerque, and D. Ongu (2023). "An ontology-based model for evaluating cloud attack scenarios in CATS – a serious game in cloud security". In: *2023 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, pp. 1–9. DOI: 10.1109/ICE/ITMC58018.2023.10332371.

Zhao, T., U. Lechner, M. Pinto-Albuquerque, D. Ongu, and T. Gasiba (2024). "COPYCAT: Applying Serious Games in Industry for Defending Supply Chain Attack". In: *Innovations for Community Services*. Cham: Springer Nature Switzerland, pp. 321–336.

Appendix

Cloud Game: Scenario 1

1 Mission Statement

Your task in this challenge is to secure the cloud assets against a certain kill chain by using defense action cards to build a defense plan. Each game consists of an attack scenario that comprises three steps of attack. Your goal is to select defense cards and place them correctly in either the owner role or the manager role to defend against the attack. When you finish planning, hit "Submit" to evaluate your defense. Remember there is no 100% security. The goal is to successfully stop the attack with a 90% of chance. Understanding the score evaluator will help you create better strategies and improve your gameplay. Let's dive in!

2 How it works?

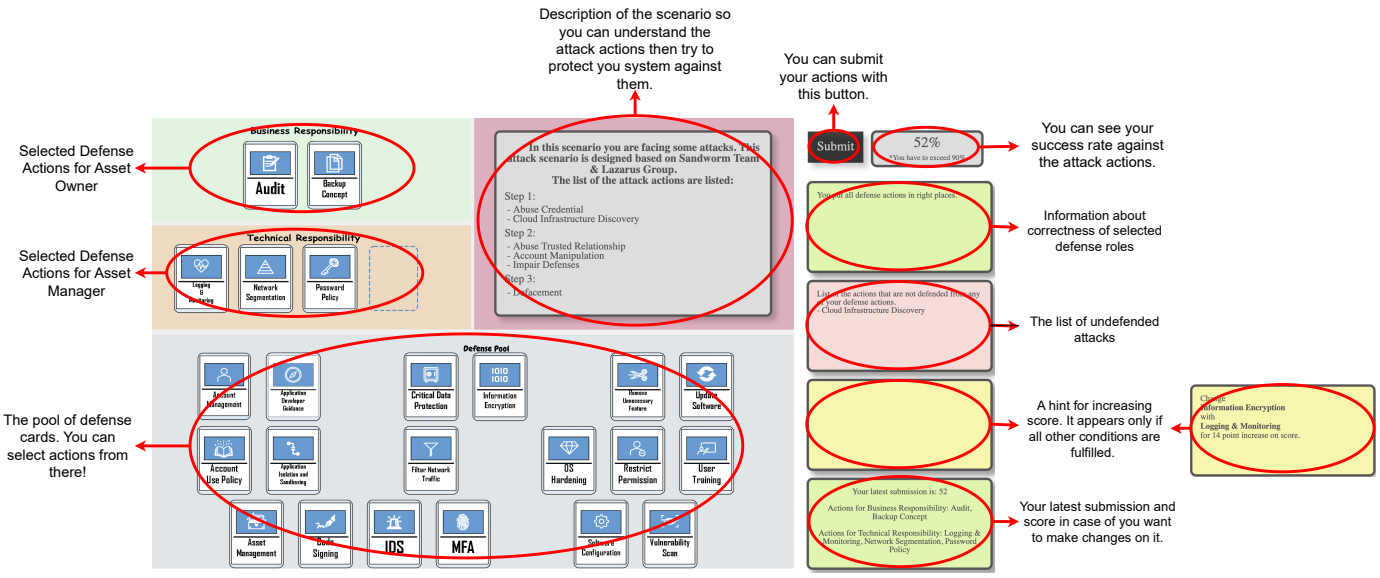


Figure 1: A screen shot of the game interface.

- The interface shown in Figure 1 is the game board. At the bottom "Defense Pool" there are 23 cards to choose from. By dragging the cards to "Responsibility of Asset Owner" or "Responsibility of Asset Manager", you assign the job to an asset owner or asset manager - cloud security is a team work!
- The text in the red area describes the attack your cloud asset is facing with. The attacker takes a step-by-step approach.
- By hitting the "Submit" button, your defense plan will be evaluated and the chance of your defense to withstand the given attack is shown on the top right corner.

- Once the chance is higher than 90%, all defenses placed in correct role, and each attack card is affected by at least one defense card, a pop-up with give you the flag.
- The first Info-Board on the right show the feedback from evaluator: have you assigned the defense job to the right role?
- The second Info-Boards on the right shows which attack cards do not affected by any of your defenses.
- The third Info-Boards on the right gives you a hint to increase your final result.
- The last Info-Board on the right shows your lasted submission just as a reference for you.Is there any attacks that currently not covered?

3 What happened in scenario 1?

This scenario is revised based on activities from the notorious threat groups Lazarus Group and Sandworm Team. Here goes the attack:

- Step 1: Gain Access
  In this step the attackers tries to collect information to gain an initial access to the system. They have two attack actions in parallel:
  - Abuse credential The attackers obtain and abuses account credentials to access the system.
  - Cloud Infrastructure Discovery They also try to discover resources that are available within the environment.
- Step 2: Launch Attack
  In this step the attackers tries to move towards their victim and launch an attack to cause damage. They have three attack actions in parallel:
  - Abuse Trusted Relationship Attackers breach the organization to access the protect resource in the environment.
  - Account Manipulation To maintain access to victim system, attackers manipulate the stolen account to open a backdoor.
  - Impair Defenses Attackers maliciously modify the components of a victim environment in order to hinder defensive mechanisms.
- Step 3: Make Impact
  In this step is the end move of the attacker. They decide how they can get the most out of their attack.
  - Defacement Attacker modifies the homepage of enterprise network to cause a panic and gain fame.

Cloud Game: Scenario 2

1 Mission Statement

Your task in this challenge is to secure the cloud assets against a certain kill chain by using defense action cards to build a defense plan. Each game consists of an attack scenario that comprises three steps of attack. Your goal is to select defense cards and place them correctly in either the owner role or the manager role to defend against the attack. When you finish planning, hit "Submit" to evaluate your defense. Remember there is no 100% security. The goal is to successfully stop the attack with a 90% of chance. Understanding the score evaluator will help you create better strategies and improve your gameplay. Let's dive in!

2 How it works?

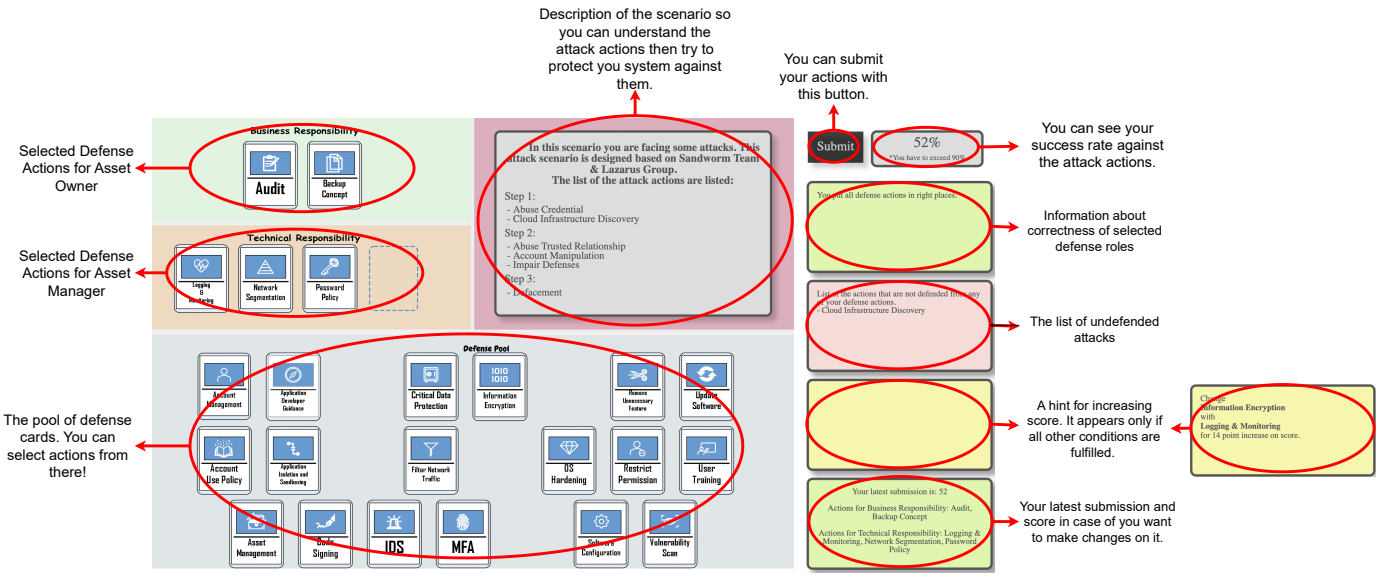


Figure 1: A screen shot of the game interface.

- The interface shown in Figure 1 is the game board. At the bottom "Defense Pool" there are 23 cards to choose from. By dragging the cards to "Responsibility of Asset Owner" or "Responsibility of Asset Manager", you assign the job to an asset owner or asset manager - cloud security is a team work!
- The text in the red area describes the attack your cloud asset is facing with. The attacker takes a step-by-step approach.
- By hitting the "Submit" button, your defense plan will be evaluated and the chance of your defense to withstand the given attack is shown on the top right corner.

- Once the chance is higher than 90%, all defenses placed in correct role, and each attack card is affected by at least one defense card, a pop-up with give you the flag.
- The first Info-Board on the right show the feedback from evaluator: have you assigned the defense job to the right role?
- The second Info-Boards on the right shows which attack cards do not affected by any of your defenses.
- The third Info-Boards on the right gives you a hint to increase your final result.
- The last Info-Board on the right shows your lasted submission just as a reference for you.Is there any attacks that currently not covered?

3 What happened in scenario 2?

This scenario is revised based on activities from the panic-stricken threat group TeamTNT and their vicious malware Hildegard, as well as Lucifer. Here goes the attack:

- Step 1: Gain Access
  In this step the attackers tries to collect information to gain an initial access to the system. They have two attack actions in parallel:
  - Network Service Discovery The attackers try to get a listing of vulnerable remote hosts.
  - Cloud Infrastructure Discovery They also try to discover resources that are available within the environment.
- Step 2: Launch Attack
  In this step the attackers tries to move towards their victim and launch an attack to cause damage. They have three attack actions in parallel:
  - Impair Defenses Attackers maliciously modify the components of a victim environment in order to hinder defensive mechanisms.
  - Abuse Trusted Relationship Attackers breach the organization to access the protect resource in the environment.
  - Infrastructure Manipulation Attackers creates infrastructure to take advantage of the computing power of the resources.
- Step 3: Make Impact
  In this step is the end move of the attacker. They decide how they can get the most out of their attack.
  - Resource Hijacking Attackers leverage the stolen resource to earn virtual currency.

Cloud Game: Scenario 3

1 Mission Statement

Your task in this challenge is to secure the cloud assets against a certain kill chain by using defense action cards to build a defense plan. Each game consists of an attack scenario that comprises three steps of attack. Your goal is to select defense cards and place them correctly in either the owner role or the manager role to defend against the attack. When you finish planning, hit "Submit" to evaluate your defense. Remember there is no 100% security. The goal is to successfully stop the attack with a 90% of chance. Understanding the score evaluator will help you create better strategies and improve your gameplay. Let's dive in!

2 How it works?

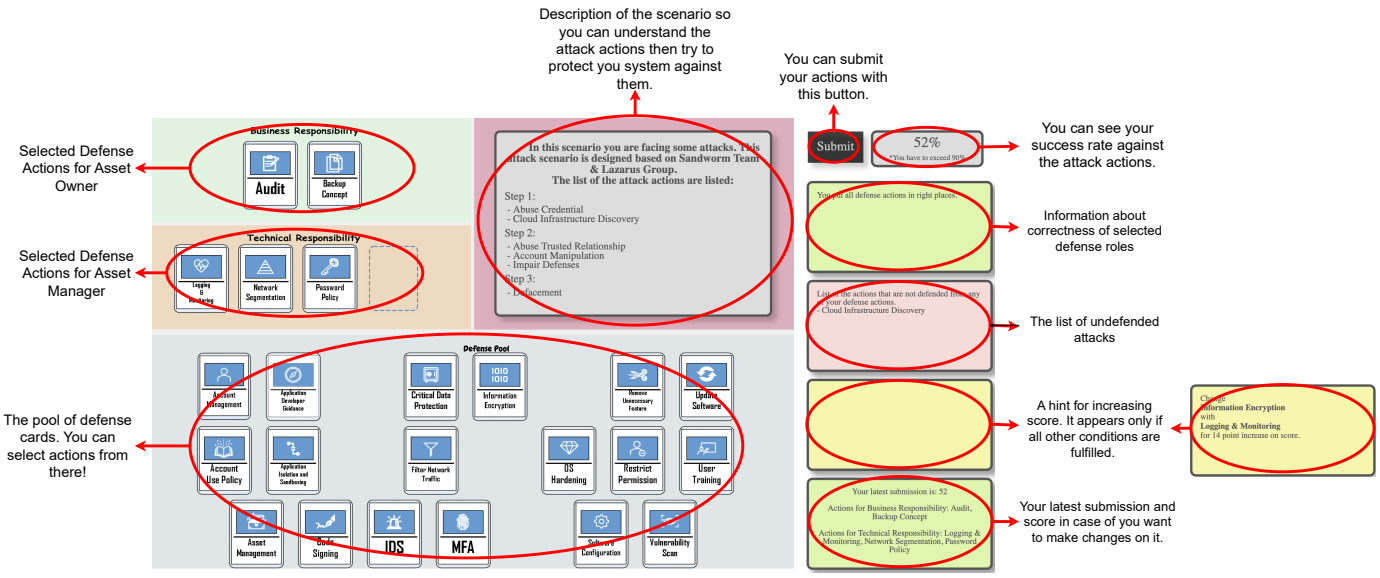


Figure 1: A screen shot of the game interface.

- The interface shown in Figure 1 is the game board. At the bottom "Defense Pool" there are 23 cards to choose from. By dragging the cards to "Responsibility of Asset Owner" or "Responsibility of Asset Manager", you assign the job to an asset owner or asset manager - cloud security is a team work!
- The text in the red area describes the attack your cloud asset is facing with. The attacker takes a step-by-step approach.
- By hitting the "Submit" button, your defense plan will be evaluated and the chance of your defense to withstand the given attack is shown on the top right corner.

- Once the chance is higher than 90%, all defenses placed in correct role, and each attack card is affected by at least one defense card, a pop-up with give you the flag.
- The first Info-Board on the right show the feedback from evaluator: have you assigned the defense job to the right role?
- The second Info-Boards on the right shows which attack cards do not affected by any of your defenses.
- The third Info-Boards on the right gives you a hint to increase your final result.
- The last Info-Board on the right shows your lasted submission just as a reference for you.Is there any attacks that currently not covered?

3 What happened in scenario 3?

This scenario is revised based on activities from the well-known threat group APT 28, APT 29 and menuPass. Here goes the attack:

- Step 1: Gain Access
  In this step the attackers tries to collect information to gain an initial access to the system. They have two attack actions in parallel:
  - Exploit Public-facing Application Attackers try to take advantage of a vulnerable internet-facing computer.
  - Brute Force Interface is protect by password. The attackers apply brute force attack to crack it.
- Step 2: Launch Attack
  In this step the attackers tries to move towards their victim and launch an attack to cause damage. They have three attack actions in parallel:
  - Abuse Trusted Relationship Attackers breach the organization to access the protect resource in the environment.
  - Cloud Storage Breach Attackers steal data from cloud storage object which is mis-configured.
  - Impair Defenses Attackers maliciously modify the components of a victim environment in order to hinder defensive mechanisms.
- Step 3: Make Impact
  In this step is the end move of the attacker. They decide how they can get the most out of their attack.
  - Denial of Service Attackers exhaust the resources to cause a disruption in manufacturing.

Cloud Game: Scenario 4

1 Mission Statement

Your task in this challenge is to secure the cloud assets against a certain kill chain by using defense action cards to build a defense plan. Each game consists of an attack scenario that comprises three steps of attack. Your goal is to select defense cards and place them correctly in either the owner role or the manager role to defend against the attack. When you finish planning, hit "Submit" to evaluate your defense. Remember there is no 100% security. The goal is to successfully stop the attack with a 90% of chance. Understanding the score evaluator will help you create better strategies and improve your gameplay. Let's dive in!

2 How it works?

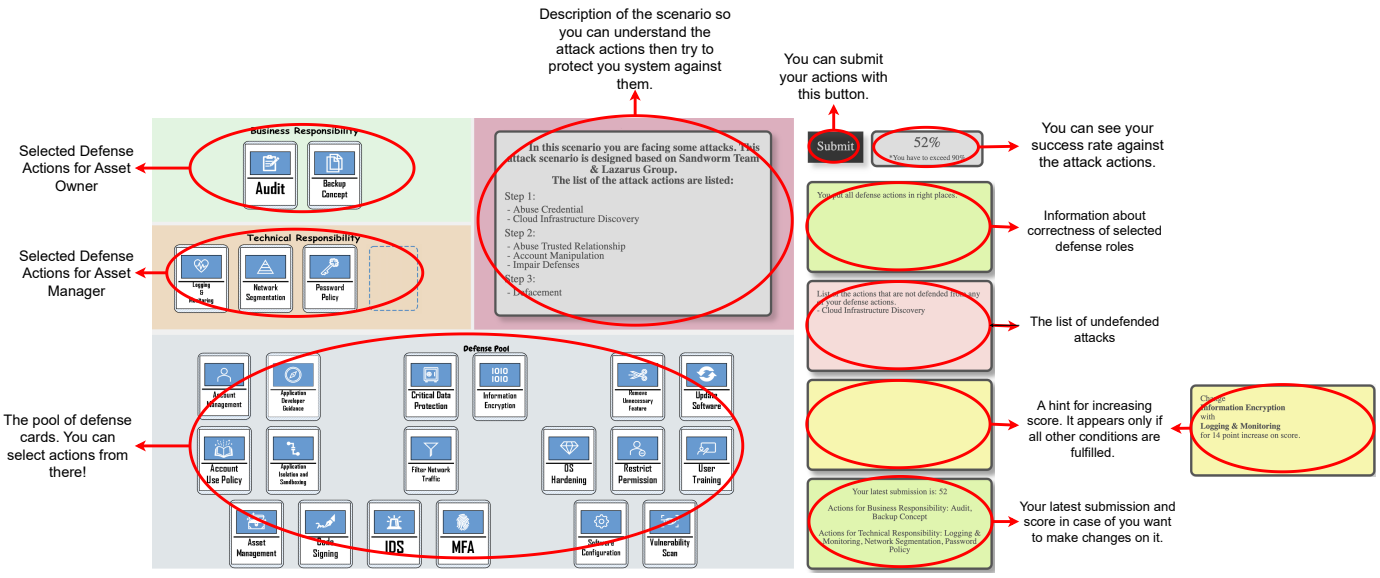


Figure 1: A screen shot of the game interface.

- The interface shown in Figure 1 is the game board. At the bottom "Defense Pool" there are 23 cards to choose from. By dragging the cards to "Responsibility of Asset Owner" or "Responsibility of Asset Manager", you assign the job to an asset owner or asset manager - cloud security is a team work!
- The text in the red area describes the attack your cloud asset is facing with. The attacker takes a step-by-step approach.
- By hitting the "Submit" button, your defense plan will be evaluated and the chance of your defense to withstand the given attack is shown on the top right corner.

- Once the chance is higher than 90%, all defenses placed in correct role, and each attack card is affected by at least one defense card, a pop-up with give you the flag.
- The first Info-Board on the right show the feedback from evaluator: have you assigned the defense job to the right role?
- The second Info-Boards on the right shows which attack cards do not affected by any of your defenses.
- The third Info-Boards on the right gives you a hint to increase your final result.
- The last Info-Board on the right shows your lasted submission just as a reference for you.Is there any attacks that currently not covered?

3 What happened in scenario 4?

This scenario is revised based on multiple hacking activities against cloud assets, making use of the ransomware Pysa. There involved threat groups includes, but not limited to: Pysa, APT39, APT41, Fox Kitten, Rocke, Operation Wocao. Here goes the attack:

- Step 1: Gain Access
  In this step the attackers tries to collect information to gain an initial access to the system. They have two attack actions in parallel:
  - Exploit Public-facing Application Attackers try to take advantage of a vulnerable internet-facing computer.
  - Network Service Discovery The attackers try to get a listing of vulnerable remote hosts.
- Step 2: Launch Attack
  In this step the attackers tries to move towards their victim and launch an attack to cause damage. They have three attack actions in parallel:
  - Cloud Storage Breach Attackers steal data from cloud storage object which is mis-configured.
  - Impair Defenses Attackers maliciously modify the components of a victim environment in order to hinder defensive mechanisms.
  - Monitoring Escaping Attacker exfiltrate data by transferring the data to another cloud account they control on the same service to avoid typical file transfers/downloads and network-based exfiltration detection
- Step 3: Make Impact
  In this step is the end move of the attacker. They decide how they can get the most out of their attack.
  - Resource Hijacking Attackers leverage the stolen resource to earn virtual currency and blackmail the victim to pay ransom, otherwise the stolen data will be exposed.

Cloud Game: Scenario 5

1 Mission Statement

Your task in this challenge is to secure the cloud assets against a certain kill chain by using defense action cards to build a defense plan. Each game consists of an attack scenario that comprises three steps of attack. Your goal is to select defense cards and place them correctly in either the owner role or the manager role to defend against the attack. When you finish planning, hit "Submit" to evaluate your defense. Remember there is no 100% security. The goal is to successfully stop the attack with a 90% of chance. Understanding the score evaluator will help you create better strategies and improve your gameplay. Let's dive in!

2 How it works?

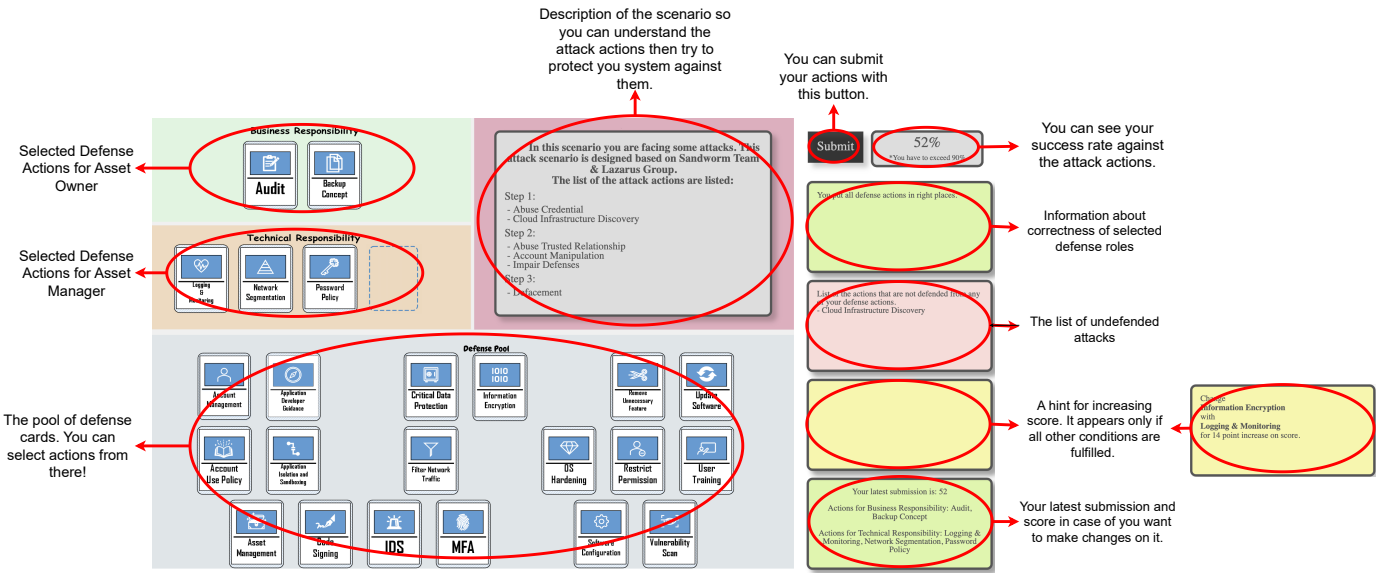


Figure 1: A screen shot of the game interface.

- The interface shown in Figure 1 is the game board. At the bottom "Defense Pool" there are 23 cards to choose from. By dragging the cards to "Responsibility of Asset Owner" or "Responsibility of Asset Manager", you assign the job to an asset owner or asset manager - cloud security is a team work!
- The text in the red area describes the attack your cloud asset is facing with. The attacker takes a step-by-step approach.
- By hitting the "Submit" button, your defense plan will be evaluated and the chance of your defense to withstand the given attack is shown on the top right corner.

- Once the chance is higher than 90%, all defenses placed in correct role, and each attack card is affected by at least one defense card, a pop-up with give you the flag.
- The first Info-Board on the right show the feedback from evaluator: have you assigned the defense job to the right role?
- The second Info-Boards on the right shows which attack cards do not affected by any of your defenses.
- The third Info-Boards on the right gives you a hint to increase your final result.
- The last Info-Board on the right shows your lasted submission just as a reference for you.Is there any attacks that currently not covered?

3 What happened in scenario 5?

This scenario is designed based on the historically most-chosen cards from the previous player-versus-player mode. Traditionally, the attacker teams have a better chance to win, but let's try to beat them today! Here goes the attack:

- Step 1: Gain Access
  In this step the attackers tries to collect information to gain an initial access to the system. They have two attack actions in parallel:
  - Abuse credential The attackers obtain and abuses account credentials to access the system.
  - Cloud Infrastructure Discovery They also try to discover resources that are available within the environment.
- Step 2: Launch Attack
  In this step the attackers tries to move towards their victim and launch an attack to cause damage. They have three attack actions in parallel:
  - Abuse Trusted Relationship Attackers breach the organization to access the protect resource in the environment.
  - Monitoring Escaping Attacker exfiltrate data by transferring the data to another cloud account they control on the same service to avoid typical file transfers/downloads and network-based exfiltration detection
  - Impair Defenses Attackers maliciously modify the components of a victim environment in order to hinder defensive mechanisms.
- Step 3: Make Impact
  In this step is the end move of the attacker. They decide how they can get the most out of their attack.
  - Resource Hijacking Attackers leverage the stolen resource to earn virtual currency and blackmail the victim to pay ransom, otherwise the stolen data will be exposed.

Cloud Game: Scenario 6

1 Mission Statement

Your task in this challenge is to secure the cloud assets against a certain kill chain by using defense action cards to build a defense plan. Each game consists of an attack scenario that comprises three steps of attack. Your goal is to select defense cards and place them correctly in either the owner role or the manager role to defend against the attack. When you finish planning, hit "Submit" to evaluate your defense. Remember there is no 100% security. The goal is to successfully stop the attack with a 90% of chance. Understanding the score evaluator will help you create better strategies and improve your gameplay. Let's dive in!

2 How it works?

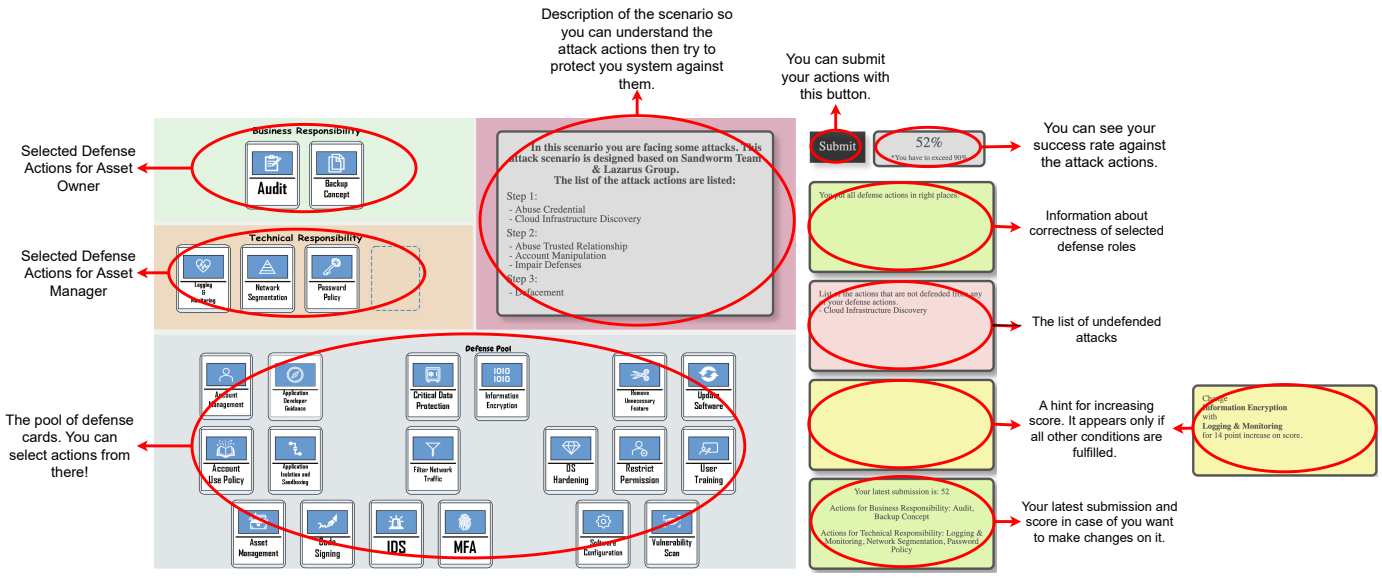


Figure 1: A screen shot of the game interface.

- The interface shown in Figure 1 is the game board. At the bottom "Defense Pool" there are 23 cards to choose from. By dragging the cards to "Responsibility of Asset Owner" or "Responsibility of Asset Manager", you assign the job to an asset owner or asset manager - cloud security is a team work!
- The text in the red area describes the attack your cloud asset is facing with. The attacker takes a step-by-step approach.
- By hitting the "Submit" button, your defense plan will be evaluated and the chance of your defense to withstand the given attack is shown on the top right corner.

- Once the chance is higher than 90%, all defenses placed in correct role, and each attack card is affected by at least one defense card, a pop-up with give you the flag.
- The first Info-Board on the right show the feedback from evaluator: have you assigned the defense job to the right role?
- The second Info-Boards on the right shows which attack cards do not affected by any of your defenses.
- The third Info-Boards on the right gives you a hint to increase your final result.
- The last Info-Board on the right shows your lasted submission just as a reference for you.Is there any attacks that currently not covered?

3 What happened in scenario 6?

This scenario is a difficult one. The attacks actions come from different angle, please try you best to build the defense plan and protect your cloud asset. Here goes the attack:

- Step 1: Gain Access
  In this step the attackers tries to collect information to gain an initial access to the system. They have two attack actions in parallel:
  - Network Service Discovery The attackers try to get a listing of vulnerable remote hosts.
  - Cloud Infrastructure Discovery They also try to discover resources that are available within the environment.
- Step 2: Launch Attack
  In this step the attackers tries to move towards their victim and launch an attack to cause damage. They have three attack actions in parallel:
  - Exploit Unused Region Attacker creates cloud instances in unused geo-graphic service regions in order to evade detection.
  - Cloud Storage Breach Attackers steal data from cloud storage object which is mis-configured.
  - Abuse Trusted Relationship Attackers breach the organization to access the protect resource in the environment.
- Step 3: Make Impact
  In this step is the end move of the attacker. They decide how they can get the most out of their attack.
  - Resource Hijacking Attackers leverage the stolen resource to earn virtual currency and blackmail the victim to pay ransom, otherwise the stolen data will be exposed.