FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING

Anssi Antila
Jouni Annamaa

HACK THE ROOM: AN AUGMENTED REALITY GAME FOR NON-EXPERTS TO LEARN ETHICAL HACKING

Bachelor's Thesis
Degree Programme in Computer Science and Engineering
June 2022

Antila A., Annamaa J. (2022) Hack the Room: An Augmented Reality Game for Non-Experts to Learn Ethical Hacking. University of Oulu, Degree Programme in Computer Science and Engineering, 51 p.

ABSTRACT

The shortage of cybersecurity skills caused by a widespread talent drought is having a significant economic impact on organizations globally. Several initiatives have been implemented to address this deficit, providing new educational pathways for novice and advanced students. Recently, ethical hacking gamification platforms and Capture the Flag (CTF) online games have risen in popularity, offering fun and engaging content that motivate beginners to acquire offensive and defensive cybersecurity skills. However, the use of augmented reality (AR) applications for cybersecurity skill development remains mostly unexplored.

Against this backdrop, the overall aim of the thesis is to examine whether CTF games in AR can improve learning outcomes in information security and enhance security situational awareness. Specifically, we explore how AR gamification impacts training and overall experience in the context of ethical hacking tasks. To achieve this, we have created *Hack the Room*, which is an ethical hacking game developed in Unity, where players use Linux terminals to solve CTF-style tasks. The game can be used for learning key cybersecurity concepts vital for organizations, and target users who have no previous cybersecurity experience, and need to be retrained for future-proofing organizations.

In the game, the player has to use simple simple Linux terminal commands like listing files in directories and reading files stored in virtual machines hosted in the cloud (CSC Pouta) to reach the predetermined tasks. Each playthrough lasts 20 minutes and features three tasks. The game can be modified or made more difficult by changing the tasks in the virtual machine. The main goal of the game is to complete all of the tasks in the game.

Our gamification concept was evaluated in a field experiment that included six participants divided into two groups, an expert group (N=3) and a non-expert group (N=3). The expert group responded to a questionnaire that assessed their situational awareness during the game, while the non-expert group responded to a questionnaire that evaluated learning outcomes. The participants reported positive learning outcomes and high situational awareness after playing the game.

Keywords: ethical hacking, gamification, augmented reality, escape rooms, capture the flag (CTF), cybersecurity, information security.

Antila A., Annamaa J. (2022) Hack the room: An Augmented Reality game for non-experts to learn ethical hacking. Oulun yliopisto, Tietotekniikan tutkinto-ohjelma, 51 s.

TIIVISTELMÄ

Pula tietoturvaosaamisesta vaikuttaa taloudellisesti organisaatioihin maailmanlaajuisesti. Tämän puutteen korjaamiseksi on tehty useita aloitteita, joissa tarjotaan oppipolkuja aloitteleville sekä edistyneemmille oppillaille. Eettisen hakkeroinnin pelillistämisalustat sekä Capture the Flag- (CTF) (suom. lipunryöstö) verkkopelit ovat lisänneet suosiotaan viime vuosina ja ne tarjoavat hyvän mahdollisuuden vasta-alkajille opetella tietoturvahyökkäämistä ja -puolustamista. Lisätyn todellisuuden hyödyntämistä tietoturvakoulutuksessa ei ole kuitenkaan tutkittu laajalti.

Tässä kandidaatin tutkinnossa käsitellään lisätyn todellisuuden hyödyntämistä CTF-peleissä sekä sitä, miten lisätty todellisuus vaikuttaa tietoturvallisuuden ja turvallisuuden tilannetietoisuuden oppimiseen. Käsittelemme erityisesti, miten lisätyn todellisuuden pelillistäminen vaikuttaa harjoitteluun sekä yleiseen kokemukseen eettisissä hakkerointitehtävissä. Tämän mahdollistamiseksi loimme *Hack the Roomin*, joka on Unityssä kehitetty kyberturvallisuuspeli, jossa pelaajat käyttävät Linux-terminaaleja läpäistäkseen lipunryöstö-tyyppisiä tehtäviä. Sitä voidaan käyttää työkaluna henkilöiden tietoturvaan tutustuttamiseen, kouluttamiseen ja uudelleen opettamiseen.

Pelin tehtävät koostuivat yksinkertaisista tehtävistä, joissa käytettiin Linux-komentoja, kuten tiedostojen listaamista ja -lukemista. Jokainen pelikerta on 20 minuutin pituinen ja sisältää kolme tehtävää. Peliä voi muokata tarpeiden mukaan, esimerkiksi nostaa vaikeustasoa muuttamalla pelin virtuaalikonetta. Pelin käyttämä virtuaalikone sijaitsee CSC Pouta-palvelimella.

Kehittämämme pelillistämiskonsepti evaluoitiin kenttäkokeella. Kokeessa oli 6 osallistujaa, jotka jaettiin kahteen ryhmään. Ryhmät koostuivat asiantuntijoista ja henkilöistä, joilla ei ollut aiempaa kokemusta eettisestä hakkeroinnista. Asiantuntijoiden ryhmä vastasi kyselyyn, joka mittasi heidän tilannetietoisuuttaan ja toinen ryhmä kyselyyn, joka mittasi heidän oppimistaan pelissä. Kenttäkoe osoitti sekä positiivisia oppimistuloksia, että korkeaa tilannetietoisuutta pelissä.

Avainsanat: eettinen hakkerointi, pelillistäminen, lisätty todellisuus, pakohuoneet, lipunryöstö, tietoturva

TABLE OF CONTENTS

FOREWORD

This Bachelor's thesis and project was done for the Applied Computing Project 1 (ACP1), 521041A in University of Oulu. In addition to the main authors Anssi Antila and Jouni Annamaa, Jaakko Ohrankämmen contributed to this thesis and project. We would like to thank our thesis supervisor Dr. Panos Kostakos, the teaching assistants Mikko Korkiakoski and Saeid Sheikhi, the ACP1 course coordinator Timo Ojala and all of the participants in our field experiment. Without these people this thesis would have not been possible to write.

Oulu, June 1st, 2022

Anssi Antila
Jouni Annamaa

LIST OF ABBREVIATIONS AND SYMBOLS

CTF	Capture the flag
UI	User Interface
AR	Augmented reality
VR	Virtual reality
HMD	Head mounted display
LiDAR	Light detection and ranging
UNSDG	United Nations Sustainable Development Goals
CCDCOE	Cooperative Cyber Defence Centre of Excellence
IP	Internet Protocol
VM	Virtual machine
SSH	Secure Shell Protocol

1. INTRODUCTION

Over the past years, the general public has become concerned about cybersecurity; however, the average computer users are unaware of the common tactics, techniques, and procedures (TTPs) that enable these cyber threats and attacks. Cyber-attacks are considered arbitrary, abstract, and something the everyday computer user does not encounter. The goal of this project was to develop an escape game that bridges this gap between the average user (victim) and common procedures used by threat actors to help raise awareness of security risks. This project's outputs help end-users learn essential ethical hacking skills through a gamified learning experience that enables players to gain improved familiarisation of otherwise unknown and often mystical cybersecurity concepts. Gamification applies typical elements of gaming, such as point-scoring, into an otherwise dull product or service. Specifically, the gamification of ethical hacking concepts is achieved by integrating skills and knowledge transfer tasks into augmented reality (AR) escape room games. augmented reality provides an accessible, motivating and easy to use interface for learning and enhancing learning experiences. Gamification has been studied in academia and has proven to be a successful tool in incentivizing and motivating learners to take in new and challenging concepts [1].

Ethical hacking tasks in the project are simple and understandable for the average computer user. Tasks range from scanning ports to reading and finding files with common Linux commands. By providing basic examples of ethical hacking, it is ensured to provide the technology as something fundamental and concrete instead of abstract cybersecurity ideas. Our proposed solution involves tasks executed in an augmented reality version of the Linux terminal developed in the Unity [2] game engine. Mobile devices are used for playing the escape room game and the live Linux terminals are accessed via scanning the playing room with the mobile device. The playing room is scanned into the Unity game engine and access points for Linux terminals are set. The game also utilizes real-world objects as hints and potential access points in the game. For example, a hint may be received in a text file after completing a task. These hints may include information about the room or references to real-world items like "the hedgehog lies behind the television". The user could then scan the area behind the television in a room and access another terminal. Thus, the player makes lateral moves along the cyber-physical environment to collect clues for solving the escape puzzle.

The project aims to educate users on cybersecurity. Even if the users do not actively participate in information technology-related work regularly, the project helps to understand better the use of computers and computer security. Educating users can reduce the risk of possible threats to companies if the employees are familiar with basic hacking concepts and attack procedures. Building a risk-aware culture for cybersecurity helps decrease the risks for attacks and malicious intent. Human vulnerabilities make up a substantial part of the risks in cybersecurity related issues [3].

Furthermore, developing strong cybersecurity awareness contributes positively to several United Nations Sustainable Development Goals (UNSDG), including good health, quality education, gender equality, economic growth and ending poverty [4]. Increasing digitization in the healthcare sector helps in growing dividends, and better cybersecurity is needed to aid in securing patient data from attacks. The same

cybersecurity-based solution is also needed for securing gender equality since online resources require strict privacy controls and supervision that can prevent further risks and abuse to women. Cybersecurity in education helps reach the UN's goal of quality education by securing educational products that ICT has enabled in a larger scale than before (e.g. Telepresence, remote learning). Economic growth is preserved with cybersecurity by securing payment systems like online banking and mobile payment systems. Ensuring control of intellectual property and the availability of financial systems is crucial in the economy. Economic growth is supported by the benefits of digitization and increasing trust in ICT systems, this aids in UN's goal of ending poverty by providing access to information on the internet. Cybersecurity is a critical part in UN's goal by ensuring access to information systems for all. Finally, access to food and medication depends heavily on resilient supply chains.

Previous efforts in cybersecurity gamification exist. Online platforms like Hack the box [5], capture the flag [6] and war games [7] all exist to educate those interested in cybersecurity. However, these are not popular or accessible ways to teach cybersecurity concepts to the average computer user. Serious games have also been the subject of academic research and have been found to enable competent teaching and learning experiences in cybersecurity curricula [8]. Hack the box also provides an educational service for universities to use for learning. Capture the flag (CTF) cybersecurity games have been previously used for teaching technical concepts to students [6]. CTFs are a computer security game in which data, also called "flags" is hidden in vulnerable software that is then to be found by participants. CTFs are one of the oldest methods of cybersecurity gamification in addition to war games. Finally, cyber wargaming is increasingly used in evaluating the reliance of national infrastructure. The Estonian based NATO Cooperative Cyber Defence Centre of Excellence (CCDCOE) is a training and knowledge facility that provides cooperation and information sharing between NATO, its allies and partners. The facility provides an exercise called Locked Shields which is a large complex cybersecurity defence exercise and war game simulation dedicated as a practice exercise for national cyber defenders. Locked Shields simulates severe cyber threats on national IT systems [9].

This project was completed using augmented reality (AR) to make the game as interesting as possible for everyone. This technology allows the game to be played using only mobile devices eliminating the need for computers or other escape room props, because large props are expensive and cumbersome to deploy when needed. The inclusion of Light Detection And Ranging (LiDAR) sensors to mobile phones allow powerful AR experiences without expensive equipment. AR will be used due to its immersive nature and to make the project interesting for a large audience. Finally, the combination of AR with environment tracking features also allows us to make an escape room without any props.

2. RELATED WORK

2.1. Ethical Hacking

Ethical hacking is a rapidly growing knowledge area enabling blue teams (defenders) to discover vulnerabilities in an application, system or organisational infrastructure. After obtaining the necessary legal permissions, researchers can use ethical hacking skills to prevent malicious outsider attacks on information systems by legally hacking into the system in a way that an outsider hacker would [10]. As shown on Figure 1, ethical hacking is composed of five phases: reconnaissance, scanning, gaining access, maintaining access and analysis [11]. Ethical hacking skills, enables cybersecurity experts to analyse and assess a customer company's computer system. The *Hack the Room* project emulates different tasks of these phases in a fun and engaging learning experience. Ethical hackers are often referred to as white-hat hackers. The majority of solutions used in securing web applications are based on ethical hacking principles and methods. With strong background knowledge on the TTPs likened to specific attacks, it is possible to strengthen defense of an information system. The term ethical hacking was coined in 1995 by IBM's John Patrick [10].

Standard ethical hacking tools have been made easily accessible and available for a wider audience in the recent years. Kali Linux [12], an ethical hacking Linux distribution containing standard ethical hacking tools was released in 2013 [12]. Figure 2 shows the main screen with the penetration testing tools available in the Kali Linux distribution. The original distribution of Kali Linux features tools such as Nmap [13], Lynis [14] and Aircrack-ng [15]. Kali Linux is marketed and directed at penetration testing specialists and hobbyists and is often used in ethical hacking games.

Ethical hacking is integrated with most Cybersecurity majors and has been listed as a key teaching topic under the current cybersecurity knowledge areas in the 2017 ACM Cybersecurity Curricula report [16]. The knowledge areas included in the curricula content of cybersecurity are data, software, connection, human and organisational security. These knowledge areas provide the proficiency students need to achieve in cybersecurity, regardless of the main program focus. Ethical hacking encompasses multiple of these knowledge areas; and, therefore, it is a key focus in cybersecurity education [16].

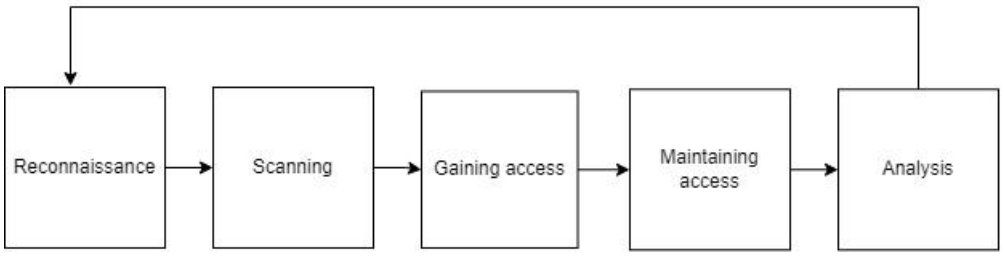


Figure 1. Example flow of ethical hacking.

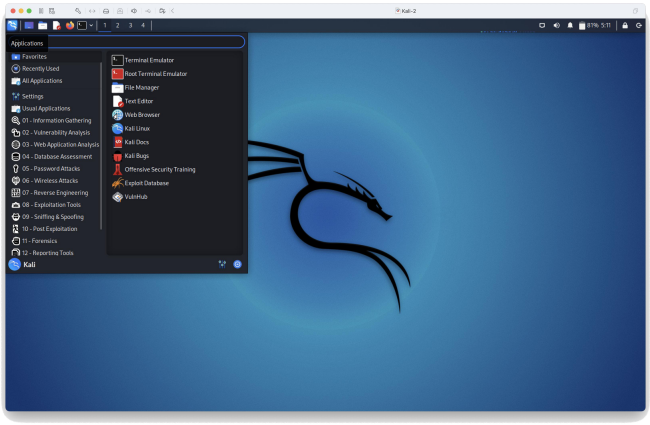


Figure 2. The user view of the penetration testing tools available in Kali Linux.

2.2. Training by Gaming

Numerous online resources and platforms such as Hackthebox [5], TryHackMe [17], OverTheWire [18], or SANS Cyber Ranges [19] offer gamified cybersecurity training. The more popular Hackthebox is an online platform where individuals and organizations can train their offensive (red team) and defensive skills (blue team) through virtual machines that are updated by the community [5]. The boxes (i.e VMs) are divided into easy, medium, hard, or insane difficulty. All of the boxes have a certain set of vulnerabilities that are most commonly met in real life. The players' objective is to gain access to two flags in these boxes: one user flag and one root flag. OverTheWire offers wargames for the user to learn and practice cybersecurity [18]. In cybersecurity, wargames are challenges where participants try to gain access to a system by exploiting its vulnerabilities. OverTheWire and Hack the box both make it easy for beginners to start learning with easy steps and small increments in difficulty. This aspect, positively reinforces learning and makes it easy for individuals to monitor their progress and their level of knowledge.

Combining cybersecurity training with escape room mechanics is another emerging trend. For example, companies such as Living Security [20], Thales [21], and Infosequre [22] offer cybersecurity training through escape rooms. The common factor between these escape rooms is an easy and fun way to introduce groups such as office workers to common cybersecurity themes and threats.

In general, learning through gamification has been shown to bring beneficial outcomes in almost any age group and subject [23]. Researchers in [23], examined how participants from different age groups learned various subjects through playing video games. They concluded that gamification creates positive learning outcomes regardless of the type and subject of game that was played. The study also noted that participants learned, as long as the game was designed to be interesting and relevant enough for the target audience.

Similar observations have been reported for AR games targeting cybersecurity training. In a recent study [24], a group of participants were exposed to the CybAR game. CybAR is an AR game made for smart phones that focuses on common cybersecurity threats, such as usage of public wifi, weak passwords, or lack of

awareness about social engineering. The game's design is based on technology threat avoidance theory (TTAT). TTAT explains how and why in voluntary settings, an individual avoids IT threats [25]. The study [24] conducted a questionnaire on their participants. The results showed that majority of participants agreed on the game's helpfulness on learning about cybersecurity.

2.3. Escape Games

Escape rooms are puzzle games in which the participants solve tasks and puzzles to "break" out of a locked room [26]. These are often implemented in commercial physical rooms that are provided by companies specialising in these experiences. Previous research efforts in escape room gamification exist. Laurea University of Applied Sciences has developed a virtual reality escape room game called Mysteeri 24/7 in collaboration with Kajaani University of Applied Sciences and Häme university of applied sciences. The escape room project was developed to assist with vocational rehabilitation [27]. The Mysteeri 24/7 game's main objectives are learning daily management skills, learning skills and key skills required to work in a profession. The project is aimed at young people at the risk of social exclusion [28]. The Mysteeri24/7 project combines VR immersion with the intensity of escape game to keep the players interested and invested in the escape room game.

According to Steven M. LaValle [29], Virtual reality is the introduction of artificial stimuli to an organism, while the organism has as little knowledge of the stimuli as possible. There are many ways of introducing this stimuli to the organism, but today is most often introduced using Head Mounted Displays (HMD) that have screens to introduce visual stimuli to the target and can transport the organism to a completely virtual environment with large amount of presence.

Other augmented reality and mixed reality escape rooms have been developed before and are in commercial use or deployed for research purposes. A mixed reality escape room game called AMELIO was developed for use in team-building exercises [30] and was studied academically. In AMELIO, the participants used 3D-glasses to navigate the tasks. Mixed reality escape rooms combine augmented and virtual realities with the real world. In an entirely VR environment, the real world surroundings are not considered.

2.3.1. Augmented Reality

Augmented reality is a variation of virtual reality, where users are not immersed in a completely virtual environment. In augmented reality, users are in a partially virtual reality where digital assets are superimposed to the image of real world. If done correctly, the virtual assets appear natural so that virtual and real objects appear to exist in the same place.

Augmented reality also gives new exiting opportunities for teaching and learning. AR allows utilizing innovative technologies in education. AR is a great tool, but like all tools it is only as useful as the implementation [31]. Because mobile technologies are becoming more and more powerful, AR is becoming more and more accessible

[32]. Furthermore, AR applications no longer require expensive HMDs, since 3D assets can now be tracked in the real word from nearly any mobile device.

2.3.2. World Tracking

There are multiple prevailing ways of implementing AR in an application. One way is creating a "target" that can be a printed picture, a physical object or a QR code. The target is then tracked through the camera, and a digital image is displayed on the viewing device. One popular way of applying AR is using the Global Positioning System (GPS) to track the users location in the world, when the user is in the correct location, the application can then use the device camera to make the AR experience complete. Another way to apply AR to an application utilizes a 3D scan of the environment, also known as a Vuforia [33] Area Target, to apply virtual assets to the room. This application then uses the 3D scan to track the location of the AR device in the playing area. In this project, we used the LiDAR sensors of a mobile device to create the Area Target, thus eliminating the need for placing additional physical objects in the playing area.

Table 1 provides a summary of how *Hack the Room* scores against other serious games. The Web column identifies if the game is available online via a web browser. The VR and AR columns indicate if the game utilizes either VR or AR technologies. The accessibility column describes if the users need specific equipment to play the game. Lastly, the mobile column describes if the game is tied to a specific location.

Tool Name	**Web**	**VR**	**AR**	**Accessible**	**Mobile**
Mysteeri 24/7	✓	✗	✗	✗	✗
Hackthebox	✓	✗	✗	✓	✗
TryHackMe	✓	✗	✗	✓	✗
OverTheWire	✓	✗	✗	✓	✗
SANS Cyber Ranges	✓	✗	✗	✓	✗
The Living Security Cyber Escape Room	✗	✗	✗	✗	✓
CybAR	✗	✗	✓	✓	✓
Hack the Room	✗	✗	✓	✓	✓

Table 1. Comparison table between related projects.

3. DESIGN

3.1. Starting off and First Steps

The purpose of this project was to create an AR environment to educate people on ethical hacking. To achieve this, we created an AR game using the Unity [2] game engine, akin to the browser-based Hack The Box [5] where users use ethical hacking skills to solve puzzles. The game developed is designed to be educational, enabling skill and knowledge transfer without the present of a teacher/tutor. The design reflects this requirement by giving instructions to the user and gradually introduce key terms and ethical hacking techniques.
Furthermore, the game development involved the use of a LiDAR sensor to create a 3D scan of the environment where different computer terminals are placed for the user to interact with. While the game was built for mobile platforms, there should be no reason why AR HMD's (e.g. Hololens 2) could not be used to play this game.
First steps in this project were to get a working implementation of the LiDAR scan and an User Interface (UI) where users can interact with the virtual terminals. Once the environment was working, puzzles were designed and implemented. The overall steps of the game and the targeted awareness areas are depicted on Figure 3 and 4.

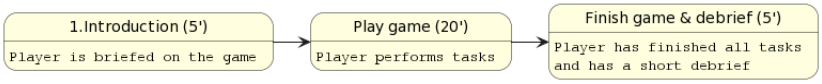


Figure 3. Steps of game and time estimates.

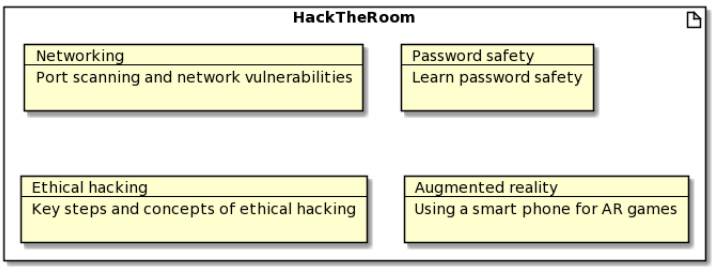


Figure 4. Awareness areas of *Hack The Room*.

3.2. Game Design

Hack the Room's game design follows constructionist game design to promote learning a new concept through grounded and practical approach [23]. Implementing the work flow from Figure 1, the practical approach could be done with implementing a real Linux terminal and the grounded approach could done by developing the game on handheld Android or iOS devices. This setup introduces the player to a live Linux terminals which they would use through the game.

To further promote learning, the game should be fun and immersive enough to promote a flow experience in player [34]. This can be done by telling the player a clear objective with simple instructions. For example, tell the player about the ethical hacking flow from Figure 1 with hints on what commands the player should start with. The players prior experience has an effect on experiencing a flow state [34]. The game should offer a challenge based on players skill level. This was done by making a time limit, which opts the more experienced players to challenge themselves to play the game faster than others. Figure 5 shows how challenge needs to scale with the player's experience to achieve a flow state.

The gameplay supports these requirements and makes the game fun [34]. The gameplay introducs a treasure hunt mechanic, where the player needs to find the hidden terminals with their device and explore the terminal for finding flags to further progress in the game. This enables a simple linear level progression system with scaling challenges.

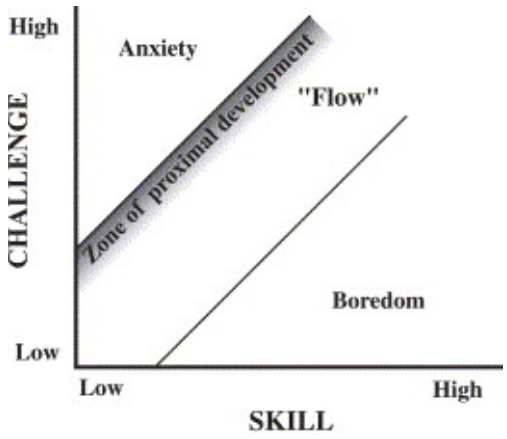


Figure 5. The three areas show how with more skill the player has, the more challenging tasks need to be for the player to achieve a "Flow" state. Figure taken from [34].

3.3. Software

Figure 6 shows how users may use the program. Users are divided into players and game providers. The game is configured and set by a game provider. Firstly, the game provider needs to scan the room in which players will be playing. The scanning will be done with a phone or other device with a LiDAR sensor through Vuforia Area Target Creator application [35]. Then Vuforia Engine will turn the data from the scanned surface to an Vuforia Area Target, where AR elements can be placed in the room using the Unity game engine. A test example is shown on Figure 7.

The play area should be big enough for all the necessary AR elements and for the player to move freely. The AR elements could be simple doors, boxes, or books that need to be unlocked. The game provider should keep in mind the duration of game (see Figure 3) and that there are enough interactive objects for covering all awareness areas (see Figure 4), when setting the AR elements. After scanning the room, the

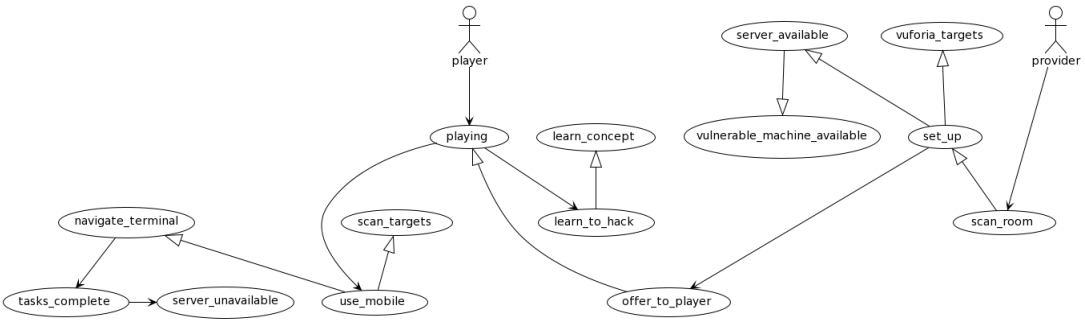


Figure 6. Use Case Diagram of *Hack the Room*.

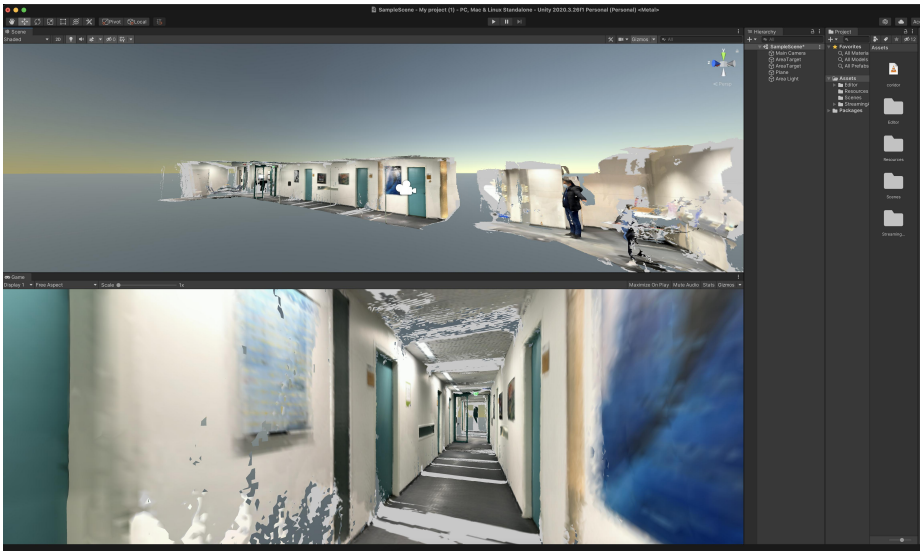


Figure 7. Example scan of various spaces in UBICOMP lab made with iPhone 13 pro inbuilt LiDAR sensors.

game provider needs to make sure that all the AR elements are in suitable places and the room is properly scanned. When the game provider is satisfied with the room's configuration, players can be invited to play the game.

By pressing the play button, the game starts. The player is greeted with simple instructions on what is happening and what they should do. After reading the instructions, the player plays the game by looking around the room, scanning targets with their phone and interacting with them through an embedded terminal. The embedded terminal takes simple Linux terminal commands that the user writes from the keyboard of the phone or through some pre-configured buttons that have command lines. The inputs are sent from the game engine to a remote server, checking the inputs and returning the outputs.

The game ends after the player has completed all the simple tasks that train ethical hacking and cybersecurity skills. The tasks, for example, could be simple password fetching from different sources (i.e Linux server terminals), so a password from one VM is required to gain access to another VM, so the difficulty level may rise appropriately. After finishing the game, the player will have learned something new about ethical hacking and cybersecurity.

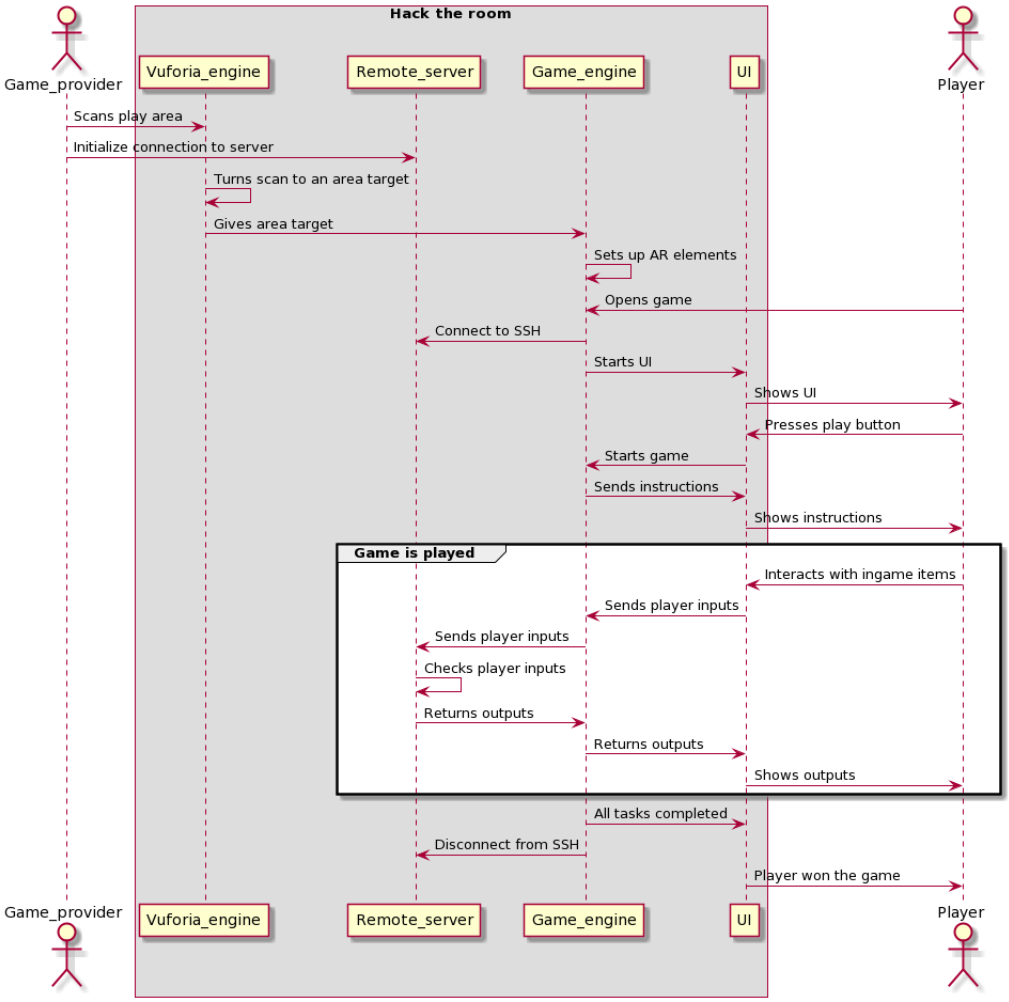


Figure 8. Sequence Diagram of *Hack the Room*.

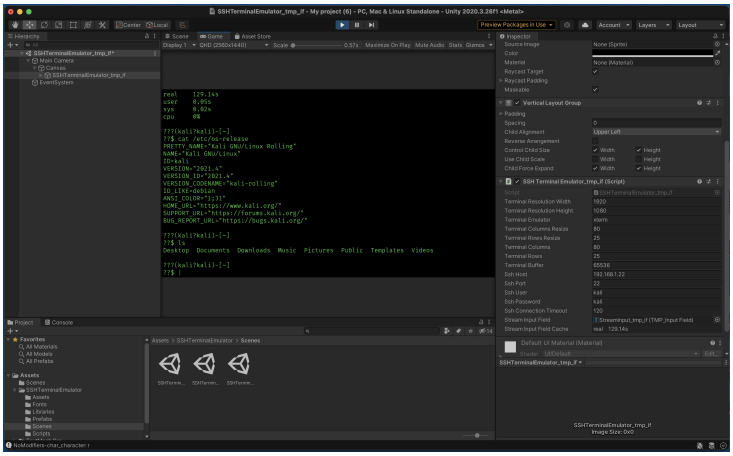
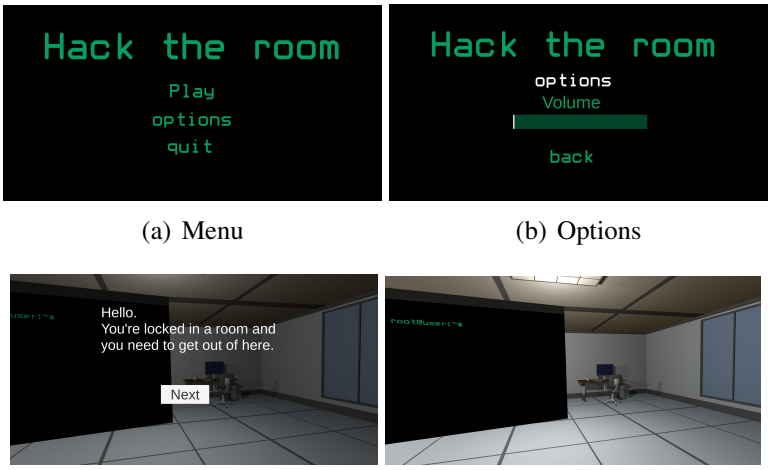


Figure 9. Live SSH Terminal Emulator in Unity connected to the Kali VM shown on Figure 2.



(a) Menu (b) Options

(c) After pressing play button. (d) After reading the instructions.

Figure 10. Initial design of *Hack The Room's* interface.

3.4. Interface

The interface takes into account that players are playing on their mobile phones. To achieve user-friendly experience, the interface was designed as simple as possible. The game starts in the main menu, as seen on Figure 10(a). The main menu has three buttons. The first button is for playing the game, the second button is for options, and the third button is for quitting the game. Figure 10(b) shows the options menu, the menu has all the settings the player may tinker with. The volume slider is just for design purposes and other settings will be added later. Choosing the play button starts the game, and the player is greeted with a short message seen on Figure 10(c) that explains the situation. After understanding how to play/win the game, the player can start playing the game to solve the puzzles on Figure 10(d) using the terminal shown on Figure 9.

3.5. Analysis of Design

It was necessary to aim for simplistic design choices when designing the project since users use the game without technical experience in the IT field 11. These simplistic choices include using a mobile phone for playing, easy to navigate menus, easy hacking tasks and simplifying typing into the terminal. The system is accessible by using a smartphone to play instead of traditional HMDs or AR glasses. This also lessens the number of devices used for providing the game, making the overall system less prone to device failure [32].

AR provides a unique way to create feedback, presence and immersion into the project. This is important for a game that focuses more on the learning experience and less on the actual gameplay provided. AR technology provides the player more stimulus compared to normal games, thus increasing immersion. Greater immersion can help with more joy gained from playing the game [32]. This helps in learning the concepts provided by the game subconsciously. Feedback provided by this design comes from tips and help provided by the interface when completing the task. AR

helps the learner to refocus their attention to the game, since it has a versatile visual UI that can provide notes and tips [32].

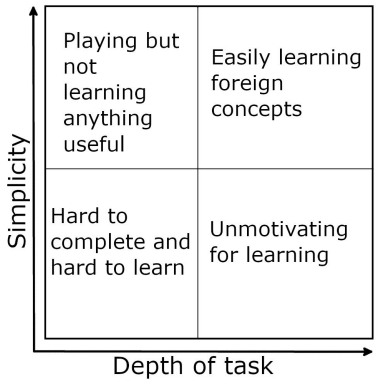


Figure 11. Design simplicity and depth of task.

3.5.1. Design Evaluation

The data for the evaluation was gathered by conducting a controlled study. We enlisted subjects that have little to no knowledge about ethical hacking, since this project is aimed at people with no knowledge of ethical hacking. We also recruited subjects who are experts in the subject matter in order to have a meaningful baseline.

Individual participants were handed a consent form and asked to fill in their age, gender and answer basic questions about themselves and their knowledge of ethical hacking. After answering the questionnaire, they were given a mobile device and a brief overview of the basic user interface and the goal of *Hack the Room*. Before starting the experiment, participants were asked if anything is unclear and if they have any questions regarding the experiment.

All participants were given the same mobile device, and they were put into a room with no distractions and a generally a minimal physical setup to simulate and test how well the software works by itself without external help. Participants were asked to play *Hack the Room* for a set amount of time. Conducting researcher was silently monitoring the experiment and taking observational notes. The researcher was not talking with the participants unless they were asked a question regarding the research procedure or if the participant faced a technical problem with the experiment equipment. Right after playing the game, while the experience is still fresh in their memory, the participants were asked to fill questionnaire, measuring the gaming experience, ethical hacking skill and knowledge acquisition, and overall learning experience.

All the surfaces and devices used in the experiment were sanitised between participants, to ensure a safe environment for participants. While education is important, the ultimate goal of the project was to create an enjoyable way of learning about ethical hacking, that is approachable to people with little technical knowledge on the field.

3.5.2. Risk Assessment

The design of *Hack the Room* has a few potential bottlenecks that require extra attention in the implementation phase. The first being the use of terminals on a mobile phone. If a user is typing into the terminal, they have to hold the mobile device in place to see text on the AR terminal and at the same time type commands into it. This can cause additional physical strain to the user. To combat this, we made the game to be played in portrait mode and put the terminals at appropriate heights to prevent this. This issue was of medium difficulty, since it does need extra care and planning to prevent it.

Another bottleneck to consider in our design and the overall process of the game is device failure. The design is implemented so the minimal amount of devices is used for playing and the technical tasks such as scanning the room are left to the game provider. This way, the users playing the game are narrowed down only to one device. The providers are more well equipped in fixing issues in the playing process compared to the users. Therefore, it is important to let the provider handle everything outside playing the actual game. Smartphone device failure and inadequate AR functions in smartphones used by the participating individuals have to be considered [32].

By implementing AR as the core aspect of its design, *Hack the Room* brings a large amount of new concepts to the player at the same time. This is where an issue for the game bringing forward too many new concepts at once to the player comes forward. The users first have to get used to the AR-system itself before they can properly pivot their attention towards the ethical hacking tasks [32]. This is a medium scale pedagogical problem and is actively prevented by simplifying the game more than normal computer games would be simplified.

4. IMPLEMENTATION

4.1. Implementation Process

The project adopted a rapid application development method to overcome bottlenecks in implementing our design. The method was adopted because after doing initial LiDAR scans of the physical environment, we quickly noted significant differences between playing the game on a mobile phone and in Unity's simulation. The main issue was with the AR camera's tracking, which misplaced some 3D objects and had difficulties tracking the device's position to its surroundings.

Rapid application development or RAD was introduced by James Martin. The development methodology focuses on rapidly developing applications through frequent iterations, emphasising on software usability and user feedback [36]. The method can be divided to the four stages shown on Figure 12. First stage is to define the project requirements, second is to build prototypes, third is to test the prototypes and build a working model, and fourth step is to review the model and repeat from step one if the model does not fulfill the requirements or the requirements have changed during development [37].

Hack the Room's requirements were to make an escape room type of game about ethical hacking for people who lack prior knowledge on ethical hacking. The goal of the game is to complete tasks in a given time frame. These requirements were to be implemented in a short weekly prototyping cycles. These prototypes were then combined to a working model that was tested on site. After receiving feedback from users, the product was polished to meet the new requirements.

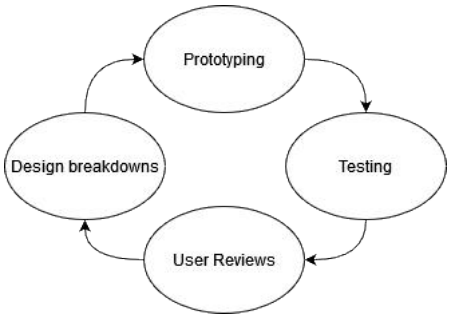


Figure 12. Rapid application development flowchart.

4.2. Implementation of Game Design

Hack The Room's game design was relatively simple to implement. During the design process we took great care in making sure that the aims, goals, and learning outcomes of the game are clearly defined and conceptualised. This gave our Rapid application process a purpose and we generally knew what we wanted to implement.

To make sure that we promote a flow state [34], we play-tested the game as frequently as possible while developing it. The largest deviation from the initial design was the choice to remove all unnecessary elements from the game space. While

these objects can add wonder and excitement to the game, they introduce unnecessary information that might not directly contribute to the target learning outcomes. Removal of the unnecessary elements from the game space, and showing them only when relevant to the current task, made sure that the player knows that everything needs to be interacted with and the player can focus more on the task. While implementing the design we tried to keep the educational aspect in mind and all design decisions made during game were made to either make the gameplay experience better or to improve the educational focus of the project.

4.3. System Architecture

The main system of *Hack the Room* consists of the base game, a virtual machine deployed in the cloud, and a smartphone that is used for interfacing with the game. *Hack the Room* does not implement persistent data management. Files used for the tasks are persistent on the VM but *Hack the Room* itself does not manage or save any data. The game is a one shot experience for players, therefore saved data is not required for playing. The VM is used for the implemented ethical hacking tasks. It is interfaced via a Unity SSH terminal asset [2]. An overview of the system architecture is depicted on Figure 13.

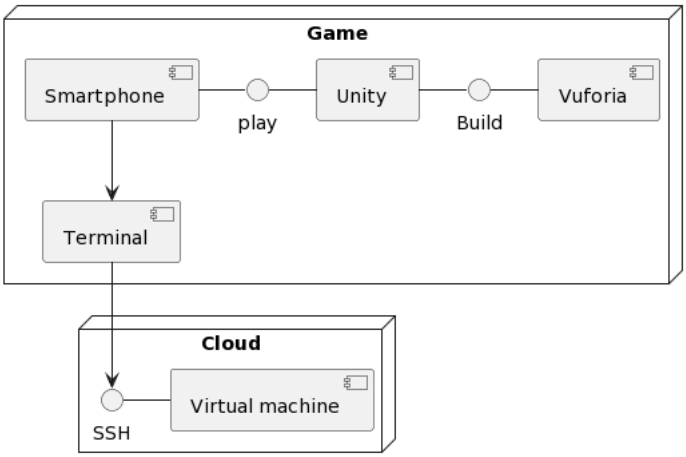


Figure 13. Overview of system architecture.

The operational functions of the game, such as scanning objects and using the Area Target for positioning objects, are implemented using the Vuforia AR Engine [33]. The Vuforia Engine is initialized at the start of the game. Vuforia is used inside Unity when the game is built on the smartphone and then deinitialized when the player quits the game.

Hack the Room starts when the game is opened on the player's smartphone. At the start of the game, the Vuforia Engine is initialized, and the SSH connection is created when the player opens the first terminal. The game can be quit at any time, but it does not save the game state, so the player would have to restart the experience. The game ends when the experience is completed or the timer runs out. At game end, the Vuforia

Engine is deinitialized, and the SSH connection is closed. States of the system and their respective functionalities are depicted on Figure 14.

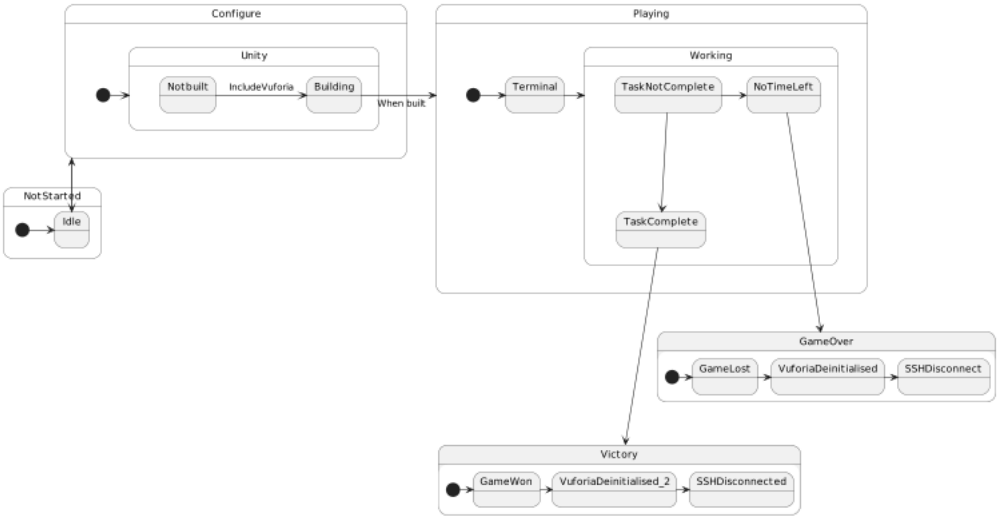


Figure 14. State diagram of system architecture.

4.4. Vuforia Engine

Vuforia Engine [33] is an AR engine made by PTC Inc. [38]. The engine supports various kinds of object tracking [39]. The most notable ones are: i) image tracking, where an image is tracked and virtual objects can be placed near it; ii) model targets, where real life objects are recognized by pre-existing 3D models; and iii) Area Targets, where tracking is based on LiDAR scanned area. This project uses Area Targets for tracking the in-game terminals. The play area is scanned using the Area Target Creator App [35]. The application configures the scan to an Unity compatible file. We decided to scan one of UBICOMP's [40] offices and one hallway for a play area shown on Figure 15. These areas were selected because they had the best fit for following the best practices when preparing and scanning an environment [41].

We used Apple iPhone 13 as a scanner for the project. The iPhone 13 has a LiDAR sensor that supports Vuforia's Area Target Creator App. The application is suitable for generating Area Targets for spaces up to 50 m^2 [35] or 5 minutes of scanning. Figure 16 shows how the lasers from the sensor create a 3D scan of the environment. Our play area exceeded these limitations, so we tried to make a combination of Area Targets or Multi Area Target [42]. Unfortunately, the Multi Area Target did not work as planned, since the tracking offset was too much. This lead us to making a compromise and we scanned an area that fit the Area Target Creator App limitations.

After scanning, the application lets you build the scan to an Area Target and check how Vuforia Engine will track the environment. Figure 17 shows how the engine will track the environment. The Area Target then can be moved to Unity where you can add your 3D objects that will be tracked and scaled to the to fit the environment.

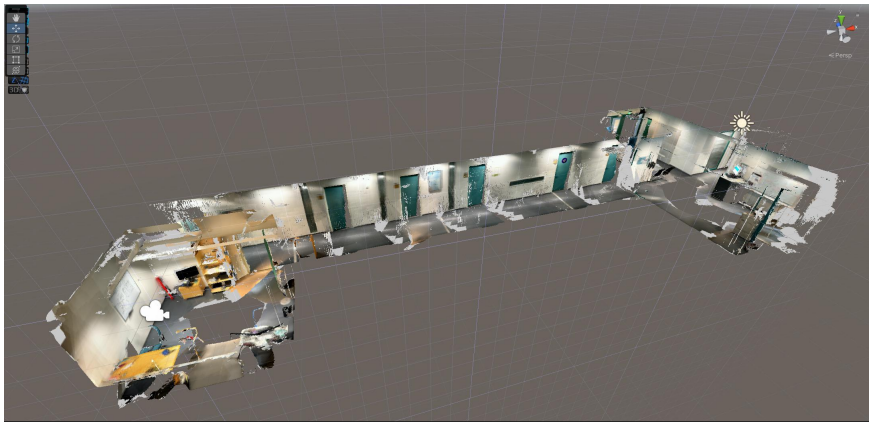


Figure 15. Play area of *Hack the Room*.

Figure 16. The lighter green areas indicate that these areas are closer to the scanner than the darker purple areas. Source: Vuforia's photo from Area Target Creator App [35].

4.5. Unity

Unity game engine [2] offers an easy way to modify Vuforia Area Targets and add 3D objects, which players can interact with. Other benefit of using Unity in this project are Unity engine's functions. These functions eased implementing necessary functionalities and basic game logic for the 3D objects. Other useful aspect of Unity is their supported Asset store [43], where we got the in-game SSH Terminal Emulator [44]. The emulator uses SSH.NET [45] and VtNetCore [46] as a back end and native Unity UI elements to show the output. This makes the Terminal work on our target platforms Android and iOS.

The in-game terminal opens with a preconfigured user and password. The player connects to the VM by clicking the connect button 18. After connecting the terminal script initializes and starts the terminal. The initialization consists of creating the terminal, creating a data consumer that forwards the user inputs to the back end, and creating a SSH client. After initialization, the terminal listens the user inputs and transforms them from ASCII code to bytes. The bytes are fetched from a dictionary and forwarded to the back end.

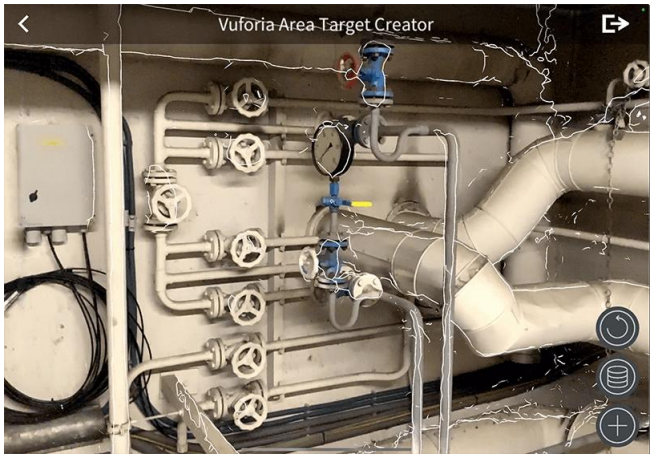


Figure 17. The white lines show how Vuforia Engine tracks environment when it recognizes the area. Source: Vuforia's photo from Area Target Creator App [35].

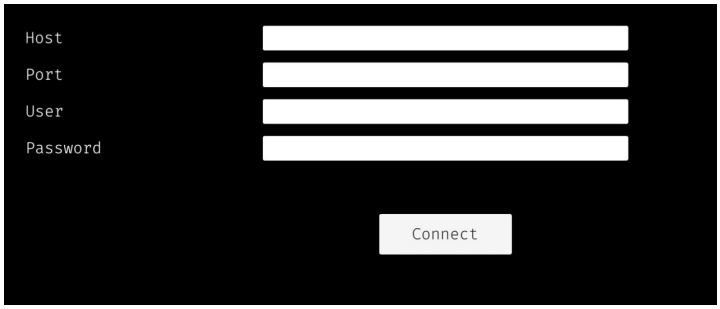


Figure 18. In-game welcome terminal.

The back end consists of two dynamically linked libraries or DLLs [47]. The DLLs are VtNetCore [46] and SSH.NET [45]. VtNetCore is a terminal emulator library for the .NET Standard 2.0 framework and acts as a translator between user inputs and terminal outputs. The SSH.NET does the functionality part by supporting the various SSH terminal features. The DLLs then gives the output to the unity UI elements, which show the output to the user. Figure 19 shows how the terminal works from a macro perspective.

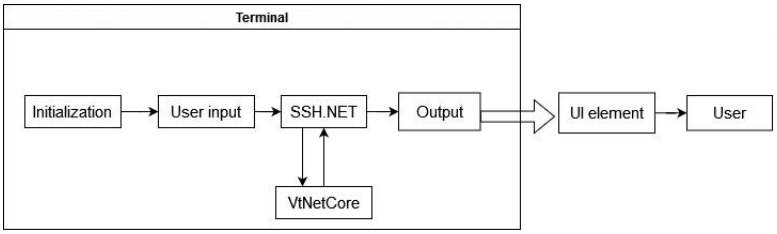


Figure 19. In-game terminal data flow.

4.6. Virtual Machine

Hack the Room connects to a Kali [12] Linux virtual machine via a SSH terminal emulator Unity asset. The OS version of Kali Linux implemented for *Hack the Room* is 2022.1 and it was downloaded from the official source [12]. In the virtual machine, three tasks are implemented in which the player retrieves passwords to continue playing. The tasks are as follows: In the first task the player reads a readme.txt file using the `cat` command. From this file the player retrieves the first password. The second task has the user move to another directory with a `cd` command. From this directory the player can then retrieve the needed passcode after answering a question about cybersecurity for the next task. The tasks are implemented incrementally and the player uses the previous task's commands to aid in executing the next task. In the last task the user scans the ports of another Kali Linux virtual machine. After scanning, the user answers questions based on their findings. The respectable tasks are references as user account names in the virtual machine. For example, Task 1 connects to a user called `hack1`.

The tasks are done within the virtual machine, but clues are placed in the surrounding of the player in the AR environment. A virtual machine was implemented to simulate a regular non-AR CTF game. By using a virtual machine, making the tasks was easier than making a "fake" terminal that would have needed to be developed from the scratch. Furthermore, virtual machines enable us to develop tasks by simply placing files, directories and scripts in the operating system and opening ports for the Nmap scan [13].

4.7. Security and Privacy

Hack the Room does not store or use personal data. During the evaluation, time elapsed is measured via a timer in the game. This information can not be related back to a player that has played the game. Wifi and/or 4G connectivity is required to play this game. The internet connection is used to access the VM during gameplay.

A key part of the *Hack the Room* game is its VM component. The VM is secured via a password and the instance can be paused when not used to prevent unauthorised access. The VM's file system does not contain any information that can be used with malicious intent. The use of a VM poses a risk since the VM image is loaded with Kali Linux's default attack tools. For this reason the VM has to be kept secure. If an outsider gains access to the system, they can edit or replace files used to play the game (readmes, script files) with malicious software that can lead to gaining access on the player's mobile phone via a [48] payload exploit. To mitigate these threats, the VM's default SSH port 22 was remapped to port 2222 to prevent unauthorised access through SSH connections [49]. The SSH service has been bound to specific IP addresses using firewall rules to further prevent unauthorised access from outsiders. With these security measures, a threat scenario of outsider attacks via a SSH connection can be mitigated. The VM is secured so it can only be connected to from authorised IP-addresses. These addresses include the developers' IP-address and the evaluation stage wifi router's IP-address. These addresses are whitelisted using security groups.

The VM of *Hack the Room* is secured in a cloud server (CSC Pouta) provided by CSC IT Center for Science [50] and is accessed via a floating IP-address assigned to the instance of the machine in the cloud. The virtual image was uploaded to CSC Pouta and launched as an instance to run from the cloud service. The instance was then assigned a floating IP to make connecting via SSH from other machines possible. Backups of the machine can be taken by creating snapshots of the VM's current image. Snapshots save the current instance's memory and drive states. These backups can then be downloaded locally or be saved in the cloud to restore an old instance if the VM is misconfigured or happens to break. Firewall rules are set in the CSC web interface.

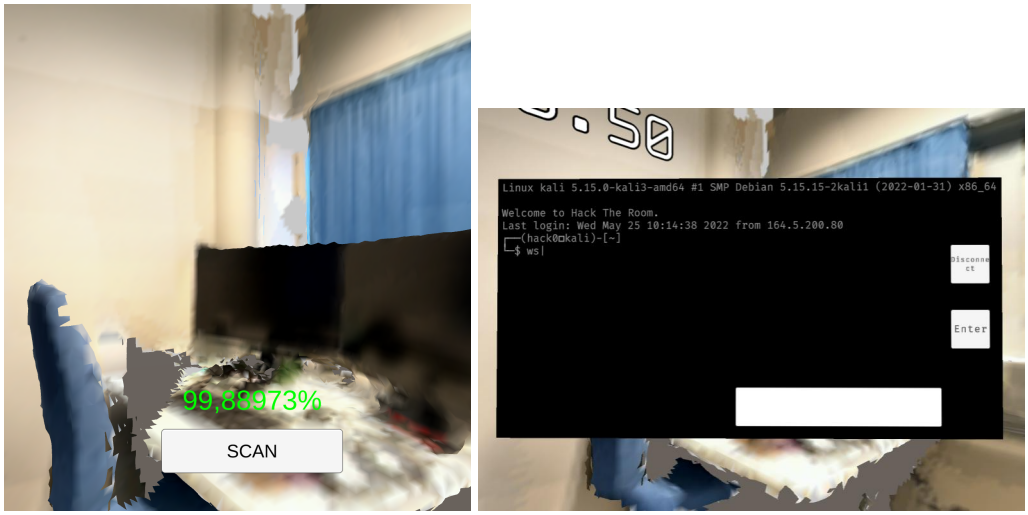
For the evaluation stage, we explored a threat scenario of insider attacks to the *Hack the Room* VM. A test user with unsupervised access can create malicious files to the VM if not monitored. If the test user succeeds in creating malicious files, the next test user after them can fall into the trap. To mitigate this, each test user has to be monitored during gameplay for security reasons. The Unity and VM log files have to be reviewed after each experiment to ensure no malicious activity by the players has occurred during gameplay. After each experiment, the next test user can SSH into a fresh installation to ensure that insider threats are negated.

4.8. User Interface

User interface for this project was implemented as minimalist as possible to avoid cluttering the phone screen with unnecessary information and to keep only the relevant information visible. When the user approaches an object (i.e. an AR terminal) a proximity indicator appears on the phone screen to indicate in black font that a scannable object is near. When the smartphone is directed towards the object the percentage score of the proximity indicator gets closer to 100. As shown on Figure 20(a), when the smartphone is almost directed towards the object the text turns green and button appears. Subsequently, the user can "scan" the object to reveal and interact with it.

The main part of the user interface are the terminals from which users can send console commands to the virtual machine seen on Figure 20(b). This part of UI is a floating console with input field and two buttons, "Enter" and "Send input" on the right side of the console window. User can use the phone's virtual keyboard to write commands to the console input field. The *Enter*button sends the input to the console window and executes the command.

One possible way to further simplify the design is to get rid of the input field and replace it with buttons that input the command needed automatically. This could be easier for less tech savvy users, since it limits the options giving the player clarity for the task at hand and is less intimidating.



(a) Game While scanning for terminal. (b) Terminal UI.

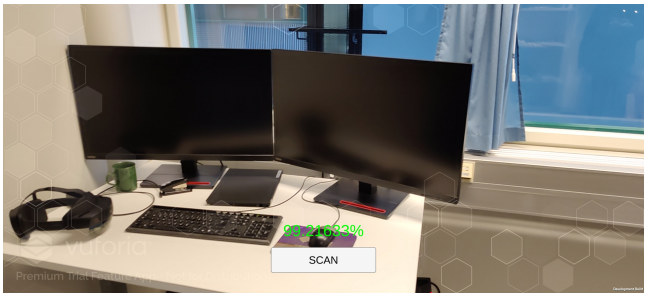
Figure 20. Game UI.

4.9. Game Flow

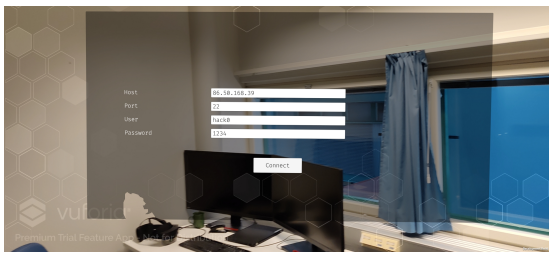
Hack the Room follows simple linear gameplay with clues and instructions that help the player play through the game. We anticipated that in a 20 minutes gameplay, users can complete three tasks without instigating too much fatigue from holding the mobile device. The tasks are described below.

As seen on Figure 21(a), the game starts with simple objective describing the mission of the player in the game. Figure 21(b) shows how the player is instructed to search for the first terminal, which is close to the monitors in the starting room. After scanning, the terminal is opened and the player starts the game with simple `ls` command. Figure 21(c) and 21(d) show the scale of the terminals.

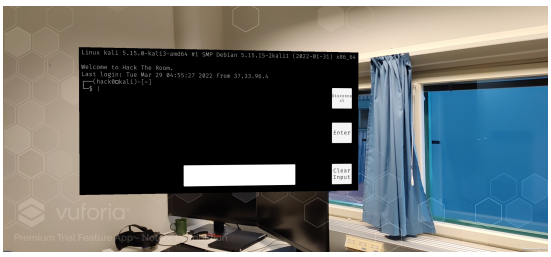
(a) After pressing play.



(b) New terminal found.



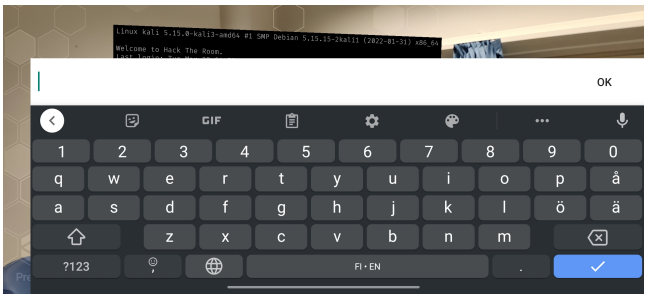
(c) After pressing the scan button.



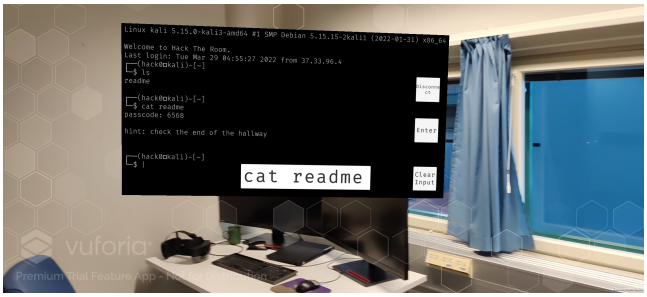
(d) After connecting.

Figure 21. Screenshots from a end user device. The Vuforia watermark is from using the Basic Plan [51].

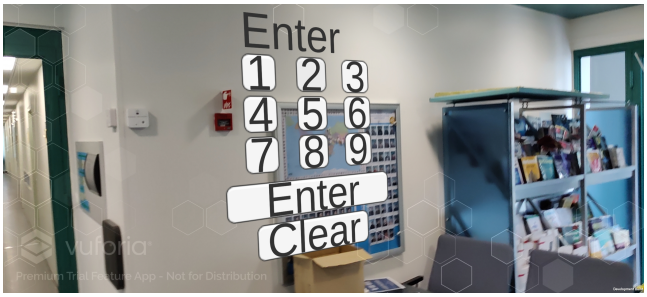
The game is played by writing Linux terminal commands using the end user's virtual keyboard. The mobile device's keyboard limits some of the terminal's functionalities, such as simple keyboard shortcuts. After playing the game, the player walks to the numpad that works as a level changer. The numpad only accepts correct flags and it indicates if the correct flag was given by turning the input to green. If the player puts a wrong flag, the numpad turns the input to red. These functionalities are depicted on Figure 22.



(a) End user device's keyboard.



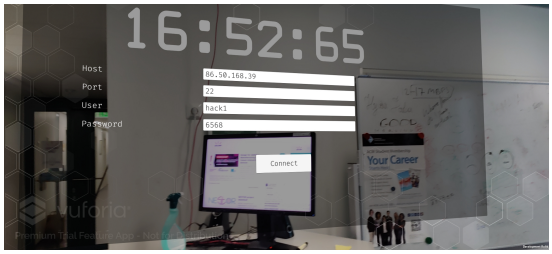
(b) Terminal after playing the first task.



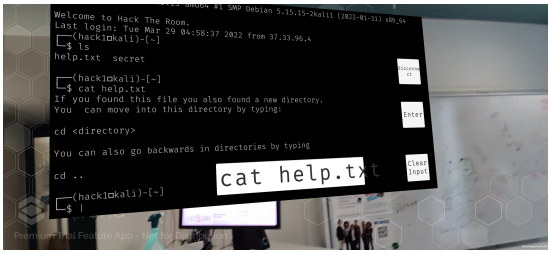
(c) The in-game numpad.

Figure 22. Screenshots from a end user device.

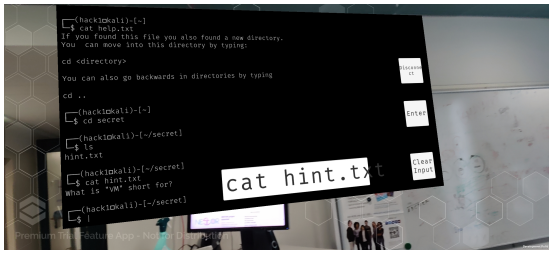
Figure 23 shows the user's view during the second task. The task introduces a new command for the player. The new command is `cd` and lets the player change to a different directory. With this new knowledge, the player should be able to find a clue for acquiring the next flag. Figure 24 shows the last task, which is a simple `Nmap` scan [13]. The `Nmap` scan provides info for answering the last question to win the game. The player loses when the in-game timer runs out.



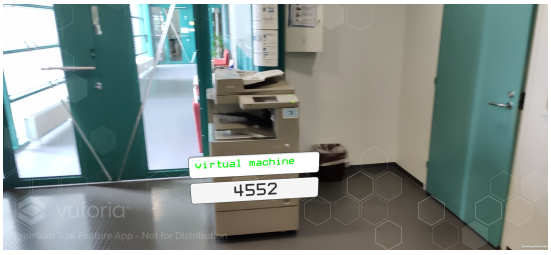
(a) After finding second terminal with in-game timer on the background.



(b) Moving to a new directory.



(c) Found the clue for the next flag.



(d) After answering the question right.

Figure 23. Screenshots from a end user device.

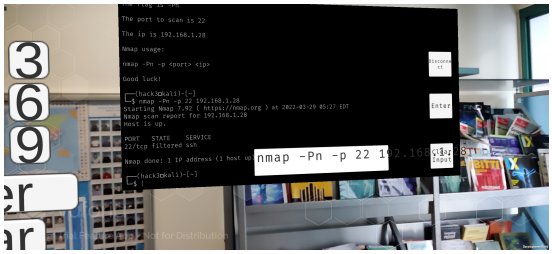
4.10. Risk Assessment

This implementation includes risks for playing comfort and UI usability. Playing an AR game can cause dizziness and discomfort to the player. Discomfort is an important factor to consider since the player's comfort affects user experience and learning outcomes either positively or negatively. The 3D motion sickness is caused by vergence-accomodation conflict [52]. Vergence-accomodation conflict can be circumvented by using long viewing distances, matching display and focal distance, and having reliable depth cues [52]. Players of *Hack the Room* should view objects such as the terminal window from a comfortable distance. The terminal must not be rendered too big for the user or be played in a narrow space, for example the corridor.

Hack the Room's UI has issues regarding its AR elements. The player has to hold their phone still and aimed at the terminal while typing commands if they are not comfortable with what they are writing. This can be uncomfortable, but can be fixed with having the terminal at a comfortable height for the player to see. Another issue is that the terminal input methods can be unresponsive at times when playing.

The implementation of a cloud-hosted VM also presents some challenges. The tasks implemented in the VM are of beginner difficulty, but can still be confusing for players that are not familiar with Linux terminals. Inexperienced users might mistype commands or get stuck in the terminal window. This can be circumvented by having clear instructions and help tips in the game.

The implementation of Area Targets includes risks for object tracking failures in the game. For example, when a wall or a part of a room does not offer any references for the smartphone camera, the AR object cannot be tracked [39]. This renders the long corridor used in *Hack the Room* almost useless since it consists of mainly blank walls. Object tracking failures may hinder the game experience if the application has not been tested thoroughly before play. These object tracking failures can be avoided



(a) After reading instructions and doing the nmap scan.

(b) After answering the question related to nmap scan.

(c) After winning the game.

Figure 24. Screenshots from a end user device.

by placing the objects in places where there exist differences in the room, for example a table or other objects that have affected the scan.

The application of *Hack the Room* utilizes the camera for the whole 20 minute playtime. This extended use of the camera and its tracking features drains the battery of the smartphone quickly and can cause heat issues if use is prolonged. There are no hardware side methods to avoid this, but using 1-2 other smartphones if many players play after each other can help with battery issues. The unused smartphones can be charged while the others are not in use. Heat from the application being strenuous to run cannot be circumvented, but when played in turns with other devices, the phone can be let cool down.

5. EVALUATION

5.1. Evaluation Plan

To evaluate the educational impact of the developed AR application, we designed a field experiment with eight participants (three females and five males) split into two groups: technology experts and non-experts. A preliminary pilot study with two non-expert participants was conducted before the actual field experiment.

The non-expert group included users who do not have prior training in IT or cybersecurity. Participants in the non-expert group were tasked to evaluate low-level learning outcomes of the game. After playing the game, the non-expert participants were asked to fill out a post-experiment questionnaire with recall questions about technical concepts shown in the experiment. The post-experiment questionnaire was based on Bloom's taxonomy [53] and it aims to gauge the extent to which the game helped the participants learn new cybersecurity concepts.

Respectively, the second group of participants (cybersecurity experts) was selected in order to evaluate situational awareness during the experiment and help us establish a credible base line (i.e. that the AR application has some value related to cybersecurity tasks). Situational awareness was measured by answering a post-experiment questionnaire. To evaluate the situational awareness of the expert users, we used the Situation Awareness Rating Technique (SART) questionnaire [54]. The questionnaire measured various mental loads including engagement, stability, awareness, and concentration.

Overall, the test procedure followed the steps outlined on Figure 3, including i) a quick introduction to the game where participants read the instructions and asked introductory questions, ii) about 20 minutes of actual game play, and iii) answering a short questionnaire depending on which group the player was assigned.

In addition to questionnaire data, we collected timestamped game logs from each playthrough to gauge the performance of users. Each time a game object was scanned by the user, the timestamp event was saved to a log file. By logging the time it takes to complete different sections of the game, we could further evaluate which parts of the experiment were the hardest and which ones were not. These logs were then referenced when analysing the evaluation results. For example, the logs helped us identify how much time a player has spent on task 2, while the post-experiment questionnaire identified the player's ability to accurately recall the learned concept (i.e correctly recall the function terminal commands).

The experiment started with the participants signing a consent form and answering demographic questions about their age, gender and previous experience with AR and cybersecurity. The participants then read the rules and general instructions of the game. These included information about safety measures and usage of commands. The rules include clauses regarding the unsolicited creation of files on the VM or using unauthorised commands. The participants were permitted to have the instruction paper with them while playing to remind themselves of the commands and their usage. Before starting their playthrough, all participants received basic training on how to scan the first terminal.

5.2. Setup

The tests took place in the Center for Ubiquitous Computing (UBICOMP) premises at the University of Oulu (see Figure 15) using the tester's own mobile devices. The mobile devices were relatively new android phones using an Android 11 or 12 operating system and had the newest version of ARCore [55] installed. During evaluation the mobile device was connected to a Wireless Dual Band 4G LTE Router, ensuring proper security measures for the VM. This simplified the networking process, so that additional ports do not have to be opened to play the game during evaluations.

The questionnaire based on Bloom's taxonomy [53] measured how much the non-expert participants remembered from the experiment as well as how they experienced the performance of the game. The questions asked on Figure 30 were as follows:

1. On what port was the SSH service running?
2. What commands were used in the terminal?
3. What part of the game was related to the virtual machine?
4. What was `nmap` used for?
5. What does the `cd` command do?
6. How would you generalize what you learned during the experiment?
7. On a scale of 1 to 10, how would you rate the performance of the game?
8. Did you run into any issues when playing the experiment?

The SART questionnaire [54] on Figure 31 captures self-reported cognitive functions of the users during the playthrough, including: situation arousal, concentration, information quantity, spare mental capacity, instability, variability, division of attention, and familiarity with the situation. Arousal measured how much the participants enjoy the situation; concentration measured how much they had to concentrate to do it; information quantity measured how much information the participants could gather from the situation; and spare mental capacity measured how much mental capacity they had to use to complete the experiment e.g. could they think of anything other than the experiment while doing it. Furthermore, instability measured situation instability, e.g. was the situation unpredictable; variability measured how much the situation changes; division of attention measured how many aspects the participants had to concentrate on at once and; familiarity measured how mutual the participants were with the situation compared to previous experienced situations.

6. RESULTS

6.1. In-Game Metrics

On average, the non-experts were 22-year-old males with no prior experience on AR applications or cybersecurity. As shown on Figure 25 non-experts average playtime was around 15 minutes where the most time was taken on second task. After finishing the game, on average they got five out of six recall questions correct.

Participants in the expert group were on average 29-year-old (2 males and 1 female) with experience on AR applications and cybersecurity. Figure 26 shows how on average, experts playtime was 13 minutes with most time spent on third task. On the SART questionnaire shown on Figure 31, experts gave on average 4.0 out of 7.0 on instability, complexity, and division. Variability was given 4.6, spare 5.0, information quantity 5.3, familiarity 5.6, and 6.3 on concentration and arousal.

6.2. Post-Experiment Questionnaire

6.2.1. Non-Expert Group

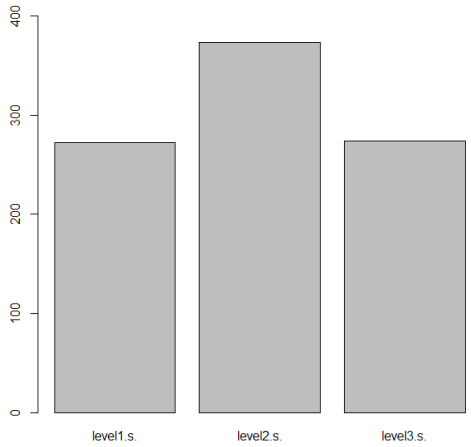
The group reported good recollection of concepts introduced during the game. The questions seen on Figure 30 were rated as 1 point per question. All of the participants got questions 1 and 2 correct. On average, questions 3, 4 and 5 were answered right with one participants out of three answering these questions incorrectly. Question 6 was answered right by all participants in the non-expert group (see Figure 25(b)). On average, points gained in all of the questions were above 0.5.

6.2.2. Expert Group

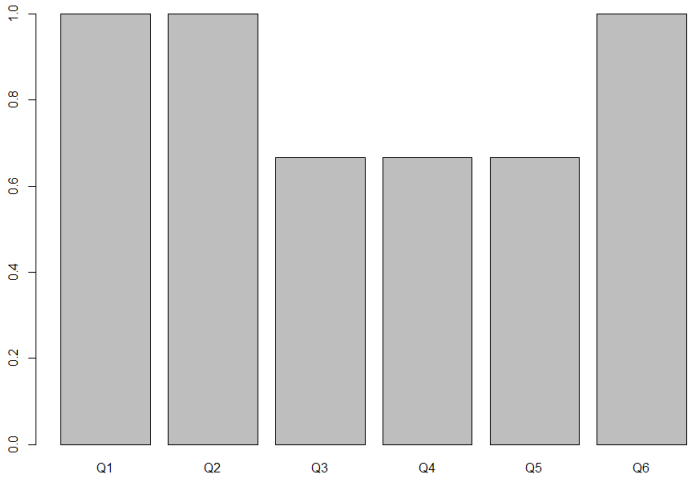
The group reported high arousal and concentration in the situation, average instability, complexity and division of attention in the situation and above average spare mental capacity, information quantity and familiarity as seen on Figure 4. Arousal had the average score of 6.3 with the highest score of 7 that was given. Concentration also had the average score of 6.3 with the highest score given being 7. The last aspects measured were instability, complexity, variability, familiarity and division of attention. The instability score given was 4 by average. The average score given for instability was 4. The average score for variability was 4.6. The average score for familiarity was 5.6. The average score for division of attention was 4.

6.3. Analysis of Results

As seen on Figure 25 and 26, the experts used less time on tasks on average with smaller variance than non-experts. This shows how familiarity with Linux terminal commands and cybersecurity tools, the game is easier to play and more quickly finished. In the non-experts group, there was a bigger variance between finishing tasks

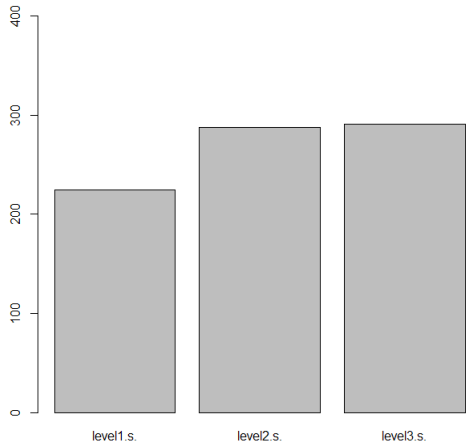


(a) Bar plot of the average time spent on a level with non-experts. Tasks are on the x-axis as levels and time is on y-axis in seconds.

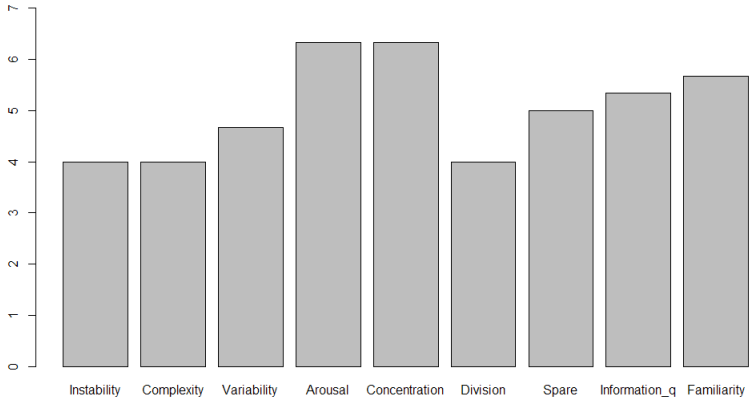


(b) Bar plot of the average score of per question answered by a non-expert regarding cybersecurity. Note that Q7 and Q8 measure the performance of the game and therefore not included in the bar plot.

Figure 25.



(a) Bar plot of the average time spent on a level with experts. Tasks are on the x-axis as levels and time is on y-axis in seconds.



(b) Average scores given by experts on a situation awareness rating technique questionnaire.

Figure 26.

as compared to the expert group. The higher variance is most likely caused by change in task difficulty. The non-experts had no prior concept on how to proceed on after finishing first task. This shows in the more time used on second task which introduces a new command and is a step further in terms of difficulty. The third task is similar to the second task, but has different objectives. This indicates how the non-expert player most likely by third task has learned the pattern on how to ethically hack and what steps they are supposed to take, thus spending less time on the task.

The *Hack the Room* system is averagely scalable. The current implementation uses Vuforia Area Targets as its main functions to provide the playing area. Area Target scanning is limited to a 5 minute scanning time if done as a one shot scan. Multiple Area Targets can be combined in Unity using the Vuforia multi-target feature. This feature was not implemented in *Hack the Room* as the play area was sufficiently small enough for the 5 minute scan limit. The game can be implemented with using Vuforia model targets to allow for better scalability across multiple locations. This way you only need to place scanned objects in the real world instead of scanning the actual world around you. A bottleneck in this implementation would be a secure wifi-connection since the game needs an internet connection to play the game. You would either have to use a municipal wireless network, which is vulnerable for eavesdropping and security attacks or play it with a mobile internet connection.

Playing *Hack the Room* as a group can be a considerable implementation of the game's basic gameplay. This allows teamwork to further make the game more enjoyable for players. Multiple devices can be used to play *Hack the Room* simultaneously as the VM allows multiple SSH connections at one time. Regular escape rooms are almost exclusively group experiences. Teamwork can enhance learning outcomes.

Hack the Room's VMs are stored in a cloud service and have implemented firewall rules to prevent unauthorised access. The game does not store critical data. The VMs can be compromised if the firewall rules are changed to being open from everywhere but as long as they are controlled carefully, the VMs can not be shut down by an outsider. Security and privacy was discussed earlier in this thesis.

6.3.1. Non-Expert Group

As seen on Figure 25, most of the questions were answered right, and only questions 3,4 and 5 had any wrong answers from 1 participant. Unfortunately there is no certain way of knowing if the participants knew the answers from prior experiences, or if they learned the answers from playing *Hack the Room*. Only way to gauge the prior knowledge is the questionnaire they answered prior to playing the game, since that questionnaire asked them if they had prior knowledge of cyber-security and only 1 participant reported prior knowledge from hobbies. This would mean that at least the 2 participants that reported no prior cyber security knowledge learned something while playing the game.

The questions that were answered wrong were: What part of the game was related to the virtual machine?, What was `nmap` used for? What does the `cd` command do? All of these questions on Figure 30 are more in-depth understanding of the tools and the terminology, not questions that had obvious answers in the *Hack the Room* game

and that could be the reason why everyone did not answer correctly. The rest of the questions had answers that were very prevalent in-game. For example, Question 1 was "On what port was the SSH service running?" and that answer was on the screen multiple times while playing the game. Unfortunately there is no way of verifying if this is why some questions were answered wrong with the limited data set.

6.3.2. Expert Group

High arousal and concentration in the situation may have been effected by the use of AR in the game experience. These are important aspects in measuring an AR based serious game as the game needs to teach the participants but also be enjoyable as an experience. Concentration in the situation is important for keeping engagement in the game. In *Hack the Room* the participants were very focused on the smartphone when playing. This was noticeable when following the participants around while they played the game. Many of the participants forgot to check their environment for clues, for example look for an answer to an in-game question that was answered in the instructions paper.

The average score in instability can be explained with game performance. The game had some tracking issues due to an imperfect and inaccurate Area Target environment. These Area Target imperfections caused objects to disappear at times. For future projects, a higher quality Area Target scan must be used to ensure better game performance. The participants reported average scores for complexity in the game. Average scores of complexity mean that the system may lack some parts that could be simplified but on average are simple enough that system complexity did not present an issue. The complexity score can be explained with the use of a real SSH terminal asset instead of a self-coded text-adventure like system. A Linux terminal may feel unfamiliar when used inside an AR environment. Using an AR asset terminal instead of a normal Linux terminal can break the familiarity of a regular non-AR Linux terminal.

Above average scores were reported in spare mental capacity, familiarity and information quantity. The participants kept a fair amount of spare mental capacity while playing the game. The game's level of difficulty and the way the experiment was set up affected this. Many of the participants were given hints during gameplay even if they belonged in to the pro-group. This may have lowered the scores given for spare mental capacity. Having spare mental capacity during gameplay keeps the player interested in the game without the experience being too taxing. If the experiment is too taxing, the participant might quit halfway through. Giving hints to participants lowers the amount of thinking they have to do themselves when playing the experiment and if *Hack the Room* was offered as a service, hints would have to be given to players as the target audience of *Hack the Room* is not naturally familiar with Linux terminals.

The participants reported that they were averagely familiar with the system. The above average scores can be explained with the expert group being previously familiar with Linux terminals and AR games. Two-thirds of the pro-group participants reported that they were previously familiar with AR systems. The information quantity received in the experiment was scored above average. The expert participants of the sample did not receive new information related to cybersecurity concepts but instead gained knowledge about AR systems.

7. DISCUSSION

In comparison to the initial target setting, *Hack the Room* shows promising results. With the current experimental setup, we were able to obtain adequate results that confirm our working hypothesis that AR games could improve learning outcomes and provide good situational awareness in serious games for cybersecurity. These results apply to other subjects outside of cybersecurity.

Hack the Room confirms the findings and concepts of related works. Hackthebox [5] and OverTheWire [18] use similar serious game concepts to *Hack the Room*. Previous cybersecurity training escape rooms such as Living security [20], Thales [21] and Infosequre [22] offer services for similar target audiences (corporate workers). These related projects show that *Hack the Room* can be a viable option in the market for cybersecurity oriented escape room games. Training gamification has been shown to be beneficial for all age groups [23]. While *Hack the Room* supports these claims, the limitations of the field experiment are discussed below.

The sample size evaluated in this field experiment was of very small size (8 total participants). A larger sample size would have allowed us to recruit participants from a larger age pool and of different social strata. A larger and more representative sample of participants would have provided more robust results that can be generalized to represent a large part of the population. More generalized results could have provided useful data that could have helped us design the game more towards its intended audience (non-experts). This could have further helped us productize *Hack the Room* into a sellable product.

Due to the small size sample we have only calculated descriptive statics for two groups taking into account average scores, enabling us to make summations about the human subjects we have measured in the experiments. A larger sample size would have also allow us to use inferential statistics (point, confidence and interval estimates) and therefore use the data from our sample to infer general properties/parameters of much larger population.

Since the game made was a proof-of-concept prototype, the interface could not be refined to a refined version. A refined version should include scalable terminals and terminals that rotate so that they are at a 90° degree angle from the viewer when playing. This would make playing the game easier. For this project a self-coded terminal-like interface could not be made due to time constraints but is a considerable option for productising the game. This type of terminal would be easier to use for non-experts.

For this project, a Vuforia Multi-Area Target [42] could have been implemented to have the scan be of higher quality. This feature is not completely finished in Vuforia [33] and therefore was not used for this project. For this reason, we were limited to scan the play area within the 5 minute scan time limit offered by the Vuforia Area Target scanner application [35]. The UBICOMP premises have motion detection doors that make the scan inconsistent with the real world, and cause issues in the game (objects disappearance).

7.1. Future Work

Hack the Room shows promising results for integrating AR technology into serious games. Ethical hacking is only one teachable topic of many that can be taught with AR serious games. Other subjects can include subjects such as anti-bullying, politics, or everyday tasks similar to the Mysteeri 24/7 project [27]. Situational awareness results provided by the field test show that AR provides high situational engagement to the users and can be used as an effective baseline for validating the concepts under investigation.

Minimalism is important when making serious games using AR technology. Simplification can provide better learning outcomes. The field test concluded in *Hack the Room* shows good learning outcomes for an AR serious game with hard to grasp concepts for non-experts. Better results for learning outcomes and be achieved by teaching easier subjects and using a simpler interface for completing tasks in future projects. Future iterations, can explore the use of smart glasses and HMD to enable improved mobility of the tester.

Productising AR serious games or researching how projects such as *Hack the Room* translate to real world offerable services such as escape rooms can be a good topic for future projects. This thesis does not cover concepts such as marketing, consulting or offering said services. The field experiment covers a simple framework that can be implemented for serious game escape room products.

8. CONCLUSIONS

Hack the Room was made to evaluate if AR can enhance learning of cybersecurity concepts. The game was made using Unity [2], a SSH terminal emulator asset [44], Kali Linux virtual machines [12] and the Vuforia Engine [33]. Tasks in *Hack the Room* included the use of ethical hacking tools such as reading files and scanning ports. The goal was to evaluate if implementing AR technology can enhance learning outcomes and situational awareness in a serious game. For evaluation, a field experiment was conducted with 8 participants consisting of non-experts and experts. The non-experts answered a questionnaire based on Bloom's taxonomy [53] and the experts answered the Situation Awareness Rating Technique (SART) questionnaire [54]. The results showed high situational awareness and average remembrance of concepts introduced in the game. According to our results, AR has a good probability for enhancing engagement, immersion and learning outcomes in serious games.

9. REFERENCES

[1] Hamari J. & J. K. (2013) Social motivations to use gamification: An empirical study of gamifying exercise , p. 3.

[2] Unity. URL: https://unity.com/. Accessed 14.2.2022.

[3] Goel S., K. W. & E. D. (2017) Got phished? internet security and human vulnerability 18, pp. 22–44.

[4] Appendix: The sdgs and cybersecurity. URL: https://www.newamerica.org/cybersecurity-initiative/reports/securing-digital-dividends/appendix-the-sdgs-and-cybersecurity///. Accessed 2.2.2022.

[5] Hack The Box. URL: https://www.hackthebox.com/about-us. Accessed 24.1.2022.

[6] McDaniel L., Talvi E. & Hay B. (2016) Capture the flag as cyber security introduction , p. 1.

[7] Wikipedia contributors (2021), Wargame (hacking) — Wikipedia, the free encyclopedia. URL: https://en.wikipedia.org/w/index.php?title=Wargame_(hacking)&oldid=1056454617, [Online; accessed 2-February-2022].

[8] Boopathi K., Sreejith S. & Bithin A. (2015) Learning cyber security through gamification , p. 1.

[9] Nato cooperative cyber defence centre of excellence. URL: https://mil.ee/en/landforces/ccdcoe/. Accessed 2.2.2022.

[10] Palmer C. (2001) Ethical hacking , p. 2.

[11] What is ethical hacking? URL: https://www.eccouncil.org/ethical-hacking/. Accessed 25.1.2022.

[12] Kali linux release history. URL: https://www.kali.org/releases/. Accessed 26.1.2022.

[13] Nmap. URL: https://nmap.org/. Accessed 14.2.2022.

[14] Lynis. URL: https://cisofy.com/lynis/. Accessed 14.2.2022.

[15] Aircrack-ng-. URL: https://www.aircrack-ng.org/. Accessed 14.2.2022.

[16] Force C.J.T. (2017) Cybersecurity curricula 2017. • Association for Computing Machinery, pp. 23–78.

[17] Try Hack Me. URL: https://tryhackme.com/. Accessed 14.2.2022.

[18] OverTheWire. URL: https://overthewire.org/wargames/. Accessed 24.1.2022.

[19] SANS Cyber Ranges. URL: https://www.sans.org/cyber-ranges/. Accessed 1.3.2022.

[20] The Living Security Escape Room. URL: https://www.thecyberescape.com/cyber-home36277718. Accessed 14.2.2022.

[21] Thales Cyber Escape Room. URL: https://www.thalesgroup.com/en/cyber-escape-room. Accessed 14.2.2022.

[22] Infosequre. URL: https://www.infosequre.com/security-awareness-escape-room. Accessed 14.2.2022.

[23] Kafai Y.B. & Burke Q. (2015) Constructionist gaming: Understanding the benefits of making games for learning. Educational Psychologist 50, pp. 313–334. Cited By :155.

[24] Alqahtani H. & Kavakli-Thorne M. (2020) Design and evaluation of an augmented reality game for cybersecurity awareness (cybar). Information 11. URL: https://www.mdpi.com/2078-2489/11/2/121.

[25] Liang H. & Xue Y. (2010) Understanding security behaviors in personal computer usage: A threat avoidance perspective*. Journal of the Association for Information Systems 11, pp. 394–413. URL: https://www.proquest.com/scholarly-journals/understanding-security-behaviors-personal/docview/734860834/se-2?accountid=13031.

[26] Wiemker M. Elumir E. C.A. (2015) Escape room games: "Can you transform an unpleasant situation into a pleasant one?" , p. 2.

[27] Mysteeri 24/7. URL: https://www.laurea.fi/hankkeet/m/mysteeri-247/. Accessed 27.1.2022.

[28] Mysteeri 24/7. URL: https://www.hamk.fi/projektit/mysteeri-24-7//. Accessed 26.1.2022.

[29] LaValle S.M. (2016) VIRTUAL REALITY. Cambridge University Press.

[30] Warmelink H., Haggis M., Mayer I., Peters E., Weber J., Louwerse M. & Heijligers B. (2017) Amelio: Evaluating the team-building potential of a mixed reality escape room game , p. 3.

[31] Azuma R.T. (1997) A survey of augmented reality. Presence: teleoperators & virtual environments 6, pp. 355–385.

[32] Wu H.K., Lee S.W.Y., Chang H.Y. & Liang J.C. (2013) Current status, opportunities and challenges of augmented reality in education. Computers & education 62, pp. 41–49.

[33] Vuforia. URL: https://developer.vuforia.com/. Accessed 14.2.2022.

[34] Kristian K. (205) Digital game-based learning: Towards an experiential gaming model 8, pp. 13–24. URL: https://doi.org/10.1016/j.iheduc.2004.12.001.

[35] Vuforia Area Target Creator App. URL: https://library.vuforia.com/creating-area-targets/vuforia-area-target-creator-app. Accessed 11.3.2022.

[36] Beynon-Davies P. & Holmes S. (2002) Design breakdowns, scenarios and rapid application development, vol. 44. 579-592 p. URL: https://www.sciencedirect.com/science/article/pii/S0950584902000782.

[37] Rapid application development (RAD) for beginners. URL: https://powerapps.microsoft.com/en-us/rapid-application-development-rad/. Accessed 14.3.2022.

[38] PTC Inc. URL: https://www.ptc.com/en/about. Accessed 11.3.2022.

[39] Vuforia Engine Overview. URL: https://library.vuforia.com/getting-started/vuforia-features. Accessed 11.3.2022.

[40] UBICOMP. URL: https://www.oulu.fi/en/center-ubiquitous-computing. Accessed 11.3.2022.

[41] Vuforia Best Practices for preparing and Scanning an Environment. URL: https://library.vuforia.com/area-targets/best-practices-scanning-environment. Accessed 11.3.2022.

[42] Combining Mulitple Area Targets. URL: https://library.vuforia.com/develop-area-targets/combining-multiple-area-targets. Accessed 11.3.2022.

[43] Unity Asset Store. URL: https://unity3d.com/quick-guide-to-unity-asset-store. Accessed 11.3.2022.

[44] SSH Terminal Emulator. URL: https://assetstore.unity.com/packages/tools/gui/ssh-terminal-emulator-185376. Accessed 11.3.2022.

[45] SSH.NET. URL: https://github.com/cavalrytactics/SSH.NET.git. Accessed 11.3.2022.

[46] VtNetCore. URL: https://github.com/cavalrytactics/VtNetCore.git. Accessed 11.3.2022.

[47] What is a DLL. URL: https://docs.microsoft.com/en-us/troubleshoot/windows-client/deployment/dynamic-link-library. Accessed 19.3.2022.

[48] Hacking Android phone remotely using Metasploit. URL: https://irfaanshakeel.medium.com/hacking-android-phone-remotely-using-metasploit-43ccf0fbe9b8. Accessed 23.5.2022.

[49] Mathew, K., Tabassum, M., and Lu Ai Siok, M. V. (2014) A study of open ports as security vulnerabilities in common user computers , p. 1.

[50] Overview. URL: https://docs.csc.fi/computing/overview/. Accessed 21.3.2022.

[51] Vuforia Engine pricing. URL: https://www.ptc.com/en/products/vuforia/vuforia-engine/pricing. Accessed 29.3.2022.

[52] Hussain, M., Park, J., Kim, H. K., Lee, Y., & Park, S. (2021) Motion sickness indexes in augmented reality environment , pp. 1155–1160.

[53] Krathwohl D.R. (2002) A revision of bloom's taxonomy: An overview. Theory Into Practice 41, pp. 212–218. URL: http://www.jstor.org/stable/1477405.

[54] Sart. URL: https://ext.eurocontrol.int/ehp/?q=node/1608. Accessed 19.4.2022.

[55] Arcore. URL: https://developers.google.com/ar. Accessed 19.4.2022.

10. SUPPLEMENTARY MATERIALS

Study description

1. Procedures: In this experiment you will be asked to use an augmented reality application and fill a questionnaire when done.
2. Time involvement: The experiment takes up to 20 minutes of gameplay. After the gameplay you will fill a questionnaire that takes roughly 5 minutes.
3. Participant's rights: Your participation in this study is voluntary and you have the right to withdraw your consent or discontinue participating at any time. Your identity will not be disclosed in any published material and written material resulting from the study.

Authorization for the use of the collected data for research purposes

This form is intended to inform you about how your results and information will be used or disclosed in the study. Your information will only be used in accordance with this authorization form and the informed consent form and as required or allowed by law. Please read the following carefully before signing this authorization form.

1. You do not have to sign this authorization form. But if you do not, you will not be able to participate in this research study.
2. If you decide to participate, you are free to withdraw your authorization regarding the use and disclosure of the results information (and to discontinue any other participation in the study), except to the extent that the law allows us to continue using your information (e.g. necessary to maintain integrity of research).
3. Your observations, comments and information are pseudonymized, so that the data cannot be linked to your person.
4. The following students from the University of Oulu are authorized to use your results in connection with this study as described above: Anssi Antila (aantila19@student.oulu.fi), Jouni Annamaa (jannamaa19@student.oulu.fi) and Jaakko Ohrankämmen ((johranka@student.oulu.fi).

Hack the room bachelor's thesis study, University of Oulu

Signature of participants / date

Researcher signature

Figure 27. Field experiment consent form.

Introduction questionnaire

Answer the questions below before starting the experiment:

Age:

Gender:
- ☐ Male
- ☐ Female
- ☐ Other

Do you have previous experience in augmented reality systems?

Do you have previous experience in cyber security?

Do you have any physical limitations or injuries that can affect game experience e.g. back pain or walking troubles?

Figure 28. Experiment introductory questionnaire.

Experiment instructions

Introduction:
This experiment simulates a hacking situation in an Augmented Reality (AR) game using a Virtual Machine (VM). You will perform three tasks with varying difficulties using a Linux Virtual Machine (VM). We suggest reading the information provided by the terminal carefully.

The experiment starts in the main menu of the game. After the main menu, you will read three panels of text that give general instructions on what to do next in the game. The first task is performed using only the ls and cat commands.

The game uses a scanning function to search for hidden terminals. The scanning function utilizes a proximity percentage to see if you are close to a hidden terminal. When the proximity percentage is at 100%, a “SCAN” button will appear. Press this button to open the terminal.

If a found terminal is lost or disappears, try to look around if the smartphone sees other 3D objects in the room. If it doesn’t, ask for help.

This instructions sheet has useful commands for the game, keep it with you while playing. Please read through this paper carefully.

Functions:

Scanning:
Use the proximity percentage to see how close you are to a terminal and press the scan button when the percentage is at 100%.
Note: the game may lag/jitter at times when scanning, this is normal and means the game is loading objects into the room. If a terminal disappears, you can move the smartphone to cause the jitter so objects load back into the room.

Virtual machine (VM):
The Virtual machine behaves exactly like a normal Linux terminal. The in-game terminal connects into a live virtual machine located online. Everything in the VM is case sensitive, for example if inserting a command it has to be “ls” and not “Ls”.

Numpad:
Input passcodes obtained from the terminals and input fields into this.

Figure 29. Field experiment instructions.

End of experiment questionnaire

Learning questions:

On what port was the SSH service running?

What commands were used in the terminal?

What part of the game was related to the virtual machine?

What was nmap used for?

What does the cd command do?

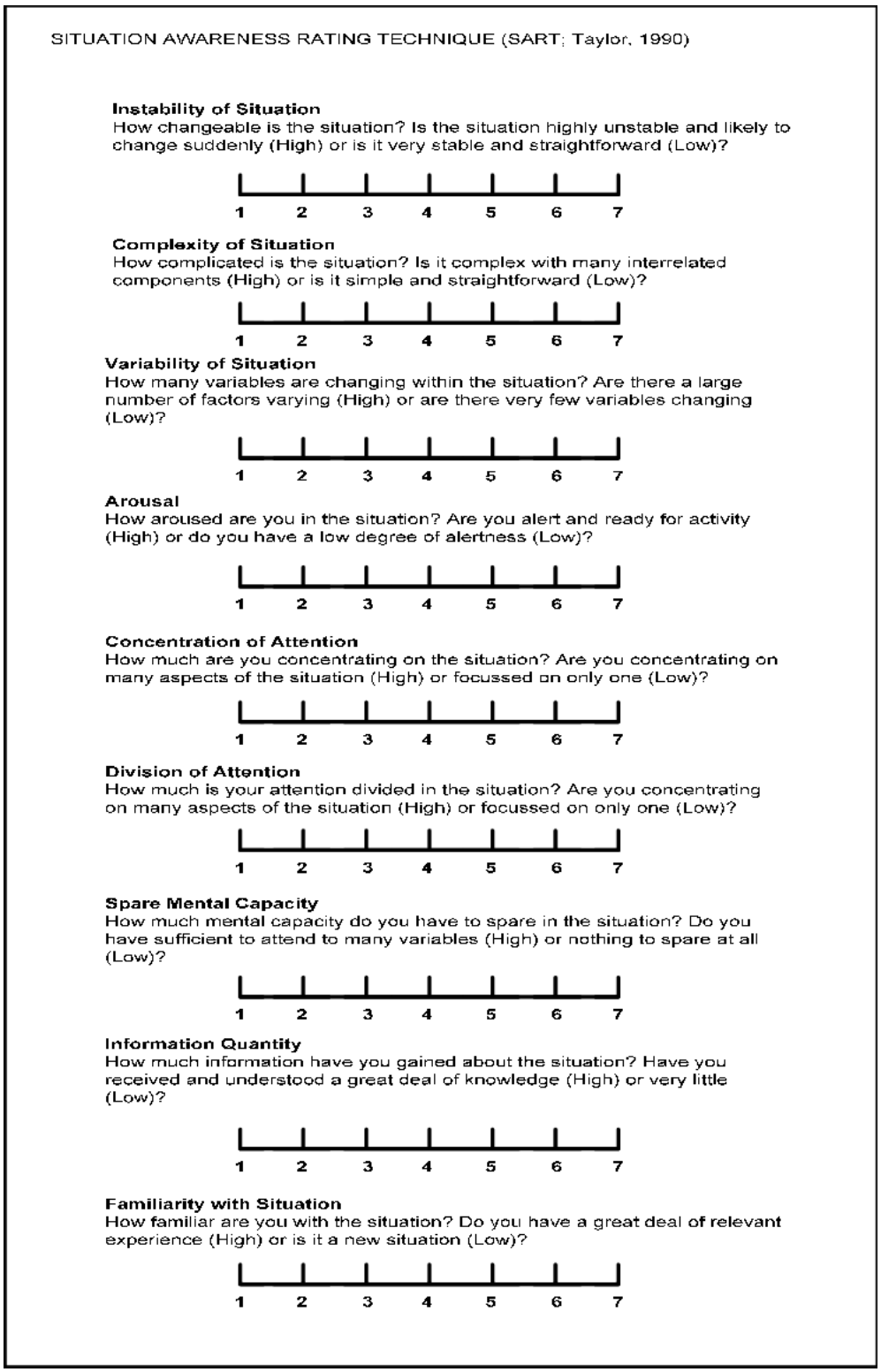
How would you generalize what you learned during the experiment?

Performance questions:

On a scale of 1 to 10, how would you rate the performance of the game?

Did you run into any issues when playing the experiment?

Figure 30. End of experiment Bloom's taxonomy questions.



SITUATION AWARENESS RATING TECHNIQUE (SART; Taylor, 1990)

Instability of Situation
How changeable is the situation? Is the situation highly unstable and likely to change suddenly (High) or is it very stable and straightforward (Low)?

1 2 3 4 5 6 7

Complexity of Situation
How complicated is the situation? Is it complex with many interrelated components (High) or is it simple and straightforward (Low)?

1 2 3 4 5 6 7

Variability of Situation
How many variables are changing within the situation? Are there a large number of factors varying (High) or are there very few variables changing (Low)?

1 2 3 4 5 6 7

Arousal
How aroused are you in the situation? Are you alert and ready for activity (High) or do you have a low degree of alertness (Low)?

1 2 3 4 5 6 7

Concentration of Attention
How much are you concentrating on the situation? Are you concentrating on many aspects of the situation (High) or focussed on only one (Low)?

1 2 3 4 5 6 7

Division of Attention
How much is your attention divided in the situation? Are you concentrating on many aspects of the situation (High) or focussed on only one (Low)?

1 2 3 4 5 6 7

Spare Mental Capacity
How much mental capacity do you have to spare in the situation? Do you have sufficient to attend to many variables (High) or nothing to spare at all (Low)?

1 2 3 4 5 6 7

Information Quantity
How much information have you gained about the situation? Have you received and understood a great deal of knowledge (High) or very little (Low)?

1 2 3 4 5 6 7

Familiarity with Situation
How familiar are you with the situation? Do you have a great deal of relevant experience (High) or is it a new situation (Low)?

1 2 3 4 5 6 7

Figure 31. SART questionnaire.